Risk and Engineering Knowledge Integration in Cyber-physical Production Systems Engineering

Felix Rinker[1,2], Kristof Meixner[1,2], Sebastian Kropatschek[3], Elmar Kiesling[4], Stefan Biffl[2,3]

[1]*CDL-SQI*, [2]*Institute of Information Systems Engineering, TU Wien*, and [3]*CDP*, Vienna, Austria

Email: {first.last}@tuwien.ac.at, {first.last}@acdp.at

[4]*Institute of Data, Process, and Knowledge Engineering, WU Wien*, Vienna, Austria

Email: elmar.kiesling@wu.ac.at

***Abstract*—In agile Cyber-physical Production System (CPPS) engineering, multi-disciplinary teams work concurrently and iteratively on various CPPS engineering artifacts, based on engineering models and Product-Process-Resource (PPR) knowledge, to design and build a production system. However, in such settings it is difficult to keep track of (i) the effects of changes across engineering disciplines, and (ii) their implications on risks to engineering quality, represented in Failure Mode and Effects Analysis (FMEA). To tackle these challenges and systematically co-evolve FMEA and PPR models, requires propagating and validating changes across engineering and FMEA artifacts. To this end, we design and evaluate a *Multi-view FMEA+PPR (MvFMEA+PPR) meta-model* to represent relationships between FMEA elements and CPPS engineering assets and trace their change states and dependencies in the design and validation lifecycle. We evaluate the MvFMEA+PPR meta-model in a feasibility study on the quality of a screwing process from automotive production. The study results indicate the MvFMEA+PPR meta-model to be more effective than alternative traditional approaches.**

***Index Terms*—agile cyber-physical production system engineering, failure mode and effects analysis, product process resource knowledge, multi-view modeling, multi-disciplinary engineering**

I. INTRODUCTION

Cyber-physical Production Systems (CPPSs), like automated car factories, aim at the flexible production of customizable products interacting with their physical environment to adapt to uncertain conditions [1], [2]. Their engineering has become agile in the sense that engineers from several disciplines work iteratively and in parallel to develop Product-Process-Resource (PPR) assets, such as product designs, production process models, and production resource plans, using PPR modeling approaches [3], [4]. In agile engineering processes, engineers are likely to update their PPR designs often an engineering project [5]. Complex business use cases and advanced software support as envisioned in Industry 4.0 (I4.0), require such agile processes and hence, integrated engineering knowledge and processes [6].

Integrated knowledge and processes are essential to detect and address quality implications and risk factors in CPPS engineering. In this context, *Failure Mode and Effects Analysis (FMEA)* is an established method in early project phases for evaluating the effects of potential failures of system components, assessing risk factors, and detecting and isolating faults [7]. However, in its current document-driven or paper-based form, FMEA interferes with one of its key purposes. That is, to reduce design effort by identifying potential design flaws and mitigate risks in the early stages of system design. Furthermore, in late project phases, i.e., after requirements analysis, design, and development, the re-validation of FMEA results is often limited; late changes in the design documentation are often not considered. Therefore, maintaining an FMEA often requires error-prone and laborious reconstruction of knowledge from design documentation [8]. Consequently, it is desirable to link FMEA causes and effects to the PPR elements to make instance dependencies explicit and to facilitate co-evolution and change re-validation.

In the following, we present challenges that hinder the efficient co-evolution and re-validation of FMEA with PPR assets in multi-disciplinary CPPS settings. Fig. 1 shows an FMEA model with a failure mode and a cause on the left-hand side and a PPR model on the right-hand side. The PPR model represents a screwing process with three input products and one output product and three production resources that execute the process. Concepts in the models can have properties, such as *torque*, to represent production or quality criteria.

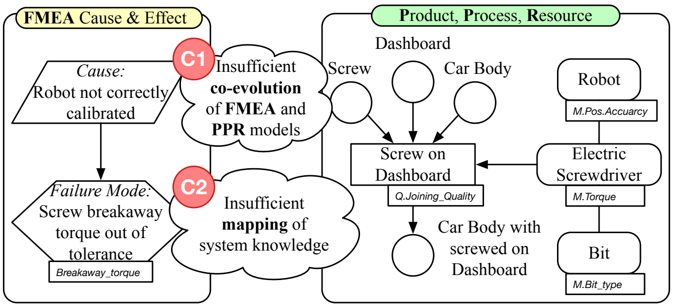

Fig. 1. Challenges of FMEA and PPR knowledge co-evolution.

Challenge 1. *Insufficient co-evolution of FMEA and PPR models.* Typically FMEA and PPR models are developed separately or scattered across heterogeneous artifacts [9]. This opens semantic gaps between concepts used by stakeholders, who conduct FMEA, and other domain experts in Production Systems Engineering (PSE) projects [10]. Bridging these gaps requires domain knowledge, which is often not documented explicitly. This makes the process of comparing and aligning FMEA models and PSE artifacts inefficient and prone to error.

Challenge 2. *Inefficient mapping of system knowledge.* FMEAs shall reuse concepts from PPR models to use shared knowledge. However, it remains unclear how to re-validate FMEA elements efficiently after PSE artifact changes [11]. To this end, it is necessary to automate capabilities such as configuring *(i)* a multi-view PPR model that integrates and preserves the PSE stakeholder perspectives; *(ii)* change dependencies in and between PPR assets and FMEA model elements; *(iii)* a data integration process that can propagate updates from local stakeholder views in artifacts to PPR asset property values; and *(iv)* analyzing which FMEA model elements require re-validation due to PPR asset changes.

The remainder of this paper is structured as follows. Section II summarizes related work on Knowledge Management in CPPS engineering and FMEA. Section III explains the research approach. Section IV introduces an illustrative use case for evaluation, and requirements for efficient FMEA re-validation in CPPS engineering. Section V outlines the Multi-view FMEA+PPR (MvFMEA+PPR) meta-model for the representation of knowledge and the steps for efficient FMEA re-validation after changes to PSE artifacts. Section VI reports on a feasibility study with a MvFMEA+PPR model instance based on a real-world industry use case and discusses the research results and limitations. Section VII concludes and delineates future work.

