Submission for the ASME 2016 International Manufacturing Science and Engineering Conference
MSEC2016
June 27- July 1, 2016, Blacksburg, Virginia, USA

MSEC2016-8658 DRAFT

KNOWLEDGE-BASED PRODUCTION PLANNING WITHIN THE REFERENCE PLANNING PROCESS SUPPORTING MANUFACTURING CHANGE MANAGEMENT

Benjamin Gernhardt
University of Hagen
Hagen, Germany

Tobias Vogel
University of Hagen
Hagen, Germany

Mohammad Givehchi
Royal Institute of Technology
Stockholm, Sweden

Lihui Wang
Royal Institute of Technology
Stockholm, Sweden

Matthias Hemmje
University of Hagen
Hagen, Germany

KEYWORDS
Function Blocks, DPP, CAPP, WPIM, Process Planning, Process-oriented Knowledge Management, Semantic Knowledge Representation, Mediator

ABSTRACT

The manufacturing of a product takes place in several partial steps and these mostly in different locations to save tax or to use the best providers. Therefore, in the era of ***Internet of Things (IoT)*** and modern ***Intelligent Production Environments (IPE)*** are going to be inevitably based on a cloud-based repository and distributed architecture to make data and information accessible everywhere as well as development processes and knowledge available for worldwide cooperation. Semantic approaches for knowledge representation and management as well as knowledge sharing, access, and re-use can support ***Collaborative Adaptive Production Process Planning (CAPP)*** in a flexible, efficient, and effective way. Thus, semantic representations of such CAPP knowledge integrated into a machine readable process formalization is a key enabling factor for sharing such knowledge in cloud-based knowledge repositories supporting CAPP scenarios as required for e.g., ***Small and Medium Enterprises (SMEs)***. When such contributors work together on a product component production planning, they exchange component production and manufacturing change information between different planning subsystems which require, e.g., a standardized product-feature- and production-machine feature representation. These data exchanges are mostly based on applying the already established ***Standard for the Exchange of Product model data (STEP)*** for the computer-interpretable representation and exchange of product manufacturing information. Furthermore, the planning process can be supported by so-called ***Function Block (FB)*** based knowledge representation models, serving as a high-level planning-process knowledge-resource template. Web-based and at the same time Cloud-based tool suites, which are based on process-oriented semantic knowledge-representation methodologies, such as ***Process-oriented Knowledge-based Innovation Management*** (German: **W**issens-basiertes **P**rozess-orientiertes **I**nnovations **M**anagement, **WPIM**) can satisfy the needs of representing such planning processes and their knowledge resources. In this way, WPIM can be used to support the integration and management of distributed CAPP knowledge, as well as its access and re-use in ***Manufacturing Change Management (MCM)*** including ***Assembly-, Logistics and Layout Planning (ALLP)***. Therefore, also a collaborative planning and optimization for mass production in a machine readable and integrated representation is possible. On the other hand, that knowledge can be shared within a cloud-based semantic knowledge repository. To integrate all these functionalities, this paper introduces a new method, called ***Knowledge-based Production Planning (KPP)*** and outlines the advantages of integrating CAPP with ***Collaborative Manufacturing Change Management (CMCM).*** In this way, an enabling basis for achieving ALLP interoperability in Distributed Collaborative Manufacturing and Logistics will be demonstrated.

INTRODUCTION, MOTIVATION, AND PROBLEM STATEMENT

The general concept of developing a knowledge-based and process-oriented CAPP support by using the WPIM method as a basis was proposed in [1]. The WPIM approach offers the possibility of modeling and representing innovation processes in a machine-readable semantic format and furthermore enables annotating the process representation in a semantic way with further planning content and knowledge resources. This whole representation structure can then later be accessed by means of semantic queries. However, so far WPIM has only been applied in domains like component design and distributed production planning. It also includes ***Product Life Cycle Management (PLM)*** support but it has not yet been practically applied in the domain of CMCM as a means of supporting ALLP in Collaborative Manufacturing and Logistics scenarios.

In parallel to the development of WPIM, Wang et al. have introduced a method for representing web-based ***Distributed Process Planning (DPP)*** activities in [3], [4] and [5]. In the following we will use slightly adapted excerpts from [3] to introduce the necessary concepts and rationale of the DPP method. The DPP method includes also the concepts of ***Meta Function Blocks (MFBs), Execution Function Blocks (EFBs) and Operation Function Blocks (OFBs)***.

Furthermore, Helgoson et al. explain in [6] that “Today, machining-feature based approaches combined with ***Artificial Intelligence (AI)*** based methods are the popular choices for process planners”. Their introduced approach is already based on a DPP modeling-method but does not yet support machine-readability and semantic interoperability of such models as it could be achieved by utilizing representations as available in nowadays semantic web technologies and as e.g., supported by WPIM. This means, while the proposed DPP approach is very useful and valid in terms of representing the product and machining features within MFBs, EFBs, and OFBs it does nevertheless not yet support semantic-web based cross-organizational and cross-domain knowledge sharing. This holds especially true for the areas of CMCM as a means of optimizing conceptually planned component production processes and at the same time enabling them to be re-used as ALLP knowledge representations.

However, this is necessary to make such knowledge more widely available e.g., to be shared in collaborations of SMEs within CAPP, CMCM, and ALLP activities. Furthermore, the relevant CMCM and ALLP knowledge to be integrated with the machine readable CAPP knowledge representations is currently not available in a machine-readable semantic representation at all.

Furthermore, the interoperability of such a CMCM and ALLP knowledge representation with technologies of the semantic-web and therefore with other applications and tools, like e.g., from the area of AI and ***Machine Learning (ML)***, cannot easily be achieved.

Moreover, this CMCM and ALLP knowledge cannot easily be automatically shared, managed, accessed, exchanged, and re-used within collaborations that take advantage of cloud-based semantic repositories of CAPP-, CMCM-, and ALLP-knowledge. Besides, it is aiming at optimizing a conceptual CAPP instance for component production to also include ALLP and its optimization, ideally beyond planning domains and across organizational borders. If such a semantic and process-oriented integration of CAPP- and ALLP support based on a CMCM knowledge representation utilizing semantic-web technologies would exist, it could be very well supported by other semantic-web enabled technologies and in this way become interoperable. Therefore, an integration of cloud-based semantic CAPP and CMCM knowledge repositories with other e.g., AI, CAPP-, and MCM-support technologies could be achieved by means of integrating them based on the semantic web software development paradigm.

