

Document downloaded from:

<http://hdl.handle.net/10251/78092>

This paper must be cited as:

Solano García, L.; Romero Subirón, F.; Rosado Castellano, P. (2016). An ontology for integrated machining and inspection process planning focusing on resource capabilities. *International Journal of Computer Integrated Manufacturing*. 29(1):1-15.
doi:10.1080/0951192X.2014.1003149.



The final publication is available at

<http://dx.doi.org/10.1080/0951192X.2014.1003149>

Copyright Taylor & Francis

Additional Information

An ontology for integrated machining and inspection process planning focusing on resource capabilities

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

^{1,3} *Departamento de Ingeniería Mecánica y de Materiales, Universitat Politècnica de València, Camino de Vera s/n, 46022 Valencia, Spain.*

² *Departamento de Ingeniería de Sistemas Industriales y Diseño, Universitat Jaume I, Avda. Vicente Sos Baynat, 12071 Castellón, Spain*

* Corresponding autor 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: fromero@esid.uji.es Tel.: +34964728209; fax: +34964728170

³ E-mail address: prosado@mcm.upv.es Tel.: +34963877000 Ext. Int.: 76221; fax: +34963877629

An ontology for integrated machining and inspection process planning focusing on resource capabilities

The search for and assignment of resources is extremely important for the efficient planning of any process in a distributed environment, such as the Collaborative Product Integrated Development process. These environments require a degree of semantic interoperability, which currently can only be provided by ontological models. However, the ontological proposals centered on Resources for Machining and Inspection Process Planning have a limited reach, do not adopt a unified view of machining and inspection, and fail to express knowledge in the manner required by some of the planning tasks, as is the case with those concerned with resource assignment and plan validation. With the aim of providing a solution to these shortcomings the MIRC (Manufacturing and Inspection Resource Capabilities) ontology has been developed, as a specialist offshoot of the Product and Processes Development Resources Capabilities ontology. This ontology considers resource capabilities to be a characteristic of the resource executing any activity present in an Integrated Process Plan. Special attention is given to resource preparation activities, due to their influence on the quality of the final product. After describing the MIRC ontology, a case study demonstrates how the ontology supports the process planning for any level, approach or plan strategy.

Keywords: resource capabilities ontology; inspection and machining resources; integrated process planning; resource assignation; process plan validation

1. Introduction

As is widely recognized, ontologies allow distributed knowledge to be used and shared, while guaranteeing semantic interoperability and integration between different applications and agents (Cai, M., W. Y. Zhang, and K. Zhang 2011), and thereby facilitating collaboration between all the parties involved. However, the proposals made up to now for extended and complex environments, characterized by inter-

functional interaction, are partial and only consider some of the outlooks needed to achieve effective integration, in environments defined by a holistic management of the company (Kantola 2009). This is the case with the Collaborative Product Integrated Development processes, in which integration of the ontologies focusing on product, processes and resources is absolutely necessary (Zdravkovic and Trajanovic 2009; Ramos 2010; Honggen et al. 2012).

One of the most proven ways to integrate ontologies is based on the use of foundational ontologies (Oberle et al. 2007). Particularly, Descriptive Ontology for Linguistic and Cognitive Engineering (DOLCE), described by Masolo et al. (2003), has served as the basis for a number of ontologies used in the domain of design and manufacturing, such as ADACOR (ADaptive holonic COntrol aRchitecture) (Borgo and Leitao 2007) or the PPDRC (Product and Processes Development Resources Capabilities) ontological model (Solano, Rosado, and Romero 2013), which represents Resource Capabilities in the Product and Processes Development process.

As with the proposal by Newman and Nassehi (2009), in the PPDRC ontology a resource is an entity characterized by the set of its capabilities. However, an important difference is that in the PPDRC ontology the capabilities of a resource are linked to the execution of activities, adopting concepts from DOLCE, Activity Theory (Kuutti 1995) and Process Specification Language – PSL – (ISO 2004b).. From the first two, it takes the social and agentic character of its entities, while from PSL it takes the semantic richness necessary to support any activity planning and execution that falls within the collaborative framework proposed by Rosado and Romero (2009).

Upon these foundations, PPDRC reformulates the concepts and axioms linked to the resources and their capabilities that are present in generic proposals, such as MANDATE (ISO 2004a), ISA-95 (ISA 2000), ADACOR and TOVE (Fadel, Fox, and

Gruninger 1994), or that belong to specific proposals for the service-oriented manufacturing domain (Ning, Tian-guo, and Wen-jian 2010; Ameri et al. 2011). With the exception of MANDATE, they all characterize the resource from only a partial perspective, focusing on operational capabilities (such as production capacity, throughput time, cost per unit, etc.) with the objective of supporting production planning and control activities, but ignoring the technological capacities, which are fundamental for decision-making in process planning. These representations, which see capabilities as static and individual properties inherent in the resource and independent of state and conditions of use, have been improved in the PPDRC. For this purpose, it considers that: a) resources can participate in the execution of an activity with different roles; b) the capability associated to a resource executing an activity depends on the type of activity; c) capability is conditioned by the input objects' fulfilling certain requirements; d) capabilities change as a consequence of the resource preparation activities and other uncontrolled causes that may occur throughout its lifetime; e) the resource and its capabilities can be considered at different levels of aggregation (section, cell, machine, etc.) in order to cover the needs of the different levels of process planning (aggregated, supervisor or operational); and f) a complex resource shows behavior which goes beyond the sum of its individual parts. Thus, the PPDRC differs from others proposals by allowing the reasoning and the integration of planning, programming and control of intelligent systems. Furthermore, it can support web manufacturing services, like the validation of resource selection in a process plan, which are not covered by other ontologies centered only on resource assignment.

In the machining and inspection process planning domain, and particularly at the supervisor level (setup selection and sequencing, assignment of machines and tools, process plan validation, etc.) few proposals can be found which deal with the technical

characteristics of resources in sufficient detail. Vichare et al. (2009) define a resource model concerned with setups and the dimensional and geometric validation of solutions, but it does not include the precision of the kinematic characteristics of resources and so it does not establish relationships with manufacturing process and product models. Newman and Nassehi (2009) refer to the status of devices over time, and define a resource capability profile as an aggregation of the individual profiles corresponding to the tool, fixture, machine axes and part family production policies; likewise, they also propose monitoring and prediction strategies to update the current status of the resource capability profile. In the area of inspection, research contributions are scarce (Martínez-Pellitero et al. 2011) and do not deal with the representation of measurement capabilities of resources.

From the review of publications that was carried out, it is clear that: a) the approach adopted by the generic proposals, including the PPDRC, does not respond to the specific needs of integrated, collaborative planning focusing on the supervisor level; and b) the specific proposals for machining and inspection have a limited reach and do not represent an integral approach. To solve these shortcomings, the MIRC ontology (Manufacturing and Inspection Resource Capability), which is described in this paper, specializes the concepts and predicates of the PPDRC ontology in order to represent the capabilities of resources in machining and inspection operations and in the preparation activities involved therein. To achieve this objective it is important to take into account the difficulty inherent to the activities involved in machining and inspection integrated process planning in order to establish a set of machining and inspection operations and their sequence for satisfying all the quality requirements of the machined part. This process will be influenced by a number of diverse factors, among which it is worth highlighting the integrated assignment and configuration of manufacturing and

inspection resources. This is especially important in Virtual companies in which different agents participate in the definition of the process plan, using and exchanging shared knowledge about geometric deviations of the process and manufacturing resources, which constitute one of the most critical considerations in the successful definition of integrated machining and inspection plans.

Another relevant aspect is the integration of process planning and execution-controlling activities, which allows the final quality to be improved by taking advantage of the inspection data (in-process and post-process) in order to adapt the process plans to the actual resource capabilities. To do so, information on configuration and traceability of resources is essential (González et al. 2009).

The design of the MIRC ontology must ensure support to several queries related to collaborative and distributed process planning referring to types of machining and inspection operation; capabilities needed for an operation; resource configuration, their participation in occurrences, etc. These queries, which are known as *competency questions*, can be seen in the case study (section 4) centered on the planner's task of calculating or estimating the resource capability values bearing in mind the influence factors. Previously, in sections 2 and 3, the proposals adopted (PPDRC ontology and MIRC ontology) are shown.

2. PPDRC ontology

On the first level of the PPDRC ontology, three types of mutually exclusive entities can be found: *Object*, *ActivityOccurrence* and *Region* (Fig. 1). These entities are specializations of *Endurant*, *Perdurant* and *Quality* respectively, which together with *Abstract* are the four entities at the first level of DOLCE. An *Object* is a tangible or intangible entity with existence (Solano, Romero, and Rosado 2010).

ActivityOccurrences are executions of activities and they use, produce and transform objects. Finally, the entity *Region* quantifies, via a field and a value, the qualities of the other entities.

