



Towards an ontological infrastructure for chemical batch process management

E. Muñoz, A. Espuña, L. Puigjaner*

Chemical Engineering Department – CEPIMA, Universitat Politècnica de Catalunya, Av. Diagonal 647, Pab. G-2, 08028 Barcelona, Spain

ARTICLE INFO

Article history:

Received 19 August 2009

Received in revised form

16 December 2009

Accepted 19 December 2009

Available online 4 January 2010

Keywords:

Ontology

Knowledge representation

Knowledge sharing

Batch process

Decision-support systems

ABSTRACT

A crucial step for batch process improvement and optimization is to develop information structures that streamline data gathering and, above all, are capable of integrating transactional data into a system using the analytical tools that are developed. Current trends in electronics, computer science, artificial intelligence and control system technology are providing technical capability that greatly facilitates the development of multilevel decision-making support. In this paper, we present the batch process ontology (BaPrOn), wherein different concepts regarding batch processes are categorized and the relationships between them are examined and structured in accordance with ANSI/ISA-88 standards, which provide a solid and transparent framework for integrating batch-related information. This paper also focuses on systematic integration of different actors within the control process. The proposed approach bases the conceptualization through the ANSI/ISA-88 representation, providing the advantage of establishing a more general conceptualization of the batch process domain. The capabilities of the envisaged ontological framework were assessed in a test bed PROCEL pilot plant: scheduling-monitoring and control-rescheduling was closed, information quality was accessed by knowledge description, and an optimum decision-making task was performed. The ontological structure can be extended in the future to incorporate other hierarchical levels and their respective modeling knowledge.

© 2009 Elsevier Ltd. All rights reserved.

1. Introduction

The chemical process industry (CPI) must face an ever-changing environment to meet current market needs. The management of chemical batch plants involves collecting and processing huge amounts of data, which are subsequently exploited. These data can be viewed as a valuable source of information for decision-making, regardless of the use of its analysis (such as in preformulation and new process development, supply chain management, scheduling, process control, fault analysis, etc.).

For 40 years, companies have been developing management information systems to help end users to exploit data and models, with the final objective of use in discussions and decision-making. Current global competition has made some of these decisions (related to aspects of manufacturing such as economic efficiency, product quality, flexibility and reliability) essential for the viability of the enterprise. Decision-support systems (DSS) are computer technology solutions that can be used to support complex decision-making and problem solving (Shim et al., 2002).

DSS are defined as computer-aided systems at the company management level that combine data and sophisticated analytic models to support decision-making (Simon & Murray, 2007). Clas-

sic DSS design is comprised of components for sophisticated database management capabilities with access to internal and external data, information, and knowledge; modeling functions accessed by a model management system; simple user interface designs that enable interactive queries, reporting, and graphing functions; and optimization by mathematic algorithms and or intuition/knowledge. Much research and practical design effort has been conducted in each of these domains.

Fig. 1 describes what probably came to be the most commonly used model of the decision-making process in a DSS environment. Typically, the phases overlap and blend together, with frequent looping back to earlier stages as more is learned about the problem, as solutions fail, and so forth. A first step is the recognition of the problem or an opportunity. Once the problem has been recognized, it is defined as a term that facilitates the creation of the model. Some authors consider that the emphasis is on the next two steps: model development and alternatives analysis. Then, the choice is made and implemented. As a final step and if necessary, a new recognition is performed. Obviously, no decision process is this clear-cut in an ill-structured situation (Shim et al., 2002).

There has also been a huge effort in the DSS field to build a group support system (GSS) or collaboration support systems to enhance the communication-related activities of team members who are engaged in computer-supported cooperative work. The communication and coordination activities of team members are facilitated by technologies that can be characterized along the three

* Corresponding author.

E-mail address: luis.puigjaner@upc.edu (L. Puigjaner).

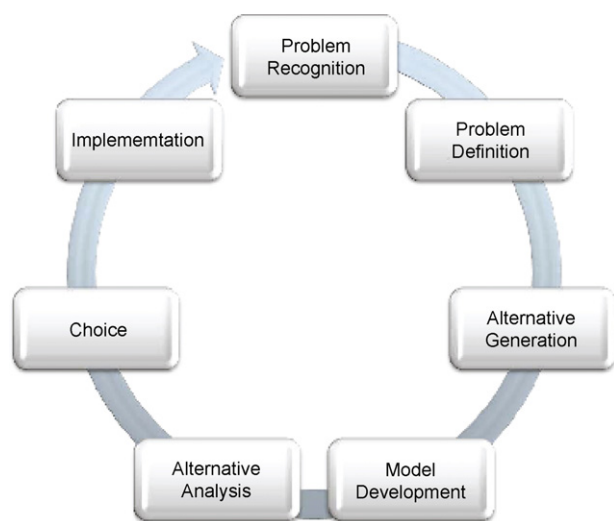


Fig. 1. Decision-making process.

continua of time, space, and level of group support (Alavi & Keen, 1989; Shim et al., 2002). Teams can communicate synchronously or asynchronously; they may be located together or remotely; and the technology can provide task support primarily for the individual team member or for the group's activities. These technologies are utilized to overcome the space and time constraints that burden face-to-face meetings, to increase the range and depth of information access, and to improve group task performance effectiveness, especially by overcoming "process losses".

From the previous discussion, it is clear that the need for infrastructures that continuously and coherently support fast and reliable decision-making activities related to the production process is now of paramount importance (Venkatasubramanian et al., 2006). This need is more evident when we consider recent activity in the fields of data warehousing, online analytical processing (OLAP), data mining and Web-based DSS, followed by the treatment of collaborative support systems and optimization-based decision support (Shim et al., 2002). It is quite common for batch process activities to have large databases. Hence, an enormous amount of information is created, stored and shared and it may be hard to find the right information when it is required. Furthermore, because of the possible use of different computer languages and differences in conceptualization, the interoperability between information in different systems is one of the most critical aspects in the daily operation of many organizations.

Indeed, shrinking profits have made it essential to exploit large databases (as companies need to manufacture many non-cyclical products with complex recipes, etc.) using non-generic/blind methods. One key aspect is information extraction, which should result in the extraction of information quality. Information quality can be defined as precise information in terms of time, content, and clarity (Eppler, 2006). A common problem is that this data extraction process may be performed using blind methods in many cases. However, the performance of such methods can be drastically improved by combining them with knowledge or expertise of the process.

Information is data that is processed to be useful. Knowledge is the application of data and information through the development of a system which models, in a structured way, the experience gained in some domain. Knowledge exists as soon as human interaction is or has been made available in any step of the product/process development (Gebus & Leiviskä, 2009). In recent years, there has been an effort to create knowledge with a minimum human interface, either in a straight and formal way (e.g. expert systems) or

in a conceptual manner. The use of multiple models to represent detailed and abstract knowledge of chemical processes has been taken into account recently. In particular, this knowledge representation enables us to identify process sections together with their function, objectives and relations within the process. This enables the automatic generation of alternative views of the process, organized in a hierarchy of different levels of abstraction. The approach is similar to reverse engineering.

Other requirements that a knowledge-representation system should meet are natural representation of the physical reality/model (items of equipment/devices), automatic generation of abstraction levels to identify which sections of the process can be potentially improved, and the implementation of computer reference model sets (Rodríguez-Martínez et al., 2004). The computer reference model is a large-scale, hierarchically arranged computer system that integrates plant management, plant production scheduling, inventory management, individual process optimization, and unit process control for all of the plant's operating units as a whole. The reference model defines requirements that are common to all implementations, but it is independent of the specified requirements of any particular implementation that are applicable to existing processes.

The way to address these problems is to reduce or eliminate conceptual and terminological confusion and come to a shared understanding. Such an understanding can function as a unifying framework for the different viewpoints of an ontology that has been adopted to develop an integrated framework, through the definition and semantic description of data and information. This is the basis for modeling the different forms of knowledge that are to be organized or the contextualized information that can be used to produce new meanings and generate new information.

Ontologies constitute a means of specifying the structure of a domain of knowledge in a generic way that can be read by a computer (formal specification) and presented in a human-readable form (informal specification). Moreover, ontologies are emerging as a key solution to knowledge sharing in a cooperative business environment (Missikoff & Taglino, 2002). Since they can express knowledge (and the relationships in the knowledge) with clear semantics, they are expected to play an important role in forthcoming information-management solutions to improve the information search process (Gruber, 1993; Obrst, 2003). Recently, several ontologies have been developed in the field of chemical processes. They include OntoCAPE, which is a formal, heavyweight ontology for the domain of (chemical) process engineering. In the chemical domain, the design, construction, and operation of chemical plants are considered the major engineering activities (Bayer & Marquardt, 2004; Morbach, Yang, & Marquardt, 2007; Morbach, Wiesner, & Marquardt, 2009). Another well-known ontology is ontological informatics infrastructure for pharmaceutical product development and manufacturing (POPE), which is an informatics framework in the pharmaceutical product development domain, to support decision-making in the entire process, including drug formulation design, process simulation and process safety analysis for the active pharmaceutical ingredient process, as well as drug product development (Venkatasubramanian et al., 2008; Zhao et al., 2006). BioPortal is an open repository of biomedical ontologies. In this portal, researchers in biomedical informatics can submit their ontologies and access them using the BioPortal user interface or Web services (Board et al., 2009). This allows searches for terms and biomedical resources, and also provides comments within ontologies or the option of adding ontology mappings.