In consequence, this insight requires the application of semantic technologies to knowledge sharing and mediation in CAPP and CMCM to support overall ALLP processes. In this way process-oriented semantic representations of CAPP knowledge in which the ***Product Features (PFs)*** and ***Machining Features (MFs)*** are formalized within MFBs, EFBs and OFBs like domain-specific representations i.e., domain models of the DPP knowledge domain could support not only the CAPP knowledge domain but also the CMCM knowledge domain and the re-use of the planning and change management knowledge in the ALLP scenario at the same time.

Based on this insight, in our preceding paper "Supporting Production Planning through Semantic Mediation of Processing Functionality" [31], we have already been describing in detail the applying and implementing necessary DPP and semantic-web integration approach for KPP within a so-called ***Mediator Architecture (MA)*** [14]. Such MAs are typical for distributed implementations of semantic-web repositories and are solving semantic integration challenges as well as integrating several local knowledge sources into a global, potentially cloud-based, semantic repository. Furthermore, we have already explained that this can then be considered a semantic and cloud-based CAPP-knowledge repository which has been implemented in a very (technologically) open and distributed way. From the point of view of WPIM, we have also demonstrated the domain models for MFBs, EFBs and OFBs can be covered by a semantic integration in this repository with the existing WPIM domain concepts of **WPIM-Master Processes**, -**Process Instances**, -**Tasks** and –**Activities.** Thus, it supports the integration of WPIM- and DPP-based knowledge modeling, as well as the semantic representation of DPP knowledge to become available as a knowledge-based support to CAPP activities. We are now reviewing this result in the light of other related work support ALLP interoperability as well as possible integrations with MCM and especially CMCM features.

To support this analysis of related work, we will start in the area of currently emerging international standards aiming at supporting CAPP and at the same time aiming at supporting MCM. Only recently the ProSTEP iViP Association [29] published a White Paper, called “Modern Production Planning Processes” [27]. This Paper is based on the currently emerging ISO/DIS 18828 Standard [30] and aims at representing an end-to-end reference process that can be adapted to individual needs. This formal process, so called ***Reference Planning Process (RPP)***, is a recommendation and should be used in this paper as a basis for a proof of concept implementation evaluating and validating our KPP approach as a possible reference implementation of RPP. In addition, KPP applies even to another recommendation of ProSTEP iViP, called "Manufacturing Change Management (MCM)" [28]. MCM deals with the management of changes during production and continuously collaborates with the Production Planning Phase and vice-versa. Therefore, we will also review this recommendation in the light of being evaluated and validated by means of KPP as a reference implementation of an in- and output process for MCM. That can achieve learning about optimization opportunities in the area of assembly, logistics and layout dimensions of the overall planning and change management processes.

Therefore, in the remainder of this paper, KPP will be explained in more detail and we will show the interoperability, usability, and in general the usefulness of KPP as an input and output process for MCM which in this way supports bridging the gap between CAPP and CMCM processes. Our analysis and validation will aim at including an estimation of possible impacts from such an integration based on these recommendations and their integration and reference implementation with other relevant standards.

To start with and referring to our previous paper [31], we will revisit very briefly and in summary the FB concept as well as the related concept of MFBs, EFBs, and OFBs. Furthermore, the State of the Art of the FB-based production planning models and the proposed DPP method will be revisited. We briefly describe the State of the Art w.r.t. Process Ontologies and the WPIM-Ontology as well as the basics of the mediator technology. In the following modelling chapter of this paper, we will carry out a mapping of the DPP modeling approach of Wang et al. [3] to the WPIM-Ontology and we will introduce the method of KPP and its conceptual system, architecture which combines the advantages of these technologies in a hybrid approach based on the three level DPP mediation process described our previous paper [31]. In the following use case and evaluation design chapter we are going to discuss the KPP method and architecture w.r.t. its capability to serve as reference implementation of the currently emerging ISO/DIS 18828 Standard [30]. This takes place in order to semantically wrap, mediate, and integrate the overall DPP planning processes including its integration with the overall RPP and its utilization as an input to the CMCM support and management of the relevant CMCM knowledge resources to be re-used as an input to all following ALLP scenarios. Finally, conclusions and an outline of future work are provided.

STATE OF THE ART AND RELATED WORK

The following paragraphs will briefly summarize the State of the Art of FB based DPP modeling as well as the RPP and MCM models. The section is based on a slightly adapted excerpt from [3], the WPIM-based semantic process-modeling and the recommendations [27] [28] of ProSTEP iViP. It also introduces the necessary concepts of information integration and mediation as well as of mediator architectures as a background for the integration and mediation approach to be applied for the integration of DPP and WPIM. Furthermore, the relevant STEP standard will be introduced.

Function Blocks

FBs are initially defined in the IEC 61499 standard [7], which explains the usage, development, and implementation of FBs in distributed industrial process-measurement and -control systems in a component-oriented approach [8]. IEC 61499 was developed jointly from the existing concepts of FB diagram in the Programmable Logic Controllers (PLC) language standard IEC 61131-3 [9] and standardization work concerning Fieldbus [9]. An FB is an event-triggered component containing algorithms and an Execution Control Chart (ECC) with inputs and outputs of data and events. A literature review related to the FB related research targeting the areas of machining and assembly is available in [3] [4].

Distributed Process Planning

As outlined in more detail in [3], the three core components of the DPP are namely the planning processes of ***Supervisory Planning (SP), Operation Planning (OP)*** plus a new ***Execution Control Planning (ECP)***. These are explicitly modeled in a conceptual ***ICAM Definition for Function Modeling (IDEF0***, where '***ICAM***' is an acronym for ***Integrated Computer Aided Manufacturing***) process formalization model together with their inter-relationship and dataflow.

Meta, Execution and Operation Function Blocks

MFBs are used in this research to encapsulate machining sequences (of setups and machining features) and only contains generic information about process planning of a product. It is a high-level process template, including, e.g., suggested cutting tool types and tool path patterns, for subsequent manufacturing tasks.

EFBs are the FBs that are ready to be downloaded to a specific machine. Basically, an EFB can be created by instantiating a series of MFBs associated with a task. Each manufacturing task corresponds to its own set of EFBs, so that the monitoring functions can be conducted for each task unit.

