Journal Title
XX(X):1–16
©The Author(s) 2026
Reprints and permission:
sagepub.co.uk/journalsPermissions.nav
DOI: 10.1177/ToBeAssigned
www.sagepub.com/
SAGE

ProQ-KG: Integrated Cyber-Physical Production System Knowledge Graph for Quality Issue Analysis

Kabul Kurniawan[1], **Sebastian Kropatschek**[4,3], **Elmar Kiesling**[2], **Oskar Gert**[4], **Dietmar Winkler**[4,3], **Thilo Sauter**[3], **and Stefan Biffl**[3]

Abstract

Analyzing complex quality issues in the context of modern production systems is a challenging task that requires the coordination of multiple stakeholders and the integration of their knowledge across multiple domains. This heterogeneous knowledge necessary for quality issue analysis can be organized based on products, processes, and production system resources (PPR) on the one hand, and potential failure modes and effects (FMEA) on the other hand. In this paper, we introduce an integrated ontology that unifies Product, Process, Resource (PPR) and Failure Mode and Effect Analysis (FMEA) modeling approaches, and to provide a foundation for *(i)* the construction of integrated FMEA-PPR knowledge graphs, and *(ii)* the coordination of quality issue analysis among heterogeneous domain experts using knowledge-graph-based methods. Furthermore, we propose ProQ-KG, a knowledge-graph-based framework that facilitates quality issue analysis and multi-view coordination across diverse stakeholders. We implement our KG-based quality issue analysis approach and demonstrate its feasibility by means of three real-world use-case applications in collaboration with an industrial partner from the automotive industry. Our evaluation shows that the integrated approach makes it easier for stakeholders to structure and share their mental models as well as to navigate the integrated knowledge to identify root causes of quality issues.

Keywords

CPPS, Knowledge Graphs, Industry 4.0, FMEA, PPR, Quality Issue Analysis

Introduction

Cyber-physical Production Systems (CPPSs) have been characterized as complex production systems that integrate both physical and digital components (1). This integration enables flexibility and allows the production system to adapt to the needs of complex multi-variant products. The complexity of these systems arises from the entanglement of numerous processes, products, materials, and resources that all affect the quality of the final product. Achieving and maintaining high-quality products in Cyber-physical Production System (CPPS), therefore, becomes increasingly challenging. This is exacerbated by the fact that the engineering and operation of CPPSs typically involve numerous stakeholders from different domains (e.g., mechanical, electrical, software, etc.). Their diverging views, knowledge, and interpretations can hinder efficient coordination, analysis, and mitigation of quality issues.

In CPPSs, analyzing product quality issues and finding potential root causes of a failure is critically important in order to prevent high unplanned costs, product rework, project delays, etc. However, finding the causes of quality issues becomes increasingly difficult in the CPPS context due to the vast amount of data scattered across domains and resources.

Research in the domain of production quality issue analysis has mostly focused on FMEA (2; 3; 4). FMEA is an engineering technique widely used in manufacturing domain to define, review and identify potential failures, their causes and effects on systems, designs, processes, or services (3). FMEA provides a rigorous method for "capturing" failure-cause-effect information in a structured manner. In this context, FMEA provides an approach to identify possible causes of a failure and determine appropriate mitigation actions (5). However, FMEA is typically highly document-centric and aimed at descriptiveness for human consumption rather than machine-interpretability for automated agents (6). Consequently, laborious reconstruction of knowledge from design documentation is commonly necessary.

Given the emergence of complex production systems such as CPPS, a purely traditional FMEA approach is considered less effective and efficient for quality issue analysis. CPPSs typically involve multiple engineering domains and coordinating tasks across them in a collaborative environment requires a common understanding of different stakeholders, which, however, largely remains an unresolved challenge

---

[1] Department of Computer Science and Electronics, Universitas Gadjah Mada, Indonesia
[2] Institute of Data, Process and Knowledge Management, WU Vienna, Austria
[3] Inst. of Information Systems Engineering, Technische Universität Wien, Karlsplatz 13, 1040 Vienna, Austria
[4] Austrian Center for Digital Production, Seestadtstraße 27/19, 1220 Vienna, Austria

Corresponding author:
Kabul Kurniawan, Department of Computer Science and Electronics, Universitas Gadjah Mada, Yogyakarta, Indonesia.
Email: kabul.kurniawan@ugm.ac.id

(7). To this end, resolving quality issues requires a comprehensive approach that not only integrates heterogeneous domain knowledge but also provides effective and efficient collaboration of multiple engineering disciplines (8)(9).

Knowledge Graphs (KGs) have been widely used in various domains including manufacturing (10). They provide a semantically flexible data model (through graph representation) and a machine-interpretable and understandable format. This makes it easier to integrate heterogeneous sources and infer new unseen information. Several approaches to address the limitations above by means of KGs exist (11; 12; 13; 14; 15). However, they typically do not consider complex production systems that involve integration and linking across multi-disciplinary engineering domains such as CPPS. The lack of linking among those domains can inhibit efficient collaboration for quality issue analysis, leading to additional risk and defects in the CPPS (16).

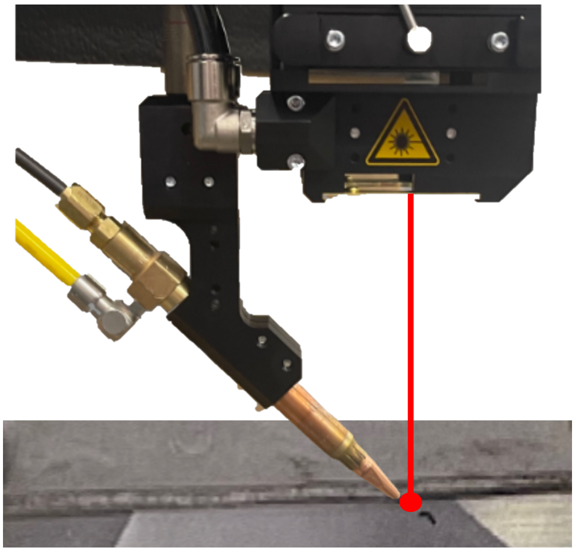

Figure 1. Laser Beam Welding Illustration

Motivating Example We conducted a study involving domain analysis at a major supplier of aluminum parts for the automotive industry in Europe. The study focused on weld seam quality in a CPPS for producing structural car parts using laser beam welding. Figure 1 illustrates the welding process that consists of several elements, such as *(i)* the *laser beam* that heats the material to join two pieces; *(ii)* the *filler material supply* for delivering filler material; and *(iii)* the *gas nozzle* to furnish welding gas. Maintaining alignment for the gas shield is crucial for consistent penetration depth.

Weld seam quality is influenced by a large variety of factors; relying on a single analysis perspective is not sufficient for discovering the causes and effects of a failure. For example, mechanical engineers may have their own perspective regarding the influence of stained protection glass against the laser power and the weld seam quality. However, there is no view on the possible cause-and-effect relationship from other domains, such as electrical or software engineering. Therefore, the collaborative identification of these causes is a time-consuming and difficult process, as it involves several stakeholders, including *Quality Managers, Process Experts, Mechanical Engineers, Data Curators, and Operators*.

To address these challenges, we introduce a KG-based approach that integrates FMEA with PPR knowledge for cross-domain and multi-view quality issue analysis. It extends our prior work (8) with a semantic model and shows how the resulting knowledge graph can serve as a coordination artifact and provide a foundation for production quality issues analysis. Specifically, our contributions are as follows: *(i)* formalized model and concept for integrated and collaborative quality issue analysis. *(ii)* standardized vocabularies that provide a basis for domain knowledge representation; *(iii)* KG-based architectural framework and its prototype implementation; *(iv)* Use-case applications derived from real-world production system scenarios as part of the evaluation.

The remainder of this paper is structured as follows: Section 2 reviews related work and highlights research gaps; Section 3 puts forth a set of requirements as a basis of our proposed solution; Section 4 discusses the conceptual model of our approach; Section 5 introduces the implementation of conceptual mode by means of ontology and KG; Section 6 discusses the prototypical implementation of our approach and the application scenarios; and Section 7 highlights the advantages, limitations and future work; Finally, we conclude with an outlook on future research in Section 8.

Related Work

In this section, we review three streams of related work that form the foundation of our approach: *(i)* Failure Modes and Effect Analysis (FMEA) and Quality 4.0; *(ii)* Multi-view model-based quality analysis; and *(iii)* Ontological/Knowledge Graph-Based Quality Issue Analysis.

FMEA & Quality 4.0

Quality issue analysis is critical for ensuring the reliability of processes and resources as well as the quality of products in PPR model. FMEA is a prominent approach for conducting quality issue analysis in manufacturing. FMEA has been recognized as a well-established method to identify known and potential failures, their corresponding cause and effect, and their prioritized mitigations (2; 3). However, whereas FMEA is typically well supported in individual engineering disciplines, conducting quality analyses in complex production systems across disciplines remains a challenge e.g. due to fragmented knowledge on cause-effect relationships that come from different domains and inconsistent documentation standards across manufacturing environments (17).

Building upon traditional FMEA limitations, several researchers have proposed enhancements to address modern manufacturing challenges. Razouk and Kern (25) focused on improving the consistency of FMEA documents in semiconductor manufacturing, addressing the critical issue of documentation standardization across complex production environments. Okazaki et al. (26) developed a framework to support failure cause identification through generalization of past FMEAs, demonstrating how historical knowledge can be leveraged to improve current analysis. This approach addresses the reusability challenge in FMEA by creating

Table 1. Comparison of Graph-based Approaches. (*N/A*): not available, (✓): provided, (✗): not provided.