In general, the aforementioned ontologies offer alternatives for sharing knowledge in the process and engineering domains. However, many of the ontologies that have been developed to date are focused on existing product development processes. In other words, these approaches create an ontology that is suitable

for on-going processes. However, when some standard structures are taken into account, the opportunity to improve data, information and knowledge is enormous. This work presents the development of a generic ontology following ANSI/ISA-88 standards (International Society for Measurement and Control, 1995, 2006, 2007; Shirasuna, 2007), which allows infrastructure to be created that should be general enough to be applied to any batch system. These standards were developed on the basis of the Purdue reference model, which is a detailed collection of the functional requirements of the generic information management and control to run a batch manufacturing process (Williams, 1989). Additionally, the proposed ontology may be used as a straightforward guideline for standardizing batch process management and control. Moreover, it offers a robust structure for sharing knowledge, which may help to make more appropriate decisions that improve business performance.

2. Analysis requirements

The problem of batch process management has been studied in many works (Mendez, Cerda, Grossmann, Harjunkoski, & Fahl, 2006; Rippin, 1983; Shah, 1998; Subrahmanyam, Bassett, Pekny, & Reklaitis, 1995). These works have made a rich and vast description of such problems and have pointed out open issues where research efforts may provide significant improvements, in which the in-depth description aids understanding.

This section aims to describe the particular elements involved in current ontology development, which affect its subsequent performance assessment.

2.1. Plant requirements

Effective production is very important in today's global competitive environment. In the case of the batch process industry, multi-product and multipurpose plants, as well as continuous or semi-continuous processes, manufacture a variety of products through a sequence of operations that share available resources, intermediate products, and raw materials. The efficient use of such resources can be analyzed at different levels:

- If we focus on the planning area and control level, a process system involves multiple and interrelated activities that are performed at single or multiple sites, with different durations and amounts of information. Generally, information flows from the marketing department to the manufacturing department, which determines the production schedule that is needed to meet the sales strategies. In its most general form, the scheduling problem requires information that is related to the configuration of the plant (the available equipment units and resources), the product recipes (the set of processing tasks and resources required to manufacture a given product), precedence relationships between materials and final product requirements (demands and related due dates) (International Society for Measurement and Control, 1995).

At this level, a closed-loop framework is proposed for the decision-making task of batch chemical plants. This framework integrates both scheduling and control and is based on ANSI/ISA-88 standards (BaPrOn). Integration enables future events to be taken into account at scheduling level, in order to minimize the occurrence of critical situations during the execution of the process. It also provides any required information from the process measurements that are made by the control and fault diagnosis system. On the other hand, the opportunity for reactive scheduling allows the process to respond under unexpected schedule deviations or abnormal events.

- In a batch plant production environment, the occurrence of unpredictable events is usually unavoidable. Such events may be related to external market factors or to the intrinsic plant operation, and include equipment breakdowns and variable operation times. Despite the uncertainty in the environment, the scheduler has to make some decisions in order to start production and to face uncertainty when an abnormal event occurs.
- In addition, the integration of a control and monitoring system into process management helps to provide the process state information opportunely at different levels in the decision-making hierarchical structure, thus reducing the risk of incidents and improving the efficiency of the reactive scheduling by updating the schedules in the most effective way, which improves the process yield.

Unexpected events or disruptions can change the system status and affect its performance. Deviations from the original schedule and information about equipment breakdowns that is provided by the control and monitoring system will eventually trigger rescheduling. However, the schedule that is generated will be assessed according to the new plant situation. Thus, if some modifications are made, the newly created schedule will be translated into some control recipes for the actual process. Consequently, ontology can also be used to ensure the robustness of the running plan in the system.

The rescheduling system allows different dispatching rules, optimizers and objective functions to be selected, according to the process knowledge. Alternative rescheduling techniques (recalculate a new robust schedule, update operation times, reassignment, etc.) are evaluated and a system should select the most suitable ones, according to the objective function that is adopted. Optimization algorithms may be included, depending on the interest of the decision maker and the required reaction time.

2.2. Recipe requirements

This integration approach follows the ANSI/ISA-88 batch control standard, which differentiates between four types of recipes: general, site, master and control. The general and site recipes are general recipes that are outside the scope of the control system. At control level, the information recipes are the master and control recipes.

Master recipes are derived from site recipes and are targeted at the process cell. A master recipe is a required recipe level; without it, control recipes cannot be created and batches cannot be produced. Master recipes take into consideration the equipment requirements within a given process cell. They include the following information categories: header, formula, equipment requirements and procedure. Control recipes are batches that are created from master recipes. They contain the product-specific process information that is required to manufacture a particular batch of product. They also provide the detail needed to initiate and monitor equipment procedural entities in a process cell.

2.3. Control activity requirements

The control activity model from International Society for Measurement and Control (1995) (Fig. 2) provides an overall perspective of batch control and shows the main relationships between the various control activities. The control activities define how equipment in the batch manufacturing plant will be controlled.

Using this approach, recipes can be modified without changing the code of the PLCs and DCS that run the basic regulatory control. Moreover, recipes can run on different sets of equipment.

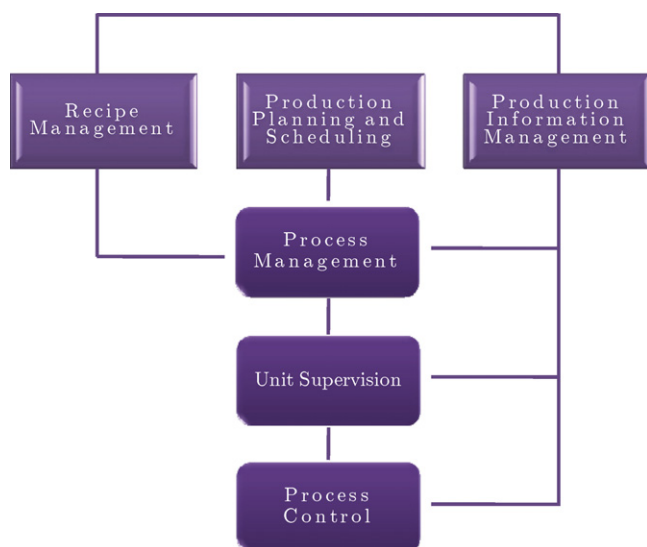


Fig. 2. Control activity model.

The use of the ANSI/ISA-88 standard significantly reduces costs and implementation time through:

- Effective utilization and optimization of plant equipment, which maximizes total plant production capabilities.
- Reductions in total validation costs and production down-time via separate validation of recipe procedures and equipment.

2.4. Final problem statement

In summary, the lack of integration among the control system's hierarchical levels does not allow complete optimization of scheduling. Information is needed from different hierarchical levels when an important change is required. However, the desired change cannot be made unless the system be enough robust. In general, at planning level, the schedule is optimized with the static information contained in the recipes, and a semi-improvement is made. In addition, the need to integrate the different modeling approaches in a hierarchical decision-support system means that consistent terminology and concepts must be used to improve the communication and collaboration tasks over the entire system.

3. Proposed approach: ontology-based infrastructure

An ontology defines the basic terms and relations that comprise the vocabulary of a topic area as well as the rules for combining terms and relations to define extensions to the vocabulary (Neches et al., 1991). It is understood as an explicit specification of a conceptualization (Gruber, 1993). However, since Gruber's definition, various improvements have been made to the definition of "an ontology". For example, an ontology has been described as a hierarchically structured set of terms for describing a domain that can be used as a skeletal foundation for a knowledge base (Swartout, Neches, & Patil, 1993).

Ontologies provide the shared and common domain structures that are required for the semantic integration of information sources. Though it is still difficult to find consensus among ontology developers and users, there is some agreement about protocols, languages and frameworks. Ontologies are hierarchical domain structures that provide a domain theory, have a syntactically and semantically rich language, and a shared and consensual terminology (Klein, Fensel, Kiryakov, & Ognyanov, 2002).