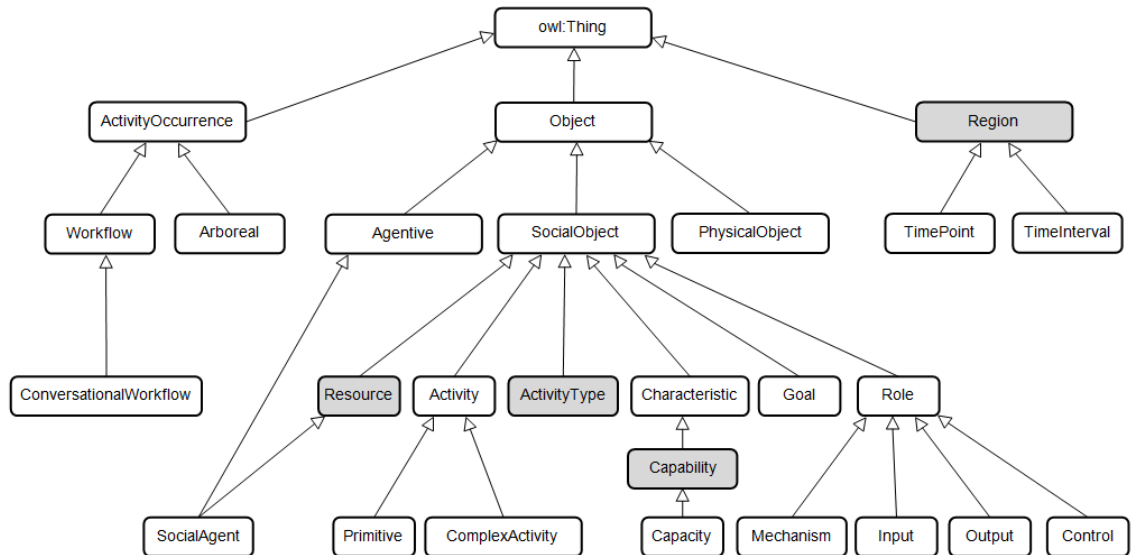


Figure 1. Taxonomy of the PPDRC ontology.

In the PPDRC ontology, social objects are of great importance and consist of shared descriptions which allow mutual understanding between members of a community (Ferrario and Oltramari 2004). *SocialObjects* include: *Activities*, *ActivityTypes*, *Resources*, *Roles* and *Characteristics*. *Activities* are the basic entities that make up the process plan and, when carried out (*ActivityOccurrences*), represent the execution of the plan. The entity *ActivityType* represents the types to which *Activities* belong. *Resources* are objects which have the competence or ability (capability) to carry out an activity, and reach a particular level of performance in this execution. A resource is a social object that has capabilities and is linked to a physical object. Role is description of how a physical object participates in the execution of an activity, and may be: *Mechanism*, *Input*, *Output* or *Control*. *Characteristic* is an entity whose individuals express the qualities of other individuals. Any object can have a relationship with

Characteristic via the predicate *characterizedBy* (*Object*, *Characteristic*), while the predicate *parametrizes* (*Characteristic*, *Region*) expresses the relationship existing between a characteristic and the regions that quantify it (Fig. 2).

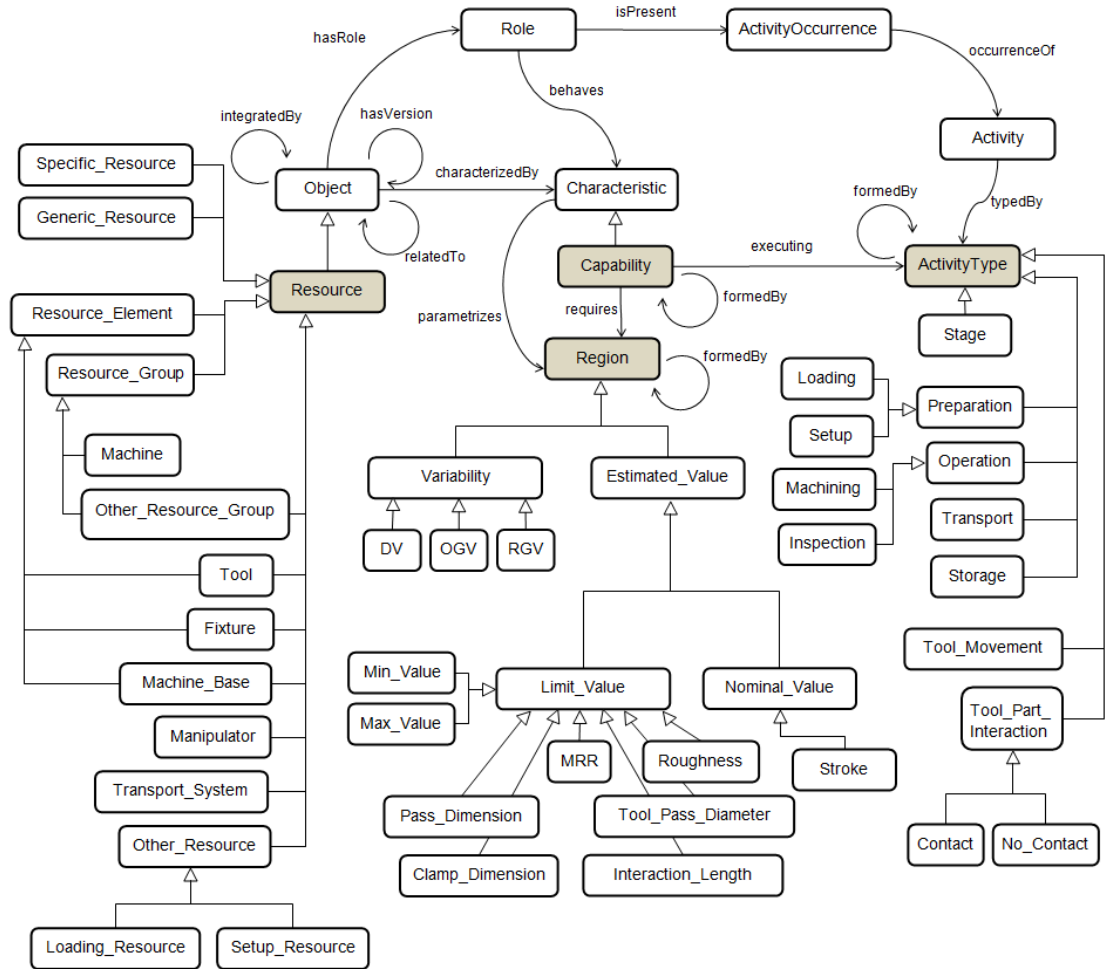


Figure 2. Predicates of the PPDRC ontology, and taxonomy of the entities *Resource*, *Region* and *ActivityType* in the MIRC ontology.

The entity *Capability* is a specialization of *Characteristic* that characterizes the use of a resource executing a particular type of activity. A capability is the ability to carry out a type of activity with a level of performance that is quantifiable via regions. The relationship that exists between a capability and its regions is expressed through the predicate *parametrizes* (*Capability*, *Region*), which has two specializations:

parametrizes_Occ and *parametrizes_Object*. In the first type, the regions of the resource capabilities (production rates, power, time, dimensions, etc.) are not transmitted to the object on which the activity is carried out, while the regions linked to *paramatrizes_Object* are transmitted.

To consider the influence of the characteristics of the object that is transformed in the activity, the ontology uses the predicate *requires (Capability, Region)*, which expresses the resource's capability requirements to the initial object characteristics. Therefore, the predicate *requires* implies the existence of restrictions on the object with *Input* role so that the capability of the resource can be considered at its specified value. A particularly interesting object role in the PPDRC ontology is the *Mechanism*, because these objects support execution of the activity and influence the quality of the result (output). Furthermore, a physical object can fulfill various roles in the same occurrence, for example, input and mechanism. The object that has an *Output* role in the execution of an activity obtains a characteristic quantified by the capability regions of the resource (regions linked to *parametrizes_Object*). Finally, the object with the *Control* role specifies the execution conditions required to produce the desired results.

In general, the changes that are produced in a physical object when an activity is executed are shown by the regions of its characteristics. However, on occasions these changes are not reflected in the physical characteristics of the object, as they only affect its social dimension. (as the resulting characteristics of a part verification activity). Similarly, changes in a resource are shown by the regions of its capabilities, and may be due to logical changes, changes in data or changes to location of the resource. These modifications to objects (or resources) that are not accompanied by physical changes lead to versions of the object (or resource) which are related through the transitive predicate *hasVersion (Object, Object)*. The versions of an object may correspond, for

example, to its current situation, compared to its future planned state, or the optimum capability provided by the manufacturer compared to the real capability, which becomes known after the verification of the results obtained.

This predicate allows the management of the historical information and traceability associated to resource changes.

Finally, another relevant aspect in the PPDR ontology refers to complex resource configurations management. The predicate *relatedTo (Object, Object)* shows the relationships that exist between physical resources that are physically connected. Those connections between elemental resources are represented by interfaces, which are associated to certain types of activity (assembly, fixing, setup, etc.) and their characteristics are quantified by the resource capability executing that activity.

3. Description of the Manufacturing and Inspection Resource Capabilities ontology

As mentioned earlier, the PPDR ontology can be tailored to diverse application domains, such as that of technological planning. This is the case of the MIRC ontology, which can tailor activities and capabilities of manufacturing resources with the aim of supporting the knowledge required to take the decisions concerning the selection, assignment and preparation of resources during the development of a machining and inspection integrated process plan. In the PPDR, a process plan is seen as a collection of planned activities to be executed with resources that have a mechanism role, with the purpose of producing a part (*Object*) with particular characteristics.

Among the various activity types, one can find the following (Fig. 2): a) transform the physical characteristics of the part (such as *Machining Operation*); b) obtain information on the physical characteristics of the part (such as *Inspection Operation*); c) modify the location, state or storage conditions of the part (such as

Transport and *Storage*); and d) create or modify the characteristics of the resources that participate in any activity of the plan (such as *Preparation*).

To reach this goal, the *Resource*, *Capability*, *Region* and *ActivityType* entities are specialized (Fig. 2). The following subsection is dedicated to the description of the first three entities, and to the conceptualization and representation of the activity entity. In the remaining subsections, the specific aspects of activity types *Operation* and *Preparation* that are relevant in the manufacturing and inspection process planning are described.