Related Papers	UseCase Domain	Artifacts Involved	Problem Focus	Data Model /Representation	MD/KI	MV/SC	R & S	Query Language
Kropatschek, et al (8)	Laser Welding	FMEA, PPR	RCA, QIA	Unspecified	✓	✓	N/A	N/A
Biffl, et al (17)	Screwing System	FMEA,PPR	RA	Unspecified	✓	✓	N/A	N/A
Winkler, et al (16)	Truck Assembly	PPR	RA, RCA	N/A	✓	N/A	N/A	N/A
Hoffmann, et al (18)	Laser Welding	FMEA, PPR	RA, RCA	Unspecified	✓	N/A	N/A	N/A
Biffl, et al (19)	Laser Welding	FMEA, PPR	RA	Unspecified	✓	N/A	N/A	N/A
Dittmann et al. (11)	N/A	FMEA	RCA	Ontology	✗	✗	N/A	F-Logic
Koji, et al. (12)	Welding	FMEA	KE	Ontology	✗	✗	N/A	N/A
Ebrahimipour et al. (13)	N/A	FMEA	KE	RDF/OWL	N/A	N/A	✓	N/A
Leitao et al. (20)	Washing Machine	PPR	KE	OWL	N/A	N/A	✓	N/A
Rehman et al. (15)	Automotive	FMEA	QIA	RDF/OWL	N/A	N/A	✓	SPARQL
Polenghi et al. (21)	Food Industry	PHA, PHM	MSP	RDF/OWL	✓	N/A	✓	SPARQL
Hodkiewicz, et al. (22)	Heating System	FMEA	QIA	RDF/OWL	N/A	N/A	N/A	N/A
Martinez, et al. (23)	Power Transformers	N/A	RCA	RDF/OWL	✗	✗	✓	SPARQL
Bachhofer et al. (24)	Injection Molding	FMEA	RCA	RDF/OWL	✓	✗	✓	SPARQL
Our Approach	Laser Welding	FMEA, PPR	RCA, QIA	RDF/OWL	✓	✓	✓	SPARQL

Note: MD/KI (Multi-Disciplinary / Knowledge Integration), MV/SC (Multi-View State Coordination), R & S (Reusability & Sharing), RA (Risk Assessment), RCA (Root Cause Analysis), QIA (Quality Issue Analysis), Knowledge Extraction (KE).

systematic methods to extract and apply lessons learned from previous failure analyses.

Recent developments in Industry 4.0 and Quality 4.0 have emphasized the integration of data analytics and knowledge-driven approaches in quality management systems. Salah et al. (27) proposed risk prioritization using modified FMEA analysis tailored for Industry 4.0 contexts, incorporating new risk factors introduced by digital transformation. Bousdekis et al (28) provide a literature review of data analytics in Quality 4.0, demonstrating the shift towards more sophisticated, data-driven quality assurance methods.

Multi-View Model-Based Quality Issue Analysis

In dynamic and flexible production systems such as CPPSs, the production system engineering landscape typically involves multiple engineering disciplines, such as mechanical, electrical, and software engineering. It requires an integration of heterogeneous models and tools. Consequently, traditional FMEA cause-effect hypotheses are often difficult to test due to a lack of interdisciplinary links. To tackle this challenge, Biffl, et al (17) proposed the CPPS Risk Assessment (CPPS-RA) approach that represents FMEA and links to CPPS Engineering Network (CEN), an integrated multi-domain multi-view model to represent and identify the possible cause-effect relationships.

Cross-domain modeling facilitates collaboration among different domain experts by capturing and representing and integrating their heterogeneous knowledge. To develop interconnection between the domains, the *three-view-concept* is introduced, i.e., PPR (29; 30). The concept provides the ability to those classes and their relationship that are widely used in the manufacturing domain, i.e., *products* are produced by a certain engineering *process* that involves (multiple) resources. However, the PPR issues are typically addressed individually without explicitly linking their dependencies. Therefore, it can lead to overseen dependencies that increase the effort for risk management e.g., failure detection and mitigation. To address this challenge, PPR Asset Network (PAN) (16) was introduced to add dependencies to engineering assets (i.e., PPR) forming an engineering network. It represents an overview structure of engineering systems and their dependency that set as foundation for downstream tasks e.g., risk mitigation, failure detection, etc.

Building on these foundations, Hoffmann et al. (18) advanced this stream by integrating data-driven and knowledge-driven analytics for interdisciplinary production risk exploration. These approaches highlight the shift toward hybrid frameworks where multi-view models are complemented with analytical reasoning for proactive quality management. Biffl et al. (19) further demonstrated methods for configuring and validating multi-aspect risk knowledge in Industry 4.0 information systems, showing how multi-view coordination can be operationalized in industrial contexts.

Extending the existing CEN and the foundational PAN approaches, the FMEA - PPR Asset Issue Analysis (FPI) (8) was introduced to represent cross-domain knowledge dependencies and multi-view coordination assets to guide quality issue analysis. It integrates and maps the existing FMEA knowledge to the PPR and facilitates collaborative, cross-domain quality issues analysis by propagating investigation state markers. Thereby, it makes the identification of potential causes of quality issues efficient. In this paper, we aim to concretize the FPI approach by introducing ontology for FPI and providing a framework for knowledge graph construction and applying knowledge graph-based quality issue analysis based on Semantic Web technology.

Ontological/Knowledge Graph Based Quality Issue Analysis

The use of the Ontological and KG-based approach in addressing quality issues in production systems and manufacturing has been widely explored by many researchers (4; 31). They include the use of Ontological-based FMEA (11; 12; 13; 14; 15; 22), KG-based approach for root cause analysis (23; 24), Ontological process and quality control (20)

Dittmann et al. (11) introduce the combination of *knowledge engineering* techniques (ontologies) and *quality engineering* techniques (FMEA) in order to address major limitations of traditional FMEA, such as the lack of *explicit knowledge* and *reusability*. The proposed ontologies facilitate domain knowledge integration, structurization, and reuse. It provides the possibility to build an integrated manufacturing system from heterogeneous fields.

Further promoting such knowledge sharing, Koji, et al. (12) proposed a knowledge transformation system from an extended functional model and FMEA sheet based on ontology engineering. The developed transformation systems can facilitate engineers' activities such as design, reliability analysis, and redesign. Building on similar principles, Xiuxu and Yuming (14) proposed an ontology-based representation method for FMEA knowledge, enabling unified management, sharing, reuse, and retrieval of dispersed FMEA knowledge to support manufacturing process quality analysis and decision-making.

Ebrahimipour et al. (13) proposed an ontology based on standard data integration (i.e., ISO-15926) to improve the representation of knowledge produced during FMEA. It enables engineers to use more informative queries to find relevant information during the safety analyses. Similarly, Rehman et al. (15) proposed an ontology for Process Failure Mode and Effect Analysis (PFMEA) that aims to enable better management of all existing automotive knowledge that is reusable and shareable. The proposed ontology facilitates automated reasoning and data retrieval for the automotive domain.

Hodkiewicz, et al. (22) proposed an ontological approach to make classes, relationships, and hierarchy from spreadsheet-based FMEA explicit. It provides the ability to automatically infer dependency through system hierarchy, e.g. the final failure effect at the system level is propagated from components failure via Web Ontology Language (OWL)-Description Logic (DL) reasoning.

Martinez, et al. (23) proposed KG-based approach for root-cause analysis of failures in the industrial domain based on three points i.e., semantic reasoning, rule-based failure classification, and graph query. Bachhofer et al. (24) proposed KG-based framework that enhances quality management on the shop floor in the injection molding industry. It extends the existing FMEA ontology proposed by Rehman et al. (15) with knowledge from *injection process adjustment protocols*, standard procedure definitions for injection parameter adjustment.

Polenghi et al. (21) developed a multi-attribute ontology-based criticality analysis framework specifically for manufacturing assets in maintenance strategy planning. Their approach demonstrates how ontological modeling can support complex decision-making processes by integrating multiple criteria and stakeholder perspectives in asset criticality assessment.

More recent research continues this trajectory, highlighting the industrial adoption of KGs. Martinez-Gil et al. (32) provide a comprehensive review of KG adoption in manufacturing, identifying opportunities and barriers to large-scale deployment. Wan et al. (33) further outline how KGs enable smart manufacturing by bridging semantic representation with analytics and optimization.

Ontological and KG-based methods are becoming central to quality issue analysis by enabling semantic integration, cross-domain reasoning, and collaborative decision-making. However, most approaches typically focus on either specific domains or isolated tasks and do not consider complex, broad-range engineering domains such as CPPS. In this paper, we introduce KG-based approach for quality issue analysis that facilitates a multi-disciplinary and multi-view model by extending FMEA knowledge and linking them to the PPR model to enable more comprehensive and integrated analysis across domains

System Requirements

Based on our analysis of the limitations of the state of the art (See Section 2), we identified a set of requirements (R1 - R6) for the development of a knowledge graph-based approach for quality issue analysis.

- **R1: Uniform representation** Production systems generate a large amount of data from various sources that typically have different formats and structures. Thus, the system should be able to harmonize and integrate the data from different sources in a uniform way to facilitate the analysis of quality issues.

- **R2: Integration of heterogeneous engineering domains** The production systems involve various engineering domains and disciplines – e.g., electrical, mechanical or software. Each domain has its own set of data and knowledge that needs to be integrated in order to analyze quality issues.

- **R3: Multi-view coordination assets** Production systems interact with various stakeholders with their own perspective and knowledge about the production systems (34). The system should provide coordination of cross-domain knowledge between different stakeholders.

- **R4: Knowledge evolution and change management** Production systems are subject to changes due to, e.g., design, process, and equipment changes. These changes can affect the quality of the products and services. The system should facilitate the evolution and changes in the knowledge to ensure the quality of the products and services.

- **R5: Knowledge reusability and extensibility** The production system should provide a mechanism to reuse existing knowledge in order to reduce the effort and cost of knowledge acquisition. Moreover, the system should be extensible to accommodate new knowledge that may be required in the future.

- **R6: Standardized query and rule languages** The system should provide a standardized query and rule languages to facilitate the analysis of quality issues. This includes expressions that captures dependencies between different entities in the knowledge graph and propagation between PPR and FMEA to facilitate the root cause analysis of quality issues.