Ontologies were created to help in knowledge reuse and sharing: reuse means building new applications by assembling components that have already been built, while sharing occurs when different applications use the same resources. Reuse and sharing have the following advantages: they are cost, time and resources effective (Fensel, 2003). In this work the batch process ontology (BaPrOn) is presented. In BaPrOn a conceptualization through the ANSI/ISA-88 representation provides an advantage of establishing a more general conceptualization in the batch process domain. Such a generalization is behind years of joint work by recognized batch manufacturing experts who met to define a perceptive view of batch plants organization and its corresponding hierarchy of control functions. As a consequence, following the ANSI/ISA-88, virtually all activities concerning batch processes can be properly represented.

BaPrOn is a procedural oriented ontology that supports the management of different concepts (physical models, procedures, functions and processes) in accordance with ANSI/ISA-88 batch process standards, categorizing them and examining the relationships between them. This will allow the association between the elements mentioned before, and the further identification of any information resource if it would be required.

3.1. Language

Different ontology languages provide diverse facilities. Any language used to codify ontology-underpinned knowledge should be expressive, declarative, portable, domain independent and semantically well defined. The language used in an ontology is essential for its future implementation and sharing. We adopted web ontology language (OWL), as it has good characteristics for ontologies (Bechhofer et al., 2004). OWL has been designed for use by applications that need to process the content of information, instead of just presenting the information to humans. OWL facilitates greater machine interpretability of Web content than that supported by extensible markup language (XML), resource description framework (RDF), and resource description framework schema (RDF-S), as it provides additional vocabulary along with formal semantics. The ontology formally describes the meaning of the terminology used in documents. If machines are expected to perform useful reasoning tasks on these documents, the language must go beyond the basic semantics of RDF schema.

OWL has been designed to meet this need for a web ontology language, and is part of the growing stack of W3C recommendations that are related to the semantic web.

- XML provides a surface syntax for structured documents, but imposes no semantic constraints on the meaning of these documents (XML-Core-Working-Group, 2009).
- XML Schema is a language that restricts the structure of XML documents and extends XML with data types (McQueen & Thompson, 2000).
- RDF is a data model for objects ("resources") and relations between them. It provides simple semantics for data models, which can be represented in XML syntax (Klyne & Carroll, 2002).
- Schema is a vocabulary for describing properties and classes of RDF resources. It includes semantics for generalization hierarchies of these properties and classes (Brickley & Guha, 2002).

OWL adds more vocabulary for describing properties and classes, including the relations between classes (e.g. disjointness), cardinality (e.g. "exactly one"), equality, richer typing of properties, characteristics of properties (e.g. symmetry), and enumerated classes.

The semantics in the ontology build on XML's ability to define customized tagging schemes and RDF's flexible approach to rep-

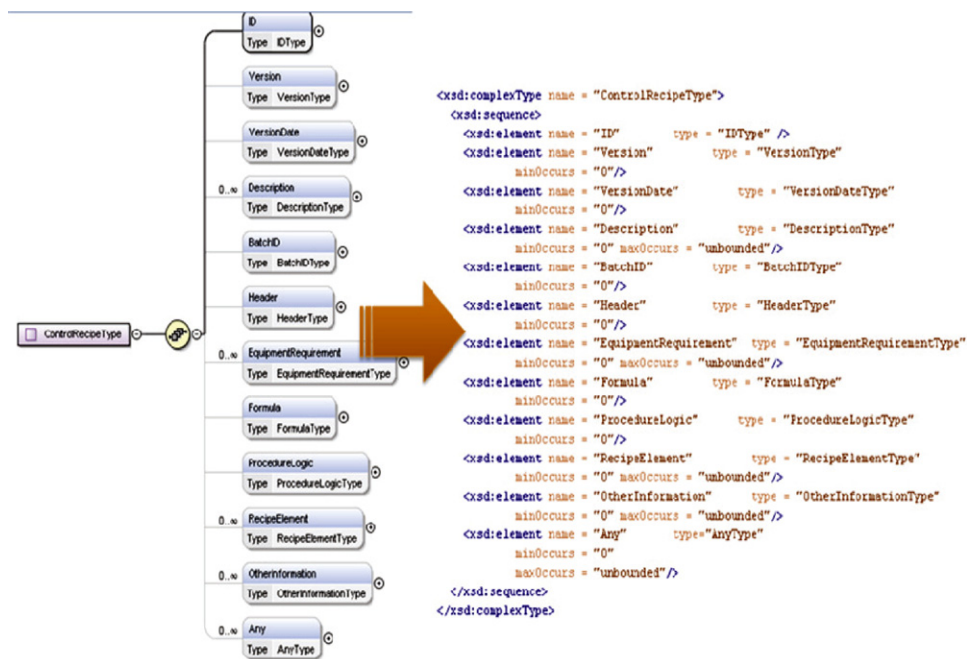


Fig. 3. XML schema for a recipe.

resenting data. This unifying aspect makes it easier to establish, through collaboration and consensus, the utilitarian vocabularies (between ontologies) needed for far-flung cooperative and integrative applications using the World Wide Web and internal servers. The uses of these languages are helpful for the first task of the ontology, which is to become a standard tool for vocabulary, format, and definitions. Restrictions and reasoning make communication possible between the different system elements. Fig. 3 shows a master recipe standard imported from the World Batch Forum (Brandl & Emerson, 2003). From left to right, a XML schema which may be more comprehensible to humans is shown, as well as XML lines that both machines and devices are capable of reading.

3.2. Methodology

Currently, numerous ontologies are being developed and used in various research areas. Each development project usually follows its own set of principles, in order to design criteria and phases in the ontology development process. The absence of standard guidelines and methods hinders the following: the development of shared and concentrated ontologies within and between projects; the extension of a given ontology by others; and its reuse in other ontologies and final applications. It is widely recognized that constructing ontologies, or domain models, is an important step in the development of knowledge-based systems (KBSS). However, there is a lack of consensus on a uniform approach to designing and maintaining these ontologies. Various methodologies exist to guide the theoretical approach that is taken, and numerous ontology building tools are available. The problem is that these procedures have not coalesced into popular development styles or protocols, and the tools have not yet matured to the degree one would expect in other software instances.

The methodology used in this paper is based in two ontology development methodologies "Methontology" (López, Gómez-Pérez, Sierra, & Sierra, 1999) and "On-To-Knowledge" (Sure & Studer, 2002).

Methontology provides support for the entire life cycle of ontology development. It enables experts and ontology makers who are unfamiliar with implementation environments to build ontologies

from scratch. Methontology identifies the following activities in the development of an ontology: specification, knowledge acquisition, conceptualization, integration, implementation, evaluation, and documentation. The life cycle of the ontology is based on the refinement of a prototype and ends with a maintenance state. The most distinctive aspect of Methontology is the focus on this maintenance stage.

In contrast, On-To-Knowledge methodology includes the identification of goals that should be achieved by knowledge management tools and is based on an analysis of usage scenarios. The steps proposed by On-To-Knowledge are: (i) kick-off, in which some competency questions are identified, potentially reusable ontologies are studied and a first draft of the ontology is built; (ii) refinement, in which a mature and application-oriented ontology is produced; (iii) evaluation, in which requirements and competency questions are checked and the ontology is tested in the application environment; and finally (iv) ontology maintenance. On-To-Knowledge stresses that the ontology modeling process should start with a definition of the abstraction level, which is strongly dependent on the usage of the ontology.

All the aforementioned methodologies have been inserted into the Plan, Do, Check and Act Cycle (PDCA), (<http://www.hci.com.au/hcisite3/toolkit/pdcacycl.htm>), which results in an ordered sequence of steps, that are easy to understand and track (Fig. 4).

3.3. Ontology layering and architecture

The architecture of BaPrOn could be considered as nominal ontology and high ontology, taking into account the factors that influence the complexity, such as concepts, taxonomy, patterns, constraints and instances, as show in Table 1.

The bases for the construction of this particular ontology include concepts and relations taken from the ANSI/ISA-88 standard. The coherence between the concepts in the conceptualization ensures that there is compatibility between the control elements at different control levels. This provides more opportunities for applications in which information is updated at the same rate as data are received.

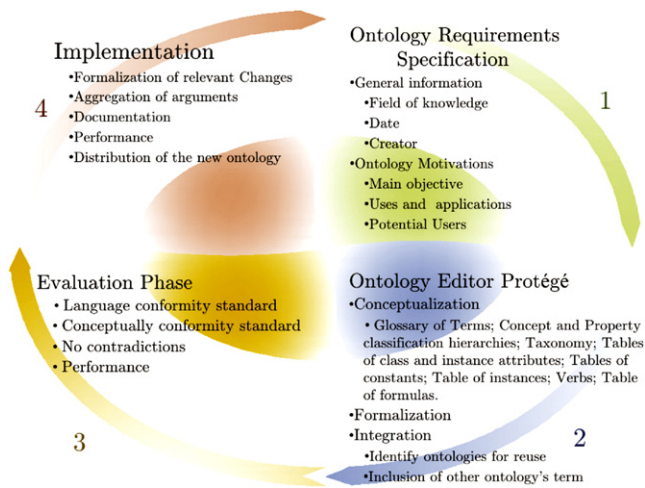


Fig. 4. Methodology for developing BaPrOn.