II. Related Work

This section summarizes related work on Knowledge Management in CPPS engineering and the Failure Mode and Effects Analysis (FMEA).

A. Knowledge Management in CPPS Engineering

To design a CPPS, engineers collaborate in a multi-disciplinary process [5]. In this work process, several engineering views, concepts, and artifacts need to be transformed and integrated in a holistic view. In PSE, project information is encapsulated in discipline- and tool-specific artifacts [12].

Discipline-specific artifacts and processes make the seamless and traceable information exchange across disciplines and stakeholders hard [6]. Yet, information exchange is required for the Industry 4.0 transformation in PSE. Wortmann *et al.* [13] deem domain-specific languages and model-driven engineering essential to support complex data-driven use cases in the Industry 4.0 context. The Reference Architecture Model for Industry 4.0 (RAMI40) uses well-known standards and technologies, like AutomationML (AML), Systems Modeling Language (SysML), Data Exchange in the Process Industry (DEXPI) and OPC Unified Architecture (OPCUA) to describe Industry 4.0 components utilizing the concept of an Asset Administration Shell [14], [15].

In this work, we take up the challenge of scattered knowledge and aim to integrate quality and engineering data.

Multi-view Modeling for CPPS engineering. Multi-view modeling [16] aims at preserving description-specific concepts and views to support collaboration and knowledge integration in Multi-Disciplinary Engineering Environments (MDEEs). Achieving multi-view models in PSE projects, requires to identify relevant information across domain-specific concepts and models that are defined as boundary objects [17]. We use Common Concepts (CCs) as boundary objects to define assets that unify all properties that stakeholders share on a particular concept [18]. Such common views on assets [18] seem promising to describe Industry 4.0 components [15]. Schleipen *et al.* [19] introduced PPR modeling, representing requirements and an integrated model in PSE. PPR modeling is based on the three main aspects of a production system: (1) *products* with their properties, (2) *processes* that produce products, and (3) *resources* that execute production processes. Meixner *et al.* [20] introduced the PPR-Domain Specific Language (PPR-DSL), a machine-readable and technology-agnostic language for PSE modeling.

We base our approach on the PPR approach to represent the engineering data and describe stakeholder views on assets and properties.

Adequate multi-view process and framework support is a major concern to support interdisciplinary PSE [6], [16]. Tunjic *et al.* [21] introduce a Single Underlying Model (SUM), a common unified model, to enable multi-view modeling environments. To populate a SUM, previously defined mappings between the common and single views, are used [22]. Rinker *et al.* [23] propose a Multi-view Model Transformation (MvMT) architecture, using AML to enable and automate an multi-disciplinary and view-specific data integration pipeline.

B. Failure Mode and Effects Analysis (FMEA)

Quality assurance is crucial in the engineering of technical systems [24]. It involves many disciplines and related engineering roles, but is mainly centered around quality engineering that uses appropriate quality models [25]. The FMEA is an engineering and quality assurance method to identify and mitigate risks and potential production failures before a customer can be effected by poor product performance [7], [26]. A typical FMEA identifies known and potential failure modes along with their corresponding causes and effects, prioritizes them, and defines corrective actions. Several FMEA types have been reported [26]. The *process FMEA* focuses on failure modes occurring during the manufacturing and/or the assembly process. The *design* and *concepts FMEA* addresses product-level or concept-level failure modes [27]. Other approaches aim at enhancing the FMEA method to identify waste modes or to monitor service quality [28], [29].

In this work, we build on the FMEA as a model that represents quality assurance data for production processes.

In multi-disciplinary engineering processes, FMEA typically starts with assembling an FMEA team of experts with relevant domain knowledge [7]. This team analyses the system's architecture, functions, and characteristics. Experts identify and assess in meetings (i) potential failure modes of the analyzed objects, (ii) the respective impact and consequences, and (iii) potential mitigation actions. The evaluation is based on the criteria severity, occurrence, and detection [7], represented by the Risk Priority Number (RPN). All steps of the analysis are documented in a comprehensive FMEA report, including a priority list of failure modes and corrective actions.

Although there are numerous tools to support the FMEA, the monitoring of artifact updates remains challenging [30]. In this paper, we explore the feasibility of representing FMEA model elements alongside with PSE model elements in a PPR network for facilitating efficient analysis and updates.

III. Research Methodology

In this paper, we address the challenges of *inefficient co-evolution and integrated knowledge* by exploring how multi-view-based coordination capabilities [31] can facilitate the efficient integration of multi-view PPR and FMEA models. In particular, we investigate the research question *"How and under what conditions do changes to properties of engineering artifacts necessitate a re-validation of FMEA elements?"* As an illustrative use case, we showcase a *screwing process* (cf. Fig. 1) with the change of a *torque value*. We focus on the engineering views (quality, mechanics, engineering, automation) and on the effect for the FMEA re-validation w.r.t. *joining quality* (cf. Section IV).

We use Design Science, extending our previous work [32], [33] by *(i)* conducting a domain analysis to identify requirements, *(ii)* designing and evaluating the *MvFMEA+PPR meta-model* to represent relationships between FMEA elements and PPR assets and *(iii)* providing a method to trace their change states and dependencies in the design and validation lifecycle. We evaluate the MvFMEA+PPR model and method in a feasibility study on the quality of a joining process.

IV. Use Case and Requirements

This section introduces the use case *FMEA Re-Validation after Changes to Engineering Artifacts* to elicit requirements for improving the efficiency of FMEA and PPR co-evolution. We report on PSE and FMEA re-validation processes abstracted from real-world use cases from system integrators of high-performance automation for car part manufacturing in Germany and Austria [33].