OFBs has the same structure as that of an EFB. However, an OFB specifies and completes EFB with more detailed, machine-specific data about machining processes and operation sequences. Moreover, operation planning module can override and update the actual values of variables in the EFB, so as to make it locally optimized and adaptable to various events happened during machining operations. Wang et al. use the two different terms of EFB and OFB in [3].

WPIM

The concept of WPIM was developed to support capturing and usage of knowledge around innovation processes [1] [2] [10]. It assumes that innovation has both a knowledge and process perspective, which needs to be used in a combined manner. Therefore, activities of a process can be annotated with resources, such as experts and documents [10]. The semantic schema of the WPIM application and the corresponding tool suite is based on the ***Resource Description Framework (RDF)*** [11] and enables semantic-based searching by using the ***SPARQL Protocol And RDF Query Language (SPARQL)***. The ***Web Ontology Language (OWL)*** [12] [13] allows to model concepts in classes and replaceable relationships between such classes of concepts. These enabling technologies provide a well-defined formal semantic representation supporting the formal description of machine readable knowledge.

With the concepts of ***Master Processes*** (German: Masterprozess, ***MP***, see **Figure 1**), ***Process Instances*** (German: Prozessinstanz, ***PI***, see **Figure 1**) as well as **Activities** and ***Tasks*** the separation of modeling and capturing generic and instance specific process knowledge is supported. In this way, the process artefact representation toolbox of WPIM allows re-using process components and their associated machine-readable knowledge resources in a seamless way.

WPIM in the Domain of Process Planning

WPIM has already been applied to represent PLM data in the field of technical products and their production processes. In both domains next to executing processes also planning processes have been modeled and used for representation. Semantics as offered by WPIM have the advantage of being easily exchangeable and machine readable. This helps e.g., to plan cross-organizational and distributed, i.e., innovative production processes based on component production, assembly including MCM support as a basis for further ALLP use case scenarios.

The following **Figure 1** describes the interaction of a MP with its PIs. If such processes need to be represented in WPIM, in a first step the user selects classes of process components and resources in the WPIM ontology repository to register an instance of a process, a process component, or a process resource. This means, the user e.g., selects the process instance, component or resource classification system to be used as the global set of ontologies into which the knowledge resource structure and contents are to be mapped. In a second step, the user selects attributes for each selected resource class for populating virtual objects in these classes with content resources. This implies, the user has also e.g., to map the attributes of the resources to specific ontologies. Thus, indicating that an attribute's contents (their range) are mapped to an ontology, such as mapping a resource attribute onto an expert ontology. Finally, the user selects the populating methods or populates the resource instances and their specific content manually.

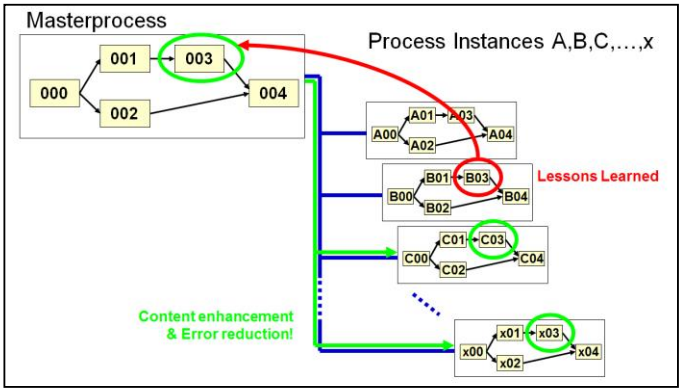

Figure 1: Master Process and Process Instances [2]

This means, the user maps the attributes of contents to classes in the ontology manually or semi-automatically using word-matching or other provided techniques e.g., map "hole" from a product feature/property ontology concept to the "drilled hole" concept in a machining feature ontology.

However, before such mappings can be established the sources' local data schemas must first be registered. For example, in our implementation we used the two activity-based schemas displayed in **Figure 1** for representing the MP and PI resources.

Master Processes

A MP is a generic high-level description of a process. In WPIM, from a data set point of view, a MP describes a data structure and attributes of a higher level template for a process. The representation approach goes beyond the sole representation of the process structural schema but describes process structures and their attributes by using semantic representations. As WPIM offers such semantic descriptions of MPs, the semantic MP schema exists as a generic and formal description of a process, independent of generated data instances during a certain execution of the process. Resources, which will be involved during execution the process. For instance, this can be experts, documents or, in the case of a CAPP adaptation, could be production machines and their production activities. In the case of integrating CAPP with CMCM activities, these representations could be extended to also include ALLP process knowledge and the corresponding knowledge resources of such processes.

Process, Activity and Task Instances

When executing a **Process**, data is gathered. WPIM describes this, from the data set point of view, as a PI. The Activity structure that exists in WPIM and is displayed in **Figure 2** is used to store all outgoing and incoming data as well as Activity states. Beyond that, WPIM also allows to describe and represent PIs including their Activities in a semantic, machine-readable format. Furthermore, WPIM PIs are ordered in a chronological way. That means, if a first instance is e.g., executed, the Lessons Learned during that execution can be stored within the higher level MP and this gathered information can be provided for the following process execution within the next PI (see **Figure 1**).

An **Activity** needs well defined inputs to generate a required output. Activities within WPIM contain one to many tasks. An instance of an Activity defines a cluster of tasks, e.g., an Activity can bundle tasks that are assigned to a single resource. Such an assignment can contain planning tasks that need to be executed by an expert (e.g., a planner) or tasks can also be assigned to a resource like a machine in order to represent the execution of a machine operation.

A **Task** structure is an action that can-not be further split into sub-actions. WPIM offers a semantic data representation to archive status and values when performing a Task. Such a Task can for example, represent an operation that can be executed by a machine and create a specified result. By having such a semantic representation containing incoming and outgoing status, progress attributes and result specification, WPIM allows to delegate a Task instance to various executing entities. An example, in the context of planning tasks, it's to finalize a plan by signing the plan and setting it into action. A signature to release a plan is a very unique task and it is obvious, that such a signing task cannot be split – either the plan is released via signature or it is not signed and therefore not released.