Table 1
Rating for the conceptualization complexity of ontologies.

Rating	Conceptual model criteria
Very low	Only concepts
Low	Taxonomy, high number of patterns, no constrains
Nominal	Taxonomy with properties general patterns available, some constrains
High	Taxonomy with properties and axioms, few modeling patterns, considerable number of constrains
Very high	Taxonomy with properties, axioms and instances, no patterns, considerable number of constrains

Logics are added to concepts, which express the relationships between the concepts and the relations. The BaPrOn ontology is logic-based, includes the logics and has six main layers. Fig. 5 introduces the six layers that are seen as the main axis of the ontology architecture.

Table 2
Some concepts from ANSI/ISA-88 part 1: models and terminology.

Name	Description	Type
Batch process	A process that leads to the production of finite quantities of material by subjecting quantities of input materials to an ordered set of processing ...	Concept
Batch	The material that is being produced or that has been produced by a single execution of a batch process. An entity that represents the production ...	Concept
Recipe	The necessary set of information that uniquely defines the production requirements for a specific product. There are four types of recipes defined ...	Concept
Header	Information about the purpose, source and version of the recipe such as recipe and product identification, creator, and issue date.	Concept
Formula	A category of recipe information that includes process inputs, process parameters, and process outputs.	Concept
Process	A sequence of chemical, physical, or biological activities for the conversion, transport, or storage of material or energy. A process consists of ...	Concept
Process stage	A part of a process that usually operates independently from other process stages and that usually results in a planned sequence of chemical or ...	Concept
Process operation	A major processing activity that usually results in a chemical or physical change in the material being processed and that is defined without consideration of the	Concept
Process action	Minor processing activities that are combined to make up a process operation.	Concept
Enterprise	An organization that coordinates the operation of one or more sites. Must contain a site.	Concept
Site	A component of a batch manufacturing enterprise that is identified by physical, geographical, or logical segmentation within the enterprise. A site may ...	Concept
Area	A component of a batch manufacturing site that is identified by physical, geographical, or logical segmentation within the site. An area may ...	Concept
Process cell	A logical grouping of equipment that includes the equipment required for production of one or more batches. It defines the span of logical control of ...	Concept
Unit	A collection of associated control modules and/or equipment modules and other process equipment in which one or more major processing activities can	Concept
Equipment module	A functional group of equipment that can carry out a finite number of specific minor processing activities. An equipment module is typically centered ...	Concept
Control module	A collection of sensors, actuators and associate processing equipment that acts as single entity from a control stand point, that can carry out basic control.	Concept

The layers represent the following:

Concepts: A concept defines a basic and abstract idea that is commonly used in an ontology domain. It is represented as a word or phrase.

Relations: A relation describes the way in which two or more concepts are interrelated. It is usually described by a verb or verb phrase (basic properties).

Basic fact types: A basic fact type is a kind of primitive sentence or fact. It is composed of concepts and relations. If the basic fact type is always true in the ontology that contains it, it can play a role as an axiom in the logic-based ontology (detailed properties).

Constraints: A constraint is the restriction that is applied to a fact type (binary or numerical restrictions for properties).

Derivation rules: These are rules, functions or operators (including mathematical calculations or logical inference) that are used to derive new facts from existing ones.

Instances: Instances have the particularities of processing and are specifications in the applications of the upper layers.

3.3.1. Concepts

The concepts were taken from ANSI/ISA-88. The first part of this standard, entitled models and terminology, defines most of the main concepts that the standard has contributed to manufacturing automation. We extracted 83 concepts with their respective descriptions, as shown in Table 2, in which the columns Name, Description, Synonyms, Acronyms and Type give the developer more information.

3.3.2. Relations

The basic relations were taken from the assertions of the models that ANSI/ISA-88 describes. These assertions show the basic relations between physical model, recipe model and procedural model and are described by entity–relationship diagrams (E–R diagrams). For ANSI/ISA-88, recipes are needed because of the set of information that uniquely identifies the production requirements for a specific product. Fig. 6(a) shows the relations between recipes types. Some models are also described, such as the procedural con-

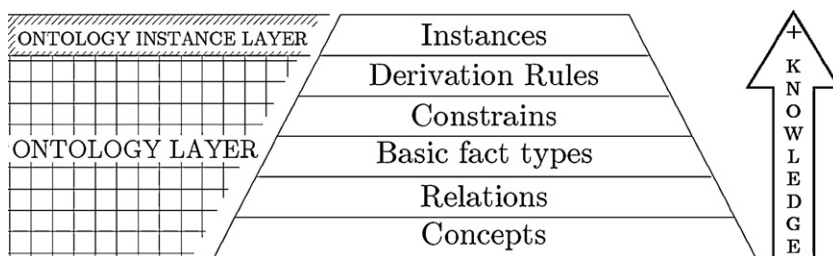


Fig. 5. Construct ontology layering.

Table 3
Recipe contents.

Concept	Description
Header	Administrative Information and process summary
Equipment requirements	Information about the specific equipment necessary to make a batch or a specific part of the batch
Procedure	Defines the strategy for carrying out a process
Formula	Describes recipe process inputs, process parameters, and process outputs
Other Information	Product safety, regulatory, and other information that does not fit in the other categories

trol model shown in Fig. 6(b), which describes the relation that occurs in a process cell. The process model shown in Fig. 6(c), which is very conceptual and describes the functionality needed to create a batch, describes the steps that must occur to make a product.

All these relations must involve the control model, process model and the physical model to accomplish process functionality. Fig. 7 shows some basic relations of the aforementioned models.

3.3.3. Basic fact types

Within ANSI/ISA-88, many facts are found that provide the necessary information for related classes in more detail. Some

examples are shown in Table 3, which describes five categories of information in a recipe.

3.3.4. Constraints

Some information about basic fact types is required. This assertion leads to the use of constraints. One example could be the following constraints, which are obtained from Table 3.

- The recipe must contain one header.
- The recipe must contain at least one procedure.
- The recipe must content at least one formula.

3.3.5. Derivation rules

This class of rules is used to link parts within the classes and properties. One example is the suggestion in the standard that “a batch management system and equipment control system should exist separately”. In other words, for the control recipe procedure and equipment control to occur, the control recipe procedure and an equipment phase must always exist.

3.3.5.1. Instances. The instances represent the particular reality of the domain. One example is a unit used in the process cell of CEPIMA's group Reactor 1 from PROCEL, which is described in Table 4.

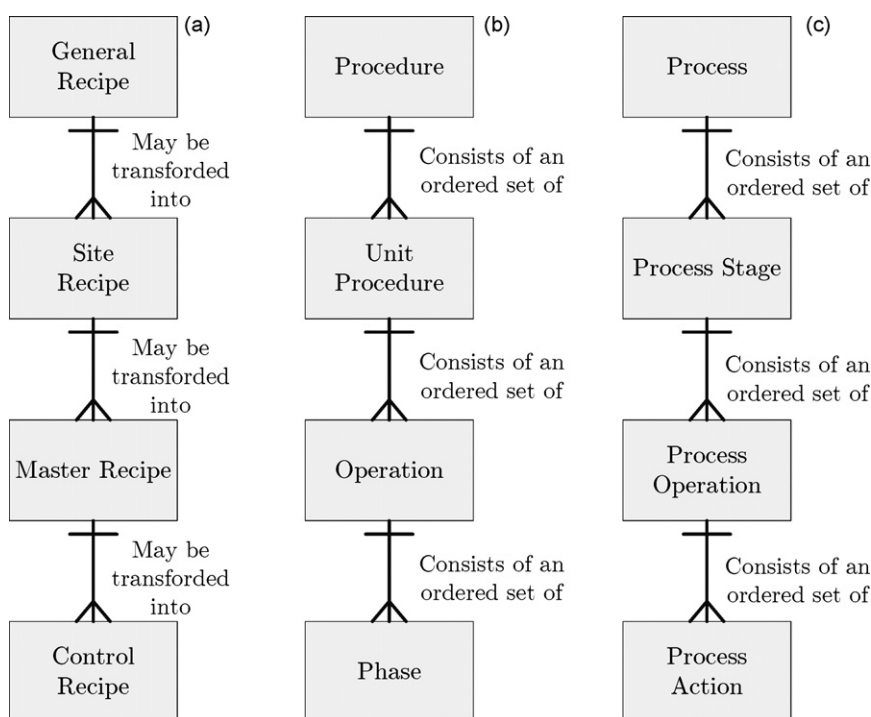


Fig. 6. ANSI/ISA-88 recipe, procedure and process models.