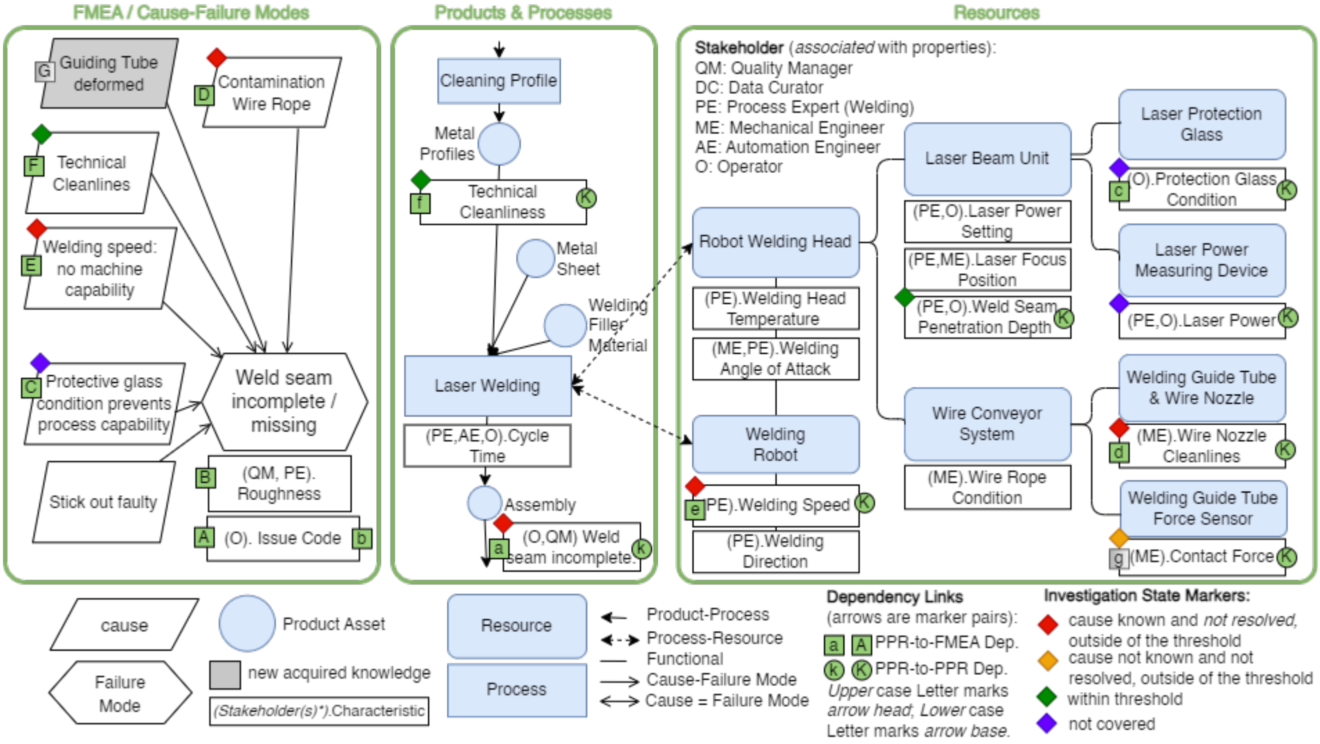

Figure 2. Laser Welding Seam Scenario represented in FPI Model.

Collaborative Quality Issue Analysis Mothodology

In this section, we introduce ProQ-KG as a foundation for collaborative quality issue analysis. The approach is grounded in links between PPR and FMEA models (introduced in our prior FPI model) that enable quality issue analysis. In this paper, we formalize this concept and explain how the approach addresses these challenges.

The FMEA-PPR Asset Issue Analysis (FPI) Approach

The PPR model represents production environments based on three fundamental concepts, i.e., *(i) Product*, input or output as resulting from the production process; *(ii) Process*, activities performed to accomplish a certain task; *(iii) Resources*, components utilized by the process to execute tasks. Furthermore, the PPR model captures the interactions between these elements in production networks.

The FMEA represents the existing knowledge of failure mode and their cause-effect relationships curated by experts.

Figure 2 shows the FPI concept implementation for the *Laser Welding Seam* Scenario described in Section 1. It integrates FMEA and PPR domain knowledge and is structured into three green boundary areas:

- **FMEA Knowledge** (left part) presents the failure modes, cause and effect knowledge derived from the scenario, e.g. a failure mode ”Welding Head wrongly positioned” is linked to its likely cause ”Angle of Attack not favorable”;
- **Processes Knowledge** (middle part) describes the process flow information of the production systems. For example, the welding seam process consists of several sub-process including ”Cleaning Profile”, ”Laser Welding” etc.
- **Products Knowledge** (middle part) provides the information of inputs and outputs of the process. For example, ”Metal Sheet” and ”Metal Profiles” as product input and ”Assembly” as product output.
- **Resources Knowledge** (right part) represents the system components involved in a process e.g. ”Welding Head”, ”Laser Beam Unit”, etc.

The FPI model represents diverse perspectives from different domain experts as depicted in Figure 2, such as *Mechanical Engineer, Operator, Quality Manager, Process Expert, FMEA Expert, etc.*

Modelling Dependency Links between Domain Knowledge

Modeling dependency links is important to establish connections and provide integration between heterogeneous domain knowledge. Once connected, they can enable the traversal of dependency pathways and support integrated quality issue analysis. We formalize six types of dependency links for FMEA-PPR model as identified in (8):

Definition 1 *(Functional Dependency).* A *functional dependency* involves the dependency between and among resources. The availability, performance and behavior of one resource directly affect the other resource.

Let $G = (V, E)$ be a graph representing a manufacturing system, where V is the set of resources, and E is the set of undirected, solid edges (-). A *functional dependency* (FD) between resources v_i, v_j is denoted by $v_i - v_j \in E$ if the availability, performance, or behavior of resource v_i directly affects resource v_j.

Example 1. In a manufacturing system represented by G, if *welding robot* (wr) has a functional dependency on *welding head* (wh), and wh has another functional dependency on the *laser beam unit* (lbu), it can be expressed as $wr - wh - lbu \in E$.

Definition 2 *(Product to Process Dependency).* A *product to process dependency* involves the connection between product properties with the way it is produced. It is important to ensure that the chosen manufacturing process is appropriate to the product specification.

Let $G = (V, P, E)$ be a graph representing a manufacturing system, where V is the set of products, P is the set of processes and E is the set of direct solid edges ($\Longrightarrow$). A *product-to-process dependency* (PPD) between product v_i and process p_j is denoted by $v_i \Longrightarrow p_j \in E$ if the chosen manufacturing process is dependent on the product specification.

Example 2. If the *laser welding* (lw) process includes three product inputs e.g., *metal profiles* (mp), *metal sheet* (ms), and *welding filler material* (wfm), it can be expressed as $\{mp \Longrightarrow lw, ms \Longrightarrow lw, wfm \Longrightarrow lw\} \in E$.

Definition 3 *(Resources to Process Dependency).* A *resources to process dependency* involves the relation between the process and the resources required to carry out those processes. The effectiveness and efficiency of a manufacturing process are closely tied to the availability and capability of these resources.

Let $G = (V, P, E)$ be a graph representing a manufacturing system, where V is the set of resources, P is the set of processes and E is the set of bidirectional, dashed edges ($\dashleftarrow\!\!\dashrightarrow$). A *resources-to-processes dependency* (RPD) between resource v_i and process p_j is denoted by $v_i \dashleftarrow\!\!\dashrightarrow p_j \in E$ if the effectiveness and efficiency of process p_j are closely tied to the availability and capability of resource v_i.

Example 3. To perform the *laser welding* (lw) process, two resources are required, e.g., *robot welding head* (rwh) and *welding robot* (wr). This can be expressed as $\{rwh \dashleftarrow\!\!\dashrightarrow lw, wr \dashleftarrow\!\!\dashrightarrow lw\} \in E$.

Definition 4 *(Cause-Failure-Mode Dependency).* A *cause-failure-mode dependency* involves the relationship between potential causes of failures to failure modes and those failure modes to the potential effects. Note that a cause that is also a failure is indicated as a bidirectional solid edge. Identifying potential cause and effect of a failure mode is a task in FMEA to avoid and reduce failures.

Let $G = (V, F, E)$ be a graph representing a manufacturing system, where V is the set of potential causes, F is the set of failures and E is the set of direct edges ($\longrightarrow$). A *cause-failure-mode dependency (CFD)* between potential causes v_i and potential failures f_j is denoted by $c_i \longrightarrow f_j \in E$ representing the relationship between causes and failure modes.

Example 4. If *welding seam incomplete failure* $(wsif)$ can be caused by *insufficient technical cleanliness* (itc) or the *Contamination wire rope* (cwr), it can be expressed as $\{itc \longrightarrow wsif, cwr \longrightarrow wsqf\} \in E$.

Note that a failure mode can also be a cause of another failure mode. In this case, the relationship between failure modes and causes is identical, i.e., *failure-mode = cause* and can be represented as bidirectional edges ($\longleftrightarrow$). As such, a failure mode that is not a cause, can be considered as an effect.

Definition 5 *(PPR to FMEA Dependency).* A *PPR to FMEA dependency* creates a relation between characteristics from both PPR and FMEA in general. Establishing direct connections between characteristics makes it easier to model cross-domain relationships for diagnosis and investigation.

Let $G = (V, F, E)$ be a graph representing a manufacturing system, where V is the set of characteristics for PPR entities, F is the set of characteristics for FMEA entities and E is the set of directed, dashed edges ($\dashrightarrow$). A *ppr-to-fmea dependency (PFD)* between characteristics of PPR entities v_i and FMEA entities f_j is denoted by $v_i \dashrightarrow f_j \in E$ to model that the identified PPR characteristic has been acknowledged in FMEA.

Example 5. The *welding nozzle cleanlines* (wnc) characteristic from the *welding-guide-tube-and wire-nozzle* resource has been acknowledged as *Contamination wire rope* (cwr) from the FMEA. This dependency can be expressed as $wnc \dashrightarrow cwr \in E$.

Definition 6 *(PPR to PPR Dependency).* A *PPR to PPR dependency* creates a relation between characteristics from both PPR and FMEA in general. Establishing direct connections between characteristics makes it easier to model cross-domain relationships for diagnosis and investigation.

Let $G = (V, E)$ be a graph representing a manufacturing system, where V is the set of characteristics for PPR entities and E is the set of directed, dashed edges ($\dashrightarrow$). A *ppr-to-ppr dependency (PrPrD)* between characteristics of PPR entities v_i and v_j is denoted by $v_i \dashrightarrow v_j \in E$ establishing PPR to PPR direct connections for diagnosis and investigation.

Example 6. The *welding-seam-incomplete* (wsi) characteristic from the *assembly* output product can be influenced by several other characteristics, such as *technical-cleanlines* (tc) characteristic from the *metal-profile* input product as well as *wire nozzle cleanlines* (wnc) characteristic from the *welding-guide-tube-and wire-nozzle* resource. This dependency can be expressed as $\{wsi \dashrightarrow tc, wsi \dashrightarrow wnc\} \in E$.

Designing Collaborative Investigation for Quality Issue Analysis

Quality issue analysis is a multifaceted and iterative process that requires the expertise of domain expert in the intricacies of their respective fields. Given the diverse nature of the PPR-FMEA model, collaboration among various experts in this diagnostic process is essential.

Investigation State Marker (ISM) To facilitate and guide this collaborative effort, we introduce the *Investigation State Marker (ISM)*. The ISM serves as an indicator of a current coordination state and can be assigned to an element or property in the FMEA-PPR model. This can be done either automatically through propagation or manually by the investigator. The markers are gradually updated and changed

Table 2. Type of Investigation Marker

Symbol	MarkerType	Indicator	Cause	Problem	Intervention	KnowledgeRef	DataRef
🟥	Red Marker	Property Deviation	Known	Unresolved	Required	FMEA, Expert	Exists
🟧	Orange Marker	Property Deviation	Unknown	Unresolved	Required	Expert	Exists
🟪	Violet Marker	Not Detected	Known / Unknown	Unknown	Maybe Required	FMEA, Expert	Not Exists
🟩	Green Marker	No Deviation	Known	Resolved	Not Required	FMEA	Exists

to represent the current state of the investigated element or properties. Table 2 shows four different categories of the ISM and how they are utilized.