Engineering process. In traditional PSE projects, engineers follow a sequential engineering process in several engineering phases, including quality engineering for system design validation and risk management with FMEA [25], [34], [35]. Due to change requests, engineers often need to work on PSE artifacts that belong to several phases (e.g., artifacts that evolve over different phases) which require flexible and agile solutions. An early-stage FMEA can be conducted based on an initial PPR model that results from basic planning. However, the FMEA model has to be refined and updated as new FMEA-relevant knowledge emerges in engineering activities along the course of the PSE project. These activities are driven by change requests, which may be triggered by engineering needs or by FMEA results and consequently require (i) changes to (validated) engineering results and (ii) the re-validation of FMEA elements that depend on changes.

Traditional FMEA model representation. FMEA knowledge can be represented in plain text, spreadsheet tables, graph modeling tools and dedicated FMEA tools. Established FMEA tools, such as APIS[1], focus on the textual description (in natural language) of FMEA concepts. Therefore, it is difficult to provide tool support for the efficient identification of FMEA elements that require re-validation.

Early FMEA can start after the initial definition of the production system, as soon as the main resources are specified. In this case, findings from FMEA can inform detail engineering to mitigate important risks early and efficiently. If the FMEA approach is applied in this context, often frequent updates are not required/executed. Consequently, findings in late project phases can lead to expensive late design changes.

Design and validation lifecycle. In the PSE process, assets and their properties have to be designed and validated. For the coordination of design and validation activities, these elements are usually assigned to *design* and *validation states* [31], e.g., "to design", "in design", "designed"; "to validate", "in validation", "validated ok", "validated with issue". Based on these states, stakeholders can describe processes/rules for the re-validation (and rework) of assets after changes. Traditional approaches often lack in explicitly defining and using design and validation states [31]. To address these shortcomings, we explore the MvFMEA+PPR approach to efficiently link FMEA concepts to PSE artifacts via an integrated meta-model.

Use Case FMEA Re-Validation after Changes to Engineering Artifacts. The aim of a screwing process is joining two or more components or materials with screws, e.g., car body and dashboard (cf. Fig. 1). A key characteristic focuses on the quality of the joining process and the joint. An common fault is an incorrect or insufficient screwing process, potentially caused by an incorrect torque caused by abrasion and friction. Friction, in turn depends on the precision (cf. Fig. 1, property *M.Pos.Accuarcy* of the resource *Robot*).

However, if the setting process does not join the rivet element properly, the friction may be insufficient to install the screw, and the desired breakaway torque might not be achieved. A setting process is only reliable, if the force *M.Torque* and the position are controlled and monitored. Furthermore, the setting speed should also be adapted to the rivet element and material.

Insufficient friction may result in the failure mode *screw breakaway out of tolerance* (cf. Fig. 1) and result in high costs or liability claims by end customers. Hence, changes of the torque or calibration of the robot may have immediate effects on the FMEA failure mode. A divergence between FMEA and PSE concepts can result in too many or too few quality checks during the production process. Hence, updates of values of related concepts in engineering views require the re-validation of FMEA model elements by involved domain experts. There may be hundreds of FMEA conditions for a machine concerning hundreds of engineering concepts in various stakeholder views. Therefore, the efficient re-validation of the FMEA model requires capabilities for the prioritization of FMEA model elements related to changes in stakeholder views

[1] APIS IQ: www.apis-iq.com/

and the grouping of FMEA concerns to involved stakeholders to conduct focused workshops for re-validation.

Requirements. Based on the use case, we identified the following requirements (Rx) for an integrated meta-model to support MvFMEA+PPR co-evolution and re-validation.

- *R1. FMEA concept representation.* The meta-model shall represent FMEA concepts – in particular failure modes, causes, their relationships and characteristics, such as severity and probability.
- *R2. PPR concept representation.* The meta-model shall represent PPR concepts [4] – in particular products, production processes, production resources, and their relationships and properties.
- *R3. FMEA-to-PPR dependency representation.* The meta-model shall represent links between FMEA concepts and PPR concepts, that are semantically similar to concepts used in the FMEA.
- *R4. FMEA/PPR change coordination representation.* The meta-model shall represent design and validation states for change coordination, such as model elements that changed or have to be re-validated after changes.
- *R5. Efficient FMEA re-validation after PPR changes.* The process shall provide capabilities for designing and instantiating an MvFMEA+PPR model. Also, the efficient identification of FMEA model elements that require re-validation shall be possible.

V. Co-Evolution of FMEA and PPR Models

This section introduces the MvFMEA+PPR meta-model and FMEA-enhanced-PPR (FMEA+PPR) approach that link FMEA models to engineering assets.

A. Multi-view FMEA+PPR meta-model

To address the requirements (cf. Section IV), we introduce the MvFMEA+PPR meta-model based on [33]. Specific concepts and relations for the initial MvFMEA+PPR meta-model have been partially derived from (a) an FMEA Ontology [36], (b) the VDI 3682 [4], and (c) the VDI 3682 Ontology-Design-Pattern [37]. We extend this initial FMEA+PPR meta-model with multi-view modeling and coordination concepts and elements derived from [23], [31]. The resulting MvFMEA+PPR meta-model is depicted in Fig. 2.

FMEA concepts. To address requirement R1 *FMEA concepts*, we derived an initial FMEA meta-model from the FMEA Ontology [36] (cf. Fig. 2, yellow classes). The initial meta-model can represent an FMEA cause effect graph. Therefore, the *FMEA* class serves as a starting point. The *FMEA* examines a *Process*, which can consist of sub processes. To model a cause-effect dependency, a *FailureMode* can be assigned to a *Process*. A *FailureMode* can be caused by another failure mode. A Risk Priority Number (RPN) *MitigationAction* and a *ControlMethod* can be assigned to a failure mode.

