

Knowledge representation for product and processes development planning in collaborative environments

Lorenzo Solano ^{1*}, Pedro Rosado ², Fernando Romero ³

^{1,2} *Departamento de Ingeniería Mecánica y de Materiales, Universitat Politècnica de València, Camino de Vera s/n, 46022 Valencia, Spain.*

³ *Departamento de Ingeniería de Sistemas Industriales y Diseño, Universitat Jaume I, Avda. Vicente Sos Baynat, 12071 Castellón, Spain*

* Corresponding author at: Departamento de Ingeniería Mecánica y de Materiales, Universitat Politècnica de València, Camino de Vera s/n, 46022 Valencia, Spain. Tel.: +34963877000 Ext. Int.: 76273; fax: +34963877629. E-mail address:

lsolano@mcm.upv.es

² E-mail address: prosado@mcm.upv.es Tel.: +34963877000 Ext. Int.: 76221; fax: +34963877629

³ E-mail address: fromero@esid.uji.es Tel.: +34964728209; fax: +34964728170

Knowledge representation for product and processes development planning in collaborative environments

Efficiency in the management of integrated product and processes development is a basic requisite to guarantee competitiveness and success for manufacturing companies. This means that operational management of activities, and human and material resources is extremely important, especially in virtual OKP (One-of-a-Kind Production) systems, and must cover related aspects of their capabilities and social character as well as assignment criteria. In this context, and to facilitate collaborative resources management, an ontology focused on resources and capabilities is proposed in this work. This ontology supports the necessary knowledge for generic and collaborative process planning, providing a shared common semantic for all the members of the virtual company. This work differs from other proposed ontologies in the area of process planning in that the resources considered are all those elements that participate in the execution of the different activity types involved in this wide and complex process. The ontology directly covers the shared, social nature of the resources, the agentive behavior of many of them and a characterization of their capabilities, thus providing specific solutions to the needs of the Collaborative Integrated Development of Products, Processes and Resources (CIDP²R) process.

Key words: Resource Capabilities, Product and Process Development, virtual OKP, Engineering Ontologies

1. Introduction

The survival of companies in the current business panorama, characterized by globalization, individualization and personalization of demand, and strong competition, depends largely on establishing flexible, adaptable, and agile organization, focusing on customer satisfaction, cooperation, learning and knowledge management and a culture

of change (Sherehiy, Karwowski, and Layer 2007) which is particularly appropriate for the OKP systems, at which this work is directed. The OKP systems include practices aimed at successful product development and manufacture in one go (Li, Xie, and Xu 2011), and involve increased agility through distributed control and the use of extremely dynamic organizational structures designed for virtual companies, and which have led to the emergence of what is known as virtual OKP companies (Xie and Tu 2011). A key element for success and for the intended agility of these company networks is the collaboration between members which must take place in an environment of exchange and sharing of knowledge and information which allows them to work in harmony and coordinate their efforts to achieve their shared objectives, adopting a process approach.

With this new approach, the company has a wide variety of process (core, support and management) going on concurrently (Mili, Tremblay, and Jaoude 2010), which need to be planned, establishing the structure of work, the activities and operations sequence and the types of resources allocated. The resources, often shared, include people, systems and information, among others. The management, assignment and configuration of these resources for each of the necessary activities will establish, to a great degree, the effectiveness of each and every one of the processes taking place.

One of the most relevant core processes in manufacturing companies in the field of engineering is New Product Development (NPD), which includes manufacturing processes. In the context of a virtual OKP company, NPD must involve a high degree of collaboration. This is what various authors have identified as Concurrent New Product Development (Büyüközkan, Dereli, and Baykasoglu 2004), Rapid Product Development process (Romero, Estruch, and Rosado 2009; Xie and Tu 2011) or Collaborative Product Development (Büyüközkan and Arsenyan 2012). These proposals, in accordance with Allied Concurrent Engineering (Chen, Shir, and Shen 2002), agree on

the need to establish a process-oriented cycle of product development, involving a high degree of integration and collaboration, enabled by a management model based on projects and supported by ICTs (Information and Communication Technologies). In recent years, to facilitate collaboration and integration in distributed environments companies have been increasingly using ontologies that allow the integration of knowledge and semantic interoperability (Poli, Healy, and Kameas 2010; Lin et al. 2011). Along these lines, both the more general proposals for explicit foundational ontologies and those defined for the different domains of product development can be found (Cheung et al. 2006; Baxter et al. 2007). It is also important to mention recent developments in enterprise ontologies which have come from a wide range of sources (O'Leary 2010). From among these it is important to mention TOVE (TOronto Virtual Enterprise project) and EO (Enterprise Ontology), as ontologies of multiple domain, or PSL (Process Specification Language) which specializes in process sub-domain. However, none of these deals directly with resource capabilities, because they have usually been centered directly on temporal aspects of the resources in a particular process. Other recent works (Rosado and Romero 2009) have shown the need to integrate the different ontologies from the areas of product, processes and resources as a basis for collaborative process planning of the Product and Processes Development process.

In response to these current needs, in this work an ontological model to represent the Resource Capabilities in the Product and Processes Development process (PPDRC) is presented. This specific ontology is one of the basic pillars of the integrated ontology that supports the Collaborative Integrated Development of Products, Processes and Resources (CIDP²R) process and provides the necessary knowledge to carry out dynamic and optimum assignment of resources for the activities of this process (design,

manufacturing, process planning, scheduling, execution coordination and control, execution, etc.), in a Service Oriented Architecture (SOA) and for any level of aggregation.

At this point it is important to note that process planning activities do not only include those defining the manufacturing process but also those defining the Product and Processes Development process itself. To achieve this goal, in the first place, a framework in which to place the key concepts of the ontology is established. Later, the entities and the predicates that constitute the core of a generic ontology to represent capabilities for the Product and Processes Development process are identified, so that all needs can be met. Finally, the implementation of the ontology is carried out in order to prove the validity of the proposal.

2. State of the art

Currently, in the domain of design and manufacturing engineering, there is a great deal of interest in knowledge management and the integration of people and systems, and ontologies are being used to develop these areas (Lin et al. 2011).

The definition of ontology given by Guarino in Uschold and Gruninger (1996) states that “an ontology is a formal description of the entities within a given domain, the properties they possess, the relationships they participate in, the constraints they are subject to, and the patterns of behavior they exhibit”. According to Borgo and Leitao (2008), ontologies can be very general (foundational ontologies) or domain dependent (core ontologies). Bräuer (2007) states that core ontologies, domain ontologies and application ontologies can be distinguished among the domain dependent ontologies. Foundational ontologies, such as DOLCE (Descriptive Ontology for Linguistic and Cognitive Engineering), create a framework for the development of ontologies, which allow a more careful definition of parts of the world, an interaction between data in

different formats (Shadbolt, Hall, and Berners-Lee 2006) and the mutual understanding and interoperability between people and machines (Poli, Healy, and Kameas 2010).

An important characteristic of any ontology is its capacity to conceptualize the reality that is being presented. This ability depends on the ontological foundations adopted. Thus, *endurantism* treats entities as objects that pass through time and are totally present at any given moment, whereas *perdurantism* assumes that entities are extended in time and space (Oberle et al. 2007). Moreover, an ontology adopts a *revisionist* position when it aims to capture the intrinsic nature of the world, while a *descriptive* approach covers the concepts based on human cognition, natural language and common sense (Bräuer 2007). Another basic ontological foundation is related to the ontology's ability to represent the entities social character, while the *functionalism* of the conceptualization indicates the capacity to express the functional considerations of the world's entities (Lambert et al. 2008). Finally, from the range of the ontological choices, *actualism* only deals with what is real (such as the execution of the activities and resource participation in the activity occurrences over time), in contrast to *possibilism*, which considers what is possible (Masolo et al. 2003), such as the assessment of resources according to their ability to participate in the activities.

The previous ontological foundations must be studied in order to consider the concepts that must be present in the PPDRC ontology. However, to establish these concepts it is also necessary to take a close look at the subject matter, which in our case is the process planning of particularly complex processes, such as a CIDP²R process.