A *Red Marker* indicates that the investigated element or property of PPR is outside the threshold. The causes of this problem are identified in FMEA, however, the issue remains unresolved. It also means that an intervention from domain experts is necessary to mitigate the problem based on the identified cause provided in the FMEA. The investigator can refer to the existing data provided by the system.

An *Orange Marker* indicates the investigated element or property of PPR also beyond the thresholds. However, the causes of this issue can not be identified or do not exist yet in FMEA. It requires an intervention from the respective domain expert to establish hypotheses on the potential causes of the problem.

A *Violet Marker* indicates an element or property has potential deviation hypothesized by an investigator. It is because no available data existed or the deviation was simply undetected due to the limited data or measurement, e.g. no sensors or measurements available. It requires consultation with the corresponding domain expert to discover possible causes and mitigation.

A *Green marker* indicates that no deviation exists in the current investigated elements or properties therefore no further intervention is required. The marker can also indicate that the existing issues have been already resolved.

Collaborative Quality Analysis Lifecycle We introduce a seven-step quality issue analysis lifecycle to support multi-view coordination during quality issue investigation in a collaborative manner (See Figure 3).

- **Step 1** *Machine Data Identification.* The first step (Step 1) of investigation begins with the identification of existing machine data (e.g., data from the edge devices) and the selection of relevant properties from the respective resource. These properties are the populated into the FPI model and knowledge graph. In this step, stakeholders from the respective domains are also assigned to corresponding elements in the model. For example, The *Data Curator (DC)* is responsible for acquiring and preprocessing machine data, while the *Process Expert (PE)* and *Automation Engineer (AE)* help determine the most relevant attributes to monitor in relation to the quality issue under investigation.

- **Step 2** *Marker Assignment.* The next step is involves assigning a marker to the production failure, often manifested by issuing a scrap code. This marker assignment requires querying and updating nodes in the knowledge graph to capture the current status of the investigation. It includes identifying elements or properties with issues and determining whether the cause is already known. For example, Figure 2 shows a *Red Marker* being triggered due to a quality issue on *Laser Welding* output, which was detected to be outside the required quality threshold. In this case, the *Quality Manager (QM)* issued a *scrap code* to initiate the investigation together with *Process Expert (PE)* and *Mechanical Engineer (ME)* to begin identifying the root cause.

- **Step 3** *Marker Investigation.* In this step, domain experts conduct an in-depth analysis of the issue, updating existing markers or assigning new ones based on the findings. For example, following the laser welding failure, the *Quality Manager (QM)* and *Mechanical Engineer* started identifying dependency links of the issue in PPR and found that it is linked to a property *Welding Position* attached to the *Welding Guide Tube & Wire Nozzle* resource. Based on further investigation, this dependency is also found to be mapped in FMEA. With this situation, another *Red Marker* is assigned to those properties. Another deviation has also been detected on property *Welding Guide Tube Force Sensor* and it is linked to the issue. However, this dependency has never been captured in FMEA, therefore an *Orange Marker* is assigned.

- **Step 4** *Cause Verification.* Subsequent steps involve verifying the hypothesized cause, addressing issues with resolution actions, marking with green, and iterating through the process. For instance, after verifying the cause for *Welding Seam Quality Issue*, the respective stakeholder (i.e, the *Mechanical Engineer (MA)* and *Automation Engineer (AE)* implemented resolution actions, while the *Quality Manager* validated the resolution. As depicted in Figure 3, Step 2 to 4 may be iterated several times to reach a validated resolution.

- **Step 5** *FPI Update and FMEA Re-validation.* The final step includes updating FPI model and re-validating the FMEA. Depending on the analysis outcome from previous step, this may involve an addition of new causes to the model (e.g., triggering marker updates from orange to red or green), inserting new FMEA causes or failure modes, as well as establishing new dependency between PPR to FMEA. The *Quality Manager (QM)* oversees these updates, ensuring alignment with organizational quality standards. In parallel, the *Data Curator (DC)* reconfigures the edge device data pipeline to accommodate monitoring for the newly identified causes.

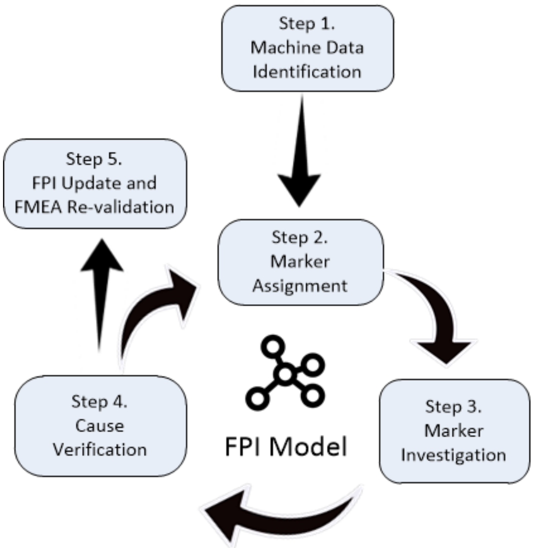

Figure 3. Marker-based Quality Analysis and Investigation Lifecycle

ProQ-KG: Knowledge Graph-based integrated quality issue analysis

In this section, we introduce an implementation of the aforementioned concept using semantic web technologies and knowledge graph approach. We start with the design methodology and development of Resource Description Framework (RDF)[*]/OWL[†]-based ontology as a standard representation for our model. Then, we explain the construction of the knowledge graph based on our proposed ontology.

Knowledge Graph

A knowledge graph is a directed, labeled graph $G = (V, E)$ where V is a set of vertices (nodes) and E is a set of edges (properties). A single graph is usually represented as a collection of triples $T = < s\ p\ o >$ where s is a *subject*, p is a *predicate*, and o is an *object*. To represent the knowledge graph in a standard way, W3C[‡] recommends RDF, a standard data model to represent knowledge graphs. In RDF, a *subject* is a resource identified by a unique identifier (URI) or a blank-node, an *object* can be both resource, blank-node or literal (e.g. String, number), and *predicate* is a property defined in an ontology and must be a URI.

To develop an ontology of domain knowledge, W3C recommends RDF-S (Resource Description Framework Schema)[§], a standard data model for knowledge representation that extends the basic RDF vocabulary with a set of classes and RDF-S entailment (inference patterns)[¶]. Furthermore, OWL[‖] is also a W3C standard for authoring ontologies. For graph querying and manipulation, W3C recommends SPARQL[**], a query language to retrieve and manipulate data stored in RDF. It offers rich expressivity for complex queries such as aggregation, subqueries, and negation.

Ontology Development

We developed a *ProQ* (Production Quality) ontology as a foundation of our KG-based issue analysis. This development of an OWL ontology was guided by the ontology 101 engineering approach (35), which is a widely used methodology for developing ontologies.

The first step was to conduct an analysis of the existing domain knowledge through workshops and reviews with our industrial partner, including a thorough review of the numerous documents they provided. This helped us identify the key concepts and relationships that needed to be represented in the ontology.

Next, we conducted a survey of several existing FMEA frameworks and related ontologies and selected (15) as a candidate for reuse. This ontology provided a good starting point for our work and helped us ensure that our ontology was compatible with existing standards and frameworks.

ProQ Ontology Overview Figure 4 shows the overview of our developed ProQ ontology. It comprises four main sub-ontologies, i.e., FMEA, PPR, IssueCode, and the integrating FPI ontology.

The FMEA ontology consists of six main classes as follows: (i) the `fmea:FailureMode` class, which represents different types of failure and is linked to (ii) the `fmea:Process` class through the `fmea:hasFailureMode` relationship. It is also linked to (iii) the `fmea:Cause` class via the `causes` relationship, representing the potential root causes that trigger failures. Each cause is connected to (iv) the `fmea:Effect` class, which represents the consequences or impacts resulting from a failure. Furthermore, the `fmea:Cause` class is linked to (v) the `fmea:ControlMethod` class through the `fmea:hasControlMethod` property, which indicates the detection or prevention mechanisms. To resolve or reduce the impact of a cause, (vi) the `fmea:MitigationAction` class is introduced and is connected to the `fmea:Cause` class via the `fmea:hasMitigationAction` property.

To cover the PPR domain, we leveraged the existing VDI 3682 (30) standard and extended the existing PPR ontology (36). It consists of three main concepts as follows: (i) the `ppr:Process` class represents the process (e.g., manufacturing steps) and consists of the `ppr:ProcessOperator` class. (ii) the`ppr:Product` class, a subclass of `ppr:State` tthat is an input connected via the properties `ppr:hasInput` and or output `ppr:hasOutput`). (iii) The `ppr:TechnicalResource` class represents tools or components (e.g., machines) that is assigned to carry out the `ppr:Process`.

---

[*] Resource Description Framework (RDF) `https://www.w3.org/RDF/`

[†] Web Ontology Language (OWL) `https://www.w3.org/OWL/`

[‡] `https://www.w3.org/standards`

[§] `https://www.w3.org/TR/rdf-schema/`

[¶] `https://www.w3.org/TR/rdf11-mt/`

[‖] `https://www.w3.org/TR/owl2-overview/`

[**] `https://www.w3.org/TR/sparql11-overview/`

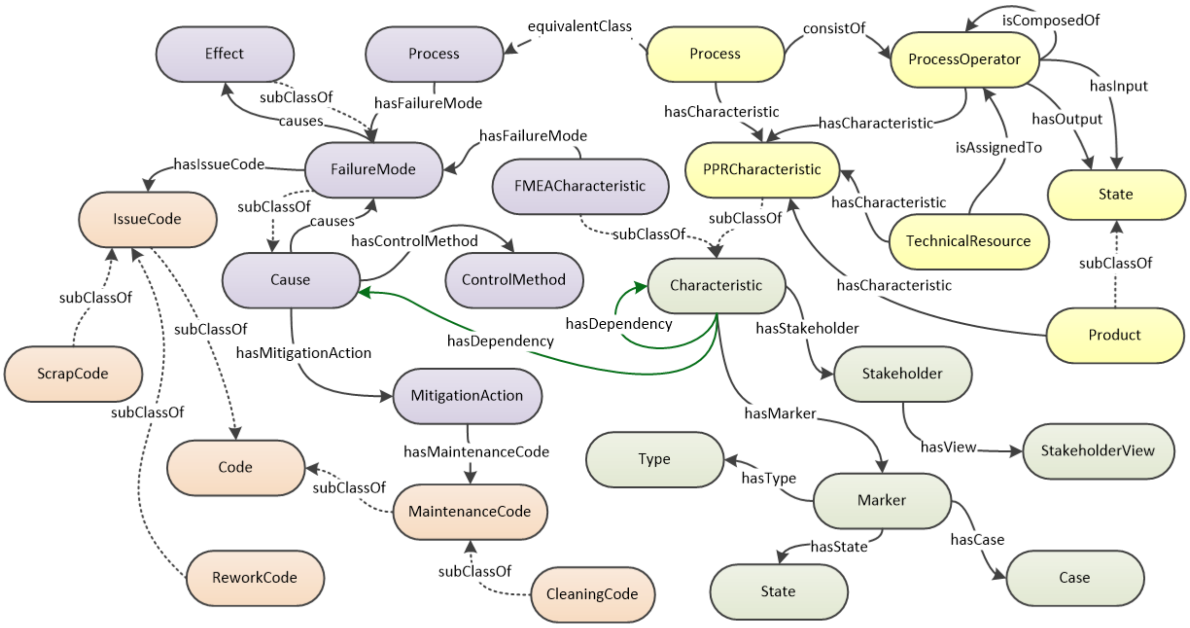

Figure 4. Integrated IssueCode (orange) FMEA-PPR Ontology with FMEA concepts (purple), PPR concepts (yellow), and integrating FPI concepts (green).