3.1. Framework of the proposal

3.1.1. Activity. Conceptualization and representations.

An essential element in the MIRC ontology is the conceptualization of *Operation* and *Preparation* activities, which is based on the following principles: a) the characteristics of the output object are obtained through the composition of the resource capabilities and through the characteristics of the execution of the activity itself (interface), which can be viewed as a link in the activity chain defining the characteristic composition; b) the capabilities of a resource are conditioned by the fulfillment of certain characteristics of the input object, which therefore participate indirectly in the activity; and c) the characteristics of the interface are regulated by the characteristics that control the activity, which also participate indirectly.

A link in that activity chain, associated to the execution of the activity, is represented in Fig. 3 (a). It shows how the characteristics of the output object, sketched by the double solid arc, depend on the characteristics of the objects that take part in the execution of an activity with *Input*, *Mechanism* and *Control* roles, sketched by the dotted arc, solid arc and solid/dotted arc respectively, and on the characteristics of the

interface of the activity, which acts as a hinge (interface) on an axis representing the execution of the activity. A three-dimensional graph, which is replaced by a plane representation (Fig. 3 (b)) in the rest of the paper, arising from the abatement of a combination of *Input* (I), *Mechanism* (M), *Control* (C) and *Output* (O) planes. In that same figure, the arc of the input object's characteristic connects with the capabilities of the resource and, similarly, that of the control object connects with that of the characteristics of the activity interface. Additionally, Fig. 3 (c) shows how object characteristics with *Input* and *Mechanism* roles are the object characteristics with an output role in the execution of the aforementioned activities, thus allowing the definition of the chains that represent the plan (e.g. Fig. 5 and 6).

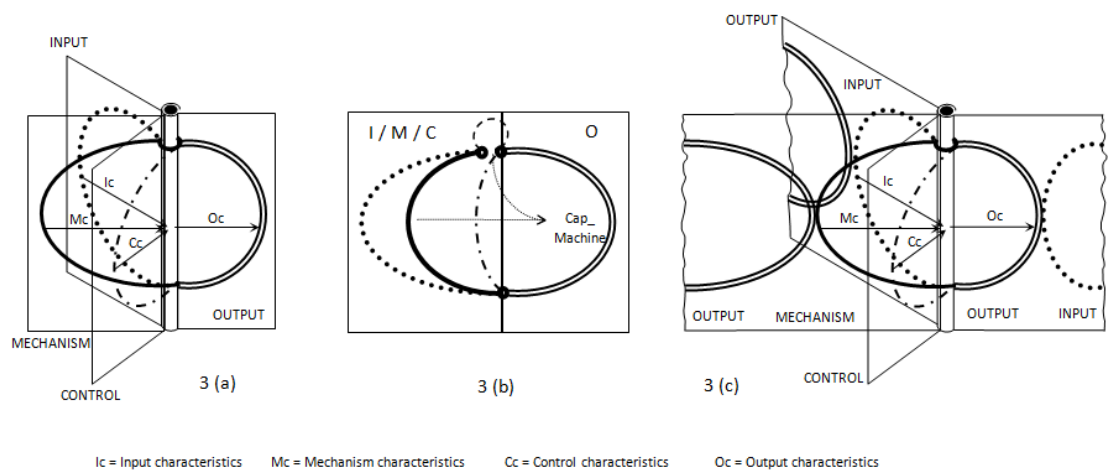


Figure 3. Spatial and plane representation of the characteristics associated to the execution of an activity.

These activity representations show that in order to guarantee some particular characteristics of the output part, it is necessary to restrict certain characteristics of the participating objects and activity interfaces. This is relevant in process planning, in which a backward strategy is usually adopted, meaning that the output object characteristics from the last activity of the plan will decide the characteristics of all the objects involved (input, mechanism, control) and activity interfaces present in the plan.

Similarly, a projected view shows arrows representing the traces of the I, M and O planes and the direction of the progress of the plan. In this representation, the objects that participate in the execution of an activity with a particular role can be seen, along with a chain of activities. This allows easy visualization of the structure of the plan's activities. These projected views complement the previous ones by making it possible to see some details about the activity chain, such as: a) that in an atomic (indivisible) activity, only one resource (simple or complex) can participate with a *Mechanism* role, as the resource characteristic is that which is transmitted to the output object; and b) that a capability of a resource executing an activity may require various characteristics of one or more input objects. Fig. 11 shows these details for a chain of preparation activities.

Although in the previous representations all the activities are atomic, the plan activities, which are understood to be a pattern of repeatable actions, can also be complex, as is the case with *Stage*. The individuals of the type *Stage* are groups of *Activities*, made up of all the machining and inspection operations that share the same resource, like: a) *Machine_Base*; b) *Machine_Base* and *Fixture*; or c) *Machine_Base*, *Fixture* and *Tool*. *Activities* that make up a complex activity, such as those of the type *Stage* can be executed sequentially and/or simultaneously. However, atomic activities can only be decomposed into other activities which are executed concurrently, as is the case for an activity of the type *Operation*, which involves the concurrent execution of atomic activities of the types *Tool_Movement* and *Tool_Part_Interaction* (Fig. 2). In process planning, the selection of resources is usually carried out at the level of these atomic activities, with the later possibility of grouping them into complex activities, which share, in a spatial and temporal sense, the same resources (plan stage) with the aim of reducing preparation time. With this in mind, the assignment of mechanism

resources is only carried out at the level of atomic activities of the type *Operation* (*Machining* and *Inspection*) and *Preparation*. Thus, it is guaranteed that any change in part characteristics is only due to the resource characteristics executing a type of activity.

When Geometric and Dimensional (GD) characteristics are the focus of analysis, it must be considered that interface characteristics are established between the active geometries during the activity – both those of the mechanism object and those of the output object. These interfaces represent the physical interaction of the object with the resource (mechanical, electromagnetic, etc.) during activity execution. For example, in the case of a *Machining* operation, both the relationship part-fixture and the relationship tool-part are considered.

3.1.2. Resource selection. Groups and capabilities.

As the selection of a specific resource capable of executing an atomic activity is usually carried out from groups of resources that have similar capabilities, the MIRC ontology considers the type *Generic_Resource* formed by all *Specific_Resources* related through the predicate *includes (Resource, Resource)*. The capability that characterizes a generic resource concerns both the types of activities that can be executed as well as the levels of achievement reached. Following on from this, a generic resource is understood to be an abstract resource whose participation in an activity implies the participation of one of the specific resources included within it.

The *Resource* has been presented as if it were an elemental entity, although it is usually integrated by other elemental resources (*Resource_Element*) or complex resources (*Resource_Group*) which act together in the execution of the atomic activity. This relationship is established through the predicate *integratedBy (Resource,*

Resource), inherited from PPDRC objects, which expresses the grouping of the social objects that make up the social object *Resource_Group*.

The specializations related to the PPDRC predicate *relatedTo* (*relatedToPart* and *relatedToTool*) express the connections between the resources that make up a resource group and the direction of the connection. This allows management of the particular resource configurations. However, it is worth noting that the characteristics of the resulting object, in this case the capabilities of a resulting resource group, depend on: a) the capabilities of the resource elements or resource groups which form them; and b) the characteristics of the resources (capabilities) and interfaces corresponding to the activities of the type *Preparation* (*Loading* or *Loading plus Setup*).

3.1.3. Object characterization. Regions.

Looking a little deeper into the quantification of capabilities through regions, it is important to remember that the values can be obtained from: a) historical generic data, coming from catalogues, manuals and reports, which consider a range of execution methods and conditions; or b) data considering specific execution conditions. In both cases, the data can correspond to different levels of aggregation of the resource and may have been obtained directly by measurement at this level or as a collection of results from lower levels. This latter case highlights the importance of having data available from elemental resources. The predicate *hasVersion* allows relationships to be established between resources that are differentiated by the values (regions) of their characteristics.

Another aspect worth highlighting is the way to quantify a characteristic. To do so, the MIRC ontology establishes three types of regions that consider the way to compose the values: *Variability*, *Nominal_Value* and *Limit_Value* (Fig. 2). Quadratic

composition rules (or similar) are applied for regions of type *Variability*, whereas in the regions of the type *Limit_Value*, the maximum or minimum values of all the regions considered are taken. Finally, in the regions of type *Nominal_Value*, the composition is carried out using the algebraic sum.

To consider dimensional and geometrical aspects, three specializations for *Variability* regions have been considered (Fig. 2): *Dimensional_Variability* (*DV*), *Own_Geometric_Variability* (*OGV*), and *Reference_Geometric_Variability* (*RGV*). *DV* regions express variability in lengths and angles. *OGV* regions express intrinsic geometric variability, for example flatness or roundness. Finally, *RGV* regions express orientation and position variability, such as parallelism or perpendicularity. Another region needed to complement dimensional and geometrical specifications is *Roughness*, which is a *Limit_Value* region.

3.2. Activities of type Operation

After the most relevant aspects of the conceptual framework of the proposal have been outlined, the current section focuses on activity of the type *Operation*. An *Operation* is a concurrent grouping of activities of types *Tool_Part_Interaction* and *Tool_Movement*. *Tool_Part_Interaction* represents the interaction between tool and part, which involves removing material or inspecting the part, and *Tool_Movement* represents the relative movement between the tool/probe and the part. A *Tool_Movement* is the combination of movements (linear or rotatory) that are defined by the machining and inspection operation strategies. In turn, the individuals of the type *Tool_Part_Interaction* can be of two types: *Contact* and *No_Contact*, depending on the characteristics of the interface between tool/probe and part (Fig. 2).

The *Resource_Group* that participates with a *Mechanism* role in the *Operations* is called *Machine* (Fig. 2) and comprises a *Machine_Base* individual and one or more *Tool* and/or *Fixture* individuals. The *Machine_Base* is characterized by the capabilities to execute *Tool_Movement* activities, while *Tool* and *Fixture* are characterized by capabilities to execute *Tool_Part_Interaction* and locating/fixing activities respectively. As was shown in the previous section, in the GD area, final characteristics of parts are determined directly by the GD capabilities of the machine that executes the operation and the GD characteristics of the operation interface, considering tool-part interaction (Fig. 4). An interface quantifies the discrepancies that exist between machine capabilities in real execution conditions and in those used in the operations that served for the quantification of machine capabilities (test and historical). This is because the estimated capabilities are always closely linked with the actual loading level and the specific type of control.