IDEF3 was one of the first methodologies used to model processes, offering a language for its graphic representation and formalizing the necessary concepts and relationships to describe processes. Many of these concepts and relationships were later included in the PSL (Process Specification Language) ontology (ISO 2004b). In PSL, as

with IDEF3, a process is a collection of activities with the restrictions that govern their relationships. The PSL ontology provides four basic concepts to describe processes and their execution (*activity*, *activity_occurrence*, *timepoint* and *object*). In PSL, an *activity* is a behavior or action that can be repeated, while an *activity_occurrence* is each of the occurrences or executions of an *activity*. The *timepoints* represent the moments in time usually linked to the beginning and end of the execution of an activity. Finally, an *object* is any entity that does not fall into any of the previous three categories (Gruninger 2009). Some ontologies for specific processes have been developed as extensions of PSL (Deshayes, El Beqqali, and Bouras 2005; Lohse et al. 2005). PSL has also served as a foundational layer for the ontological proposal of Chungoora and Young (2011).

Activity Theory (AT) is another contribution to the area of processes which covers some considerations not included in PSL, especially those related to the social character of activities. According to AT, *activities* are made up of *actions*, which in turn are made up of *operations*, whose objectives are respectively: *motive*, *goal* and *condition* (Karanasios et al. 2011). Participation in a motivated activity (with motive) means the development of an action with conscious control and an immediate objective. The concept of conscious control in AT is coherent with the BDI paradigm, in which the participants have Beliefs, Desires and Intentions (Ferrario and Oltramari 2004, Tamma et al. 2005).

PSL, in a similar way to the Resource ontology of TOVE, introduces the concept of resource as any object that is needed during the execution of an activity. Another approach to the concept of resource, from the area of manufacturing planning and execution, is the one introduced in MANDATE (ISO 2004a) which considers a resource to be any device, tool and means (including human) at the disposal of the enterprise to

produce goods or services. MANDATE focuses on resource management, and considers the two roles that the various elements can play in operations: operation objects and operation means. Moreover, MANDATE incorporates two key concepts for the characterization and use of resources: a) capability, the quality of being able to execute an activity, e.g., a group of characteristics that describes functional aspects of manufacturing resources and b) capacity, the capability of a system (or sub-system) or resource to perform its expected function, in particular in terms of amount of production.

In the literature, other ontologies that have introduced the resource concept can be found. ADACOR (Borgo and Leitaó 2008) is a core ontology covering manufacturing scheduling and control operations, and making use of the resource concept as any entity that can execute a range of operations as long as its capacity is not exceeded. Similar considerations for resources are found in the work of Lin, Harding, and Shahbaz (2004) intended to provide a manufacturing system engineering ontology for extended environments. Other ontological developments described in Tamma et al. (2005) and Rajsiri et al. (2008) reveal the importance of resources for collaboration. In these works, resources are understood to be the social use of objects in coordination and collaboration environments.

Finally, in the e-manufacturing and collaborative supply chains domains, there are some works which demonstrate the need to discover manufacturing resources and services in an efficient way to improve the agility of the manufacturing process (Jang et al. 2008; Ameri and Dutta 2008; Cai et al. 2010). With this objective in mind, ontological models based on OWL are used. However, here the proposals are focused on certain types of resources, and which describe their capabilities without considering the

agentive nature that many of them possess, as is the case with people or other intelligent agents, which are fundamental in knowledge intensive collaborative processes.

3. Ontology objectives and requirements

One of the first steps in the development of an ontology is to clearly define its scope and domain. This scope and domain enable the establishment of the ontology requirements, and help to determine the ontological foundations necessary to represent the concepts and the property and data relations to be dealt with. Following this, the requirements imposed on the ontology are addressed according to their objectives.

Process planning can include recursively other process planning operations. For instance, process planning of product development includes the process planning of: project management; technical processes and support processes. At the same time, process planning of project management includes the planning of the following project activities: project planning; project scheduling and project execution coordination and control. In turn, process planning of technical processes includes the planning of technical activities such as the following: product specifications; product development and product manufacturing. Therefore, the overall scope of PPDR ontology is much greater than those corresponding to manufacturing process planning.

The basic competence of the PPDR ontology is to model and make explicitly clear the knowledge necessary for the assignment of resources (or types of resources) to activities (or types of activities) involved in process planning in a collaborative environment. This includes the structuring of the activities required, the allocation of resources to each of them based on their capabilities and taking into account the collaborative environment, where resources may have a behavior influenced by their social character and guided by their interests. For this reason, the social character consideration of the participants in the activities and their occurrences is needed. It is

also necessary to consider the intrinsic agentic nature of certain elements that establish the control and development of the execution of activities, fixing their start and finish, orientation, strategy and objective (Lin, Harding, and Shahbaz 2004; Gangemi et al. 2005; Izhar et al. 2013). This agentic character is maintained on a cognitive level requiring a BDI approach, where all the resources that take part in activities and guide their development are agentic, and are associated with one or more objectives that set their intentions.

The required level of detail for modeling of manufacturing resource capabilities may be greater than for modeling of other resources capabilities involved in the Product and Processes Development process. For these reasons, it is necessary that the ontology to be developed provides responses to questions with a different degree of generality, referring to specific aspects or to generic questions in the domain such as: a) What are the capabilities of the resources? b) How is the process/activity defined and structured? c) What is the implication of the resources in the process/activity? d) What is the objective of the process/activity?. The responses to these questions must incorporate a certain level of knowledge beyond the simple recovery of information that has been previously stored. Rather, it is enhanced know-how obtained from relating and/or processing this information. Another requirement for the PPDR ontology is to represent all the entities of interest for the assignment of resources to activities in a collaborative and distributed context with different levels of detail. The activities involved are notably complex due to their intrinsic nature and includes activities that are developed on a physical or material level, such as manufacturing operations; and others that are developed on a mental, intellectual or cognitive level, such as a scheduling or execution, coordination and control tasks (Ferrario and Oltramari 2004; Tamma et al. 2005).

Finally, the ontology must consider complete traceability, temporal evolution and the relationship between information and data and the entities and circumstances in which they originate.

4. A resource capability ontology for product and process development

The PPDRC ontology is based on the following fundamental concepts: activities, activity_occurrences, workflows, objects, resources, capabilities, agentives, roles and objectives, which are needed in the Product and Processes Development process. This proposal satisfies the requirements expressed in the previous section and has come as a result of studying the state of the art.

4.1. Fundamental concepts of the domain

Activities represent anything that can be carried out (Kethers 2000; Lin, Harding, and Shahbaz 2004) and constitute behavior that can be repeated. This abstract concept of an activity, as a repeatable pattern, is materialized whenever an activity is executed in time and is denominated occurrence. Activity occurrences or occurrences use and produce or transform objects. In PPDRC ontology, workflows are a type of complex occurrence with a common objective shared by one or more agents, which coordinate in order to achieve this objective (Tamma et al. 2005), and which are coherent with the use of PSL for activity flow modeling as proposed by Bock and Gruninger (2005). Contrary to the approach found in PSL, in which objects are defined by exclusion with respect to the other three first level entities of the ontology (*activity*, *activity occurrence* and *timepoint*) (Qiao, Kao, and Zhang 2011), in the PPDRC ontology, an object is a tangible or intangible entity that does exist (Solano, Romero and Rosado 2010). Among these objects are social objects, or shared descriptions which allow mutual understanding

between the members of a community (Ferrario and Oltramari 2004), as is the case with the activity dealt with previously.

A resource is the description of how an object participates in the execution of an activity. As established in MANDATE, resources are characterized by the degree of qualitative or quantitative performance referred to in the concept of capability. The resource can participate in an activity occurrence according to different roles which can be grouped into four types: control; input; output and mechanism (NIST 1993). If these concepts are placed in the more general framework of a foundational ontology such as DOLCE, whose high level ontological categories are: endurant; perdurant; quality and abstract (Gangemi et al. 2005), the object and the occurrence of PPDRC ontology are covered by the concepts of endurant and perdurant respectively, which along with the concept of region (which serves to express qualities of any concept) form the basic semantics of the PPDRC ontology.

4.2. Entities and predicates of the PPDRC ontology

Taking into account the concepts mentioned previously, the first level of the PPDRC ontology is structured with three types of mutually exclusive entities: *Object*; *ActivityOccurrence* and *Region* (Fig. 1). The predicate *formedBy* (entity, entity) can be applied to any of these entities to express relationships of composition between entities of the same type: between objects; between occurrences and between regions. The predicate *formedBy* provides the mechanism to structure the entities of the ontology at different levels of detail or aggregation, and expresses transitive relationships that are limited to identifying the entities that constitute another entity, without specifying the type of relationship between them.

The next three subsections present the main entities of Object, ActivityOccurrence and Region type, and some of the predicates that are established between these entities.