We added new concepts and relationships to represent the key aspects of the FMEA-PPR domain such as the `Characteristics`, `Stakeholders`, `IssueCode`, and `Marker` Concept and *Dependency links*.

To accommodate the dependency links introduced in **??** , we developed several object properties (OP) that link between classes in ProQ ontology as follows:

- *Functional dependency (FD)* is realized by object property `fpi:hasFunctionalLink` that serves as relation between `ppr:TechnicalResources` class in PPR ontology. Thus, it has the same *domain* and *range*.
- *Resources to Process Dependency (RDP)* is realized by object property `ppr:isAssignedTo` that represents relation between `ppr:Resource` class (domain) and `ppr:Process` class (range).
- *Product to Process Dependency (PPD)* is realized by object property `ppr:hasInput` and `ppr:hasOutput` that represents relation between `ppr:Product` class (domain) and `ppr:Process` class as (range).
- *Cause-Effect Dependency (CED)* is realized by property `fmea:causes` represents relation between `fmea:Cause` class, `fmea:FailureMode` class, and `fmea:Effect` class in FMEA ontology.
- *Characteristic Dependency (CD)* is realized by property `fpi:hasCharacteristic` that represents connection between and among `Characteristic` class.

Investigation State Marker (ISM) The ISM concept is realized through the `fpi:Marker` class, which acts as a tracking label or status indicator during a quality issue investigation. It is linked to `fpi:Characteristic` class via `fpi:hasMarker` property that allow each characteristic in the FPI network to be annotated with one or more markers. The full documentation of our ProQ ontology object properties can be found in Table 3.

Implementation and Use Case Evaluation

In this section, we describe the implementation of our approach by means of a prototypical framework and demonstrate its application by means of three use case scenarios.

Prototypical Implementation

We developed a multi-step pipeline that generates a representation in RDF format from raw data and populates the resulting statements into a knowledge graphs. Figure 5 shows a diagram of the implemented pipeline.

The pipeline steps are as follows: *Data Acquisition, Data Preprocessing, Ontology Mapping & KG Construction, KG Storage and Analysis via User Interface*. Each step is described in detail below.

Listing 1: an Excerpt of Raw XML File

```
<FM-STRUCTURE-ELEMENT ID=U71A696EAF000F2 T=2023.12.07
09:18:54 F-ID-CLASS=FM-STRUCTURE-ELEMENT
   SYSCOND=+01:00>
   <LONG-NAME>
      <L-4 L=EN>Laser welding</L-4>
   </LONG-NAME>
   <SHORT-NAME SI=AUTONUMBER>1.6</SHORT-NAME>
   <FM-USERDEFINED-ATTRIBUTE-REFS>
      <FM-USERDEFINED-ATTRIBUTE-REF ID-REF=
      U8126712AB000B4 F-ID-CLASS=
      FM-USERDEFINED-ATTRIBUTE>
         U8126712AB000B4_ProcessOperator
      </FM-USERDEFINED-ATTRIBUTE-REF>
   <FM-STRUCTURE-ELEMENT-TYPE-REF ID-REF=U870E7E77800230
   F-ID-CLASS=FM-STRUCTURE-ELEMENT-TYPE>
      U870E7E77800230_Laser_welding
      </FM-STRUCTURE-ELEMENT-TYPE-REF>
   <FM-CLASSIFICATION>
   ...
```

Table 3. ProQ Ontology Object Properties

Relationship	Domain	Range	Relationship Description
`fpi:hasFunctionalLink`	`vdi:TechnicalResource`	`vdi:TechnicalResource`	Relationship that represents *Functional Dependency (FD)* that link resources to other resources in FPI Model.
`fpi:hasIssueCode`	`fmea:FailureMode`	`fpi:IssueCode`	It represents a specific failure mode associated with a particular issue code.
`fpi:hasMaintenanceCode`	`fmea:MitigationAction`	`fpi:MaintenanceCode`	It represents that a mitigation action is associated with a specific maintenance code.
`fpi:hasFMEACharacteristic`	`fmea:FailureMode`	`fpi:FMEACharacteristic`	It signifies that a particular failure mode is associated with a specific FMEA characteristic

`fpi:hasPPRCharacteristic`	`vdi:Process,`
	`vdi:Product,`
	`vdi:TechnicalResource`
		`fpi:PPRCharacteristic`	It signifies that a specific process, product or resource is associated with a particular PPR characteristic

`fpi:hasMarker`	`fpi:Characteristic`	`fpi:Marker`	Represents the association between a characteristic and a marker in the FPI Model.
`fpi:hasStakeholder`	`fpi:Characteristic`	`fpi:Stakeholder`	Represents the association between a characteristic and stakeholder in the FPI Model.
`fmea:causes`	`fmea:Cause,` `fmea:FailureMode`	`fmea:FailureMode,` `fmea:Effect`	Relationship that represent *Cause-Failure-Mode Dependency (CFD)* that link causes to failure modes in FPI Model.
`fmea:hasDependency`	`fpi:Characteristic`	`fpi:Characteristic`	Relationship that represent *PPR to FMEA Dependency (PFD)* and *PPR to PPR Dependency (PrPrD)* that link between PPR as well as FMEA characteristics in FPI Model.
`fmea:hasControlMethod`	`fmea:Cause`	`fmea:ControlMethod`	This relationship indicates that a cause is associated with a control method in the FPI Model.
`fmea:hasFailureMode`	`fnea:Process`	`fmea:FailureMode`	It represents the relationship between a process and the potential failure modes associated with it
`fmea:hasMitigationAction`	`fmea:FailureMode`	`fmea:MitigationAction`	This relationship represents the action taken to mitigate or address a particular failure mode
`vdi:hasInput`	`vdi:Process`	`vdi:Product`	Relationship that represent *Product to Process Dependency (PPD)* that link processes to products in FPI Model.
`vdi:hasOutput`	`vdi:Process`	`vdi:Product`	Relationship that represent *Product to Process Dependency (PPD)* that link processes to products in FPI Model.
`vdi:isAssignedTo`	`vdi:TechnicalResource`	`vdi:Process`	Relationship that represent *Resources to Process Dependency (RPD)* that link resources to processes in FPI Model.
`vdi:consistOf`	`vdi:Process`	`vdi:ProcessOperator`	This relationship indicates that a process consists of one or more process operators in the FPI Model
`vdi:isComposedOf`	`vdi:ProcessOperator`	`vdi:ProcessOperator`	This relationship represents the composition of one process operator from multiple other process operators in the FPI Model.

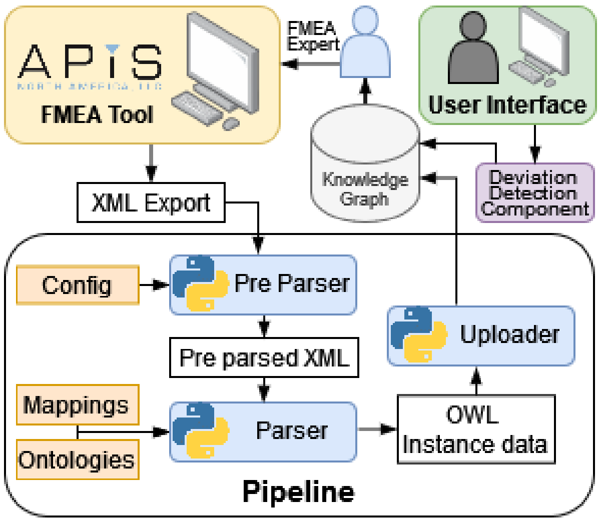

Figure 5. Implementation Pipeline

Data Acquisition The pipeline starts with FMEA data creation using a specialized tool (e.g,, APIS-IQ[††]), which is a widely used software for FMEA. The data (such as failure modes, causes, effects, etc.) is exported in a extensible markup language (XML) format. This XML serves as the primary input for the transformation process. Listing 1 show an excerpt of the mentioned XML file for the laser welding process. For this specific node, a user defined value has been added to help the parser with the ontology mapping. Although a specific tool was utilized in this case, the pipeline has been designed to accommodate XML outputs from any FMEA software.

Data Preprocessing In this step, the raw XML data is cleaned and reshaped before processing to the next step. For instance, the XML file may contains information in a format that's inconsistent (e.g., a single tag that lists multiple values separated by commas). In this case, the pre-parser handles it by splitting those values into separate entries based on rules defined in a configuration file. As a result, each value is placed into its own XML node, making the structure cleaner and ready for accurate parsing.

Listing 2: an Excerpt of the Mapping between XML and Ontology

```
{
 process: {
   xpath: ./FM-STRUCTURE-ELEMENTS/..,
   ontology: VDI3682,
   properties: {
      shortName: ./SHORT-NAME,
      longName: {
      property_label_relative_xpath: ./LONG-NAME/L-4,
              language_tag: L
           }
       },
   child_nodes_relative_paths: [
   ./FM-SE-DECOMPOSITION/FM-STRUCTURE-ELEMENT-REF,
   ./FM-SE-FUNCTIONS/FM-FUNCTION-REF,
   ./FM-SE-CHARACTERISTICS/FM-CHARACTERISTIC-REF
      ]
   }
   ..
}
```

Ontology Mapping & KG Construction In this step, the parser transforms the pre-parsed XML data into a structured RDF representation based on FPI ontology using ontology mappings. The mappings file, written in JSON format, describes which XML tags get mapped to which nodes in the ontology. The implementation of this proprietary mapping was primarily driven by the necessity for functionalities like determining the node class based on values nested within the XML object. The mapping also describes where in the XML object the properties and related nodes for a node can be found, etc. Listing 2 shows an example mapping for an XML raw data shown in Listing 1. A second mapping file describes the relationship between certain node according to the ontology. The parser uses the pre-parsed XML as input and generate RDF output.