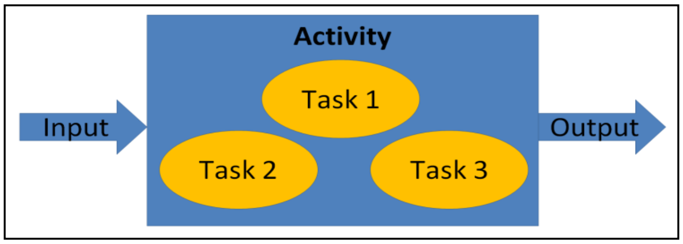

Figure 2: Visualization of an Activity as a set of Tasks

As displayed in **Figure 2**, an Activity consists of at least one up to many Tasks. These Tasks represent the transformation of an input of the Activity into an output.

Semantic integration within knowledge-based information system architectures

As outlined, e.g., in [15] data, information, and knowledge integration can be understood at varying levels of interoperability and heterogeneity. In the following, we will describe this a bit more in detail based on a slightly adapted excerpt from [15]. When trying to share distributed and heterogeneous data, a number of technical challenges must be overcome. Consider, for example, two systems having data sets that should be made interoperable. One can employ standards and technologies to overcome the various kinds of heterogeneities and to facilitate interoperability at different levels. At the systems level, one may find different operating systems (Linux, MS Windows, MacOS, etc.), different data transport protocols (FTP or HTTP, which are built on top of a stack of internet protocols called TCP/IP etc.) or higher-level protocols for discovery and interoperation of web services. System level interoperability can also be achieved at the grid or cloud service level. Grid and cloud services extend the basic web-service infrastructure and include additional features such as user authentication for secure data access. Apart from the generic issues of data access, transport and remote execution, there are also a number of application specific system level issues e.g., the choice and architecture of the mapping technology for the integration and mediation of information and knowledge resources (server-side, client-side, mixed). At the syntactic level, one has to consider heterogeneities such as different data file formats, depending on the type of content or knowledge resource and corresponding representation format of the information and knowledge representation. The Extensible Markup Language (XML) [16] provides a simple and very flexible syntax for structuring many kinds of data, metadata, content, and knowledge resources to enable their exchange. For example, this can be done in different ways, e.g. in an XML ***Document Type Definition*** (DTD) or an ***XML Schema Definition (XSD, XML Schema)*** [16] [17] to specify the allowed nesting structure and (in XML Schema) the data types of XML elements.

In this way, XML not only yields a data, information, content, and knowledge resource exchange syntax but also prescribes a schema for the exchanged resource. However, additional explicit representations of semantics such as domain specific integrity constraints have to be encoded by other means. As already outlined above, RDF [11] can be seen as an XML dialect for encoding labeled, directed graphs and in particular ontologies as an example of a standardized semantic vocabulary. For querying databases and query languages, such as the ***Standardized Query Language (SQL)*** [18] for relational databases) or ***XQuery*** (for XML databases) [19] are used, each of which come with their own syntax for query expressions. At the schema level, heterogeneities can exist because the same (or at least similar) data can be represented using vastly different schema structures (even when the same file format or syntax is used). For example, two datasets may be organized in different ways across two relational databases i.e., the table and column structure may be very different although the content (at the conceptual level) of the databases may be very similar. Similarly, different DTDs or XML Schemas can be used to describe the same data for XML databases. To overcome schema level heterogeneities, we can again apply two approaches, schema standardization or schema transformation.

For the latter, i.e., schema transformation, database query languages in general and XQuery in particular provide powerful means to express complex queries and transformations. Thus, (XML) query languages play an important role in database mediators. Finally, at the semantic level, we consider issues such as differences in terminology, different classification schemes and differences in the definition and constraints for the various concepts that are relevant to the data sets being integrated. Therefore, the main approach for reconciling semantic heterogeneities is the use of agreed-upon ontologies, which in their simplest form provide a controlled vocabulary with more or less formal descriptions of the pertinent concepts. In more sophisticated forms, ontologies include formalizations (often through logic formulas) of properties of concepts and "inter-dependencies" of concepts. Besides this, an important industry standard for ontologies is, as outlined above, OWL [20], which comes in three increasingly expressive variants: ***OWL Lite, OWL DL and OWL Full***. Based on these OWL variants, sematic representations that need to go beyond the expressiveness of RDF can be formally declared.

Mediator Architectures

Mediators are a standard approach in the construction of information system architectures. They have originally been introduced by Wiederhold in [14] as early as in 1991 when the web was still in its infancies and the semantic web did not even exist. However, since then, the use and application of these architectures in building web-based information systems supporting, data, information, and knowledge integration has grown into a de-facto standard. Database mediator systems can be used to provide uniform access to distributed heterogeneous data sets, and thereby overcome a number of the interoperability challenges mentioned above. **Figure 3** depicts a typical mediator architecture in which a number of local data sources are "wrapped" as XML sources and subsequently combined into an integrated global view. Thus, a client application or the end user is provided with the illusion of querying a single, integrated (or global) database with one integrated schema.

Mediators are software components that serve to simplify, reduce, combine, integrate, and explain data. They are mainly used for providing a common access level onto different distributed data sources. The source wrappers not only provide a uniform syntax, but also reconcile system aspects e.g., by means of a unified data access and query protocol [15].