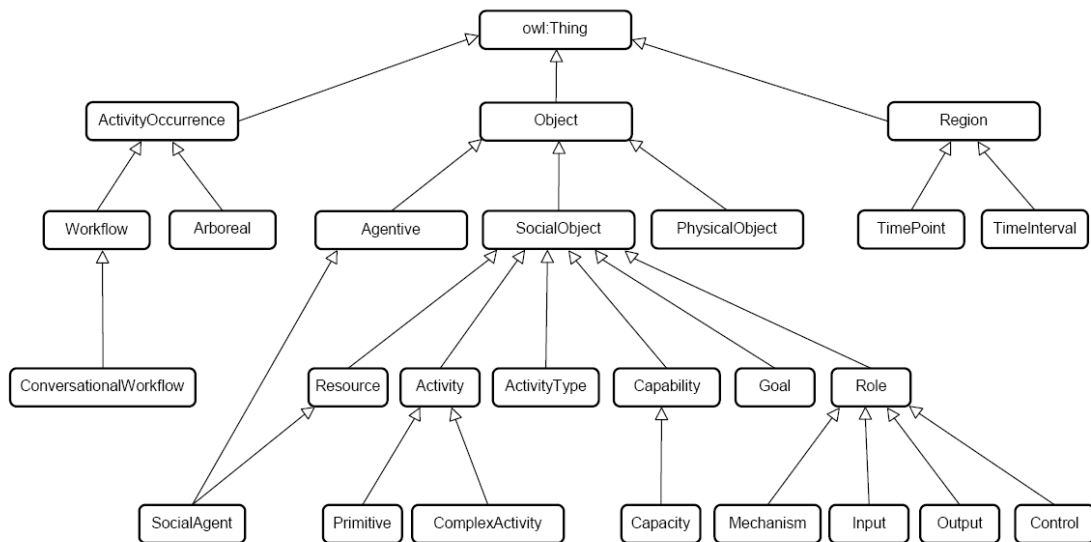


Figure 1. Taxonomy of the PPDRC ontology.

4.2.1. Objects and their predicates

The entity *Object* has three specializations: *PhysicalObject*, *SocialObject* and *Agentive*. The entity *PhysicalObject* represents things that have a physical existence, such as machines, IT systems, people, etc. The entity *SocialObject* corresponds to the concept of social object mentioned previously. Finally, the entity *Agentive* corresponds to those objects, such as people, who have beliefs, desires and intentions (BDI) and whose behavior is driven by motivation. An object can belong to several of the three last categories as these are not exclusive. In this sense, it is important to highlight the entity *SocialAgent* which belongs simultaneously to the entities *Agentive* and *SocialObject*. For example, a person is a *SocialAgent*, that is to say, a person is a social object who has an agentive character.

In the context of the PPDRC ontology, the social objects considered are: *Activity*; *ActivityType*; *Capability*; *Resource*; *Role* and *Goal*, which are all mutually exclusive, but which do not use up all the possibilities of the social objects. An *Activity* is the entity that describes the activity concept defined previously, while an *ActivityType* is the social object that represents the abstraction of a group or type of activities. *Capability* is the shared definition that characterizes the use of an object in a type of activity, expressing certain competences to execute activities of this type and, when necessary, the level of performance achieved in their execution. For example, *developing the mechanical assembly of a gear pump* is an activity type, while the competence to develop mechanical assemblies is a capability. Capabilities are related to resources through the predicate *characterizedBy* (*Resource*, *Capability*), while the predicate *executing* (*Capability*, *ActivityType*) fulfills its meaning as an entity necessarily linked to the execution of a type of activity. *Capacity* is a *Capability* expressed in terms of amount of production. For example, the capacity of a resource, such as a grinding machine, can express its rate of production (in mm^3 / min) machining flat surfaces with a determined level of roughness.

Any object that has a relationship with an individual of the entity *Capability* belongs to the entity *Resource*. Therefore, all the individuals of the entity *Resource* have the competence or ability to realize at least one type of activity. The *Capability* of a resource refers to the carrying out of a certain type of activity. For example, a person who can carry out the activity *molded piece design* is a resource. In short, any of the three categories of objects shown above (*Agentive*, *SocialObject* and *PhysicalObject*) can be a *Resource*.

The entity *Role* is a social object that is identified by the manifestation of certain capabilities in the occurrence of an activity (*ActivityOccurrence*). Resources are the

only objects that can have roles. While a capability is linked to a type of activity, an individual of the entity *Role* establishes the type of participation of a resource in a concrete activity occurrence through the predicate *behaves (Role, Capability)*. To complete its specification, the predicates *hasRole (Resource, Role)* and *isPresent (Role, Arboreal)* show the participation of a resource in the occurrence of a primitive activity, an occurrence that belongs to the entity *Arboreal* (an *ActivityOccurrence* type). A resource can participate in the same activity with different roles according to the capabilities that it exhibits on each occasion. For example, a resource that has the ability to direct, perhaps does not participate as director in the occurrence of an activity. The entity *Role* has four specializations (*Control, Input, Output* and *Mechanism*) whose individuals can only relate through *isPresent* relationships with occurrences of *Arboreal* type.

Finally, the description of the entity *Goal* represents the definition of the objective. The entities of the type *Agentive* are the only ones who can have objectives, which is expressed through the predicate *hasGoal (Agentive, Goal)*. The fulfillment of their goals is the guide to the initiatives of the agentives during the execution of activities.

4.2.2. ActivityOccurrences and their predicates

The second entity of the first level of the PPDR ontology is *ActivityOccurrence*, which, as can be seen in Fig. 2, is associated with other entities through different predicates. The relationships of their individuals establish the characteristics of an execution, for example, the object that carries out an activity, the time points in which it takes place, its restrictions and relationships with other occurrences, start and sequencing mechanisms, partial milestones and the state achieved after it is carried out (Tamma et al. 2005). The predicate *occurrenceOf (ActivityOccurrence, Activity)*

establishes the relationship between the occurrence of an activity and the activity itself. The predicate identified in Fig. 2 as *PSL_relations* represents the set of predicates that are similar to those of the PSL ontology (*after*, *before*, *earlier*, *precedes*, *hasLeaf_occ*, *hasRoot_occ*, *min_precedes*, *next_subocc*, *root*, *root_occ*, *leaf*, *leaf_occ*, *poss*, *sucessor*, etc.) which specify the relationships of sequence and precedence between the occurrences of the activities that form a process. As with PSL, the PPDRC ontology distinguishes between primitive *Activities (Primitives)* and complex activities (*ComplexActivities*). *ActivityOccurrences* of the *Arboreal* type are those *ActivityOccurrences* of primitive activities from which precedence relationships are established.

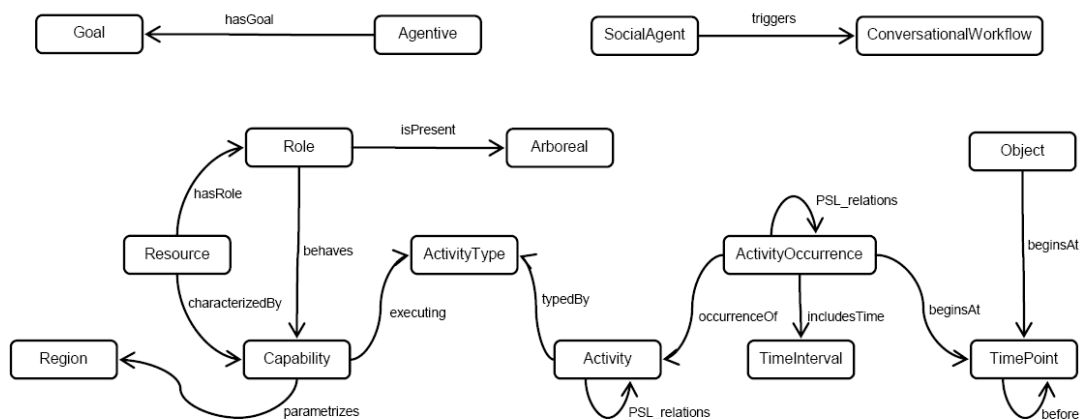


Figure 2. Predicates of the PPDRC ontology.

The *Workflow* is a type of complex *ActivityOccurrence* in which it is necessary to show its configuration. In other words, in the occurrence of a *Workflow* type activity the structure and sequence of its temporal parts must be specified. For Kethers (2000), the concepts of *action workflow* and *declaration workflow* support respectively the bilateral conversations for the development of an activity between the customer and the supplier of a process, and the bilateral conversations necessary for the development of a negotiation (Fig. 3). The entity *ConversationalWorkflow* of the PPDRC ontology groups

together these two types of *workflow*. *ConversationalWorkflow* is a type of execution of a complex activity, in which each of its stages can contain another *workflow*. *ConversationalWorkflow* is a specialization of *Workflow* whose execution can only be initiated by an individual of the entity *SocialAgent* through the predicate *triggers*. In fact, *SocialAgent* is an entity characterized by the *Capability* to trigger and manage *ConversationalWorkflows*.