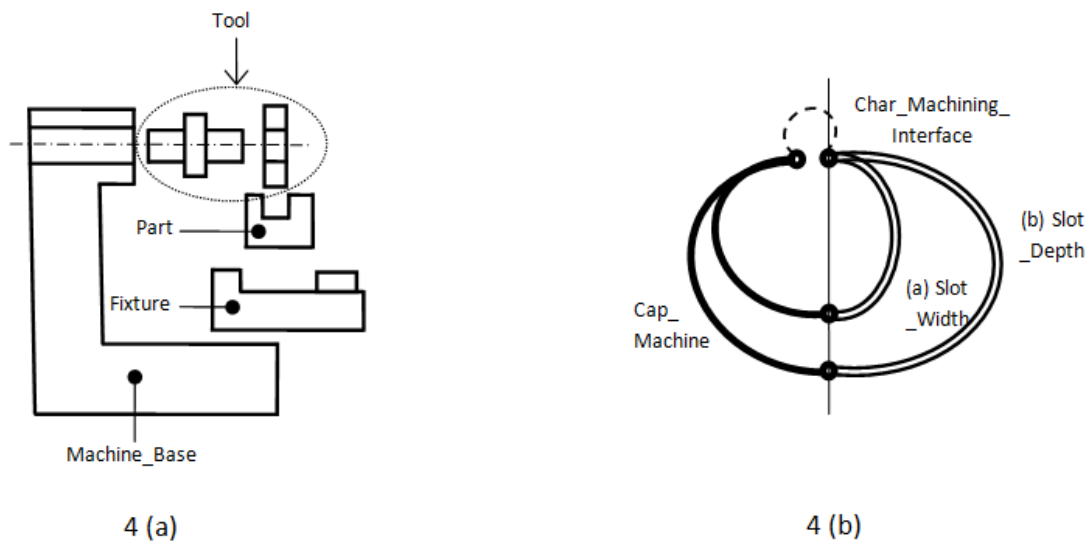


Figure 4. Physical representation of the resource participating in the machining of the slot (a) and graph with the characteristics of the machined slot (b).

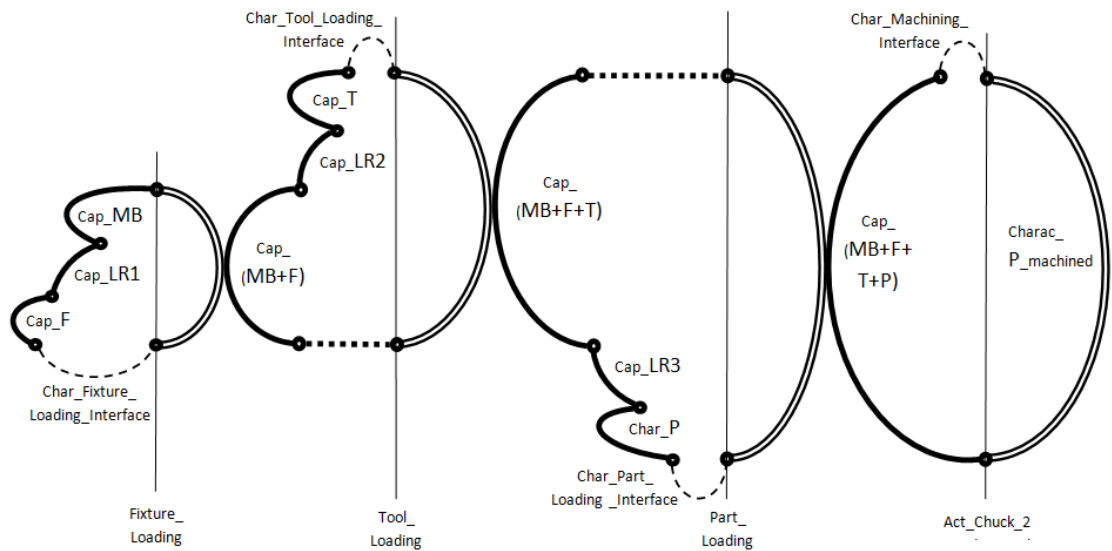
As shown in Fig 4(b), various GD characteristics of the part can be generated in a machining operation. This circumstance, which is present in the majority of the activities, leads to multi-characteristic graphs with several loops, which show the existence of different active geometrical interfaces, both in the tool-part relationship and in the part-fixture relationship (Fig. 4). Different resource capabilities are used in each loop.

In Fig. 4(b) the ends of the solid arcs represent active geometries (machined or datums) or part and resource references involved in the operation. One end of the arc representing a part characteristic is linked to the machining interface (Machining Feature), and the other is linked to a reference or active geometry of the resource.

In the case of feature's intrinsic characteristics, like slot width (Fig. 4(b)), one end of their representing arc is the machined geometry and the other one is a feature self-reference that will be made to coincide with a *Machine_Base* reference. Nevertheless, in the case of feature extrinsic characteristics, like slot depth (Fig. 4(b)), the second link corresponds to a characteristic datum, which will coincide with a local machine reference. In the next section and in the case study, it will be seen that the arc representing resource capability is composed of several arcs corresponding to individual capabilities of resources involved and interface characteristics between them.

3.3. Activities of type Preparation

As is well known, machining and inspection resource capabilities are established by a set of *Preparation* activities, needed for the formation of a *Resource_Group* and/or its modification and characterization. Hence, the *Preparation* entity has two specializations (*Loading* and *Setup*) which are not mutually exclusive (Fig. 2), thereby allowing the existence of *Preparation* activities made up of activities of both types.



MB = Machine_Base; F = Fixture; T = Tool; P = Part; LRi = Loading_Resource_i

Figure 5. Plane view in which the execution of the three *Loading* activities are linked with the execution of a *Machining* activity.

The result of a *Loading* activity is a resource that is different from those participating with an *Input* role. The output resource will have some characteristics of the input objects and will also acquire new characteristics. For example, as can be seen in Fig. 5, the resulting resource capability (Cap_MB+F) of the activity *Fixture_Loading* will depend on the characteristics of its execution ($Char_Fixture_Loading_Interface$), on the capabilities of the *Loading_Resource* executing the activity (Cap_LR1), and on certain characteristics of those input objects that compose the output resource (Cap_MB and Cap_F). In *Loading* activities, the input objects act with the roles of both mechanism and input, which is a significant difference from operation activities. This particularity can be seen in Fig. 5, where the capabilities of all participating mechanisms (Cap_MB , Cap_LR1 , and Cap_F in *Fixture_Loading*) are transmitted to the output capability (Cap_MB+F). Furthermore, in Fig. 5 it can be seen how the result of the *Fixture_Loading* activity participates as a mechanism in the execution of the

Tool_Loading activity, whose result, in turn, participates in the execution of the *Part_Loading* activity, from which the resource that finally executes the machining operation is obtained. The thick horizontal broken lines, which appear in Fig. 5 and in the following figures, represent the extension of a point on the graph. In other words, these thick broken lines do not represent characteristics.

A more compact way to represent the activity chain is shown in Fig. 6. In the graph on the left, the arcs linking the executions of the activities have been deleted, and in the one on the right all the arcs representing mechanism capabilities are drawn in a single arc (composition).

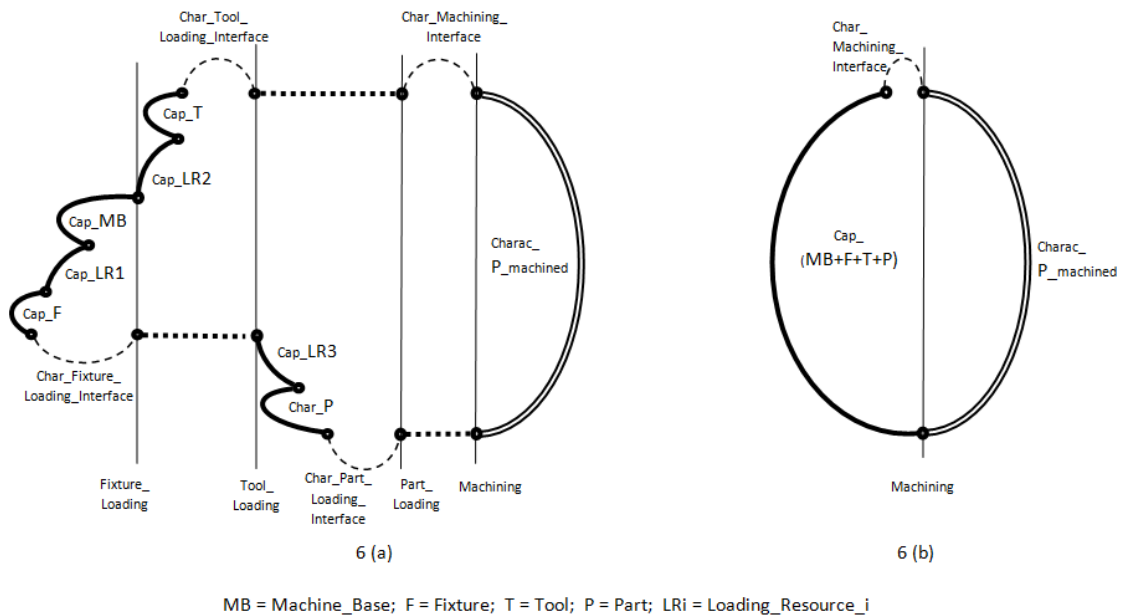


Figure 6. Compact view of detailed *Loading* activities to configure a resource for machining (a). Plane view of the *Machining* operation using the complex resource as an elemental resource (b).