PPR concepts. To address requirement R2 *PPR concepts*, the initial meta-model has been extended with PPR concepts and relations from the VDI 3682 Ontology-Design-Pattern [4], [37] (cf. Fig. 2, red classes) to enable the representation of PPR graphs. The figure omits some PPR concepts, such as *Energy*, *Information*, and *SystemBorder*, for better readability. The central class of the extended meta-model is the *Process-Operator* class. To this class *Resources* can be assigned as well as input and output *Products*, which are, according to VDI 3682, a sub class of *State*. A *ProcessOperator* consists of a *Product*, which is merged with the *Product* class of the FMEA meta-model (cf. Fig. 2, red/yellow class).

Links between FMEA and PPR concepts. To address requirement R3, the meta-model has been extended with a *BasicObject* class, from which all classes of the FMEA and PPR inherits (cf. Fig. 2, gray class). For readability, the inheritance links are not shown in the figure. To specify links between FMEA (cf. Fig. 2, green classes) and PPR concepts and characteristics, a *BasicObject* and a *Characteristic* can have *Link*s assigned. Links can be of a certain *Type* i.e., to distinguish links between PPR-tp-PPR concepts and PPR-to-FMEA concepts. The *Characteristic* class is used to define attributes in the meta-model to FMEA or PPR elements. The *ValueAttribute*, based on IEC 61360, is used to specify a property value. A *Characteristic* is assigned to a *View* to describe its context related to a *Stakeholder*.

Change coordination states of and dependencies between PPR and FMEA assets/concepts. To address requirement R4 and R5, the meta-model represents (a) change coordination states of PPR and FMEA assets and concepts and (b) coordination dependencies between PPR and FMEA assets and concepts (cf. Fig. 2, green classes). The coordination states are represented by the *Marker* class that is related to a *Characteristic*. A *Marker* has a *State* and a specific *Type* assigned. To represent domain-specific dependencies for change coordination the previously introduced *Link* concept is used. This enables the indicating of model elements to evaluate for re-validation in case of a change in a PPR or FMEA asset.

B. Multi-view FMEA+PPR re-validation method

We use the MvFMEA+PPR meta-model and approach to conduct a qualitative analysis in CPPS engineering. The MvFMEA+PPR approach consists of the following steps.

Step 1. Specify initial Multi-view FMEA+PPR model. In this step, FMEA and domain experts determine the scope of the FMEA and identify relevant PSE artifacts from use case data. They design an initial model using the concepts of the MvFMEA+PPR meta-model. Based on the initial model, they define the PPR part of the MvFMEA+PPR model in cooperation with involved stakeholders using an approach for multi-view modeling in PSE [23].

Step 2. Define FMEA-to-PPR dependencies. In this step, the FMEA expert cooperates with domain experts to collect and explicitly model re-validation dependencies, change coordination states and dependencies.

Step 3. Integrate and manage data from multi-views. In this step, a multi-view model integration pipeline [23] is defined with links between specific PSE artifacts and PPR assets [31] to extract relevant information. The MvFMEA+PPR model is instantiated in a graph database, which provides a

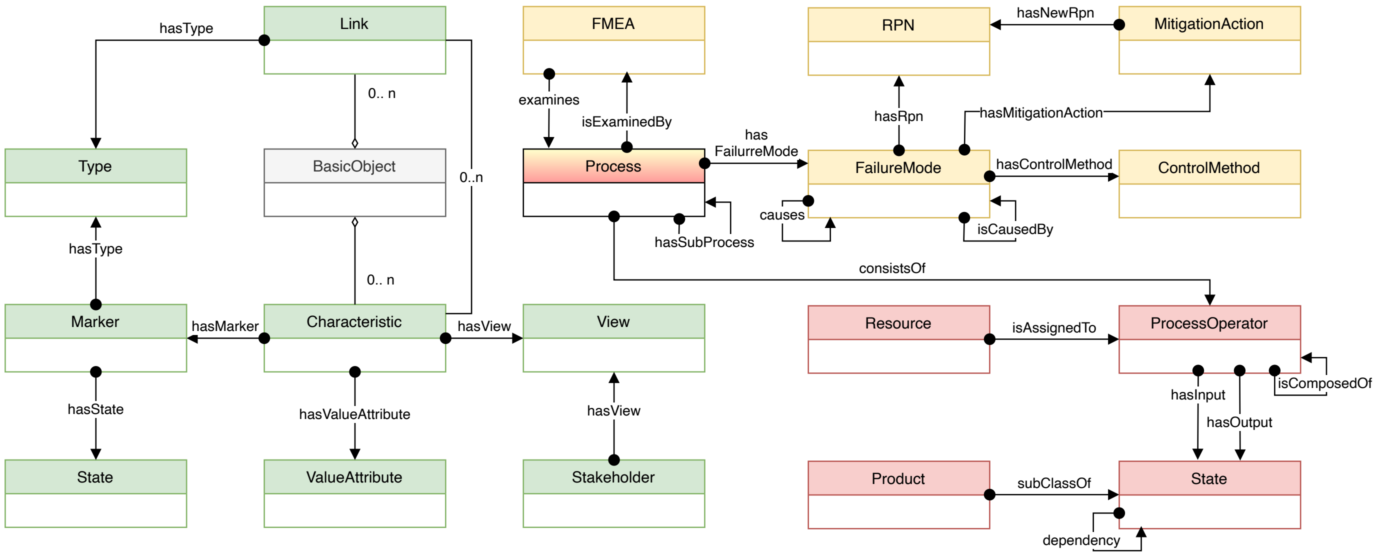

Fig. 2. Multi-view FMEA+PPR meta-model based on [33], the FMEA Ontology [36], and the VDI 3682 Ontology-Design-Pattern [37], enhanced by multi-view modeling and coordination approaches [23], [31].

foundation for reading or setting change and validation states and coordination dependencies. Furthermore, the multi-view model integration pipeline propagates changes in PSE artifacts to PSE/FMEA assets.

Step 4. Re-validate FMEA and PPR assets. In this step, the FMEA expert analyzes and marks the scope of assets for re-validating, e.g., update of PPR asset property values, using the MvFMEA+PPR model instance in graph database.