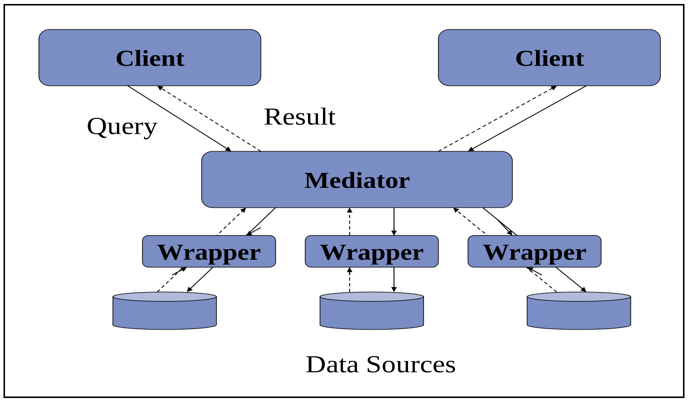

Figure 3: Mediator architecture integrating data sources

In a conventional relational, XML-, RDF-, or OWL-based mediator system, interoperability is facilitated first of all at the structural level. Differences in schema can be overcome by corresponding schema transformation as part of the view definitions for the global view. However, terminological differences or other semantic differences are not adequately handled at the purely structural e.g., XML level. To this end, source schema and contents can be registered to an ontology represented in RDF or OWL, which encodes additional "knowledge" about the registered concepts. In the next section, we will explain more in detail how by means of "ontology-enabling" the system in this way can evaluate high-level queries over concepts that are not directly in the source databases and yet indirectly linked via an ontology. The task of the mediator is, to transform queries to the global schema into queries to the local source schemata, to collect the results and to integrate and link them. The global scheme is based on a suitable data model, for which for example, XML or RDF can be used as representation. Wrappers are software components that represent the contents of a data source for the unification in another data model or schema. For example, XML wrappers are used to enable access to relational databases. The coupling between source and mediator via wrappers allows the mediator uniform access to the sources, by creating a mapping between the data model of the mediator and the data model of the local source. Also, incoming requests of the mediator can be translated into requests into the local source system.

Ontologies in Information Integration and Mediation

In information integration systems which are based on a mediator architecture, ontologies can be used to provide information at the level of conceptual models and terminologies. Thereby, facilitating conceptual-level queries against sources and resolving some of the semantic-level heterogeneities between them. In the original WPIM system, the process classification ontology and the innovation ontology are used as a global view for registering process resources and processing queries. When a resource is registered to an ontology, a mapping from the data set to the selected ontology is generated.

Wrappers use the mappings between the data source and ontology to translate queries from the global ontology to the local schema and also to translate content from the local schema to the global ontology. As explained above, the system can automatically use the subclass relation to expand concept queries when required. Note that although all system-registered ontologies can be considered as conceptual-level query mechanisms, the system can suggest suitable ontologies based on: first, the user's choice of resources and second, the sources' schema information. Database mediator systems can be used to provide uniform access to distributed heterogeneous data sets [15].

As outlined in our previous paper [31] WPIM can be applied by means of a three level mediator architecture to support DPP activities. In the following we are going to call this integrated support ***Knowledge-based Production Planning (KPP)***. KPP aims at combining and integrating distributed information and knowledge resources e.g., about product design, product features, machine and tool descriptions, machine features, and process constraints in order to create an executable plan for a certain component production task. Such activities can happen within the boundary of one organization or even across organizational boundaries. The CAPP-4-SMEs project [1] explicitly has defined and achieved the goal to research and support the field of CAPP e.g., for the use case ***where Original Equipment Manufacturers (OEMs)*** work with global partners and suppliers, which are mainly SMEs, more collaboratively to achieve entire manufacturing value chain optimization [5].

KNOWLEDGE-BASED PRODUCTION PLANNING

KPP as it has been developed within CAPP-4-SMEs is based on all the already outlined concepts, methods and technologies. It combines all the advantages to one distributed and collaborative approach for supporting production planning in a knowledge-based way by integrating production planning knowledge resources along process representations of the three levels of the planning process.

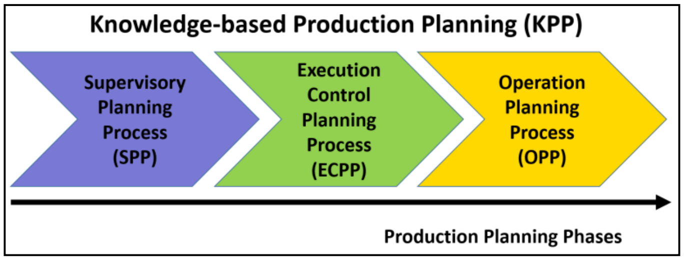

Figure 4: KPP Process Phases

Hence, it integrates the DPP planning process as well as the related resources with FBs and the semantic WPIM technology. The already presented steps of DPP were expanded and adapted to WPIM and turning **the supervisory plan** into an **execution control plan** and this one into an **operational plan** in an optimized manner. Therefore, as displayed in **Figure 4**, KPP also consists of a three-level process model. In the understanding of WPIM, the DPP planning process and resource knowledge is represented by planning activities consuming and producing planning knowledge resources. These can e.g., be FBs over all levels of CAPP activities from ***SP Process (SPP)*** activities through ***ECP Process (ECPP)*** activities to ***OP Process (OPP)*** activities. This is described in more detail in the related Paper [31].

The mediation process is also performed in a three-level MA. **Figure 5** displays the three level MA. The first mediator is called the ***SPP Mediator (SPPM)*** and integrates MFBs and other relevant and potentially distributed resources for the SPP activity. A down-stream DPP mediation can be implemented by means of two analogously derived additional mediators on the second and the third DPP level. On the second level of the MA follows then the deduced and so-called ***ECPP Mediator (ECPPM)*** which supports the above-mentioned ECPP activity. They assimilated at least an earlier iteration of the SPP-mediator as MFB and an OFB of the subsequent ***OPP Mediator (OPPM)*** (level 3) and various other relevant and potentially distributed resources. Coming from the machining-data point of view, the corresponding up-stream mediation process starts from machines with a defined need of steering information which can be harmonized by using wrappers and offering a mediated interface to clients. The third and final level of the MA of the KPP process forms the again derived ***OPPM*** and completes the mediation process. This integrates relevant and potentially distributed manufacturing knowledge resources as FBs and by the second level generated EFBs (ECPP Mediator) for the OPP activity.

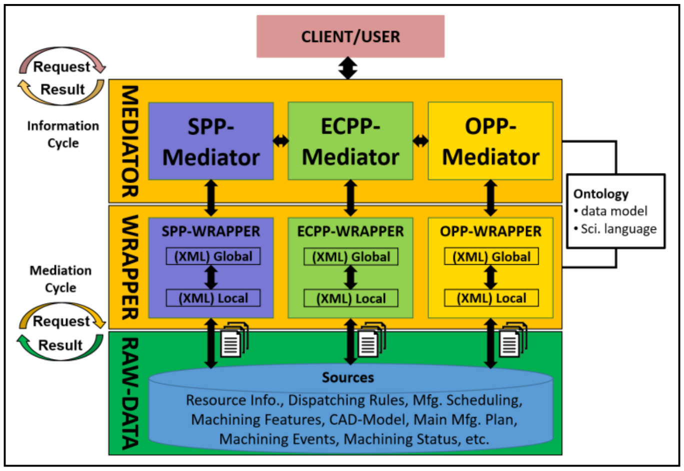

Figure 5: Conceptual Architecture of the KPP Mediator

This three-tier architecture can support an Information Process by, providing data from distributed data repositories, combining various data schemata and corresponding formats into a single semantic-enabled global schema and format. Moreover, it enables mediation process requesting, accessing, and collecting/gathering/combining data from different distributed manufacturing- and planning knowledge resources.