Listing 3: an Excerpt of RDF Output

```
@prefix vdi3682: <https://acdp.at/onto/VDI3682#> .
@prefix rdfs: <http://www.w3.org/2000/01/rdf-schema#> .
@prefix FMEA: <https://acdp.at/onto/FMEA#> .

:ProcessOperator_U71A696EAF000F2
   a :ProcessOperator ;
   rdfs:label Laser␣welding@en;
   FMEA:hasFMEACharacteristic
      FMEA:FMEACharacteristic_U71AA7B84700066,
      ..
:FMEACharacteristic_U71AA7B84700066
   a FMEA:FMEACharacteristic;
   rdfs:label Dimensional␣accuracy@en;
   FMEA:hasFailureMode
      FMEA:Failure_mode_U71AA7B902001FD;
      ..
FMEA:Failure_mode_U71AA7B902001FD
   a FMEA:Failure_mode;
   rdfs:label Dimensions␣outside␣the␣specification@en .
   ..
```

Listing 3 shows an example of RDF output (excerpt) in Turtle[‡‡] format. It shows that the extracted data from the XML are now transformed into graph representation in RDF. Each entity or node has a property and it links to another entity.

KG storage Once the RDF data is generated, it is loaded into a graph database using the uploader component, which supports multiple backends, including Neo4j and GraphDB. The uploader ingests both instance data and ontology definitions to populate the constructed knowledge graph.

Analysis via User Interface Using the knowledge graphs as a back-end, we built an application that can be use to analyze data, identify product quality issues and provide recommendations on how to resolve them. The front-end of the application was developed using *Vue.js* while the rest interface to the graph was implemented in *Python* with *FastApi* (see Appendix, Figure 8) for more detail implementation.

Use Case Evaluation

We evaluate our approach by means of three use case applications. For this evaluation, we used data provided by a major supplier of aluminum parts for the automotive industry in Europe. We conducted extensive workshops with the stakeholders to derive the necessary data, knowledge and

---

[††] `https://www.apis-iq.cn/en/software/products/`

[‡‡] RDF Turtle `https://www.w3.org/TR/turtle/`

`https://neo4j.com/`

`https://graphdb.ontotext.com/`

Table 4. Knowledge Graph Statistics

Metric	FMEA	PPR	FPI	Issue Code	Total
#Axioms	77	49	43	37	206
#Class	12	10	8	6	36
#Object Property	12	7	7	0	14
#Data Property	5	2	0	5	12
#Individual	2688	78	0	63	2829

requirements for our analysis, specifically regarding the PPR and FMEA. Table 6 and **??** (see Appendix) shows an excerpt overview of PPR and FMEA data we used for this evaluation.

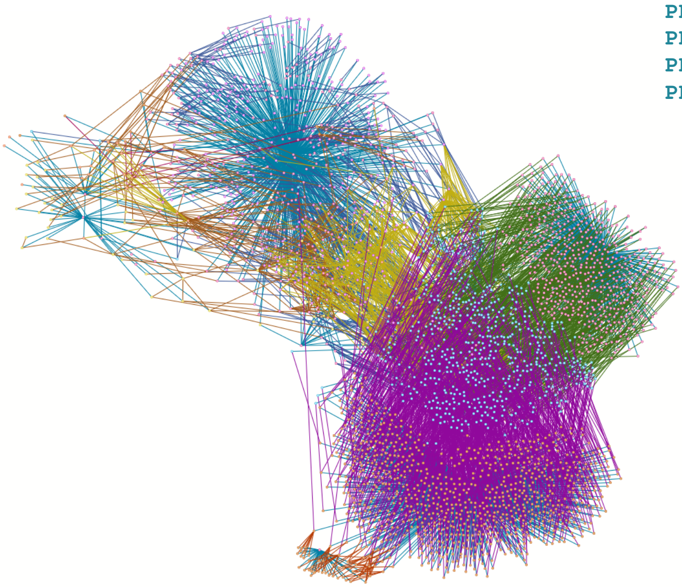

Figure 6. An Overview Visualization of ProQ-KG Knowledge Graph

We used our prototype pipeline discussed in **??** to process the data, construct the knowledge graph and perform analysis. Table 4 shows the overview statistic of the constructed knowledge graph, and Figure 6 shows the visualization overview of constructed KG with each color cluster representing a distinct subdomain. For example, the purple cluster corresponds to FMEA, which dominates with 2688 individuals and 77 axioms while the green cluster represents PPR with 78 individuals.

The use-case evaluation is structured around three key stages, each aligning with procedural steps (Steps 1 to 3) of collaborative quality analysis lifecycle (cf. Figure 3): *(i)* Marker Assignment, *(ii)* Issue Investigation, Failure Cause Analysis and Mitigation, *(iii)* Cause Verification and FMEA-Revalidation

Use-Case 1: Marker assignment In this use case, a quality analyst is responsible for identifying the current state of particular properties or characteristics in FPI network that may be contributing to an ongoing production failure. These characteristics could include measurable aspects like welding temperature, speed or positioning.

Listing 4 shows a SPARQL query that is used to assign an ”investigation marker” to a resource property/characteristic. Depending on whether the characteristic values fall within an acceptable range (`IsInRange: true`) or not (`IsInRange: false`), the quality analyst can attribute the characteristic with the corresponding marker, cf. Table 2.

For example, an ”orangeMarker” is assigned to the existing characteristic (`?c fpi:hasMarker :orangeMarker`) when the properties/characteristics are identified to be outside the threshold (`?c fpi:isInRange false`) and the cause of the deviation is not mapped yet in the FMEA (`FILTER NOT EXISTS ?c fmea:hasFailureMode ?ca`). Note that this identification of characteristic/property threshold can be done either ”automatically” by some detection component or manually by human observation.

```
PREFIX vdi: <https://acdp.at/onto/VDI3682#>
PREFIX fpi: <https://acdp.at/onto/fpi#>
PREFIX fmea: <https://acdp.at/onto/FMEA#>
PREFIX : <https://acdp.at/resources/fpi#>

    #assign orangeMarker to Characteristic ?c
    INSERT { ?c fpi:hasMarker :orangeMarker}
    WHERE {
        ?c a fpi:Characteristic.
        ?r a vdi:TechnicalResource .
        ?ca a fmea:Cause .
        #properties/characteristics that belong
         ↪ to resources
        ?r fpi:hasCharacteristic ?c .
        #properties/characteristics that are
         ↪ out of range
        ?c fpi:isInRange false .
        #the cause of deviation that does not
         ↪ exist in FMEA
        FILTER NOT EXISTS {
            ?c fmea:hasFailureMode ?ca .
        }
    }
```

Listing 4: SPARQL Query: Marker assignment based on type of marker, e.g., Orange Marker

Use-Case 2: Issue Investigation, cause analysis and Mitigation Following the identification of markers, the next crucial step involves the quality analyst investigating the issue and pinpointing potential causes of the failure. This process is facilitated through SPARQL queries, as shown in the example in Listing 5.

The construction of the cause-failure begins by establishing a connection between the `IssueCode` and `FailureMode` through the `fmea:hasIssueCode` object property as well as the `FailureMode` and `Causes` through `fmea:causes`. If necessary, links between characteristics and causes via `fmea:hasFailureMode`. The query is then filtered based on a specific issue code, such as ”P0512” in this case.

Figure 7 visually represents the resulting graph of query in Listing 5. It shows the interdependence characteristics and causes of the failure code ”Weld seam incomplete/missing”.

To identify the most critical deviated properties/characteristics that need to be addressed, the quality analyst proceeds to determine resolution action for the identified failure causes. This information will help resource engineers (such as mechanical and automation engineers) to quickly respond and solve the cause of the problem.

```
PREFIX rdfs:
 ↪ <http://www.w3.org/2000/01/rdf-schema#>
PREFIX fmea: <https://acdp.at/onto/FMEA#>
PREFIX fpi: <https://acdp.at/onto/fpi#>

    CONSTRUCT {
            ?c fmea:hasFailureMode ?ca .
            ?ca fmea:cause ?f .
    }
    WHERE {
           ?ca fmea:causes ?f .
            ?f fpi:hasIssueCode ?i .
            ?c fmea:hasFailureMode ?ca.
            ?i fpi:shortName ?il .
            FILTER (str(?il) = "P0512")
    }
```

Listing 5: SPARQL Query for characteristics and resource of a particular Issue Code

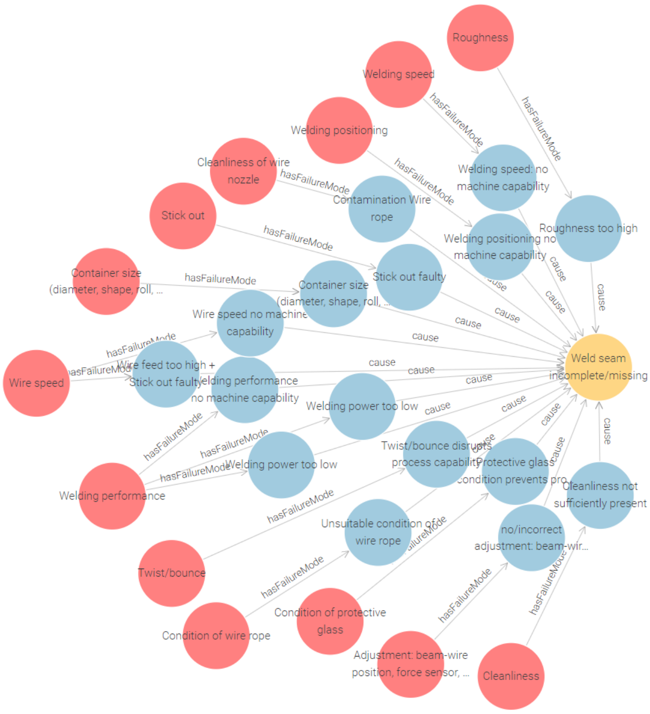

Figure 7. Resulting Graph visualization for listing 5: Characteristics (red node), causes (blue node) and failure mode (yellow node) dependency.

Listing 6 showcases a SPARQL query that matches the `Cause` with `MitigationAction` through `fmea:hasMitigationAction`, specifically focusing on the intended failure mode such as ”Weld seam incomplete/missing” in this case. Additionally, the query allows to prioritize the mitigation based on the investigation marker status (i.e., only `RedMarker` or `VioletMarker` need to be prioritized).