The FMEA expert and domain experts re-validate and improve marked elements in the model to reduce the risk of invalid assets in the FMEA model part.

The MvFMEA+PPR method provides the foundation for a re-validation of changes in a multi-disciplinary, multi-view engineering environment via an engineering graph. To investigate the feasibility of the meta-model and the method, we evaluate the approach in the following section.

VI. Evaluation and Discussion

This section demonstrates the MvFMEA+PPR meta-model's feasibility employing the illustrative use case, introduced in Section IV. In the feasibility study, we (a) instantiated the *MvFMEA+PPR meta-model* in *Neo4J* graph database following the MvFMEA+PPR method and (b) evaluate the MvFMEA+PPR model capabilities in comparison to the traditional FMEA approach with engineering artifacts and discuss the fulfillment of the MvFMEA+PPR requirements.

Model instantiation. To explore the feasibility and estimate the effort required for creating a MvFMEA+PPR model instance, we selected a typical robot cell [38]. Next, we collected typical PSE artifacts described in the FMEA, such as bills of materials, processes, resources, and their links, for several instances of the use case *FMEA Re-Validation after Changes to Engineering Artifacts* in a manufacturing work line [38].

Fig. 3 illustrates the derived instance of the FMEA model linked to a PPR network. Column *FMEA - Cause & Effect* shows an example failure mode *Screw breakaway torque out of tolerance* linked to a potential causes. Column *Products & Processes* contains a *Screw on Dashboard* process with a property *M.Torque*, automated by resources, including an *Electric Screwdriver* with the property *M.Torque*. The failure mode has a characteristic *Breakaway torque* that is linked to PPR assets and properties. Specifically, the property *M.Torque* of the process *Screw on Dashboard* and the resource *Electric Screwdriver* (orange FMEA to PPR links).

An engineering artifact change related to the *Electric Screwdriver*, e.g. via a agile model integration process described in [23], [31], results in an update of the property *M.Torque*. Consequently, the coordination state of this property is set to *changed* (red diamond marker). The dependent PPR assets and properties are marked as *to validate* (yellow diamond markers). Next, *(i)* the property of the associated failure mode characteristic *Breakaway_torque* gets marked as *to validate* by traversing the previously described FMEA to PPR link. and *(ii)* failure mode *Screw breakaway torque out of tolerance* gets marked as *to re-validate* (orange diamond markers) (cf. Cypher[2] queries in Listing 1). FMEA cause *Robot not correctly calibrated* carries a marker *validated* (green diamond) from a recent validation task.

The FMEA/PPR assets, properties, and links provide a foundation for graph database queries in *Neo4J* that answer stakeholder questions, such as *which FMEA assets are linked to a changed PPR node?* An instantiation of such a graph in *Neo4J* notation is accessible for replication in an online

[2]Cypher: www.opencypher.org/

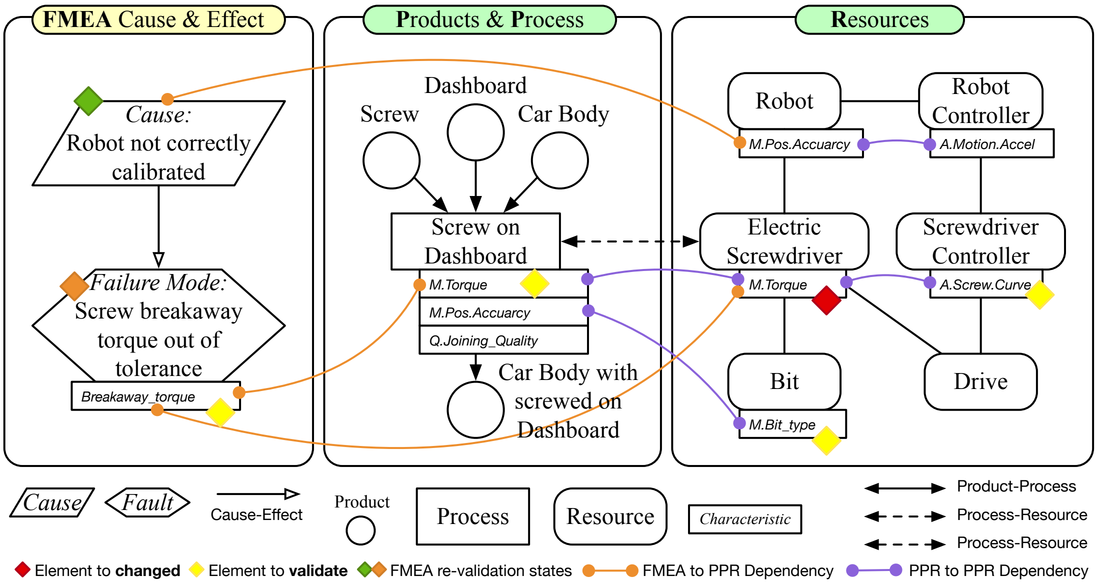

Fig. 3. FMEA model with coordination links to a PPR network for the use case *FMEA Re-Validation after Changes to Engineering Artifacts*, based on [31].

```
MATCH (startnode:Characteristic)-[edge]-(endnode)
  WHERE startnode.ChangeState = "Changed"
  SET endnode.ValidationLifeCycleState="To Validate"

MATCH (startnode:Characteristic)-[edge]-
    (endnode:FailureMode)
  WHERE startnode.ValidationLifeCycleState="To Validate"
  SET endnode.ValidationLifeCycleState="To Re-Validate"
```

Listing 1: Cypher queries for FMEA element re-validation.