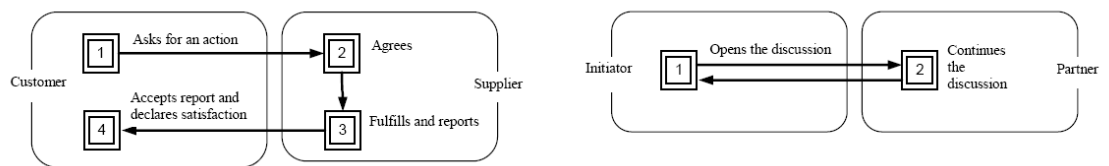


Figure 3. Action workflow and declaration workflow.

4.2.3. Regions and other predicates

The third of the top level entities of the ontology is *Region*, which represents the space in which the values (*quales*) corresponding to a quality are found. A *quale* is a particular value from the allowed values for quality. *TimeInterval* and *TimePoint* are regions that represent respectively intervals and points in time, and are inherent in the activity occurrences and in the existence of the objects. The predicate *parametrizes* (*Capability*, *Region*) expresses the relationship between a capability and its *quales*, which identifies the values or attributes of the capabilities. For example: $0,15 \mu m$ is the value of the characteristic *average roughness* (*Ra*) for the capability to obtain flat surfaces; *high complexity* is the value of the characteristic *part complexity* associated with the capability to plan the manufacturing process. All types of capabilities can be modeled with any degree of detail, and can even be quantified. However, the high degree of detail or the quantification normally required in capabilities of production means, such as *obtain flat surfaces*, is not necessarily relevant in other capabilities such

as *design parts* or *plan the manufacturing process*.

As well as the predicates already discussed, the ontology has others which serve to express the relationships between the objects (endurants) or activity occurrences (perdurants) and time: an endurant exists in time and can genuinely change over time, while a perdurant exists in time but cannot change over time. The PPDRC ontology is able to represent both the temporal space in which the occurrence of an activity is developed or in which an object exists, as well as the changes in the object over time. The first is expressed through the predicates *beginsAt* and *endsAt* (*Object or ActivityOccurrence, TimePoint*), which establish the relationship between the occurrence of an activity or the existence of an object and its *quales* of temporal region. For the second of these cases, the predicate *version* (*Object, Object*) is used, which in a similar way to MANDATE (ISO 2005), associates the different versions of the same object. The predicate *version* has various specializations, such as *upgrade* and *planned*. *Upgrade* allows reference to an object stating its new version, while *planned* associates it with its planned future state. For example, the capabilities of a resource can be updated through *upgrade*.

4.3. Implementation

In response to the requirement of supporting a Web-based OKP system, it was decided to model the ontology using a standardized and widespread language. Therefore, the modeling of the PPDRC ontology was carried out through OWL/SWRL, which are widely used within globally extended manufacturing teams, because they enhance the semantic interoperability and reuse of knowledge resources (Cai et al. 2010; Lin et al. 2011). SWRL (Semantic Web Rule Language) improves the semantic expressivity of OWL (Ontology Web Language) as it allows the expression of knowledge that cannot

be directly defined with the OWL axioms (Lin 2008).

The PPDRC ontology has been developed and edited using the ontology editor Protégé and can be consulted at (PPDRC 2013). Additionally, to check the consistency between predicates and definitions of the ontology developed, maintain the hierarchy of the ontology and allow consultation, the reasoner Pellet was used. Pellet is an open code reasoner which allows the classification and reasoning with individuals in OWL/SWRL ontologies.

In order to validate the proposal, a Java application has been implemented. The application incorporates Java libraries to operate with OWL files (OWL2 API) and to reason and query with Pellet (Pellet OWL2 API). This Application Programme Interface (API) incorporates a query engine that provides SPARQL support. Additionally, the application incorporates a front-end interaction interface that translates the user query into SPARQL and transforms the SPARQL query results into a more user readable format.

5. An illustrative example

This case study shows the process planning of a Collaborative New Product Development process in a virtual OKP Enterprise, emphasizing resources and their capabilities. All involved activities (management, technical and support) can be carried out at different levels of aggregation, defining and analyzing the activities, identifying the capabilities required for these, consulting the capabilities of the resources available and assigning the resources to concrete occurrences. For the sake of simplicity, the example only shows some of the activities necessary, and does not aim to describe an optimum management of the project. Its objective is to show how the PPDRC ontology allows the management, in a distributed form and with different levels of aggregation, of the information on resources that is necessary in New Product Development process

at different levels. The use of PPDR ontology is of special interest in process planning, scheduling and coordination and execution control corresponding to the project management.

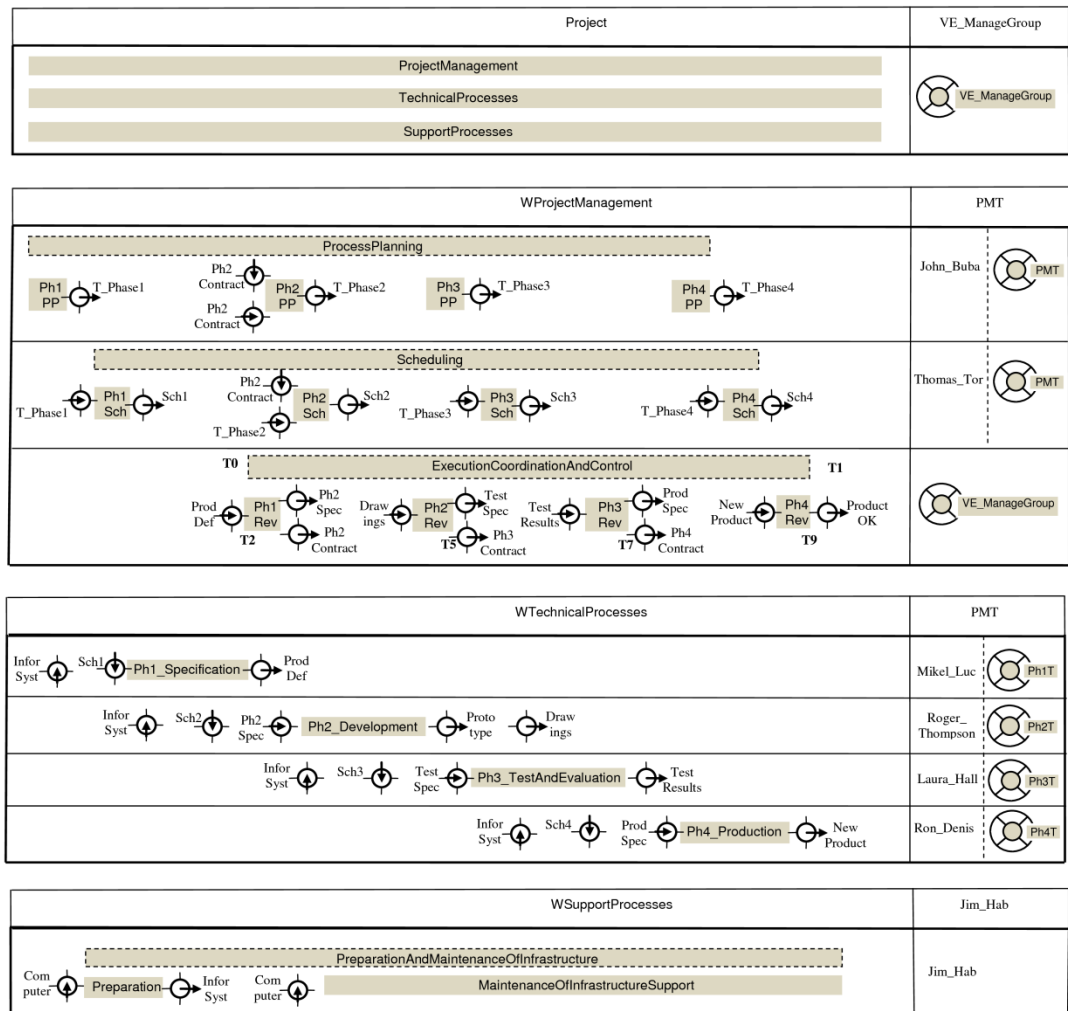


Figure 4. Execution of the complex activity Project.