KPP MANUFACTURING KNOWLEDGE RESOURCES

The whole portfolio of different types of product data related to supporting KPP activities for component production is represented in XML language format to eliminate the communication barrier between different software, hardware, and specialized tools. For this purpose, the MA in KPP uses a wrapper technology with all common machine information and -codes encoded in XML by using e.g., the STEP standard as a domain model. Initially, based on the first level KPP MA (SPPM) and with access to a corresponding sample data set (Product Component CAD-Model (**Figure 7**) and Product Feature Extraction) from the field of manufacturing is utilized to demonstrate the functionality of a wrapper for **Product Design Information** and **Product Feature Information**. In the next step, the access to a typical **Main Manufacturing Plan** (see **Figure 6**) represented as MFB from a local relational database source is implemented by means of an appropriate wrapper and will be integrated and represented in the global representation schema as STEP compliant XML code. Through the normalization via the global schema, the parent mediator can get access to manufacturing information and may offer it to the client and vice versa. Furthermore, the following code sample (see **Figure 6**) from Y. Lu et al. [22] clarifies the representation of machining information that is integrated with the production plan information in the next step of the integration. This illustrated request represents the machining feature type, typical important parameters and requirements which are important for the production process.

```
<?xml version="1.0" encoding="utf-8"?>
<job name="Job 122612" description="manufacturing job request" material="P" owner="Yuqian Lu" id="122612">
  <serviceRequirements>
    <deliveryTime>none</deliveryTime>
    <costExpectation>none</costExpectation>
  </serviceRequriements>
  <machiningFeatures>
    <machiningFeature featureType="General Open Pocket" externalId="" unit="Metric" name="General Open Pocket">
      <parameters>
          <parameter name="Dp" description="Depth of pocket">20</parameter>
          <parameter name="Rm" description="Minimum radius in concave corner">3</parameter>
          <parameter name="Wo" description="Width of open area">12</parameter>
          <parameter name="Cs" description="Corner style">Corner Break</parameter>
          <parameter name="Cb" description="Corner break">1.6</parameter>
          <parameter name="Rg" description="Floor radius">0</parameter>
          <parameter name="Wc" description="Width of 45° chamfer">0</parameter>
          <parameter name="Aw" description="Angle of wall">90</parameter>
          <parameter name="Wi" description="Smallest width of the gap">0</parameter>
          <parameter name="Vp" description="Volume of the pocket">0</parameter>
          <parameter name="Qw" description="Quality of wall surface">N10</parameter>
          <parameter name="Unit" description="Unit of Tool">None</parameter>
          <parameter name="FType" description="Solid/Indexable Tool">None</parameter>
          <parameter name="SBType" description="Shank/Bore Type">None</parameter>
      </parameters>
      <toolAssemblies>
          <toolAssembly referenceId="" />
      </toolAssemblies>
      <designModels>
          <model format="STEP">.\Models\PocketModel.stp</model>
      <designModels>
    </machiningFeature>
  </machiningFeatures>
</job>
```

Figure 6: Typical Production Data in XML format [22]

All remaining product information, manufacturing steps, requirements, and the CAD-Model which are necessary to produce a product component are now processed in a similar form in the SPP without being based on a concrete machine or tool. For example, this includes the "feature-based design", "fixture information" and "machining technology and constraints". Now the mediator respectively the wrapper has to be implemented and allows the individual access via a web interface to all parameters of a product component. Out of this, it is not only possible to generate a product component file or a feasible machining plan of one single product component in the later KPP process, but rather individual modified product components as instances.

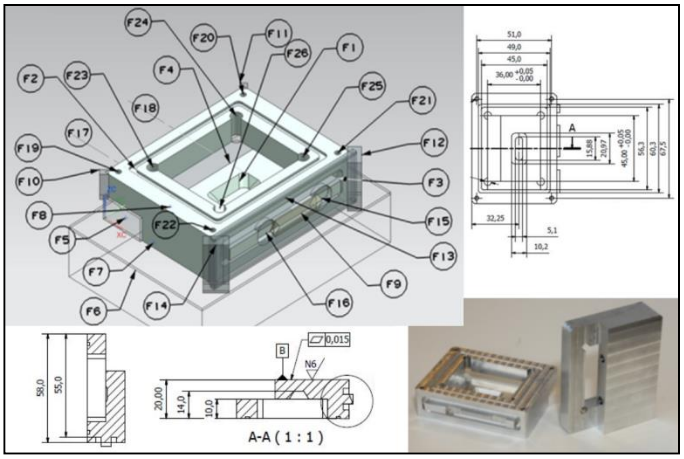

Figure 7: Exemplar Product Component Model [23] [24]

Givehchi et al. [23] [24] have introduced a method how a MFB containing product design and feature information could be processed in a DPP environment. They have shown that similar product component features can be summarized and grouped as a nested directed graph of generic setups. They show that on the basis of a simple product component example (see **Figure 7**) which is, e.g., produced from a block of aluminium raw material. For this purpose, all the important process steps and product component features were extracted from local data sources and summarized unsorted as already mentioned above.

KPP IDENTIFICATION WITHIN THE RPP USE CASE

The RPP is an end-to-end reference process that can be adapted to individual needs. Thus, it can be seen as a high level template for creating a concrete production planning process that takes individual company-specific and location-specific conditions into consideration. The following **Figure 8** displays that this process is divided into three maturity level-related phases: ***Concept Planning (CP)***, ***Rough Planning (RP)*** and ***Detailed Planning (DP)***.