In setup activities, which include a measure and/or correction, the capabilities of a resource group are quantified directly, as if a single entity were being dealt with. As can be seen in Fig. 7, in *Setup* activities, the input resources do not play a mechanism role and only connect with previous activities. This means that they do not transmit their

capabilities to the output resource (prepared resource), which is only influenced by the *Setting_Resource* (SR3 or SR4). As is generally recognized, the inclusion of this type of activity improves the capabilities of the resources.

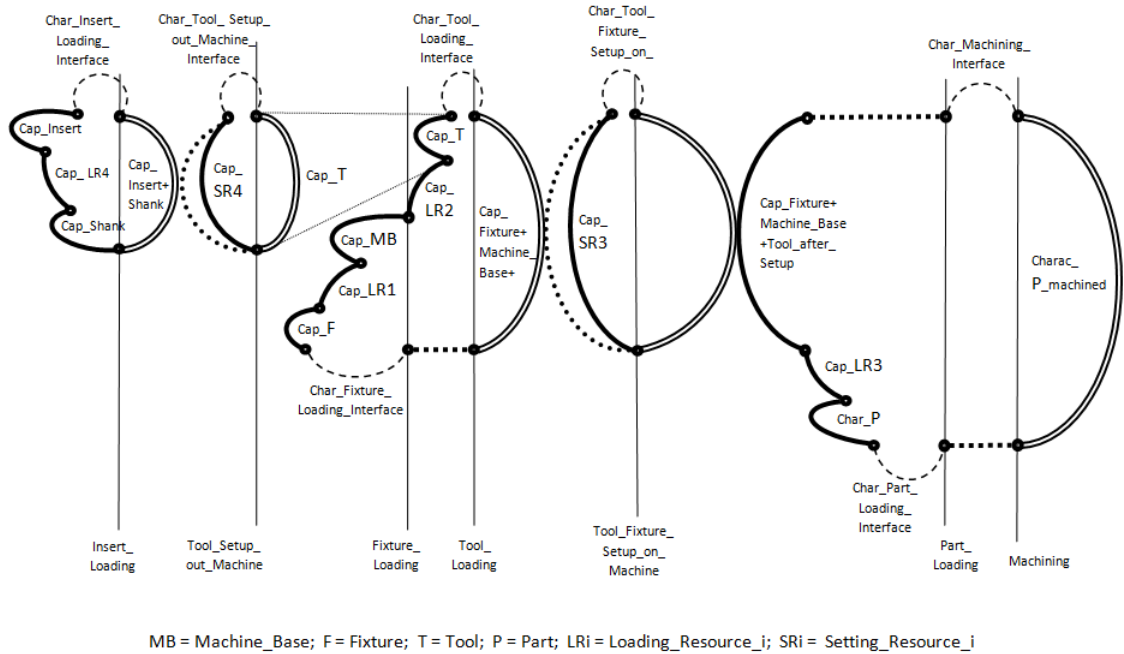


Figure 7. Chain of *Loading* and *Setup* operations to configure a resource for machining.

In the MIRC ontology (Fig. 8) various *Setup* specializations have been considered: *Setup_on_Machine* and *Setup_out_Machine*. Fig. 7 shows a *Setup_on_Machine* (*Tool_Fixture_Setup_on_Machine*) and a *Setup_out_Machine* (*Tool_Setup_out_Machine*). In the first type, a *Machine_Base* is always involved, which is not necessary in the *Setup_out_Machine*.

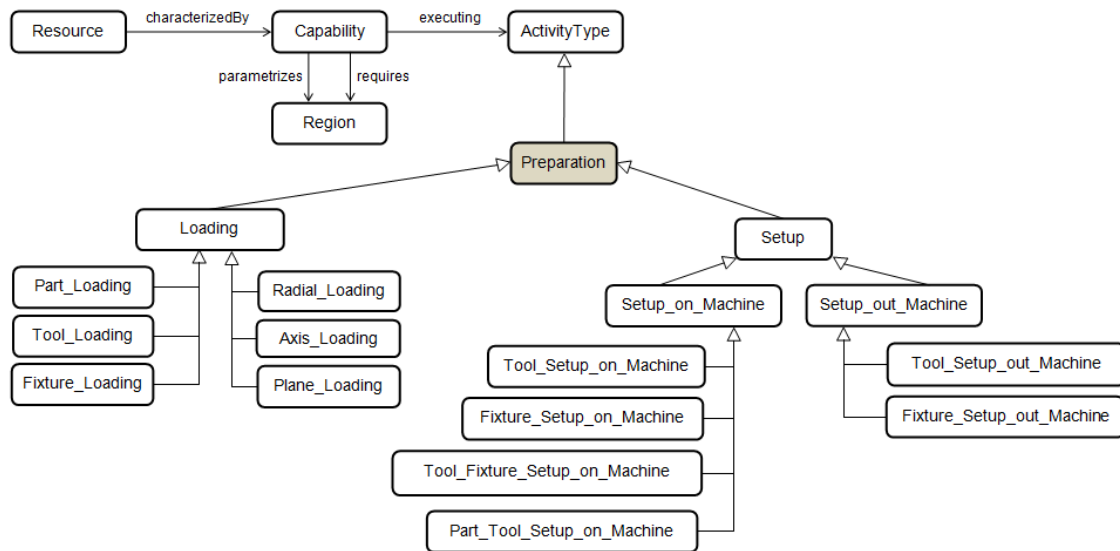


Figure 8. *Preparation* activities in the MIRC ontology.

3.4. Implementation of the ontology

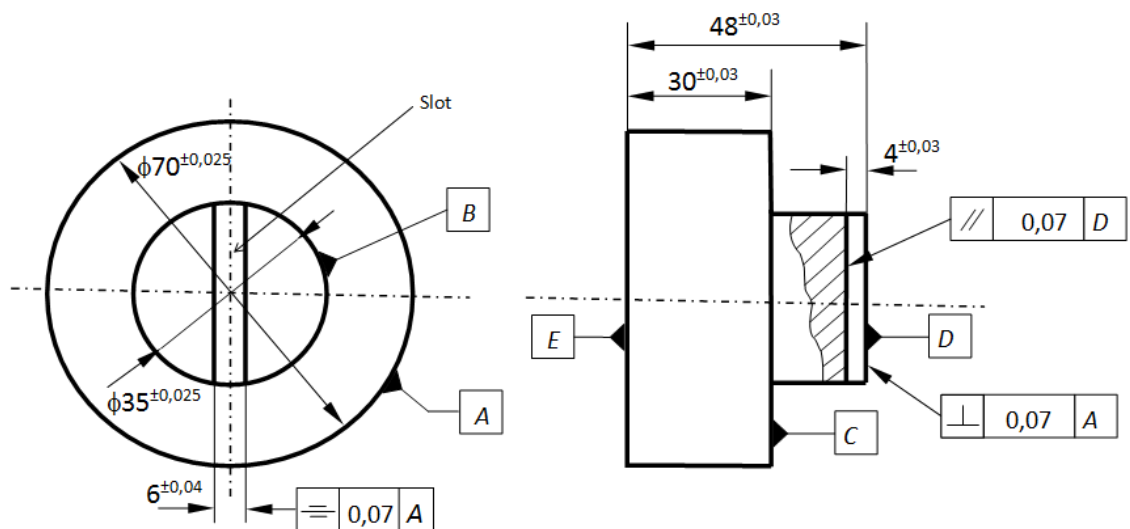
In response to the requirement of supporting a Web-based OKP system, it was decided the ontology should be modeled using a standardized and widespread language. Therefore, the modeling of the MIRC ontology was carried out through OWL (Ontology Web Language) and SWRL (Semantic Web Rule Language), adopted by World Wide Web Consortium (Cai et al. 2010; Lin et al. 2011). SWRL improves the semantic expressivity of OWL, thus allowing it to express knowledge that cannot be directly defined with the OWL axioms (Lin 2008).

The MIRC ontology has been developed and edited using the ontology editor Protégé and can be consulted at MIRC (2014). Additionally, to check the consistency of the ontology between predicates and definitions, to maintain the ontology hierarchy, and to allow consultations, the Pellet reasoner was used. Pellet is an open-source reasoner which allows classification and reasoning with individuals in the OWL/SWRL ontologies.

In order to validate the proposal, a Java application was implemented. The application incorporates Java libraries to operate with OWL files and to reason and query with Pellet, which includes query support using the SPARQL language (Mariot et al. 2007). In that way, the application is provided with a front-end interaction interface that translates the user query into SPARQL and transforms the SPARQL query results into a more user-readable format.

4. Case study

The use of the MIRC ontology can be shown with the description of the process planning of Part_1 (Fig. 9), which is manufactured in a virtual OKP context. This case study concentrates on decision-making tasks based on capability knowledge, as in the case of the selection and validation of machining and inspection resources. In this way, the aim is to show how the MIRC ontology facilitates distributed process planning, regardless of the level at which it is being carried out (aggregated, supervisor or operational), the approach adopted (variant or generative) and the strategy used (forward or backward).



Material: F1140 Ra = 0,4 μ m (external cylindrical surfaces)

Figure 9. Dimensional and geometric specifications for Part_1.

To do this, it will be shown how the different planners or software agents involved can consult knowledge and data in the MIRC ontology. The framework within which these planners carry out their work takes into account the relationship between process planning, resource planning and scheduling (Sormaz, Arumugan, and Rajaraman 2004) and is presented schematically presented in Fig. 10. The companies that make up the virtual OKP (C1, C2, C3, etc.) have the capabilities to carry out the activities of the Integrated Development of Products, Processes and Resources. These companies include resources with capabilities to execute activities of *Operation* and *Preparation* types. The data relating to these generic or specific resources and to their capabilities, together with the data on the characteristics of the part machining features, are represented in the corresponding knowledge base (KB).