Table 5 shows the complete list of causes and the corresponding mitigation action for those causes where characteristics have either `RedMarker` or `VioletMarker`. Armed with this information, the *Quality Manager* can collaborate with *Resource Engineer* to promptly address the root cause of the problem.

```
PREFIX rdfs:
 ↪ <http://www.w3.org/2000/01/rdf-schema#>
PREFIX fpi: <https://acdp.at/onto/fpi#>
PREFIX fmea: <https://acdp.at/onto/FMEA#>
PREFIX : <https://acdp.at/resources/fpi#>

SELECT DISTINCT ?marker ?cause ?mitigation
 WHERE {
   ?ch fmea:hasFailureMode ?ca.
   ?ca fmea:causes ?f .
   ?ca fmea:hasMitigationAction ?m.
   ?ch fpi:hasMarker ?ma.
   ?ca rdfs:label ?cause .
   ?m rdfs:label ?mitigation .
   ?ma rdfs:label ?marker .
   ?f rdfs:label ?fl .
   FILTER
    ↪  (str(?fl)="Weld seam incomplete/missing")
   FILTER (?marker = :redMarker || ?marker =
    ↪  :violetMarker) .
}
```

Listing 6: Mitigation action for failure cause with Red and Violet Marker

Use-Case 3: FPI Update and FMEA-Revalidation

Throughout the investigation of quality issues, new causes related to failure modes may emerge and require the discovery of new mitigation strategies. In light of these revelations, updating the existing knowledge graph with the latest insights and seamlessly integrating them becomes imperative.

Listing 7 shows a SPARQL query for updating the knowledge graph with new statements. For example, following prior investigation, the Quality Manager might discover a new cause, such as *”Guiding Tube deformed”* to an existing failure mode like *”Weld seam incomplete/missing”*. This updating process is achieved through an `INSERT` syntax followed by new statements for the novel causes (e.g., `:Cause_U81169XX :Cause_U8169XX a fmea:Cause.` and connect them with the corresponding characteristics and failure mode.

```
PREFIX rdfs:
 ↪ <http://www.w3.org/2000/01/rdf-schema#>
PREFIX fmea: <https://acdp.at/onto/FMEA#>
PREFIX : <https://acdp.at/resources/FMEA#>

INSERT  DATA
   { #create new cause
     :Cause_U8169XX a fmea:Cause .
     :Cause_U8169XX rdfs:label
        "Guiding Tube deformed"@en .
     #add characteristics to the new cause.
     :Char_U8169XX fmea:hasFailureMode
        :Cause_U8169XX .
     #add cause to failure mode
     #Weld␣seam␣incomplete/missing".
     :Cause_U8169XX fmea:causes
        :Failure_mode_U71AA7BAC700162 .
 }
```

Listing 7: New cause addition to failure mode

Table 5. Query Result Mitigation of failure cause with Red and Violet marker

Marker	Failure Cause	Mitigation Action
”Red Marker”	”Contamination Wire rope”	”Supplement maintenance plan”
”Red Marker”	”Welding speed: no machine capability”	”Define specifications for the machine/component/unit with regard to functionality and requirements”
”Red Marker”	”Welding speed: no machine capability”	”Provide proof of machine capability”
”Red Marker”	”Welding speed: no machine capability”	”Take over technical equipment in preventive maintenance
”Violet Marker”	”Protective glass condition prevents process capability”	”Implement technical solution”
”Violet Marker”	”Protective glass condition prevents process capability”	”Ensure qualification”
”Violet Marker”	”Protective glass condition prevents process capability”	”Supplement maintenance plan”
”Violet Marker”	”Protective glass condition prevents process capability”	”Carry out maintenance”

Discussion

In this section, we reflect on the results of the paper and suggest potential avenues for future work.

Summary of Findings

We summary our contributions with respect to our identified requirements (cf. **??**) as follows:

Knowledge Integration and Querying Our paper proposed knowledge integration and querying in the context of CPPS. By introducing an RDF/OWL based ontology for PPR and FMEA, we provide a standardized graph representation that harmonizes heterogeneous data sources and makes implicit knowledge explicit (addressing R1). It facilitates flexible graph representations which allow semantic integration of diverse engineering domains (e.g., linking between different properties and characteristics in PPR and FMEA) (addressing R2). Furthermore, the standard query language provides different stakeholders a unified language for issue marking, root cause analysis, and issue resolution in an efficient manner (fulfilling R6).

Collaborative Quality Issue Analysis A key challenge in quality issue analysis in CPPSs is the collaboration among stakeholders with diverse perspectives and knowledge backgrounds. Our paper addresses this challenge by providing a platform for collaborative quality issue analysis. By representing different views and knowledge of stakeholders within a shared knowledge graph, we facilitate a common understanding of the production system and thereby make it easier to coordinate system states and asset changes during quality issue analysis (addressing R3).

Knowledge Evolution, Sharing and Reusabiity Our approach creates a feedback mechanism that facilitates knowledge update and revalidation of FMEA knowledge by means of a knowledge graph. For example, when new failure causes and their assets dependencies are identified, a quality manager can identify the change and subsequently update the existing knowledge graph (addressing R4). Moreover, by representing engineering knowledge in a standard representation, the constructed knowledge (e.g., FMEA) can be shared and reused for similar analysis purposes (covering R5).

Limitation & Future Work

Although our approach shows strong potential for collaborative quality issue analysis, we identify several limitations and corresponding directions for future work:

Ontology Development and Evaluation While the ProQ ontology was developed through extensive and iterative workshops with respective domain experts, a formal evaluation using standardized methodologies such as logical consistency checks has not yet been performed. As future work, we aim to leverage Description Logic (DL) reasoners to assess inference completeness, and SHACL constraint validation to ensure the consistency of the ontology.

Reasoning-Based Analysis At present, the system focuses on semantic integration and navigation through SPARQL-based queries. While this supports structured analysis, the framework has yet to exploit more advanced reasoning capabilities. Future work will integrate rule-based reasoning for automated fault propagation, and constraint validation using SHACL to capture domain-specific quality thresholds and compliance rules.

Use Case Evaluation Our use case evaluation, conducted with a major automotive supplier, demonstrates the real-world feasibility of the ProQ-KG framework in enabling traceability, multi-view coordination, and structured root cause analysis in complex production environment i.e., Laser Welding Seam. Future work will incorporate extensive evaluation in terms of quantitative metrics including query execution time, accuracy of root cause identification and resolution effectiveness, and user interaction efficiency across different stakeholder role.

Finally, other potential future works that could further enhance the applicability of our approach including: *(i)* Integration with real-time machine data (shop floor) for monitoring purposes that enable proactive quality issue analysis and mitigation. *(ii)* Automated detection based on predefined thresholds. It could be used to improve analysis process and reduce reliance on manual (human) intervention.

Conclusions

In this paper, we proposed an integrated ontology derived from heterogeneous domain knowledge that integrates failure modes and effects with products, processes, and resources. The constructed knowledge graph makes it easier for stakeholders from different domains to share their models but also to navigate through different knowledge – i.e., to identify causal dependencies of quality issues.

W3C standard `https://www.w3.org/standards/`
`https://www.w3.org/TR/shacl/`

Acknowledgements

The financial support by the Christian Doppler Research Association, the Austrian Federal Ministry for Digital and Economic Affairs and the National Foundation for Research, Technology and Development is gratefully acknowledged. This work has been partially supported and funded by the Austrian Research Promotion Agency (FFG) via the Austrian Competence Center for Digital Production (CDP) under the contract number 881843. This work has also received funding from the Teaming.AI project in the European Union's Horizon 2020 research and innovation program under grant agreement No 95740.

References

[1] Monostori L. Cyber-physical production systems: Roots, expectations and R&D challenges. Procedia Cirp. 2014;17:9-13.

[2] Russomanno DJ, Bonnell RD, Bowles JB. Functional reasoning in a failure modes and effects analysis (FMEA) expert system. In: Annual Reliability and Maintainability Symposium 1993 Proceedings. Atlanta, GA, USA: IEEE; 1993. p. 339-47.

[3] Stamatis DH. Failure mode and effect analysis: FMEA from theory to execution. Quality Press; 2003.

[4] Wu Z, Liu W, Nie W. Literature review and prospect of the development and application of FMEA in manufacturing industry. The International Journal of Advanced Manufacturing Technology. 2021 Jan;112(5-6):1409-36.

[5] DIN EN 60812:2015-08 - Failure Mode and Effects analysis (FMEA) (IEC 56/1579/CD:2014) ; 2015.

[6] Kropatschek S, Gert O, Ayatollahi I, Meixner K, Kiesling E, Steigberger A, et al. Designing a Digital Shadow for Efficient, Low-Delay Analysis of Production Quality Risk. In: 2022 IEEE 27th International Conference on Emerging Technologies and Factory Automation (ETFA). Stuttgart, Germany: IEEE; 2022. p. 1-8. Available from: `https://ieeexplore.ieee.org/document/9921582/`.

[7] dos Santos França JB, Netto JM, Santo Carvalho JdE, Santoro FM, Baião FA, Pimentel MG. KIPO: the knowledge-intensive process ontology. Softw Syst Model. 2015;14(3):1127-57.

[8] Kropatschek S, Steuer T, Kiesling E, Meixner K, Ayatollahi I, Sommer P, et al. Analysis of Quality Issues in Production With Multi-view Coordination Assets. In: 2022 IFAC MIM; 2022. .

[9] Rinker F, Meixner K, Kropatschek S, Kiesling E, Biffl S. Risk and Engineering Knowledge Integration in Cyber-physical Production Systems Engineering. In: 2022 48th Euromicro Conference on Software Engineering and Advanced Applications (SEAA); 2022. p. 338-45.

[10] Buchgeher G, Gabauer D, Martinez-Gil J, Ehrlinger L. Knowledge graphs in manufacturing and production: A systematic literature review. IEEE Access. 2021;9:55537-54.

[11] Dittmann L, Rademacher T, Zelewski S. Performing FMEA using ontologies. In: 18th International Workshop on Qualitative Reasoning. Evanston, Illinois, USA; 2004. p. 209-16.

[12] Koji Y, Kitamura Y, Mizoguchi R, et al. Ontology-based transformation from an extended functional model to FMEA. In: 15th International Conference on Engineering Design. Melbourne, Australia; 2005. p. 323-4.

[13] Ebrahimipour V, Rezaie K, Shokravi S. An ontology approach to support FMEA studies. Expert Systems with Applications. 2010;37(1):671-7.

[14] Xiuxu Z, Yuming Z. Application research of ontology-enabled process FMEA knowledge management method. Int J Intelligent Systems and Applications. 2012:34-40.

[15] Rehman Z, Kifor C. An Ontology to Support Semantic Management of FMEA Knowledge. International Journal of Computers Communications & Control. 2016;11(4):507-21. Available from: `http://univagora.ro/jour/index.php/ijccc/article/view/1674`.