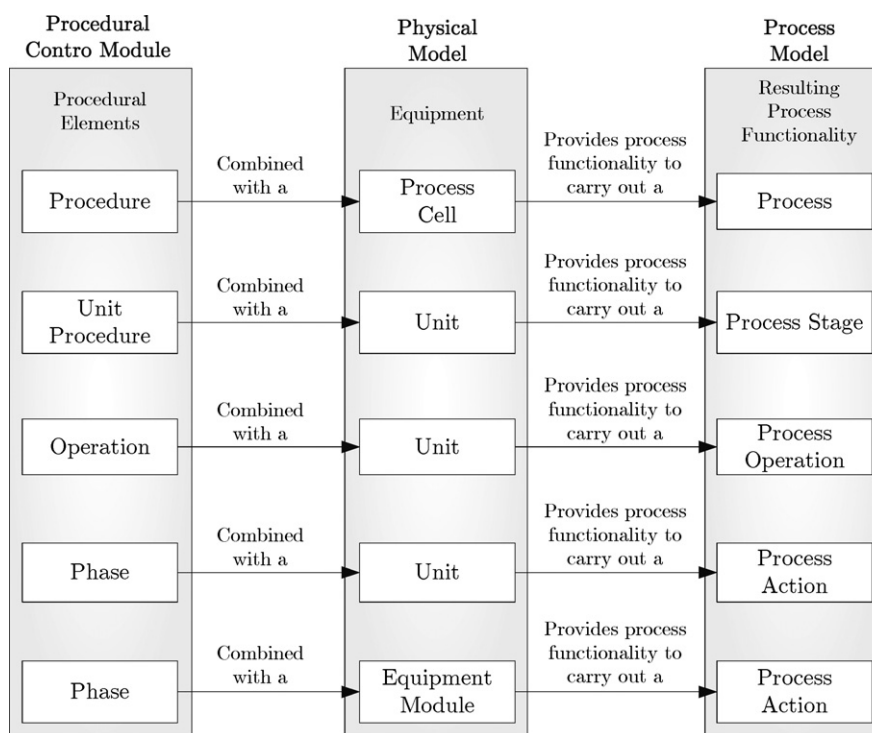


Fig. 7. ANSI/ISA-88 basic relations between models.

3.4. Implementation

We selected Protégé, a description logic reasoning system as a tool for ontology editing and knowledge acquisition (Horridge et al., 2007). This system acts as an inference engine to check data consistency and validity (<http://protege.stanford.edu/>). Protégé is a widely used open-source ontology and knowledge base editor

with a friendly user interface. Furthermore, we used Collaborative Protégé, a Protégé extension that enables users who develop an ontology collaboratively to hold discussions, chat, annotate ontology components and changes, all as an integral part of the ontology development process. From Protégé, it is possible to export ontologies to other knowledge-representation systems, such as:

- Resource description framework (RDF) is mainly intended for use in the semantic web, but it has also been described as a content management technology, a knowledge management technology, a portal technology, and as one of the pillars of e-commerce (Klyne & Carroll, 2002).
- Ontology inference layer (OIL) is intended to solve the findability problem, support e-commerce, and enable knowledge management (Horrocks et al., 2000).

Table 4
Instance description (PROCEL unit).

Concept	Description
Type	Glass reactor
Label	EQ1
Capacity	10 L
Process	Heating and holding

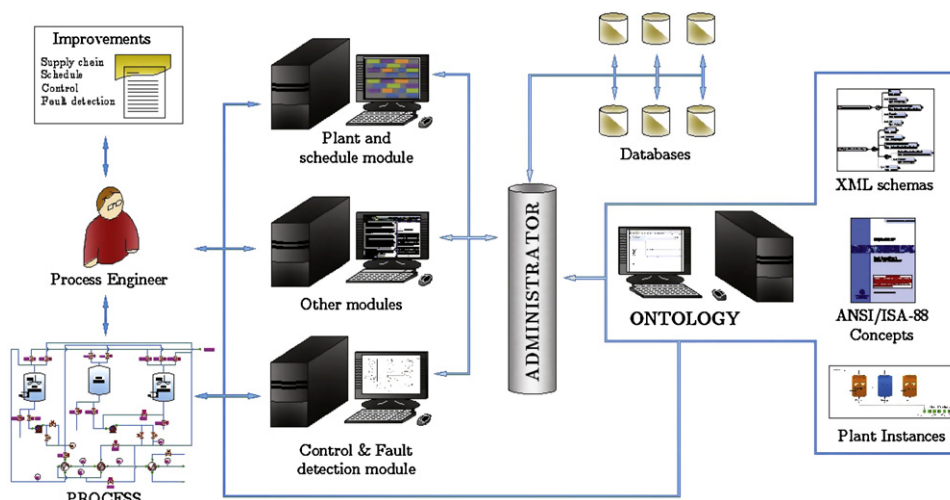


Fig. 8. Ontology modeling approach.

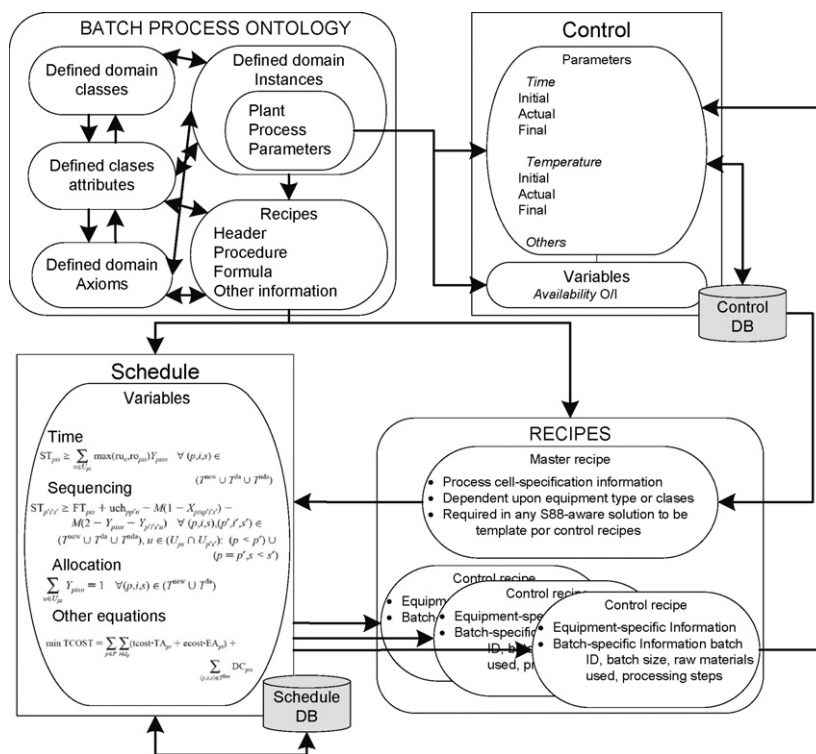


Fig. 9. System active part.

- The DARPA agent markup language (DAML) focuses on supporting the semantic web, though one would assume that it also has other uses (Pagels, 2006).

The proposed ontology is intended to promote transversal process-oriented management, to enable crossover among the different functionality silos in which businesses have typically been structured. In order to obtain (and manage) a comprehensive view

of the overall process, new modeling structures have been developed within BaPrOn. These structures can recognize the existing trade-offs and impacts of the available alternatives at the different information aggregation levels, and discard non-significant effects, through retuning the decision-making/optimization model according to the current process status. Thus, effective decisions can be made by avoiding both a greedy/myopic hierarchical decision-making structure and a monolithic optimization model (which is