An overview of the process at the Project level is given in Fig. 4, showing the activities of the technical processes (specification, development, test and evaluation and production phases) and management and operative support activities directly linked to the technical activities. In the illustration, the occurrences that make up the global occurrence Project are represented with shaded horizontal bands. If they are complex,

the band is outlined with a discontinuous line and without outline if they are primitive. The column on the right identifies the people or organizational units, who participate in occurrences with a mechanism role. Some of the teams included in the figure are: VE_ManageGroup (Virtual_Enterprise_Manage_Group), PMT (Project_Manager_Team), Ph1T (Phase1_Team), etc. As well as the human resources with a mechanism role, there are other resources that are necessary to carry out the activities which are represented together with the occurrences in which they participate using the icons in Fig. 5.

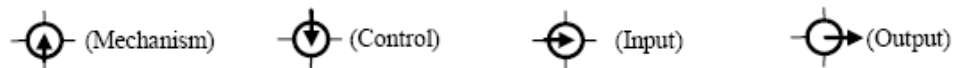


Figure 5. Symbols used to represent the roles of resources.

5.1. Project level

The occurrence Project is broken down into three occurrences (Fig. 4): ProjectManagement, TechnicalProcesses and SupportProcesses. These occurrences establish the responsibilities and conditions under which the action workflows (WProjectManagement, WTechnicalProcesses and WSupportProcesses) are triggered. In this case, the VE_ManageGroup delegates the execution of the first two workflows to the PMT team, and the third to Jim_Hab, maintaining overall responsibility for the coordination and execution control at the project level following the project management models proposed by McGrath (2004). In turn, the PMT team delegates to John_Buba the responsibility for planning the technical process and to Thomas_Tor the project scheduling. Finally, the management of technical phases is delegated to other members of the PMT, who are supported by multi-disciplinary groups created for each

phase.

The use of PPDR ontology during the course of the project begins with the process planning stage at Project Management, as an aid in defining: technical activities (phases) and support activities; the structure of deliverables (input and output resources); the roles of human resources and work teams; and the time and resource dependencies. Later, in the scheduling occurrences of Project Management, the ontology is used to assign the resources that participate according to previous roles established in the process planning, and to set other restrictions relative to execution times. The ontology must also respond to the needs of those who participate in the coordination and execution control of phases as well as their execution (WTechnicalProcesses) and support for the phases (WSupportProcesses). This last group provides support for the rest of those involved and facilitates inter-department and/or inter-company collaborative work.

The queries in this section were formulated in an application developed for this purpose, which uses the SPARQL language to interrogate the ontology. In the first query, Table 1 shows the query in SPARQL, its interpretation in natural language and the result. In the rest of the queries only the interpretation and the results are shown.

Table 1. Query 1.

Query 1	Natural language interpretation	Answer
Query in SPARQL SELECT ?Capability WHERE { ?Capability <http://www.coapp.es/ontologies/2011/0/PPDR_v1.owl#capabilityRequiredBy> ?Occurrence . ?Occurrence a (<http://www.coapp.es/ontologies/2011/0/PPDR_v1.owl#subactivityOccurrenceOf> value <http://www.coapp.es/ontologies/2011/3/Ejemplo10.owl#WTechnicalProcesses>) . }	What capabilities are required to execute the occurrences that are subactivity_occurrences of WTechnicalProcesses?	Capability ManageWorkflow C_Preliminary_Develop C_Detailed_Develop C_Product_Assembly_Develop C_Product_Elect_Develop C_Develop C_Product_Mech_Develop ManageWorkflow C_Specify ManageWorkflow C_Produce ManageWorkflow C_Evaluate

5.1.1. Process planning scenario

In this scenario (Fig. 6), the ontology must support the planner in the definition of the Project occurrences, which is carried out by refining or modifying Project model templates, defining, at different levels of aggregation, a structure of activities (WBS - Work Break Structure-).

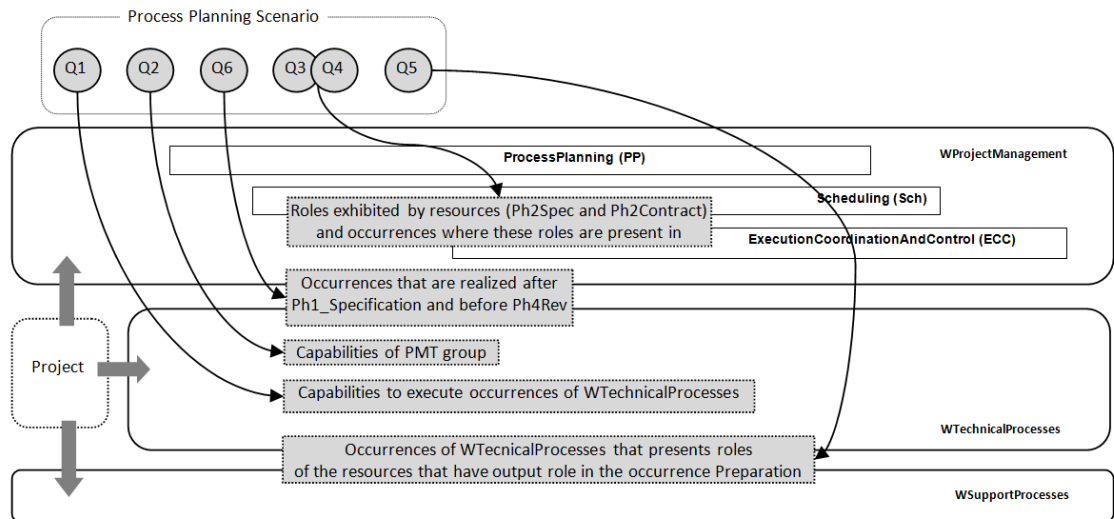


Figure 6. Examples of queries in the process planning scenario corresponding to the planning of phases of the project.

According to adopted template, the planner could consult the required capabilities for the types of activities present in WTechnicalProcesses (Table 1). Later, in order to check the suitability of the PMT group to which the WTechnicalProcesses occurrence is assigned, the planner can consult the combined capabilities of its members (Table 2).

Table 2. Query 2.

Query 2	
Natural language interpretation	Answer
Which capabilities characterize the resources that form the PMT group?	Capability C_Execute C_Evaluate C_Specify C_Preliminary_Develop C_Schedule C_Coordinate C_Control C_Coordinate_Execution C_Develop C_Plan C_Test

Results of queries can be derived from facts or assertions, or can come as the result of a process of inference. For example, query 1 results show an inferred relationship between some occurrences and the necessary capabilities by means of rule 19 (Table 3). Similarly, rule 12 allows inference that a resource is characterized by some capability, if that capability has appeared in any role played by the resource.

Table 3. Rules of inference written in SWRL.