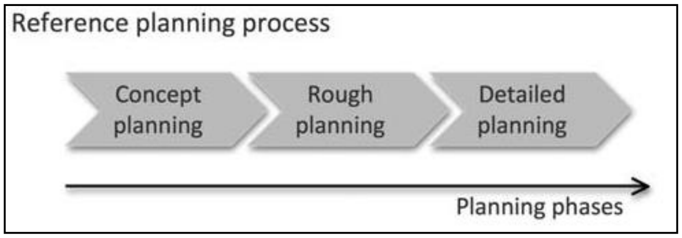

Figure 8: Reference Planning Process (RPP) by ProSTEP iViP [27]

CP identifies the cost-optimized production concept and attempts to integrate sometimes better and more innovative solutions produced by deductive approaches with time-saving, best practice-driven solutions generated by inductive approaches. In this way, the RP has been developed as a means to support planning scenarios for manufacturing and assembly planning and as a means to define and represent a certain level of automation for the systems responsible for the flow of material and information. The rough planning stage takes over the manufacturing planning-related results of the corresponding planning discipline from the concept phase and combines these if necessary with updated data or additional premises. DP developed a detailed planning scenario (manufacturing and assembly plan) and a work schedule suitable for series production.

There are different planning disciplines for each of these phases like, e.g., Manufacturing-, Assembly-, Logistics- or Layout Planning as the second organizational criterion for the structuring of the production planning process. In this way, the RPP covers not only the manufacturing planning but also the ALLP scenarios for which Manufacturing Planning based on KPP can be considered as a starting point. This means, that for the (Component) Manufacturing Planning, the KPP process can be considered as a reference implementation of the RPP in the Manufacturing dimension.

KPP SUPPORT FOR MCM

The already mentioned MCM settles between the planning phase and the actual production phase of a product (seen **Figure 9).** Furthermore, it responds to possible changes arising only during the set-up and pretesting of the planned production phase such as, e.g., the deviation of the current state of the real system from the current state of planning. In this sense, it is a downstream but also interdigitated and collaborative process of the RPP.

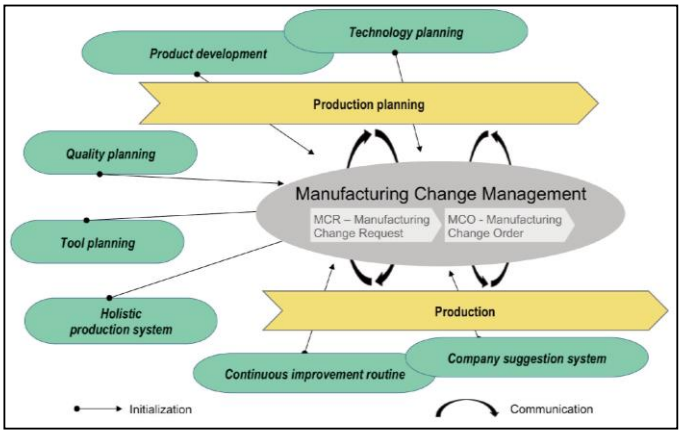

Figure 9: MCM Model by ProSTEP iViP [28]

The aims of MCM are the long-term production efficiency relies on a well-coordinated interplay between planning improvements and ongoing optimizations, the optimization of sustainable implementation of planned production processes, and the identification and minimization of discrepancies between the planning and production process and the improvement of the synchronization between production planning and production.

In this way the MCM process can be considered as a consumer of a production plan provided by KPP. At the same time, it can be considered as a producer of optimization input for production planning. This can be enabled by means of passing on production plans represented in the KPP method to the MCM process and feeding back production plans with optimization annotations from the MCM process into the KPP process. This occurs in order to update representations of Master Process that have been used as a basis for the KPP-based production plan.

CONCLUSION AND FUTURE WORK

This paper has presented the relevant State of the Art and Related Work as well as an innovative method to support KPP in the manufacturing domain based on semantic process representations producing and consuming function blocks and other relevant planning resources for manufacturing planning.

On the basis of several sample data sets, we will also demonstrate in our future work the query, search and representation for information as well as the functionality of the KPP wrapper technology for the domain of manufacturing planning. Thus, also the complete three step mediator architecture will be demonstrated. Furthermore, we will evaluate the KPP method by means of integrating it into the overall RPP process. This includes applications of KPP to Assembly, Logistic and Layout Planning as well as feeding KPP planning results into MCM infrastructures and consuming MCM-produced manufacturing change requests by means of the KPP infrastructure as a method to optimized production process templates. Therefore, future production planning activities will be enabled to take advantages of lessons learned during set-up and pre-testing as a means of optimization of production planning.

ACKNOWLEDGEMENTS AND DISCLAIMER

This publication has been produced in the context of the CAPP-4-SMEs project.

The CAPP-4-SMEs project has received funding from the European Union's Seventh Framework Program for research, technological development and demonstration under grant agreement no 314024.

However, this paper reflects only the author's view and the European Commission is not responsible for any use that may be made of the information it contains;

REFERENCES

[1] F. Miltner, T. Vogel and M. Hemmje; "Towards Knowledge Based Process Planning Support for CAPP-4-SMEs: Problem Description, Relevant State of the Art and Proposed Approach", vol. 1, International Manufacturing Science and Engineering Conference (MSEC), 2014.

[2] T. Vogel, "Wissensbasiertes und Prozessorientiertes Innovationsmanagement WPIM - Innovationsszenarien, Anforderungen, Modell und Methode,Implementierung und Evaluierung anhand der Innovationsfähigkeit fertigender Unternehmen", Dissertation, Hagen, 2012.

[3] L. Wang, G. Adamson and M. H. a. P. Moore, "A Review of Function Blocks for Process Planning and Control of Manufacturing Equipment", Journal of Manufacturing Systems, Vol.31, No.3, pp.269-279, 2012.

[4] L. Wang, W. Jin and H. Y. Feng, "Embedding machining features in function blocks for distributed process planning," *International Journal of Computer Integrated Manufacturing*, pp. 443-452, 2006.

[5] L. Wang, H. Y. Feng and N. Cai, "Architecture design for distributed process planning," *Journal of Manufacturing Systems*, pp. 99-115, 2003.

[6] M. Helgoson, L. Wang, R. Karlsson, M. Givehchi and M. Tedeborg, "Concept for Function Block enabled Process Planning towards multi-site Cloud Collaboration", International Manufacturing Science and Engineering Conference (MSEC), Vol. 1, 2014.

[7] International Electrotechnical Commission, Switzerland: Function blocks – Part 1: Architecture, IEC 61499-1, 2005.