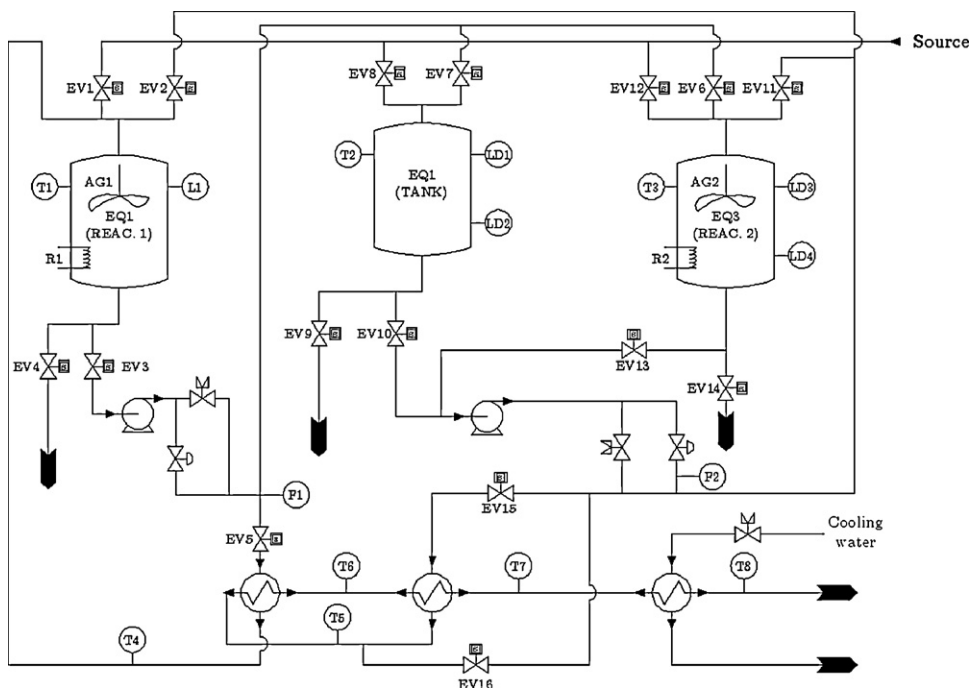


Fig. 10. PROCEL scheme of the flow shop plant of the case study.

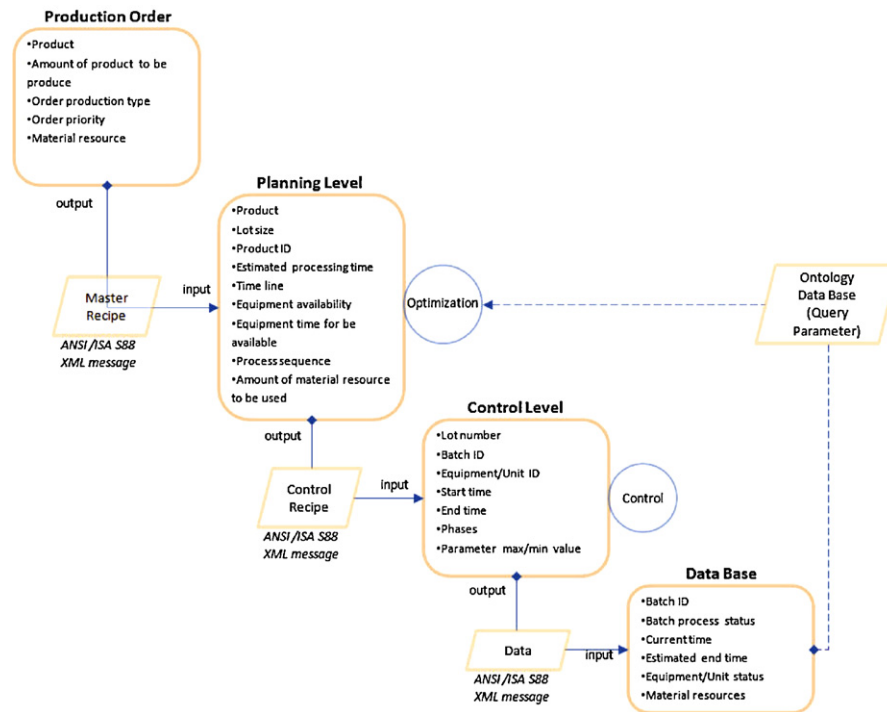


Fig. 11. Information required by the recipe at different levels.

Table 5
Batches processing times (min).

Product	i1		i2		i3	
	Unit (j)	Time	Unit (j)	Time	Unit (j)	Time
11	j1	30	j2	24	j3	50
12	j2	60	j3	12	j2	12
13	j3	59	j1	54	–	–

impossible to solve in industrial-size scenarios). The management process structures incorporated into this framework were designed following ANSI/ISA-88 standard recommendations (International Society for Measurement and Control, 1995, 2001, 2003, 2006, 2007). However, it is envisaged that this transversal ontology can be further extended to incorporate higher decision-making hierarchies (supply chain) and to include the entire life cycle of the organization, from the design stages to the delivery of final products.

4. Proposed ontology exploitation

A first application of this ontology has been implemented to “close” the typical scheduling-fault analysis-rescheduling loop (control levels 0 to 3 of the Purdue Reference Model) (International Society for Measurement and Control, 1995; Williams, 1989). The system is coordinated by an internal server acting as an information administrator that is consistent with the ontology structure. This can be achieved through the ontology web language-application program interface OWL-API, which is a Java

Table 6
Pipes transfer times (min).

From-to	Unit (j1)	Unit (j2)	Unit (j3)
Unit (j1)	0	5.8	9.6
Unit (j2)	10.4	0	5.6
Unit (j3)	10.3	11.8	0

interface and implementation for the W3C web ontology language (OWL).

The user interface interacts with a model of the ontology (batch processes domain) on the client side via a listener pattern. When the model needs to be filled with new information, the remote procedure call (RPC) module on the client side will invoke a request to the RPC module of the server, which interacts with the ontology and the collaboration APIs to provide the requested data. The aim is to make it easier for knowledge engineers and experts to manage knowledge. The working process of the ontology modeling approach is shown in Fig. 8, which illustrates the relation between different actors. In addition, it enables the user to reuse existing information from the modeling databases or to create and add new design information.

For the application, we activated the parts of the ontology that concern the management of recipe information between the schedule and control, as showed in Section 3. One of the main goals of the ontology is to act like the ANSI/ISA-88. XML recipe standards have been defined to provide an automated standardization for the generation and management of master recipes and control recipes.

Fig. 9 shows the parts of the ontology that are activated in the control process module, the schedule module and the recipe structures, which were mentioned in Section 3 and control the behavior of the interaction between real classes and properties, ruled by the axioms.

Table 7
Cleaning time (min).

Product	Unit (j)	Time
i1	j1	20
i2	j2	10
i3	j3	20
i1	j1	40
i2	j2	20
i3	j3	40
i1	j1	10
i2	j2	5
i3	j3	10

```

<OUTPUT_DATA>
<SCHEDULE NAME="Production plan" Description="Production plan for a
demand of 20 ton A and 24 ton B">
.....
<ProductionPlanInformation>
<ProductionPlanList>
<ProductionPlanListEntry ID="2" ">
<ProductionPlanListEntryType>Batch</ProductionPlanListEntryType>
<ProductID>2
</ProductID>
<ProductionPlanListEntryID="1" ">
<ProductionPlanListEntryType>Stage</ProductionPlanListEntryType>
<EquipmentID>1
</EquipmentID>
<ProductionPlanListEntryID="1" ">
<ProductionPlanListEntryType>Operation</ProductionPlanListEntryType>
<StartTime> 2.60</StartTime>
<EndTime> 3.10</EndTime>
<ProductionPlanListEntryType>Temperature</ProductionPlanListEntryType>
>
<Value> 36.79</Value>
</ProductionPlanListEntry>
</ProductionPlanListEntry>
</ProductionPlanList>
</ProductionPlanInformation>
</SCHEDULE></OUTPUT_DATA>

<ControlRecipe xmlns="http://www.wbf.org/xml/BatchML-V02"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
.....
<Parameter5>
<ID>SP_Tr</ID>
<Description>SetPoint for the temperature regulation
loop98</Description>
<ParameterType>Parameter</ParameterType>
<Value>
<ValueString>36.79</ValueString>
<!-- -->
<DataInterpretation>Parameter</DataInterpretation>
<DataType>float</DataType>
<UnitOfMeasure>°C</UnitOfMeasure>
</Value>
<Scaled>0</Scaled>
</Parameter5>
<Parameter6>
</RecipeElement2>
.....
</RecipeElement1>
</ControlRecipe>

```

Fig. 12. Temperature parameter following the ANSI/ISA-88 standard.

The process control module is responsible for controlling the phases, the actions and the specific values of the process variables of each unit as well as the transition logics between each phase inside an operation or procedure.

In contrast, the schedule module applies typical scheduling techniques and generates a priori production schedules that assume known and stationary operating conditions and demands along the entire time horizon. In the schedule, the underlying structure of the proposed mathematical model relies both on a continuous-time representation and the notion of general precedence. The generalized precedence notion extends the immediate predecessor concept to consider all the batches that belong to the same processing sequence. Broadly, this model handles allocation and sequencing decisions through different sets of binary variables, which are defined as follows:

- Y_{pisu} is a binary variable equal to one whenever task (p, i, s) , that is the s th stage for manufacturing the i th batch of product p and is allocated to equipment unit u .