	Rules
1	TimePoint(?x) , TimePoint(?y) , quale(?x, ?u) , quale(?y, ?v) , lessThan(?u, ?v) , differentFrom(?x, ?y) -> before(?x, ?y)
2	earlier(?x, ?y) -> differentFrom(?x, ?y)
3	after(?x, ?y) , before(?x, ?y) -> sameAs(?x, ?y)
4	Atomic(?act1) , Atomic(?act2) , Input(?i1) , Input(?i2) , beginsAt(?act1, ?t1) , beginsAt(?act2, ?t2) , hasRole(?r, ?i1) , hasRole(?r, ?i2) , isPresent(?i1, ?act1) , isPresent(?i2, ?act2) -> sameAs(?t1, ?t2)
5	subactivities(?x, ?y) , subactivities(?y, ?x) -> sameAs(?x, ?y)
6	behaves(?r, ?cap) , isPresent(?r, ?occ) , occurrenceOf(?occ, ?a) , typedBy(?a, ?at) -> executing(?cap, ?at)
7	beginsAt(?x, ?in) , endsAt(?x, ?fin) -> before(?in, ?fin)
8	beginsAt(?sub, ?t) , root_occ(?sub, ?oc) -> beginsAt(?oc, ?t)
9	endsAt(?sub, ?t) , leaf_occ(?sub, ?oc) -> endsAt(?oc, ?t)
10	beginsAt(?y, ?ini) , earlier(?x, ?y) , endsAt(?x, ?fin) , differentFrom(?fin, ?ini) -> before(?fin, ?ini)
11	beginsAt(?x, ?in) , endsAt(?x, ?fin) -> differentFrom(?in, ?fin)
12	Resource(?x) , behaves(?y, ?z) , hasRole(?x, ?y) -> characterizedBy(?x, ?z)
13	Atomic(?a1) , Atomic(?a2) , Input(?rol) , Output(?sal) , hasRole(?r, ?rol) , hasRole(?r, ?sal) , isPresent(?rol, ?act2) , isPresent(?sal, ?act1) , occurrenceOf(?act1, ?a1) , occurrenceOf(?act2, ?a2) -> earlier(?act1, ?act2)
14	Atomic(?a1) , Atomic(?a2) , Input(?i) , Mechanism(?m) , hasRole(?r, ?i) , hasRole(?r, ?m) , isPresent(?i, ?act1) , isPresent(?m, ?act2) , occurrenceOf(?act1, ?a1) , occurrenceOf(?act2, ?a2) -> earlier(?act2, ?act1)
15	Control(?c) , Input(?i) , beginsAt(?act1, ?t1) , beginsAt(?act2, ?t2) , hasRole(?r, ?c) , hasRole(?r, ?i) , isPresent(?c, ?act2) , isPresent(?i, ?act1) -> before(?t2, ?t1)
16	Control(?c) , Output(?o) , endsAt(?act1, ?t1) , endsAt(?act2, ?t2) , hasRole(?r, ?c) , hasRole(?r, ?o) , isPresent(?c, ?act2) , isPresent(?o, ?act1) -> before(?t1, ?t2)
17	Atomic(?a1) , Atomic(?a2) , Mechanism(?mec) , Output(?o) , hasRole(?r, ?mec) , hasRole(?r, ?o) , isPresent(?mec, ?act2) , isPresent(?o, ?act1) , occurrenceOf(?act1, ?a1) , occurrenceOf(?act2, ?a2) -> earlier(?act1, ?act2)
18	Atomic(?x) , Atomic(?y) , subactivityOf(?x, ?y) , differentFrom(?x, ?y) -> ConcurrentSuperpositionAtomic(?y)
19	executing(?cap, ?at) , occurrenceOf(?occ, ?a) , typedBy(?a, ?at) -> requiresCapability(?occ, ?cap)

In this same planning scenario, queries can also be raised about the deliverables. For example, the planner raises queries 3 and 4 referring to the roles fulfilled by the deliverables Ph2Spec and Ph2Contract (Fig. 6).

The planner may be interested in other types of resources. This is the case of Query 5, which raises a question about the WTechnicalProcesses occurrences which use the IT resources obtained in the occurrence Preparation (Fig. 6).

As well as the queries related to resources, roles and capabilities, the planner could be interested in the sequence restrictions. This is the case of Query 6 (Table 4), which involves checking the occurrences to be carried out between Ph1_Specification and Ph4Rev. The result of this query, as well as the sequence restrictions declared by the process planner, incorporates facts arising from rules 13 to 17 of Table 3. These rules introduce sequence dependencies among the primitive occurrences arising from the roles present in them. For example, as the output generated in Ph1_Specification (ProdDef) is the input required in Ph1Rev, it is inferred that the beginning of Ph1Rev will take place after the completion of Ph1_Specification (rule 13). This knowledge inference has special relevance in process planning in collaborative environments, where the planner role may be shared or distributed, meaning that different planners can consult or enter facts to the ontology.

Table 4. Query 6.

Query 6	
<i>Natural language interpretation</i>	<i>Answer</i>
Which occurrences are realized after Ph1_Specification and before Ph4Rev?	Occurrence Ph3Rev Ph3Sch Ph2_Development Ph2Sch Ph3PP Ph3_TestAndEvaluation Ph2PP Ph4PP Ph2Rev Ph4Sch Ph1Rev Ph4_Production

5.1.2. Scheduling scenario

In the scheduling scenario (Fig. 7), the scheduler carries out the assignment of resources with a mechanism role and establishes the values of the *timepoints* which define the beginning or end of certain occurrences and may create additional precedence relationships to those fixed previously in the process planning scenario. As with the planner role, the scheduler role can also be shared and/or distributed. In order to assign resources the scheduler can query (Table 5) the ontology about the people who bring together the necessary capabilities for any occurrence such as Development (Query 7) or about the resources that have a particular capability, such as C_Product_Elect_Develop (Query 8). Other similar queries are: Queries 9, 10 and 11 (Fig. 7).

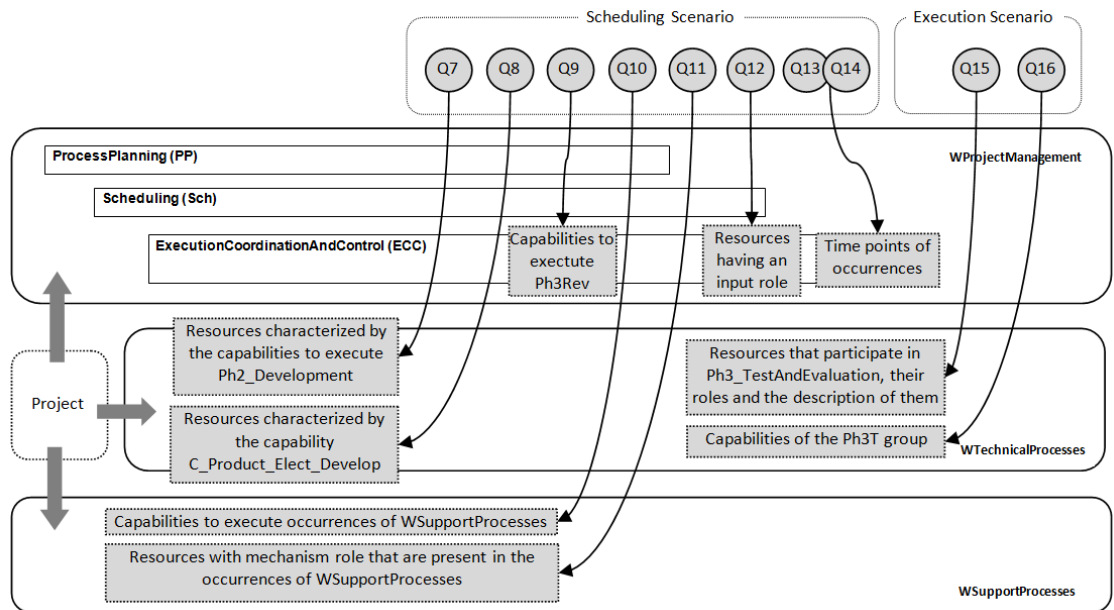


Figure 7. Examples of queries in the scheduling and execution scenarios during the scheduling of the project and during the execution and control of the project.

Table 5. Queries 7 and 8.

Query 7	
<i>Natural language interpretation</i>	<i>Answer</i>
Which resources are characterized by the capabilities required to execute the occurrence Ph2_Development?	Resource Capability Mikel_Luc C_Preliminary_Develop Stp1T C_Preliminary_Develop Tom_Lee C_Detailed_Develop Stp2T C_Detailed_Develop Frank_Cisco C_Product_Assembly_Develop Tsk3T C_Product_Assembly_Develop VE_ManageGroup ManageWorkflow Tsk2T C_Product_Elect_Develop Ana_Lupfer C_Product_Elect_Develop Roger_Thompson C_Develop Ph2T C_Develop Nick_Nolte C_Product_Mech_Develop Tsk1T C_Product_Mech_Develop
Query 8	
Which resources are characterized by the capability C_Product_Elect_Develop?	Resource Tsk2T Ana_Lupfer

In the PPDRC ontology, the resources can only be assigned to primitive occurrences, as the roles only admit `isPresent` relationships with occurrences of the *Arboreal* type. However, queries may be formulated about the resources participating in complex occurrences. For example, in Query 12 (Fig. 7) the ontology is consulted about resources with an input role that participate in the complex occurrence `ExecutionCoordinationAndControl`.

As was shown previously, the scheduler, as part of his work, may require information on the start and finish timepoints of a complex occurrence such as `ExecutionCoordinationAndControl`, and for this, he will raise Query 13 (Table 6). In this case, the result of the query is obtained from the beginning timepoint of the occurrence `Ph1Rev` (T2) and end timepoint of the occurrence `Ph4Rev` (T9) and by inferring that the beginning and end timepoints of the occurrence `ExecutionCoordinationAndControl` (T0 and T1) match with T2 and T9 respectively, as can be seen in Fig. 4.

Table 6. Query 13.