Req. / *FMEA +*	*EA*	*TS*	*MvPPR*
R1. FMEA concept representation	○	+	++
R2. PPR concept representation	−−	○	++
R3. FMEA-to-PPR dependency representation	−−	+	++
R4. FMEA/PPR change coordination representation	−−	+	++
R5. Efficient FMEA re-validation after PPR changes	−	−	+

TABLE I
FMEA RE-VALIDATION CAPABILITIES OF *FMEA+EA*, *FMEA+TS*, AND *MvFMEA+PPR* RE-VALIDATION APPROACHES.

repository[3]. The Cypher query (cf. Listing 1) sets coordination markers to PPR and FMEA elements. Using these markers, the FMEA experts and CPPS engineers can re-validate the quality of the changes of the assets and their characteristics, resp. utilize them as a basis for discussion and issue resolution.

Evaluation of FMEA re-validation capabilities. We compare the MvFMEA+PPR to the traditional approaches (a) FMEA+EA: *FMEA re-validation based on Engineering Organisations (EOs)* in a shared space, requiring manual mapping and co-evolution of FMEA models and PSE artifacts, and (b) FMEA+TS: *FMEA re-validation in Tool Suites (TSs)* that manage engineering objects in a data base as a basis for co-evolution with FMEA model versions. We used a 5-point *Likert* scale (++, +, ○, −, −−), where ++/−− indicate very high/low capabilities, to evaluate the fulfillment of the requirements in comparison with alternative approaches. Table I summarizes the results.

R1. FMEA concept representation. For all approaches, we assume the use of a best-practice FMEA tool, such as APIS, with FMEA concepts and conditions represented in natural language, possibly with references to PSE concepts. *FMEA+EA* is rated average as the FMEA concepts can refer to stakeholder views in heterogeneous Engineering Artifacts (EAs), requiring for one FMEA concept the management of references to several stakeholder views, e.g., mechanical/electrical identifiers in M-CAD/E-CAD, software identifiers in programs and configurations, which concern an Electric Screwdriver. *FMEA+TS* is rated high as one FMEA concept can refer to one engineering object, e.g., the Electric Screwdriver, which represents several stakeholder views in the tool suite data model. However, the tool suite data model covers only a limited set of stakeholder views and falls back to engineering artifacts for stakeholder views not covered by the tool suite. *MvFMEA+PPR* is rated very high as one FMEA concept can refer to PPR concepts and, if required, stakeholder views attached to a PPR asset. By design, the MvFMEA+PPR model represents the required FMEA graph concepts.

R2. PPR concept representation. FMEA+EA is rated very low as the approach concerns engineering artifacts that, in general, do not consider PPR assets. *FMEA+TS* is rated average as the EOs may represent PPR assets and their properties, but do not consider dependencies between PPR assets. Furthermore, the TS covers only a limited set of PPR assets. *MvFMEA+PPR* is rated very high as it represents all relevant stakeholder views as PPR assets and their properties. Moreover, explicit dependencies between PPR concepts represent domain expert knowledge, e.g., on change dependencies.

R3. FMEA-to-PPR dependency representation. FMEA+EA

[3] Multi_view-FMEA-PPR.neo4j:
https://github.com/tuw-qse/fmea-revalidation-resources

is rated very low as the approach considers dependencies to engineering artifacts, not PPR assets or properties. *FMEA+TS* is rated high as the approach considers dependencies to engineering objects, but with limited stakeholder views. *MvFMEA+PPR* is rated very high as the MvFMEA+PPR model explicitly represents FMEA+PPR links between FMEA and PPR concepts.

R4. FMEA/PPR change coordination representation. FMEA+EA is rated very low as change coordination is limited to engineering artifacts in the team work space and neither covers FMEA nor PPR concepts. *FMEA+TS* is rated high as change coordination concerns individual engineering objects. However, there is no consideration of a network of change dependencies and the scope of stakeholder views is limited. *MvFMEA+PPR* is rated very high as the model represents the required change coordination states, e.g., markers for representing the state of change and re-validation, missing links and dependencies of PPR and FMEA assets/concepts.

R5. Efficient FMEA re-validation after PPR changes. FMEA+EA is rated low as comparing FMEA concepts to changes in heterogeneous engineering artifacts involves significant manual effort from domain experts to identify FMEA concepts for re-validation after each change to an engineering artifact. *FMEA+TS* is rated low as the automation of FMEA re-validation in the tool suite would require adding the FMEA view to the tool suite with considerable effort to design. However, once implemented, the FMEA re-validation could become very efficient in the limited scope on engineering disciplines in the tool suite. *MvFMEA+PPR* is rated high as the approach considers the relevant scope of engineering disciplines and tools in a PSE project assuming the multi-view data logistics capabilities for efficient update of PPR assets from engineering artifacts.

Discussion. Overall, the MvFMEA+PPR approach seems well suited to provide FMEA re-validation capabilities as a basis for integrating FMEA with agile PSE. For a demonstration, the *Neo4J* graph instance enables efficient queries to analyze linked FMEA and PPR knowledge, e.g., for efficient FMEA re-validation based on queries to the graph database to select and prioritize FMEA elements for re-validation.

Limitations. The number of the re-validation concepts and dependencies used in the evaluation can be considered a limitation. Industrial scenarios may also require more detailed modeling of FMEA conditions. We plan to investigate the effectiveness of the approach in more detail in a setting that involves a larger number of concepts, dependencies and stakeholder views.

VII. Conclusion and Future Work

In agile PSE, multi-disciplinary stakeholders work on partial PPR views in engineering artifacts in an iterative parallel process towards a functional production system. In such settings, FMEA is vital to reduce the risk of PSE design errors, such as mismatches between stakeholder designs, that may be costly to resolve in late PSE stages. Therefore, is it crucial to reuse FMEA knowledge on system components from previous projects and efficiently identify FMEA elements to re-validate after updates to PSE artifacts coming from heterogeneous stakeholder views. Efficiently identifying FMEA elements for re-validation requires capabilities to *(i)* trace or propagate a change in a PSE artifact *(ii)* a common PSE object, such as a PPR asset, which reflects the integrated knowledge in the project team, and *(iii)* keep track of the change states of shared PSE objects and of FMEA elements. However, in current best-practice knowledge management in PSE, FMEA elements and PSE artifacts represent the knowledge required for FMEA re-validation incompletely. Furthermore, their meaning is difficult to interpret automatically, which makes FMEA re-validation inefficient and prone to error. Therefore, FMEA re-validation may become ineffective in agile PSE, reducing the benefit that would be expected from conducting FMEA early.