- Regarding the sequencing decisions, X_{pis}^{pis} is a binary variable that establishes the general precedence relationship between a pair of tasks (p, i, s) and (p, i, s) executed at the same processing unit (otherwise X_{pis}^{pis} is meaningless). If X_{pis}^{pis} is equal to 1, task (p, i, s) is a direct or no direct predecessor of task (p, i, s) on the waiting line for allocated unit.
- Alternatively, in the case where task (p, i, s) is processed before task (p, i, s) in the same unit, X_{pis}^{pis} takes the value zero.

Finally, in order to achieve the aforementioned process, we created a recipe schema that is ontology aligned. The recipe schema was fitted for information concerns to the master and control recipes and was built in XML language. The recipes are filled automatically by a java application, which takes the required data from the different database modules (control and schedule). In general for recipes, the unit procedure was mapped to unit instances, and operation and phase levels were mapped to equipment level instances.

Since a production order exists, data for the master recipe are built from the information found within that order. The production order also contains information that is used to create and optimize the schedule. However, control recipes are built from the information resulting from the optimization process (schedule), which converts each piece of schedule into a control recipe for a specific product, unit and batch. These recipes are sorted and sent to the control module, using the respective data and information to complete the process.

5. Case study

A batch pilot plant (PROCEL), which is a basic environment for open simulation and optimization in a real time environment package scenario, is located in the laboratory facilities at the UPC Chemical Engineering Department. PROCEL provides an appropriate scenario for evaluating the ontology performance and for studying and developing new process strategies. This case study deals with the production of three chemical products, each with different production requirements (Table 5). The production system is comprised of three principal processing units, two reactors and one tank (Fig. 10). We considered one recipe for each product. Nominal processing times, transfer times (Table 6) and cleaning times (Table 7), are considered fixed for initial scheduling. However, they will be subject to changes, if required, to react to unexpected events. The changes follow the corresponding models by which the process is updated according to the information received from the plant. In Table 8 some control set points are taking in to account, which are the maximum values for maintaining the plant under control.

BaPrOn consists of 76 classes, 115 axioms, and 86 sub-object properties. These components make the ontology reasoning and its use possible. The instances have been introduced into another

Table 8
Control set points.

Control parameter	Value
Discharge EQ1 LTlmin	2.4 L
Charge QE1 LTlmax	4.4 L
Hold tank time	60 s
SP clean EQ1 time	90 s
SP clean EQ3 time	120 s
Heat EQ1 SP T1	50 °C
Heat EQ3 SP T3	50 °C
Heat EQ1 SP R1	100%
Heat EQ3 SP R3	50%
Heat EQ1 SPAG1	100%
Heat EQ3 SP AG2	20%

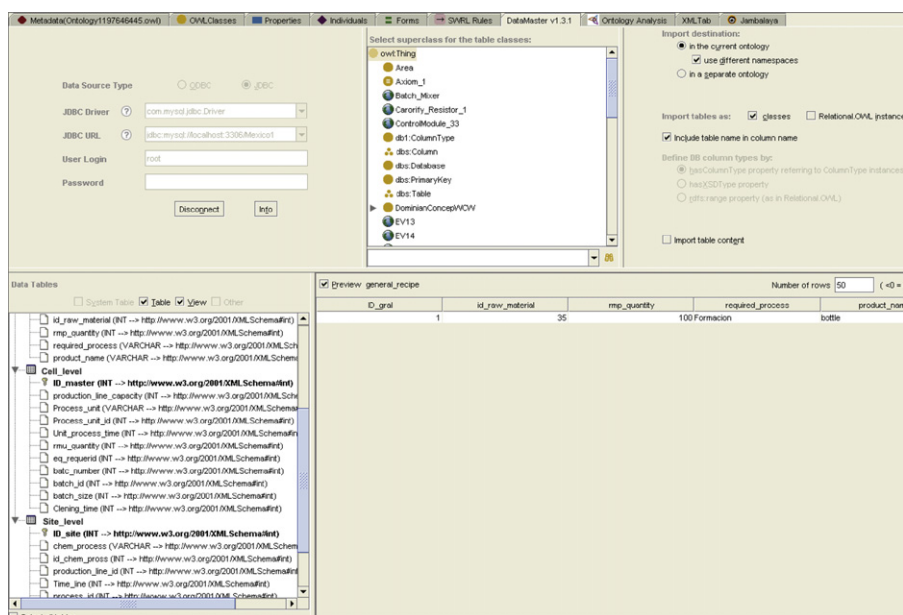


Fig. 13. Values shown by protégé from the process database.

ontology (instance layering; see Fig. 5), which is called from the main ontology as an extension. Thus, we take into account one of the main ontology principles, which is to be universal (subjective) in a particular domain, in this case the main ontology. The recipes need information founded inside the process. This information is shown in Fig. 11 and arrives to the schedule module and the control module, covering the required information.

Fig. 12 shows in a summarized way, how from the master recipe (unique file) some temperature parameter values (found from the 5 to 19 code string) are carried out to some control recipe (1 of 8 files) temperature parameters (found from 299 to 311 string line), by the application of the model standardized at these two control levels.

The control task in the units was applied to bring the processing time from the control level, through the ontology, to the schedule level. In this way variable values were updated when planning level asked for using them as input for scheduling. In order to succeed in this activity a link between data source from the external software variable and the real values captured by the control system in the process database has been created. Using the Protégé Database Manager facilitated this task. An example is shown in Fig. 13 where structured query language (SQL) process databases have been loaded for future information access.

The main BaPrOn classes and subclasses hierarchical taxonomy (Fig. 14) was built in accordance with ANSI/ISA-88 standards. Consequently, a better understanding of this standard is attained for

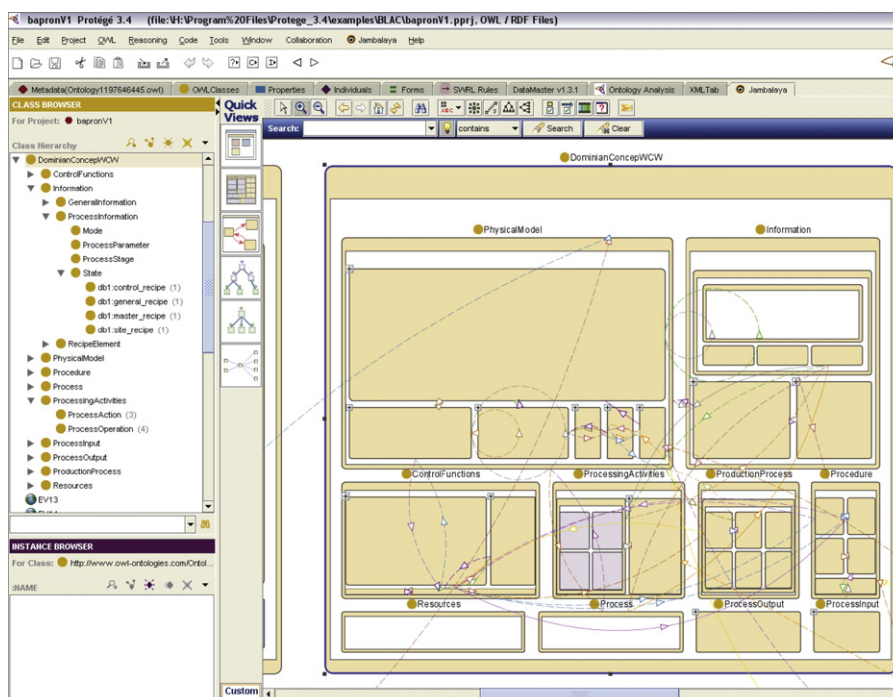


Fig. 14. ANSI/ISA-88 classes and subclasses hierarchical taxonomy.

Table 9

Comparison of process models at scheduling and control levels.

	No	Present steps	Time (min)	No	Proposed steps	Time (min)
Scheduling level	1	Input of external information source	2	1	XML information quality input	3
	2	Selection of specific and necessary information for the particular process	20	2	Capturing of information to the used particular language	4
	3	Translation and capturing of information to the used particular language, in the scheduling process	30	3	Scheduling optimization process	4
	4	Scheduling optimization process	4	4	XML quality information output	1
	5	Output as information source	2			
Control level	1	Input of external information source	2	1	XML information quality input	3
	2	Selection of specific and necessary information for the particular process	20	2	Capturing of information to the used particular language	4
	3	Translation and capturing of information to the used particular language, in the control process	30	3	Control parameter process	6
	4	Control parameter process	6	4	XML quality information output	1
	5	Output as information source	2			
		Total time	118		Total time	26

potential users, which facilitates the application and the implementation by the instantiation of the current batch process system. Another remarkable aspect is that when a system or a simple process is instantiated within the ontology structure, the relations ensure that every part or instance follows ANSI/ISA-88 standards requirements. As shown in Fig. 15, the slot information is required when a control recipe instantiation is performed.