[8] R. Lewis, "Modelling control systems using IEC 61499 – applying function blocks to distributed systems", ISBN: 0852976 796: The Institution of Electrical Engineers, 2001.

[9] International Electrotechnical Commission, Switzerland: Programmable controllers – Part 3: Programming languages, IEC 61131-3, 2003.

[10] T. Vogel and M. Hemmje, "Auf dem Weg zu einem Wissens-basierten und Prozess-orientierten Innovationsmanagement (WPIM) – Innovationsszenarien, Anforderungen und Modellbildung," in *KnowTech 2006*, Poing, CMP-WEKA-Verlag, 2006.

[11] R. Cyganiak, D. Wood, M. Lanthaler, G. Klyne, J. Carroll and B. McBride, "RDF 1.1 Concepts and Abstract Syntax," W3C Recommendation 25 February 2014, World Wide Web Consortium (W3C), http://www.w3.org/TR/rdf11-concepts/,, Feb 2014, last accessed Nov 13, 2014.

[12] W3C OWL Working Group, "OWL 2 Web Ontology Language Document Overview (Second Edition)," W3C Recommendation 11 December 2012, World Wide Web Consortium (W3C), http://www.w3.org/TR/owl2-overview/, December 2012, last accessed Nov 13, 2014.

[13] W3C, "OWL Web Ontology Language Overview,," World Wide Web Consortium,, 10 February 2004. [Online]. Available: http://www.w3.org/TR/owl-features/. [Accessed 14 November 2013].

[14] G. Wiederhold, "Mediators in the Architecture of Future Information Systems", The IEEE Computer Magazine, 1992.

[15] B. Ludäscher, K. Lin, B. Brodaric and C. Baru, "GEON: Toward a Cyberinfrastructure for the Geosciences—A Prototype for Geologic Map Integration via Domain Ontologies", Digital Mapping Techniques ’03 — Workshop Proceedings, U.S. Geological Survey Open-File Report 03–471, 2003.

[16] T. Bray, J. Paoli, C. M. Sperberg-McQueen, E. Maler and F. Yergeau, "Extensible Markup Language (XML) 1.0 (Fifth Edition), W3C Recommendation 26 November 2008, World Wide Web Consortium (W3C)," http://www.w3.org/TR/REC-xml/, November 2008.

[17] S. Gao, C. M. Sperberg-McQueen, H. S. Thompson, N. Mendelsohn, D. Beech and M. Maloney, ""W3C XML Schema Definition Language (XSD) 1.1 Part 1: Structures", W3C Recommendation 5 April 2012, World Wide Web Consortium (W3C), http://www.w3.org/TR/xmlschema11-1/," last accessed Nov 13, 2014, 5 April 2012.

[18] J. Melton, "ISO/IEC FDIS 9075-1 Information technology - Database languages - SQL - Part 1: Framework (SQL/Framework), ISO Draft International Standard, ISO/IEC JTC 1/SC 32 Data Management and Interchange," http://www.jtc1sc32.org/doc/N2151-2200/32N2153T-text_for_ballot-FDIS_9075-1.pdf, last accessed Nov 13, 2014, August 2011.

[19] J. Robie, D. Chamberlin, M. Dyck and J. Snelson, "XQuery 3.0: An XML Query Language," W3C Recommendation 08 April 2014, World Wide Web Consortium (W3C), http://www.w3.org/TR/xquery-30/, April 2014.

[20] B. Motik, B. Cuenca Grau, I. Horrocks, Z. Wu, A. Fokoue and C. Lutz, "OWL 2 Web Ontology Language Profiles (Second Edition)," W3C Recommendation 11 December 2012, World Wide Web Consortium (W3C), http://www.w3.org/TR/owl2-profiles/, last accessed Nov 13, 2014, December 2012.

[21] ISO International Standard 14649-1:2003. Industrial automation systems and integration -- Physical device control -- Data model for computerized numerical controllers -- Part 1: Overview and fundamental principles. Geneva: International Organization for Standardization. Retrieved 2008-10-27.

[22] Lu, Yuqian und Xu, Xun. PROCESS AND PRODUCTION PLANNING IN A CLOUD MANUFACTURING ENVIRONMENT. Charlotte, North Carolina, USA : ASME 2015 International Manufacturing Science and Engineering Conference, 2015. MSEC2015-9382.

[23] Givehchi, Mohammad, Haghighi, Azadeh und Wang, Lihui. PaperGeneric machining process sequencing through a revised enrichedmachining feature concept. s.l. : Journal of Manufacturing Systems, 2015.

[24] Givehchi, Mohammad, Schmidth, Bernard und Wang, Lihui. Knowledge-based operation planning and machine control by Function Blocks in Web-DPP. Porto, Portugal : Flexible Automation and Intelligent Manufacturing (FAIM), 2013.

[25] X. Xu, L. Wang and Y. Rong. “STEP-NC and Function Blocks for Interoperable Manufacturing” IEEE Transactions on Automation Science and Engineering. Vol. 3, no. 3, pp 297-308. July 2006, Digital Object Identifier 10.1109/TASE.2005

[26] X Xu, H Wang, J Mao, S T Newman, T R Kramer, F M Proctor and J L Michaloski. “STEP—Compliant NC Research: the Search for Intelligent CAD/CAPP/CAM/CNC Integration”. International Journal of Production Research. Volume 43 no. 17, 1 September 2005, pp 3703 – 3743

[27] RECOMMENDATION - Reference process for production planning PSI8, ProSTEP iVIP, March 2013 http://www.prostep.org/en/medialibrary/publications/recommendations-standards.html

[28] RECOMMENDATION - Management of Changes during Production PSI12, ProSTEP iVIP, May 2015. http://www.prostep.org/en/medialibrary/publications/recommendations-standards.html

[29] ProSTEP iViP Association e.V., https://www.prostep.org

[30] ISO International Standard DIS 18828-2:2015. Industrial automation systems and integration -- Standardized procedures for production systems engineering - Part 2: Reference process for seamless production planning International Organization for Standardization, Geneva, Switzerland (2015).

[31] B. Gernhardt, T. Vogel, M. Givehchi, L. Wang and M. Hemmje; „Supporting Production Planning through Semantic Mediation of Processing Functionality”, Vol. 1, International Conference on Innovative Design and Manufacturing (ICIDM), 2015, Auckland, New Zealand