[16] Winkler D, Novák P, Meixner K, Vyskočil J, Rinker F, Biffl S. Product-Process-Resource Asset Networks as Foundation for Improving CPPS Engineering. In: 2021 26th IEEE International Conference on Emerging Technologies and Factory Automation (ETFA); 2021. p. 1-4.

[17] Biffl S, Lüder A, Meixner K, Rinker F, Eckhart M, Winkler D. Multi-view-Model Risk Assessment in Cyber-Physical Production Systems Engineering. In: Proc. of the 9th International Conference on Model-Driven Engineering and Software Development (MODELSWARD); 2021. p. 163-70.

[18] Hoffmann D, Nowacki N, Biffl S, Kiesling E, Meixner K, Lüder A. Interdisciplinary production risk exploration: A grounded approach to integrate data-and knowledge-driven analytics. Procedia CIRP. 2023;120:1016-21.

[19] Biffl S, Kropatschek S, Meixner K, Hoffmann D, Lüder A. Configuring and Validating Multi-aspect Risk Knowledge for Industry 4.0 Information Systems. In: Guizzardi G, Santoro F, Mouratidis H, Soffer P, editors. Advanced Information Systems Engineering. Cham: Springer Nature Switzerland; 2024. p. 492-508.

[20] Leitão P, Rodrigues N, Turrin C, Pagani A, Petrali P. GRACE ontology integrating process and quality control. In: IECON: 38th Annual Conf. of Industrial Electronics Society. Montreal, Canada: IEEE; 2012. p. 4348-53.

[21] Polenghi A, Roda I, Macchi M, Pozzetti A. Multi-attribute Ontology-based Criticality Analysis of manufacturing assets for maintenance strategies planning. IFAC-PapersOnLine. 2021;54(1):55-60. 17th IFAC Symposium on Information Control Problems in Manufacturing INCOM 2021. Available from: `https://www.sciencedirect.com/science/article/pii/S2405896321009630`.

[22] Hodkiewicz M, Klüwer JW, Woods C, Smoker T, Low E. An ontology for reasoning over engineering textual data stored in FMEA spreadsheet tables. Computers in Industry. 2021;131:103496.

[23] Martinez-Gil J, Buchgeher G, Gabauer D, Freudenthaler B, Filipiak D, Fensel A. Root Cause Analysis in the Industrial Domain using Knowledge Graphs: A Case Study on Power Transformers. Procedia Computer Science. 2022;200:944-53. 3rd International Conference on Industry 4.0 and Smart Manufacturing. Available from: `https://www.sciencedirect.com/science/article/pii/S1877050922003015`.

[24] Bachhofner S, Kurniawan K, Kiesling E, Revoredo K, Bayomie D. Knowledge Graph Supported Machine Parameterization for the Injection Moulding Industry. In: Villazón-Terrazas B, Ortiz-Rodriguez F, Tiwari S Sanjuand Tiwari, Sicilia MA, Martín-Moncunill D, editors. Knowledge Graphs and Semantic Web. Springer Int. Publ.;

2022. p. 106-20.

[25] Razouk H, Kern R. Improving the consistency of the failure mode effect analysis (FMEA) documents in semiconductor manufacturing. Applied Sciences. 2022;12(4):1840.

[26] Okazaki S, Shirafuji S, Yasui T, Ota J. A Framework to Support Failure Cause Identification in Manufacturing Systems through Generalization of Past FMEAs. In: 2023 IEEE/ASME International Conference on Advanced Intelligent Mechatronics (AIM). IEEE; 2023. p. 858-65.

[27] Salah B, Alnahhal M, Ali M. Risk prioritization using a modified FMEA analysis in industry 4.0. Journal of Engineering Research. 2023;11(4):460-8.

[28] Bousdekis A, Lepenioti K, Apostolou D, Mentzas G. Data analytics in quality 4.0: literature review and future research directions. International Journal of Computer Integrated Manufacturing. 2023;36(5):678-701. Available from: `https://doi.org/10.1080/0951192X.2022.2128219`.

[29] Schleipen M, Drath R. Three-view-concept for modeling process or manufacturing plants with AutomationML. In: 2009 IEEE Conference on Emerging Technologies & Factory Automation; 2009. p. 1-4.

[30] VDI 3682. VDI Guideline 3682: Formalised Process descriptions. VDI; 2005.

[31] Buchgeher G, Gabauer D, Martinez-Gil J, Ehrlinger L. Knowledge Graphs in Manufacturing and Production: A Systematic Literature Review. IEEE Access. 2021;9:55537-54.

[32] Martinez-Gil J, Hoch T, Pichler M, Heinzl B, Moser B, Kurniawan K, et al. In: Soldatos J, editor. Examining the Adoption of Knowledge Graphs in the Manufacturing Industry: A Comprehensive Review. Cham: Springer Nature Switzerland; 2024. p. 55-70. Available from: `https://doi.org/10.1007/978-3-031-46452-2_4`.

[33] Wan Y, Liu Y, Chen Z, Chen C, Li X, Hu F, et al. Making knowledge graphs work for smart manufacturing: Research topics, applications and prospects. Journal of manufacturing systems. 2024;76:103-32.

[34] Biffl S, Musil J, Musil A, Meixner K, Lüder A, Rinker F, et al. An Industry 4.0 Asset-Based Coordination Artifact for Production Systems Engineering. In: IEEE 23rd Conference on Business Informatics (CBI); 2021. p. 92-101.

[35] Noy NF, McGuinness DL, et al.. Ontology development 101: A guide to creating your first ontology. Stanford knowledge systems laboratory tech. rep. KSL-01-05; 2001.

[36] Hildebrandt C, Köcher A, Küstner C, López-Enríquez CM, Müller AW, Caesar B, et al. Ontology Building for Cyber–Physical Systems: Application in the Manufacturing Domain. IEEE Transactions on Automation Science and Engineering. 2020.

Appendix

Table 6. Process Operator, Characteristics and Assigned Technical Resources

Process Operator	Process Characteristics	Assigned Technical Resources
Laser welding	Tightness (liquids), Technical cleanliness, Burn-through, Freedom of accusation	Welding device 11, Welding device 15
MIG welding	Dimensional accuracy, A-Mass, Presence of weld seam, Branding, Non-porous, Crack-free, Freedom of inclusion, Splash-free, Technical cleanliness, Decrater	Welding device, Welding robot (with welding source), Wire conveyor system, Machine control, Protective cabin
Pick up/drop off	Dimensional accuracy, Plant availability	Internal transport (machine slat conveyor), Gripper, Robot 12, Robot 17
Tensioning	Freedom of accusation, Plant availability	Welding device 11, Welding device 15
Welding Laser UZB	Dimensional accuracy, Plant availability, Surface finish	Welding robot (with welding head), Wire conveyor system, Machine control, Protective cabin
Record	Plant availability	Internal transport (machine)
File	Plant availability	Internal transport (machine)
Hang up	Surface finish, Plant availability	Worker
Welding	Dimensional accuracy, Plant availability	Welding device, Welding robot (with welding source), Wire conveyor system, Machine control, Protective cabin

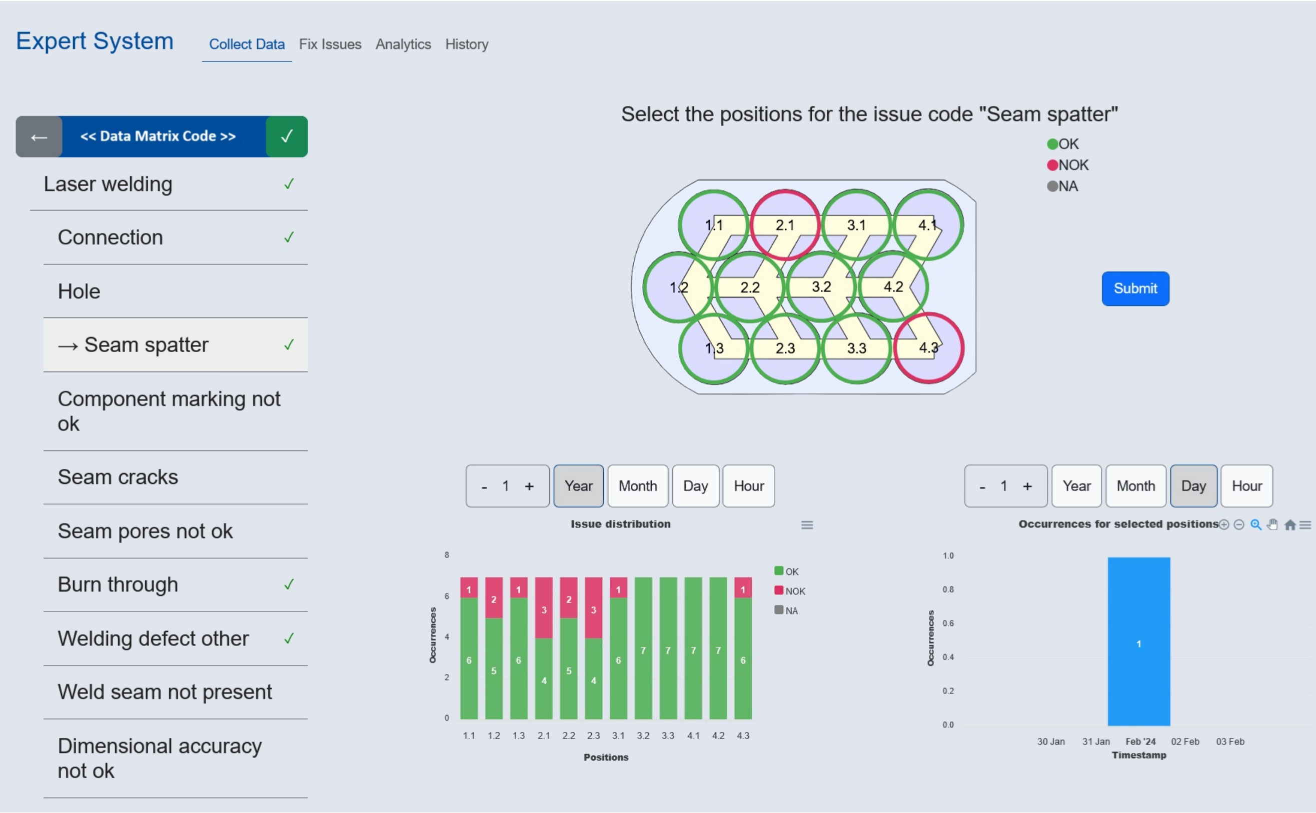

Figure 8. Expert Guidance application for collection of product quality data and guidance for solving quality issues. In this figure, the operator could locate the potential failure mode of a product with an issue code. To the left, possible issue codes are listed, on the image in the middle the user can select where on the aluminum block there are quality issues, and the two graphs show the number of issues occurring for each position as well as time points for when the issue was collected.