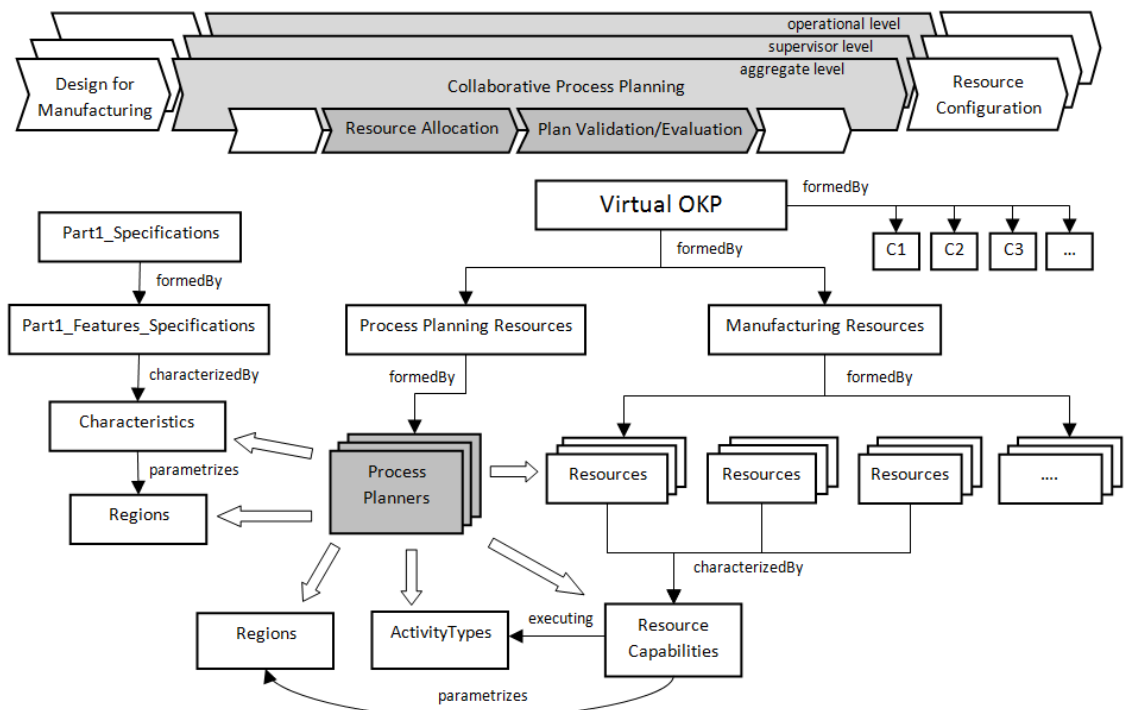


Figure 10. Schematic representation of the framework for process planning.

In the aggregated planning and adopting a backward strategy, once Part_1 has been identified as a part of type *Revolution_ with_ No_Revolution_Features* with a minimum value of 0.05 mm for its critical regions, the planner can define the alternative machining macro-plans that are valid for it. To this end, the planner will query the KB about the resources of the companies in the virtual OKP with capabilities to execute the required operations (Table 1). In this query (Query 1), the KB is asked about the generic resources that can execute the manufacturing activities for Part_1 (machining flat surfaces, cylindrical surfaces and slots) with the necessary capabilities and the blank requirements for these resource capabilities. The queries are formulated using the application developed for this purpose. In this first case, Table 1 shows the query in SPARQL, its interpretation in natural language and the response. In the rest of the queries only the interpretation and the results are shown. The two type of routes established from the results of Query 1 (“Answer”) are: a) conventional machining with in-process inspection (type_A), and b) conventional machining, grinding and post-process inspection (type_B). In both routes, the tolerance for the raw material must be less than 0.5 mm.

In the case that a type_A route is selected, the planners must define the resources needed at supervisor level. Assuming that machining of the *Slot_Feature* (Fig. 9) is the planners’ focus, they can ask about resources of type *Machine* which on some occasion have executed a *Slot_Operation* with a *DV* lower than 0.06 mm. The result of this query, corresponding to a variant approach, shows the two capable machine resources (*Machine_32GT* and *Machine_32GT_Part_1_RM*) and their components (Table 2).

If *Machine_32GT_Part_1_RM* (which includes the part) is the selected resource, the planner would need a deep knowledge about the preparation activities that were used to configure this complex resource. For this purpose, a query (Table 3) can be launched,

with the goal of obtaining the information needed to define and validate a complete plan of the preparation stage, particularly the activity sequence and their input and output objects. Later queries will allow identification of the resources that act as a mechanism in these preparation activities, thus allowing representation of the chain (Fig. 11). Finally, to validate the selection, additional queries (Table 4) must still be addressed with the aim of representing the operation multi-characteristic graph (Fig. 12) necessary for this task.

Table 1. Query 1.

Query in SPARKQL	
	<pre> SELECT DISTINCT ?Generic_Resource ?Valor_Required_Machining WHERE { ?Generic_Resource a <http://www.coapp.es/ontologies/2012/10/MRO_V3.owl#Generic_Resource> . ?Generic_Resource <http://www.coapp.es/ontologies/2011/0/CDPP_v4.owl#characterizedBy> ?Cap_Plane_Surface . ?Cap_Plane_Surface a (<http://www.coapp.es/ontologies/2011/0/CDPP_v4.owl#executing> some <http://www.coapp.es/ontologies/2012/11/Example_2_v1.owl#AT_Plane_Obtaining>) . ?Generic_Resource <http://www.coapp.es/ontologies/2011/0/CDPP_v4.owl#characterizedBy> ?Cap_Cyl_Surface . ?Cap_Cyl_Surface a (<http://www.coapp.es/ontologies/2011/0/CDPP_v4.owl#executing> some <http://www.coapp.es/ontologies/2012/11/Example_2_v1.owl#AT_Cyl_Surface_Obtaining>) . ?Generic_Resource <http://www.coapp.es/ontologies/2011/0/CDPP_v4.owl#characterizedBy> ?Cap_Slot . ?Cap_Slot a (<http://www.coapp.es/ontologies/2011/0/CDPP_v4.owl#executing> some <http://www.coapp.es/ontologies/2012/11/Example_2_v1.owl#AT_Slot_Obtaining>) . ?Cap_Plane_Surface <http://www.coapp.es/ontologies/2011/0/CDPP_v4.owl#parametrizes> ?Region_Plane . ?Cap_Cyl_Surface <http://www.coapp.es/ontologies/2011/0/CDPP_v4.owl#parametrizes> ?Region_Cyl . ?Cap_Slot <http://www.coapp.es/ontologies/2011/0/CDPP_v4.owl#parametrizes> ?Region_Slot . ?Region_Plane <http://www.coapp.es/ontologies/2012/11/Example_2_v1.owl#valor> ?Valor_R_Plane . ?Region_Cyl <http://www.coapp.es/ontologies/2012/11/Example_2_v1.owl#valor> ?Valor_R_Cyl . ?Region_Slot <http://www.coapp.es/ontologies/2012/11/Example_2_v1.owl#valor> ?Valor_R_Slot . ?Generic_Resource <http://www.coapp.es/ontologies/2011/0/CDPP_v4.owl#characterizedBy> ?Cap_Machining . ?Cap_Machining a (<http://www.coapp.es/ontologies/2011/0/CDPP_v4.owl#executing> some <http://www.coapp.es/ontologies/2012/11/Example_2_v1.owl#AT_Machining>) . ?Cap_Machining <http://www.coapp.es/ontologies/2012/10/MRO_V3.owl#requires> ?R_Req_Machining . ?R_Req_Machining <http://www.coapp.es/ontologies/2012/11/Example_2_v1.owl#valor> ?Valor_Required_Machining . FILTER (?Valor_R_Plane < 0.05) FILTER (?Valor_R_Cyl < 0.05) FILTER (?Valor_R_Slot < 0.05) } ORDER BY DESC (?Generic_Resource) </pre>
Natural language interpretation	
	<p>What generic resources are characterized by capabilities that parametrize regions with a value lower than 0.05 executing types of activities to obtain flat surfaces, cylindrical surfaces and slots, and what are the values of the regions required for each generic resource to execute machining activities?</p>
Answer	
	<pre> Generic_Resource Valor_Required_Machining LatheCenter_A_C2 0.45 LatheCenter_A_C1 0.5 GrindingMachine_B_C3 0.1 GrindingMachine_B_C2 0.15 </pre>

Table 2. Query 2.

<i>Natural language interpretation</i>	<i>Answer</i>
What resources of Machine type, included in the generic resource LatheCenter_A_C1 or in the generic resource LatheCenter_A_C2, have participated in an activity of type Slotting_In_LatheCenter executed using a capability which parametrizes a region of type DV with a value lower than 0.06 mm, and what are the resources that integrate these Machine type resources?	Machine Resource Machine_32GT Generic_Turning_Tool_A2 Machine_32GT Machine_32 Machine_32GT LatheCenter_3 Machine_32GT Chuck_2 Machine_32GT_Part_1_RM Generic_Turning_Tool_A2 Machine_32GT_Part_1_RM Machine_32 Machine_32GT_Part_1_RM Part_1_RoughMaterial Machine_32GT_Part_1_RM LatheCenter_3 Machine_32GT_Part_1_RM Machine_32GT Machine_32GT_Part_1_RM Chuck_2 Machine_32GT_Part_1_RM Part_Axis_A Machine_32GT_Part_1_RM Plane_E

Table 3. Query 3.