Query 13	
<i>Natural language interpretation</i>	<i>Answer</i>
At what time point does the occurrence ExecutionCoordinationAndControl begin, and at what time point does the occurrence ExecutionCoordinationAndControl end?	Beginning End 20 430

As with the timepoints T2 and T9, T5 and T7 were fixed by the planner as the timepoints for completion of the occurrences Ph2Rev and Ph3Rev respectively. From this information (Query 14 in Table 7), the scheduler can check the duration established for phase Ph3_TestAndEvaluation with the time interval between T5 and T7. After checking this information, the scheduler can consider different options, such as: simultaneously develop phase Ph3_TestAndEvaluation with other phases, increasing the resources assigned to this phase or modifying the time points T5 or T7 to avoid overlapping of occurrences.

Table 7. Query 14.

Query 14	
<i>Natural language interpretation</i>	<i>Answer</i>
At what time point does the occurrence Ph2_Rev end and at what time point does the occurrence Ph3Rev end?	EndPh2Rev EndPh3Rev 220 260

5.1.3. Execution scenario

During the execution of an activity, any of the people involved in it can raise queries related to the needs of the execution of these activities (Fig. 7). For example, Laura_Hall, who is responsible for the execution of Ph3_TestAndEvaluation can raise queries such as those in 15 and 16 (Table 8), in which the ontology is consulted about a range of aspects related to the execution of the occurrence.

Table 8. Queries 15 and 16.

Query 15	
<i>Natural language interpretation</i>	<i>Answer</i>
Which resources participate in the occurrence Ph3_TestAndEvaluation, what roles do those resources have and what is the description of those roles?	Resource Role Description TestResults Out16 Product evaluation results Sch3 Ctrl8 Schedule for Phase3 InforSyst Mech1 Execution support TestSpec Inp10 Product evaluation specification Laura_Hall Mech6 Person responsible for evaluation of product Ph3T Mech7 Execution of product evaluation
Query 16	
Which capabilities do the members of Ph3T group have?	Capability C_Evaluate C_Coordinate C_Test C_Coordinate C_Produce

5.2. Phase and step level

The workflows triggered at the project level produce the launch of other workflows at a phase level in which the primitive occurrences that make them up, the resources involved, or other related factors are shown. For example, Development is the occurrence of a primitive activity (Fig. 4), for which the person responsible (Roger_Thompson) triggers a complex workflow which brings about the execution of two occurrences: Preliminary and Detailed (Fig. 8). In this workflow, Roger_Thompson assigns the occurrence Phase2_WDevelopment to the team Ph2T, who delegate the process planning to Rose_Hall and the scheduling to Pitt_Moor, leaving the coordination and execution control under the responsibility of Roger_Thompson himself. The management of the occurrences Preliminary and Detailed is assigned to members of Ph2T who are supported by work teams set up for these occurrences. The rest of the resources that participate in Phase2_WDevelopment have also been represented in Fig. 8, together with the occurrence in which they participate and with an indication of their mechanism, control, input and output roles. Any of the people involved in the corresponding workflow for a phase can raise queries similar to those shown throughout this section. In a similar way, as workflow decomposition is recursive, each step of a workflow can give rise to a new workflow. In the new

workflow, participants can also consult the ontology according to their responsibilities and the scenarios that correspond to their level, which are similar to the scenarios of superior levels.

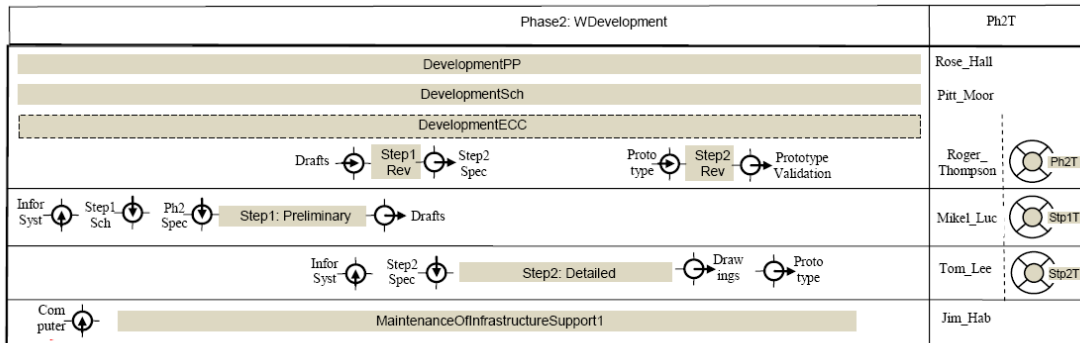


Figure 8. Workflow WDevelopment.

6. Conclusions and future lines of research

To achieve a successful result in enterprise core processes, it is vital to share information and knowledge, both in engineering activities and operational management activities. This requirement takes on a special relevance in structuring, resources assignment, scheduling, control and monitoring of all engineering activities involved in Product and Processes Development process and when these are executed in distributed and collaborative environments. In order to satisfy, this, and taking into account the diversity and complexity of these activities (co-design, co-development and co-manufacturing), the presented ontology (PPDRC) integrates concepts belonging to different ontological theories from the domain of the company and its processes (as is the case of the theory of “situation calculus”, which provides part of the semantics to PSL and TOVE, or Activity Theory) and general frameworks for the development of ontologies (DOLCE). Particular attention was paid to the incorporation of the concepts necessary to represent the social and agentive character of the resources, a fundamental aspect in collaborative processes.

In order to demonstrate the validity and general applicability of the proposal, a tool was developed that allows consultation of the ontology with queries about aspects related to planning, scheduling and control of engineering processes. In the example, the tool was applied to a New Product Development process in a virtual OKP environment, which allowed verification of whether it was able to respond satisfactorily according to the roles of the different participants (planner, scheduler, etc.).

Currently the particularization of capabilities for machining resources according to the requirements established by the CIDP²R process is being accomplished. Future lines of research are concerned with developing collaborative and expert computer aided tools that interact with the ontology through web services and represent this ontology in a foundational ontology, for later integration with other application domains.

Acknowledgments

This work has been possible thanks to funding received from the Ministry of Science and Education through the COAPP Research Project - reference DPI2007-66871-C02-01/02.

References

- Ameri, F., and D. Dutta. 2008. "A Matchmaking Methodology for Supply Chain Deployment in Distributed Manufacturing Environments." *Journal of Computing and Information Science in Engineering* 8 (1): 1-9.
- Baxter, D., J. Gao, K. Case, J. Harding, B. Young, S. Cochrane, and S. Dani. 2007. "An Engineering Design Knowledge Reuse Methodology Using Process Modeling." *Research in Engineering Design* 18 (1): 37-48. doi: 10.1007/s00163-007-0028-8.
- Bock, C., and M. Gruninger. 2005. "PSL: A Semantic Domain for Flow Models." *Journal of Software and Systems Modeling* 4 (2): 209-231. doi: 10.1007/s10270-004-0066-x.
- Borgo, S., and P. Leitaó. 2008. "Foundations for a Core Ontology of Manufacturing." In *Ontologies: a Handbook of Principles, Concepts and Applications in*

- Information Systems*, edited by Sharman, R., R. Kishore, and R. Ramesh, 751-775. Springer.
- Bräuer, M. 2007. "Design of a Semantic Connector Model for Composition of Metamodels in the Context of Software Variability." PhD diss., Technische Universität Dresden.
- Büyüközkan, G., T. Dereli, and A. Baykasoglu. 2004. "A Survey on the Methods and Tools of Concurrent New Product Development and Agile Manufacturing." *Journal of Intelligent Manufacturing* 15 (6): 731-751.
- Büyüközkan, G., and J. Arsenyan. 2012. "Collaborative Product Development: a Literature Overview." *Production Planning & Control* 23 (1): 47-66. doi: 10.1080/09537287.2010.543169.
- Cai, M., W. Y. Zhang, G. Chen, K. Zhang, and S. T. Li. 2010. "SWMRD: a Semantic Web-based Manufacturing Resource Discovery System for Cross-Enterprise Collaboration." *International Journal of Production Research* 48 (12): 3445-3460. doi: 10.1080/00207540902814330.
- Chen, Y. M., W. S. Shir, and C. Y. Shen. 2002. "Distributed Engineering Change Management for Allied Concurrent Engineering." *International Journal of Computer Integrated Manufacturing* 15 (2): 127-151. doi: 10.1080/09511920110047181.
- Cheung, W. M., D. G. Bramall, P. G. Maropoulos, J. X. Gao, and H. Aziz. 2006. "Organizational Knowledge Encapsulation and Re-Use in Collaborative Product Development." *International Journal of Computer Integrated Manufacturing* 19 (7): 736-750. doi: 10.1080/09511920500504479.
- Chungoora, N., and R. I. M. Young. 2011. "The Configuration of Design and Manufacture Knowledge Models from a Heavyweight Ontological Foundation." *International Journal of Production Research* 49 (15): 4701-4725. doi: 10.1080/00207543.2010.504754.
- Deshayes, L. M., O. El Beqqali, and A. Bouras. 2005. "The Use of Process Specification Language for Cutting Processes." *International Journal of Product Development* 2 (3): 236-253.
- Ferrario, R., and A. Oltramari. 2004. "Towards a Computational Ontology of Mind." Paper presented at the 3rd International Conference FOIS (Formal Ontology in Information Systems), Turin, November 4-6.