This paper reported on the use case *FMEA Re-Validation after Changes to Engineering Artifacts*, derived from car manufacturing with automated robot work cells, and identified a set of requirements for FMEA re-validation capabilities (cf. Section IV). To address these requirements, we developed the *MvFMEA+PPR approach* that consists of *(i)* a meta-model to represent the required knowledge for efficient FMEA re-validation, and *(ii)* a method to map FMEA elements to PPR concepts. The approach provides a foundation for the co-evolution and efficient analysis of FMEA and PPR knowledge.

A feasibility study showed a MvFMEA+PPR model instance that represents the required knowledge for analyzing the impact of changes in multi-view engineering graph. We compared the MvFMEA+PPR approach to two traditional best-practice approaches in PSE that relate FMEA elements (i) to engineering artifacts in a shared space or (ii) to engineering objects in a tool suite database. The study results encourage evaluating the MvFMEA+PPR approach in a broader context regarding usability and scalability in agile PSE scenarios of different sizes and complexity.

Future Work. We plan to investigate the usability and usefulness of the MvFMEA+PPR approach in various agile PSE settings, e.g., making implicit domain expert knowledge sufficiently explicit in FMEA with PSE models to automate analyses for the quality assurance and reuse by using semantic technologies. Due to the comprehensive scope of FMEA and PSE tasks, a model's complexity may grow considerably with the number of data elements and links. This will require research on the scalability of FMEA+PPR models.

Acknowledgement

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 supported and funded by the Austrian Research Promotion Agency (FFG) via "Austrian Competence Center for Digital Production" (CDP), contract nr. 881843. This work has received funding from the Teaming.AI project, which is part of the European Union's Horizon 2020 research and innovation program under grant agreement No 957402.

References

[1] L. Monostori, "Cyber-physical Production Systems: Roots, Expectations and R&D Challenges," *Procedia CIRP*, vol. 17, pp. 9–13, 2014.

[2] V. Gunes, S. Peter, T. Givargis, and F. Vahid, "A survey on concepts, applications, and challenges in cyber-physical systems." *KSII Transactions on Internet & Information Systems*, vol. 8, no. 12, 2014.

[3] M. Schleipen and R. Drath, "Three-view-concept for modeling process or manufacturing plants with AutomationML," in *14th IEEE Int. Conf. on Emerging Technologies and Factory Automation*. IEEE, 2009, pp. 1–4.

[4] *VDI Guideline 3682: Formalised process descriptions.*, VDI/VDE, VDI/VDE Std., 2015. [Online]. Available: https://www.vdi.de

[5] S. Biffl, A. Lüder, and D. Gerhard, Eds., *Multi-Disciplinary Engineering for Cyber-Physical Production Systems, Data Models and Software Solutions for Handling Complex Engineering Projects*. Springer, 2017.

[6] B. Vogel-Heuser, M. Böhm, F. Brodeck, K. Kugler, S. Maasen, D. Pantförder, M. Zou, J. Buchholz, H. Bauer, F. Brandl, and et al., "Interdisciplinary engineering of cyber-physical production systems: highlighting the benefits of a combined interdisciplinary modelling approach on the basis of an industrial case," *Design Science*, vol. 6, p. e5, 2020.

[7] Z. Wu, W. Liu, and W. Nie, "Literature review and prospect of the development and application of fmea in manufacturing industry," *The International Journal of Advanced Manufacturing Technology*, pp. 1–28, 2021.

[8] F. Scippacercola, R. Pietrantuono, S. Russo, A. Esper, and N. Silva, "Integrating FMEA in a Model-Driven Methodology," *DASIA 2016-Data Systems In Aerospace*, vol. 736, p. 10, 2016.

[9] Z. Huang, S. Swalgen, H. Davidz, and J. Murray, "Mbse-assisted fmea approach — challenges and opportunities," in *2017 Annual Reliability and Maintainability Symposium (RAMS)*. IEEE, 2017, pp. 1–8.

[10] L. Zheng, Q. Liu, and C. McMahon, "Integration of process fmea with product and process design based on key characteristics," in *Proc. CIRP Int. Conf. on Digital Enterprise Tech.* Springer, 2010, pp. 1673–1686.

[11] S. Kropatschek, T. Steuer, E. Kiesling, K. Meixner, T. Frühwirth, P. Sommer, D. Schachinger, and S. Biffl, "Towards the representation of cross-domain quality knowledge for efficient data analytics," in *26th IEEE Int. Conf. on Emerging Technologies and Factory Automation*. IEEE, 2021, pp. 01–04.

[12] A. Strahilov and H. Hämmerle, "Engineering workflow and software tool chains of automated production systems," in *Multi-Disciplinary Engineering for Cyber-Physical Production Systems*. Springer, 2017.

[13] A. Wortmann, O. Barais, B. Combemale, and M. Wimmer, "Modeling languages in Industry 4.0: an extended systematic mapping study," *Software and Systems Modeling*, vol. 19, no. 1, pp. 67–94, 2020.

[14] I. Grangel-González, L. Halilaj, S. Auer, S. Lohmann, C. Lange, and D. Collarana, "An RDF-based approach for implementing industry 4.0 components with Administration Shells," in *21st IEEE Int. Conf. on Emerging Technologies and Factory Automation*, 2016, pp. 1–8.

[15] Plattform Industrie 4.0 and ZVEI, " Part 1 - The exchange of information between partners in the value chain of Industrie 4.0 (Version 3.0RC01 Review)," German BMWI, Standard, Nov. 2020, https://bit.ly/37A002I.

[16] C. Atkinson, C. Tunjic, and T. Möller, "Fundamental realization strategies for multi-view specification environments," in *2015 IEEE 19th Int. Enterprise Distributed Object Computing Conf.*, 2015, pp. 40–49.