<i>Natural language interpretation</i>	<i>Answer</i>
What activities of type Preparation generate a resource that forms the resource Machine_32GT_Part_1_RM and what are the inputs and outputs of these activities?	Preparation_Activity Input_Resource Output_Resource Act_Chuck_2_Loading LatheCenter_3 Machine_32 Act_Chuck_2_Loading Chuck_2 Machine_32 Act_Generic_Turning_Tool_A2_Loading Generic_Turning_Tool_A2 Machine_32GT Act_Generic_Turning_Tool_A2_Loading Machine_32 Machine_32GT Act_Part_1_RM_Loading Part_1_RoughMaterial Machine_32GT_Part_1_RM Act_Part_1_RM_Loading Machine_32GT Machine_32GT_Part_1_RM

Table 4. Queries 4 and 5.

<i>Natural language interpretation</i>	<i>Answer</i>
Query 4 What are the resources and features of the raw part that make up the resource Machine_32GT_Part_1_RM_Setup1, and how are these resources and features linked?	Object Object_2 Generic_Turning_Tool_A2 LatheCenter_3 LatheCenter_3 Chuck_2 Chuck_2 Part_1_RoughMaterial Chuck_2 Part_Axis_A Chuck_2 Plane_E
Query 5 What features make up the part after the machining of the slot (Part_1_Final), and what are the characteristics of these features?	Object Characteristic Slot_Bottom_Plane Slot_Parallelism_to_D Plane_D Slot_Parallelism_to_D Slot_Bottom_Plane Slot_Depth Plane_D Slot_Depth Slot_Middle_Plane Slot_Symmetry_to_A Part_Axis_A Slot_Symmetry_to_A Plane_E Distance_PlaneE_PlaneD Plane_D Distance_PlaneE_PlaneD

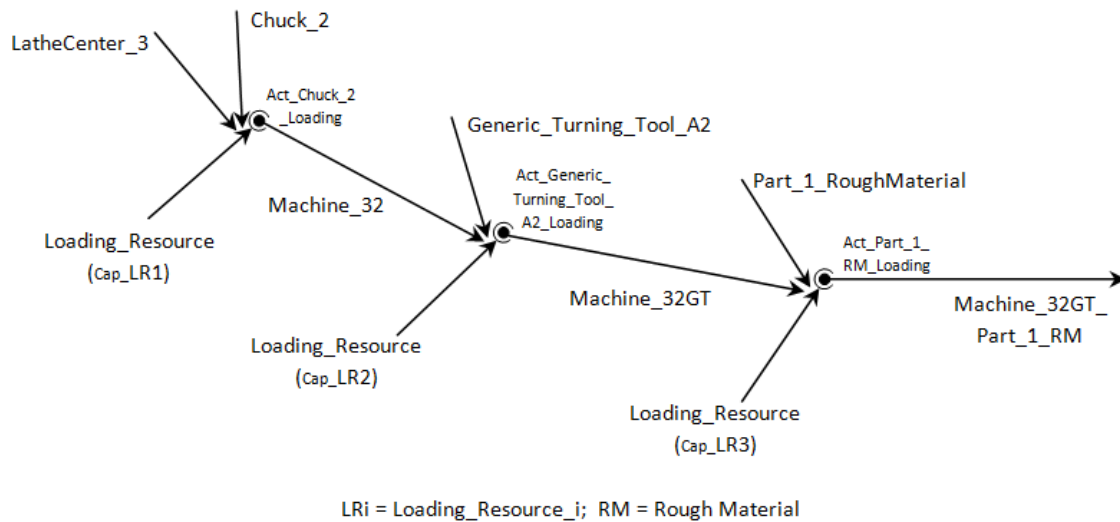
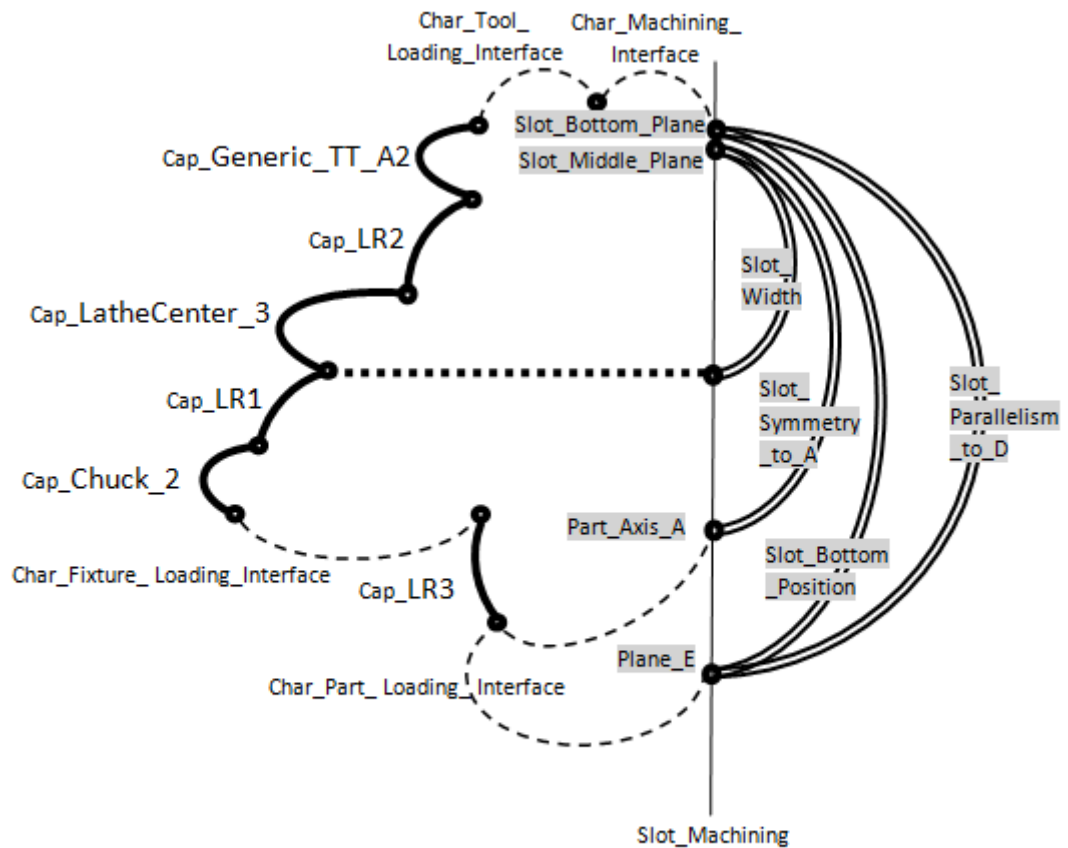


Figure 11. Projected view with the *Preparation* activities needed to obtain the complex resource Machine_32GT_Part_1_RM.

As can be seen in the graph in Fig. 12, resource capabilities, operation interface and blank characteristics (left-hand part of the graph) participate in the final characteristics of the part (right-hand part of the graph). This can be clearly seen in the loop corresponding to the Slot_Bottom_Position characteristic, in which different interface characteristics participate, particularly the one due to part loading. This Char_Part>Loading_Interface is a characteristic between the Plane_E feature and the feature of the fixture Chuck_2 with which it is matched.



TT_A2 = Turning_Tool_A2; LR_i = Loading_Resource_i

Figure 12. Recovered graph of a validated process plan for a *Slot Machining* operation.

Following a variant approach, the planner must adapt the recovered plan when a characteristic differs from the one in the case study. As the equivalent characteristic *Slot_Depth* (Fig. 13 (a)) differs from *Slot_Bottom_Position* (Fig 12), in order to close the loop a new characteristic *Distance_PlaneE_PlaneD* is introduced, which relates the feature *Slot* with *Plane_E*. This new characteristic, resulting from stacking all the characteristics present in the loop, must be compared with the capability of the complex resource in order to validate the new process stage.

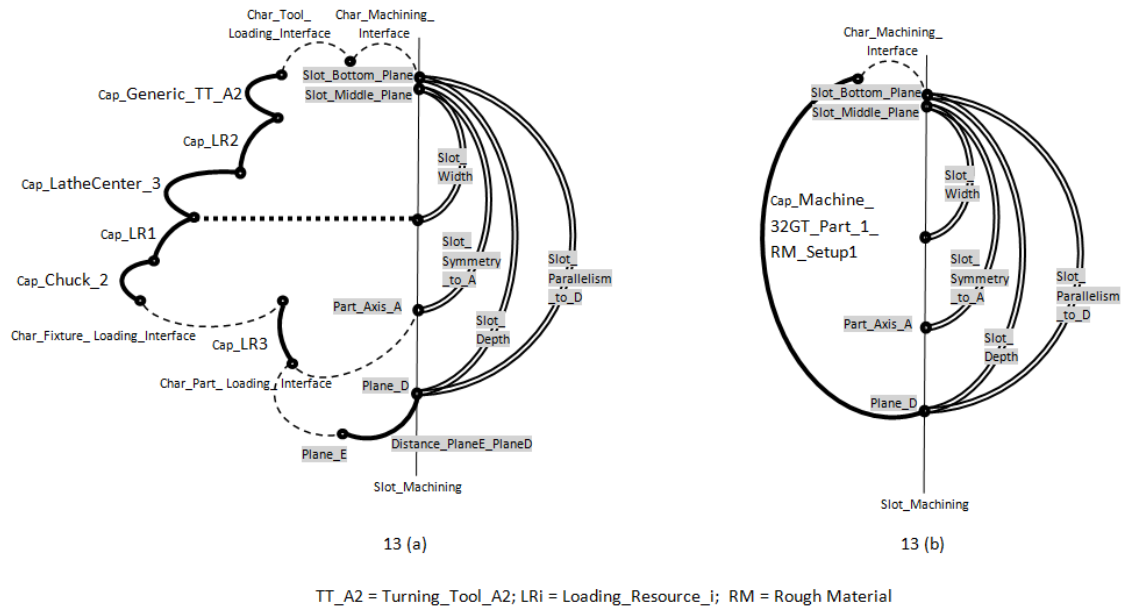


Figure 13. Operation multi-characteristics graph of: (a) initially adapted process plan for Part_1; (b) the modified process plan including an additional *Setup* operation.