- Gangemi, A., S. Borgo, C. Catenacci, and J. Lehmann. 2005. *Task Taxonomies for Knowledge Content. Deliverable D07 of the EU FP6 project Metokis*. Laboratory for Applied Ontology ISTC CNR.
- Gruninger, M. 2009. "Using the PSL Ontology." In *Handbook on Ontologies*, edited by Staab, S., and R. Studer, 423-443. Springer Berlin Heidelberg.
- ISO (International Organization for Standardization). 2004a. *Industrial Automation Systems and Integration – Industrial Manufacturing Management Data. Part 1, General Overview*. ISO 15531–1. New York: American National Standards Institute.
- ISO (International Organization for Standardization). 2004b. *Industrial Automation System and Integration – Process Specification Language. Part 1, Overview and Basic Principles*. ISO 18629–1. New York: American National Standards Institute.
- ISO (International Organization for Standardization). 2005. *Industrial Automation System and Integration – Industrial Manufacturing Management Data: Resources Usage Management. Part 32, Conceptual Model for Resources Usage Management Data*. ISO 15531–32. New York: American National Standards Institute.
- Izhar, T. A. T., T. Torabi, M. I. Bhatti, and F. Liu. 2013. "Recent Developments in the Organization Goals Conformance Using Ontology." *Expert Systems with Applications* 40 (10): 4252-4267. doi: 10.1016/j.eswa.2013.01.025.
- Jang, J., B. Jeong, B. Kulvatunyou, J. Chang, and H. Cho. 2008. "Discovering and Integrating Distributed Manufacturing Services with Semantic Manufacturing Capability Profiles." *International Journal of Computer Integrated Manufacturing* 21 (6): 631-646. doi: 10.1080/09511920701350920.
- Karanasios, S., J. Mishra, D. Allen, A. Norman, D. Thakker, and L. Lau. 2011. "Capturing Real-World Activity: A Socio-Technical Approach." Paper presented at the eChallenges e-2011 Conference.
- Kethers, S. 2000. "Multi-Perspective Modeling and Analysis of Cooperation Processes." PhD diss., RWTH Aachen.
- Lambert, D., A. Saulwick, C. Nowak, M. Oxenhan, and D. O'Dea. 2008. *An Overview of Conceptual Frameworks. DSTO-TR-2163*. Command, Control,

- Communications and Intelligence Division, Defence Science and Technology Organisation, Department of Defence, Australian Government.
- Li, B. M., S. Q. Xie, and X. Xu. 2011. "Recent Development of Knowledge-Based Systems, Methods and Tools for One-of-a-Kind Production." *Knowledge-Based Systems* 24 (7): 1108-1119. doi: 10.1016/j.knosys.2011.05.005.
- Lin, Y. 2008. "Semantic Annotation for Process Models: Facilitating Process Knowledge Management via Semantic Interoperability." PhD diss., Norwegian University of Science and Technology.
- Lin, H. K., J. A. Harding, and M. Shahbaz. 2004. "Manufacturing System Engineering Ontology for Semantic Interoperability Across Extended Project Teams." *International Journal of Production Research* 42 (24): 5099-5118. doi: 10.1080/00207540412331281999.
- Lin, L. F., W. Y. Zhang, Y. C. Lou, C. Y. Chu, and M. Cai. 2011. "Developing Manufacturing Ontologies for Knowledge Reuse in Distributed Manufacturing Environment." *International Journal of Production Research* 49 (2): 343-359. doi: 10.1080/00207540903349021.
- Lohse, N., H. Hirani, S. Ratchev, and M. Turitto. 2005. "An Ontology for the Definition and Validation of Assembly Processes for Evolvable Assembly Systems." Paper presented at the 6th IEEE International Symposium on Assembly and Task Planning: From Nano to Macro Assembly and Manufacturing, Montreal, July 19-21.
- Masolo, C., S. Borgo, A. Gangemi, N. Guarino, and A. Oltramari. 2003. *WonderWeb Deliverable D18. Ontology Library (final)*. Laboratory for Applied Ontology ISTC CNR.
- McGrath, M. E. 2004. *The Next Generation Product Development. How to Increase Productivity, Cut Costs, and Reduce Cycle Times*. McGraw-Hill.
- Mili, H., G. Tremblay, and G. B. Jaoude. 2010. "Business Process Modeling Languages: Sorting Through the Alphabet Soup." *ACM Computing Surveys* 43 (1): 4. doi: 10.1145/1824795.1824799.
- NIST (National Institute for Standards and Technology). 1993. *Draft Federal Information Processing Standards Publication 183. Announcing the Standard for Integration Definition for Function Modeling (IDEF0)*.
- Oberle, D., A. Ankolekar, P. Hitzler, P. Cimiano, M. Sintek, M. Kiesel, B. Mougouie, et al. 2007. "DOLCE ergo SUMO: On Foundational and Domain Models in the

- SmartWeb Integrated Ontology (SWIntO).” *Web Semantics: Science, Services and Agents on the World Wide Web* 5 (3): 156-174.
- O’Leary, D. E. 2010. “Enterprise Ontologies: Review and an Activity Theory Approach.” *International Journal of Accounting Information Systems* 11 (4): 336–352. doi:10.1016/j.accinf.2010.09.006.
- Poli, R., M. Healy, and A. Kameas, eds. 2010. *Theory and Applications of Ontology: Computer Applications*. Springer.
- PPDRC ontology. 2013. Accessed June
http://www.coapp.es/ontologies/2013/0/PPDRC_v1.owl.
- Qiao, L., S. Kao, and Y. Zhang. 2011. “Manufacturing Process Modelling Using Process Specification Language.” *International Journal of Advanced Manufacturing Technology* 55 (5-8): 549-563. doi: 10.1007/s00170-010-3115-3.
- Rajsiri, V., J. P. Lorré, F. Bénaben, and H. Pingaud. 2008. “Collaborative Process Definition Using an Ontology-Based Approach.” In *Pervasive Collaborative Networks*, edited by Camarinha-Matos, L. M., and W. Picard, 205-212. Boston: Springer.
- Romero, F., A. Estruch, and P. Rosado. 2009. “A Framework for the Development of an IT Platform for Collaborative and Integrated Development of Product, Process and Resources.” Paper presented at the 13th International Research/Expert Conference. Trends in the Development of Machinery and Associated Technology, Hammamet, Tunisia, October 16-21.
- Rosado, P., and F. Romero. 2009. “A Model for Collaborative Process Planning in a Engineering and Production Network.” Paper presented at the 13th International Research/Expert Conference. Trends in the Development of Machinery and Associated Technology, Hammamet, Tunisia, October 16-21.
- Shadbolt, N., W. Hall, and T. Berners-Lee. 2006. “The Semantic Web Revisited.” *IEEE Intelligent Systems* 21 (3): 96-101.
- Sherehiy, B., W. Karwowski, and J. K. Layer. 2007. “A Review of Enterprise Agility: Concepts, Frameworks, and Attributes.” *International Journal of Industrial Ergonomics* 37 (5): 445-460. doi: 10.1016/j.ergon.2007.01.007.
- Solano, L., F. Romero, and P. Rosado. 2010. “An Ontological Approach for Manufacturing Resources Modeling.” Paper presented at the 21st International DAAAM Symposium, Zadar, Croatia, October 20-23.

- Tamma, V., C. Aart, T. Moyaux, S. Paurobally, B. Lithgow-Smith, and M. Wooldridge. 2005. "An Ontological Framework for Dynamic Coordination." Paper presented at the 4th International Semantic Web Conference, Galway, Ireland, November 6-10.
- Uschold, M., and M. Gruninger. 1996. "Ontologies: Principles, Methods, and Applications." *Knowledge Engineering Review* 11 (2): 93-136.
- Xie, S., and Y. Tu. 2011. *Rapid One-of-a Kind Product Development. Strategies, Algorithms and Tools*. Springer.