[17] S. L. Star, "The Structure of Ill-Structured Solutions: Boundary Objects and Heterogeneous Distributed Problem Solving," in *Distributed Artificial Intelligence*, L. Gasser and M. N. Huhns, Eds. Elsevier, 1989, pp. 37–54.

[18] F. Rinker, L. Waltersdorfer, K. Meixner, and S. Biffl, "Towards Support of Global Views on Common Concepts employing Local Views," in *24th IEEE Int. Conf. on Emerging Technologies and Factory Automation*. New York, USA: IEEE, 2019, pp. 1686–1689.

[19] M. Schleipen, A. Lüder, O. Sauer, H. Flatt, and J. Jasperneite, "Requirements and concept for plug-and-work," *at-Automatisierungstechnik*, vol. 63, no. 10, pp. 801–820, 2015.

[20] K. Meixner, F. Rinker, H. Marcher, J. Decker, and S. Biffl, "A domain-specific language for product-process-resource modeling," in *26th IEEE Int. Conf. on Emerging Technologies and Factory Automation*. IEEE, 2021, pp. 1–8.

[21] C. Tunjic and C. Atkinson, "Synchronization of projective views on a single-underlying-model," in *Proceedings of the 2015 Joint MORSE/VAO Workshop on Model-Driven Robot Software Engineering and View-based Software-Engineering*, 2015, pp. 55–58.

[22] J. Meier, C. Werner, H. Klare, C. Tunjic, U. Aßmann, C. Atkinson, E. Burger, R. H. Reussner, and A. Winter, "Classifying Approaches for Constructing Single Underlying Models," in *MODELSWARD, Revised Selected Papers*, ser. Communications in Computer and Information Science, vol. 1161. Springer, 2019, pp. 350–375.

[23] F. Rinker, L. Waltersdorfer, K. Meixner, D. Winkler, A. Lüder, and S. Biffl, "Continuous Integration in Multi-view Modeling: A Model Transformation Pipeline Architecture for Production Systems Engineering," in *MODELSWARD*. SCITEPRESS, 2021, pp. 286–293.

[24] B. Illés, P. Tamás, P. Dobos, and R. Skapinyecz, "New challenges for quality assurance of manufacturing processes in industry 4.0," in *Solid State Phenomena*, vol. 261. Trans Tech Publ, 2017, pp. 481–486.

[25] M. Foehr, "Integrated consideration of product quality within factory automation systems," Ph.D. dissertation, Otto-v.-Guericke University Magdeburg, FMB, 2013. [Online]. Available: http://dx.doi.org/10.25673/3977

[26] D. H. Stamatis, *Risk Management Using Failure Mode and Effect Analysis (FMEA)*. Quality Press, 2019.

[27] K. D. Sharma and S. Srivastava, "Failure mode and effect analysis (FMEA) implementation: a literature review," *Journal of Advanced Research in Aeronautics and Space Science*, vol. 5, pp. 1–17, 2018.

[28] P. Chuang, "Incorporating disservice analysis to enhance perceived service quality," *Industrial Management & Data Systems*, vol. 110, no. 3, pp. 368–391, Jan. 2010.

[29] R. Victor B. de Souza and L. Cesar R. Carpinetti, "A FMEA-based approach to prioritize waste reduction in lean implementation," *International Journal of Quality & Reliability Management*, vol. 31, no. 4, pp. 346–366, Jan. 2014.

[30] C. Spreafico, D. Russo, and C. Rizzi, "A state-of-the-art review of FMEA/FMECA including patents," *Computer Science Review*, vol. 25, pp. 19–28, Aug. 2017.

[31] S. Biffl, J. Musil, A. Musil, K. Meixner, A. Lüder, F. Rinker, D. Weyns, and D. Winkler, "An Industry 4.0 Asset-Based Coordination Artifact for Production Systems Engineering," in *23rd IEEE Int. Conf. on Business Informatics*. IEEE, 2021.

[32] S. Biffl, A. Lüder, K. Meixner, F. Rinker, M. Eckhart, and D. Winkler, "Multi-view-Model Risk Assessment in Cyber-Physical Production Systems Engineering," in *MODELSWARD*. SCITEPRESS, 2021, pp. 163–170.

[33] F. Rinker, S. Kropatschek, T. Steuer, E. Kiesling, K. Meixner, P. Sommer, A. Lüder, D. Winkler, and S. Biffl, "Efficient FMEA Re-Validation: Multi-view Model Integration in Agile Production Systems Engineering," CDL-SQI, Inst. for Information Systems Eng., TU Wien, Technical Report CDL-SQI 2021-13, Nov. 2021. [Online]. Available: https://doi.org/10.34726/2362

[34] I. Graessler, J. Hentze, and C. Oleff, "Systems engineering competencies in academic education : An industrial survey about skills in systems engineering," in *13th Annual Conference on System of Systems Engineering*. IEEE, 2018, pp. 501–506.

[35] K. Pätzold, *Product and Systems Engineering/CA* Tool Chains*. Springer International Publishing, 2017, pp. 27–62.

[36] Z. Rehman and C. Kifor, "An Ontology to Support Semantic Management of FMEA Knowledge," *International Journal of Computers Communications & Control*, vol. 11, no. 4, pp. 507–521, 2016.

[37] C. Hildebrandt, A. Köcher, C. Küstner, C.-M. López-Enríquez, A. W. Müller, B. Caesar, C. S. Gundlach, and A. Fay, "Ontology Building for Cyber–Physical Systems: Application in the Manufacturing Domain," *IEEE Transactions on Automation Science and Engineering*, vol. 17, no. 3, pp. 1266–1282, 2020.

[38] K. Meixner, A. Lüder, J. Herzog, D. Winkler, and S. Biffl, "Patterns For Reuse In Production Systems Engineering," *International Journal of Software Engineering and Knowledge Engineering*, 2021.