If this evaluation is positive, the stage plan thus recovered can be used directly. However, if the evaluation is negative, the planner must propose chain modifications to adjust the value of the capability of the resource to an acceptable value. To do so, the planner can follow two alternative methods: a) substitute one of the resources for another which has better capabilities; or b) incorporate *Setup* activities, as was indicated in section 3.3, allowing substitution of a loop portion by a single link.

In this case the second option is chosen, incorporating the *Part_Tool_Setup_on_Machine* activity into the plan. This alternative, which can be seen in Fig. 13 (b), will only be valid if the measuring capability (uncertainty) of the resource that executes the *Setup* activity is less than 0.03 mm. This is a condition that is fulfilled if the setting is carried out using the on-machine measuring capabilities of the resource Machine_32GT.

Once the configuration of the resource that allows the critical slot characteristic (Slot_Depth) to be obtained has been defined, the validation will be extended to the rest of the critical characteristics of other features to be obtained in the same stage.

The case study has been centered on resource assignment and validation tasks at aggregated and supervisor levels. The MIRC ontology also supports knowledge representation relating to the sequencing and grouping of operations, a function that it inherits from the PPDRC ontology.

5. Conclusions and future work

The MIRC ontology is a specialization of the PPDRC ontology which details the activities and capabilities of manufacturing resources with the aim of supporting the activities that are executed during machining and inspection process planning. The fact that the PPDRC ontology is based on the foundational ontologies, PSL and DOLCE, gives the MIRC ontology the semantic inter-operability required for its integration with other specific ontologies considered within the collaborative integrated framework.

Thus, the most relevant contribution of this proposal is its capacity to represent the social and agentive character of the manufacturing resources, aiming to achieve integration of the process planning of both the product definition process and the integrated machining and inspection process.

Other relevant aspects that characterize the MIRC ontology are: a) the homogeneous treatment given to the different activities (machining and inspection operations and preparations), to the objects that participate in the execution of the activities (raw material and product in process, machines, tools, people in charge of preparation, etc.) and to the characteristics (capabilities of resources, characteristics of the part, etc.); and b) the way in which the characteristics of an output object are

quantified, which means adopting values for the capabilities of the resource that executes the activity and the setting of specific values for the input and control objects.

To check the validity of the proposal a case study has been presented showing how the process planner can consult a KB, supported by the MIRC ontology, to obtain the knowledge necessary for effective decision-making concerning resource assignment tasks and validation of the solutions used. Moreover, it is obvious that the query formulation is extremely flexible, thus allowing the planner to work using different approaches (variant and generative) and strategies (forward and backward) at each level of process planning (aggregated, supervisor and operational).

To support the planner's tasks, an original graphical representation has been developed, valid for different process plan approaches and levels of detail. This representation helps visualize how GD capabilities and characteristics data from the MIRC participate and are transmitted along the execution of plan activities. This representation is the essential component of a methodology based on new concepts and relations about part characteristics, resource capabilities and interfaces, which represent the activity execution characteristics governed by the control characteristics.

Future lines of research include: a) extending the scope of the MIRC ontology to cover complex resources with a level of aggregation higher than those of type *Machine*, such as workstations, workcells and virtual workcells; b) looking more deeply into the nature and typologies of the objects with a *Control* role and their influence on the quantification of activity execution; and c) broadening the characteristics studied, which in this first part of the research has been limited to the domain of dimensional and geometric quality, to include other resource capabilities (power, range of advances, cutting speed, material removal rate, etc.) related to different plan efficiency indicators.

Acknowledgments

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

References

- Ameri, F., Ch. McArthur, B. Asiabanpour, and M. Hayasi. 2011. "A Web-based Framework for Semantic Supplier Discovery for Discrete Part Manufacturing." Paper presented at the 39th SME/NAMRC (North American Manufacturing Research Conference), Corvallis, Oregon, June 13-17.
- Borgo, S., and P. Leitao. 2007. "Foundations for a Core Ontology of Manufacturing." In *Ontologies: A Handbook of Principles, Concepts and Applications in Information Systems*, edited by R. Sharman, R. Kishore, and R. Ramesh, 751–775. New York: Springer.
- 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.
- Cai, M., W. Y. Zhang and K. Zhang. 2011. "ManuHub: a Semantic Web System for Ontology-based Service Management in Distributed Manufacturing Environments." *IEEE Transactions on Systems, Man and Cybernetics, Part A: Systems and Humans* 41 (3): 574–582. doi: 10.1109/TSMCA.2010.2076395.
- Fadel, F. G., M. S. Fox, and M. Gruninger. 1994. "A Generic Enterprise Resource Ontology." Paper presented at the 3rd IEEE Workshop on Enabling Technologies. Infrastructure for Collaborative Enterprises, Morgantown, April 17–19.
- 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.
- González, F., F. Romero, G. Bruscas, and S. Gutiérrez. 2009. "Modelado de Actividades para el Desarrollo Integrado de Planes de Mecanizado e Inspección en Entornos Distribuidos y Colaborativos." Paper presented at the 3rd Manufacturing Engineering Society International Conference, Alcoy, Spain, June 17–19.

- Honggen, Z., J. Xuwen, P. Baojun, and Z. Xiaojun. 2012. "Ontology Theroy Based Process Planning and Supporting System under E-manufacturing." *Przeegląd Elektrotechniczny* 88 (1b): 107–110.
- ISA (Instrument Society of America). 2000. *Enterprise-Control System Integration. Part1: Models and Terminology*. ANSI/ISA–S95.00.01, North Carolina: American National Standards Institute.
- 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.
- Kantola, J. I. 2009. "Ontology-based Resource Management." *Human Factors and Ergonomics in Manufacturing* 19 (6): 515–527. doi: 10.1002/hfm.20181.
- Kuutti, K. 1995. "Activity Theory as a Potencial Framework for Human-computer Interaction Research." In *Context and Consciousness: Activity and Human Computer Interaction*, edited by B. Nardi, 17–44. Cambridge: MIT Press.
- Lin, Y. 2008. "Semantic Annotation for Process Models: Facilitating Process Knowledge Management via Semantic Interoperability." PhD diss., Norwegian University of Science and Technology.
- 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.
- Mariot, P., J. P. Cotton, C. Golbreich, A. Berger, and F. Vexler. 2007. "Querying Multiple Sources with OWL Ontologies: an Exploratory Study in an Automotive Company." Paper presented at the 3rd International Workshop OWL: Experiences and Directions (OWLED), Innsbruck, Austria, June 6–7.
- Martínez-Pellitero, S., J. Barreiro, E. Cuesta, and B. J. Álvarez. 2011. "A New Process-based Ontology for KBE System Implementation: Application to Inspection Process Planning." *International Journal of Advanced Manufacturing Technology* 57 (1-4): 325–339. doi: 10.1007/s00170-011-3285-7.

- Masolo, C., S. Borgo, A. Gangemi, N. Guarino, and A. Oltramari. 2003. *WonderWeb Deliverable D18. Ontology Library (final)*. Trento: Laboratory for Applied Ontology ISTC CNR.
- MIRC ontology. 2014. Accessed February.
http://coapp.webs.upv.es/ontologies/2014/0/MIRC_v1.owl.txt
- Newman, S. T., and A. Nassehi. 2009. "Machine Tool Capability Profile for Intelligent Process Planning." *CIRP Annals - Manufacturing Technology* 58 (1): 421–424.
- Ning, G., J. Tian-guo, and L. Wen-jian. 2010. "Research on Manufacturing Resource Domain Ontology Integration based on OWL." Paper presented at the 2nd International Workshop on Database Technology and Applications (DBTA), Wuhan, China, November 27–28.
- Oberle, D., A. Ankolekar, P. Hitzler, P. Cimiano, M. Sintek, M. Kiesel, B. Mougouie, S. Baumann, S. Vembu, M. Romanelli, P. Buitelaar, R. Engel, D. Sonntag, N. Reithinger, B. Loos, H.-P. Zorn, V. Micelli, R. Porzel, C. Schmidt, M. Weiten, F. Burkhardt, and J. Zhou. 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.
- Ramos, L. 2010. "Toward Dynamic Ontologies for the Industrial Manufacturing Domain." Paper presented at the 4th International Workshop on Ontology Dynamics, Shanghai, China, November 8.
- 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, October 16–21.
- 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.
- Solano, L., P. Rosado, and F. Romero. 2013. "Knowledge Representation for Product and Processes Development Planning in Collaborative Environments." *International Journal of Computer Integrated Manufacturing*. Available online: 10 Sep 2013. doi: 10.1080/0951192X.2013.834480.

- Sormaz, D. N., J. Arumugan, and S. Rajaraman. 2004. "Integrative Process Plan Model and Representation for Intelligent Distributed Manufacturing Planning." *International Journal of Production Research* 42 (17): 3397–3417.
- Vichare, P., A. Nassehi, S. Kumar, and S. T. Newman. 2009. "A Unified Manufacturing Resource Model for Representing CNC Machining Systems." *Robotics and Computer-Integrated Manufacturing* 25 (6): 999–1007.
- Zdravkovic, M., and M. Trajanovic. 2009. "Integrated Product Ontologies for Inter-organizational Networks." *Computer Science and Information Systems* 6 (2): 29–46. doi: 10.2298/csis0902029Z.