5.1. Results

BaPrOn was successfully applied to a complex scenario of closing the scheduling–monitoring and control–rescheduling loop tanks for agile and quick information sharing and exchange among diverse control levels (0, 1, 2 and 3 levels from the Purdue CIM reference model) within the loop. As a result of the knowledge description, access to information quality was improved. The best decision-making task was achieved in this particular case study when the specified data from the control and fault diagnosis level were car-

ried at the scheduling control level. The following are the most significant benefits found in the case study:

Attending to knowledge diversity the technical effort required to deal with the process system and its representation along the knowledge found at different levels, ensured that all parts within the ontology were easily accessible. This is a particular and explicit way of representing the knowledge by the content format and the content type attributes found in the ontology structure classes.

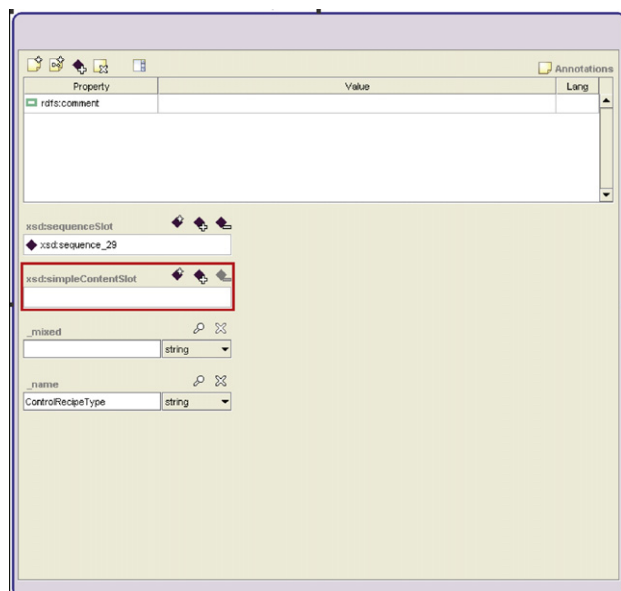
These improvements were brought about by making this information visible and readily available to diverse entities (human and computers) at different control levels. In addition, as a result of testing the ontological system with the real database structure, we could apply a strengths, weaknesses, opportunities and threats (SWOT) analysis. This enabled us to identify potential modifications in the database structure, to improve it and make it more efficient.

Information within the recipe steps and process information (real values stored and referenced to the batch in a database), raw materials, associated characteristics, and quantities consumed are automatically logged to the batch journal. When this is integrated with batch historian tools, process data can be assigned to batch context and used to generate reports, comparisons and trends for detailed batch analysis.

While the instantiation of the case study is made, the process becomes standardized automatically, as established by ANSI/ISA-88. This provides several automation benefits, including a reduction in the implementation time and cycle times. It also allows for improved (batch-to-batch) product consistency, improved product and process quality management, and better cost accounting capability.

The monitoring information is inside the ontological structure. This means that we have the guidelines for better control, as the monitoring information shows which elements are the sources or are affected in a fault detection process.

Finally, we made a contribution to the separation of recipe procedure and equipment control logic, and provided a clearer definition of manufacturing processes and production requirements. The use of common terminology and manufacturing models simplifies communication between customers and suppliers in the system.

**Fig. 15.** Protégé required parts in order to instance the process.

5.2. Ontology performance

The use of BaPrOn enabled the integration of the scheduling task and the fault diagnosis system (FDS) and supplied the process state information to different levels in the decision-making hierarchical structure, which led to the following improvements:

Behavior: The efficiency of the reactive scheduling was improved by updating the schedules in the most effective way, which leads to an increase in process yield.

Language: The information presented proper, common and standardized language within the hierarchical levels, regardless of its origins.

Documentation: Ontology provides a certain and easy description for formal (read by a computer) and informal (presented in a human-readable form) specifications of the system elements' content.

Storage: Efficient access is provided to process databases using a database management system and to establish the links and locations of these databases.

Navigation: The Web-Protégé portal provides the option of interaction among potential users of the ontology, by interchanging some notes about the different parts.

A comparison between the process of how different levels interact in a traditional way and the proposed one, are shown in Table 9. We can see an important improvement of 20% decrease in the number of steps for carrying out each process. The major improvement appears when a comparison of implementation time required by the two models is made showing a 78% of time saved.

6. Conclusions and future work

The fields of ontology, knowledge management and decision-support systems have matured significantly in recent years. Currently, the integration of various fields and technological solutions facilitates knowledge creation, storage and sharing in a specific domain, and improves the effectiveness of decision-support systems.

The creation of a bridge between the batch process and related information enabled formal and informal models, knowledge and experience to exploit different optimization tools through the support of common methods, languages and tools that simplify the generation and processing of conceptual models.

The BaPrOn ontology, which was created using the ANSI/ISA-88 standards as a guideline, performed well at capturing common understanding in conceptual design and helping to utilize the relations that are mined from the databases. This helps users to obtain data and information in a proper, fast and standard way through an internal web page portal.

The creation of consensual methodology aided the development of the ontology and made it easy and friendly to use. We considered On-To-Knowledge and Methontology, which are two of the most commonly used existing methodologies.

With the application of the ontology, in a short part of batch control process, the follow advantages can be mentioned:

- The systematic integration of different actors within the control process. In particular planning (scheduling) and plant floor control (unit control).
- This integration in a major application could allow that access that the information be not a constrain that limits the plant management capacity.
- It provides a common language for better communication about automation opportunities and manufacturing requirements.

- It creates a modular and internally consistent standard that can help reduce engineering cost, and it can finally serve as a management tool.
- It constitutes an appropriate tool to help filling the existing gap of the missing implementation technology of the ANSI/ISA-88 standard, that is to say, the utilization of an ontology as a form of knowledge representation.

The Methodology proposed versus the actual state of the art: nowadays many improvements concerned to the ontologies study area and its application is in process. However, actually there is not a certain methodology for their development. This work tries to ensure an ordered quality manner to develop an ontology. This is made through the inclusion of the two most cited methodologies within a well known continuous improvement tool PDCA cycle. From this point of view this work is an ongoing successful effort to improve ontology process development in a friendly way.

In addition, this ontology opens the way for achieving successful flexible control in adapting and recognizing different elements found through the hierarchy models that are associated with manufacturing multilevel control systems.

Finally, this work represents a step forward to support the integration (not just “communication”) of different software tools applicable to the management and exploitation of plant database information, resulting into an enhancement of the entire process management structure.

The ontological structure can be extended in the future to incorporate other hierarchical levels and their respective modeling knowledge. Concepts of functions (order processing, detailed production scheduling, production control, quality assurance, etc.) can thus be developed according to the ANSI/ISA-95 standards, and the whole supply chain (demand, capacity expansion, inventory control, etc.) can be improved.

Acknowledgements

The financial support received from the Dirección General de Educación Superior Tecnológica (DGEST), Academy Excellence Program, reference 072007004—E.A. from México and the Spanish Ministry of Science and Innovation (MICINN) through the project “Advanced Support System for Flexible Chemical and Petrochemical Production Processes” (DPI2006-05673) are fully appreciated.

References

- Alavi, M., & Keen, P. G. W. (1989). Business teams in an information age. *Information Society*, 6(4), 179–195.
- Bayer, B., & Marquardt, W. (2004). Towards integrated information models for data and documents. *Computers and Chemical Engineering*, 28, 1249–1266.
- Bechhofer, S., van Harmelen, F., Hendler, J., Horrocks, I., McGuinness, D. L., Patel-Schneider, P. F., et al. (2004). *OWL web ontology language reference website*. Available at <http://www.w3.org/TR/2004/REC-owl-ref-20040210/>
- Board, N. S. A., et al. (2009). *Biportal reference website*. Available at <http://biportal.bioontology.org/>
- Brandl, D., & Emerson, D. (2003). Batch markup language batchml. *World Batch Forum*.
- Brickley, D., & Guha, R. (2002). *Rdf vocabulary description language 1.0: RDF schema*. Available at <http://www.w3.org/TR/2002/WD-rdf-schema-20021112/>
- Eppler, M. J. (2006). *Introducing the notion of information quality* (2nd Ed.). Berlin/Heidelberg: Springer., pp. 1–16.
- Fensel, D. (2003). *Ontologies: A silver bullet for knowledge management and electronic commerce*. Springer.
- Gebus, S., & Leiviskä, K. (2009). Knowledge acquisition for decision support systems on an electronic assembly line. *Expert Systems Applications*, 36(1), 93–101.
- Gruber, T. R. (1993). A translation approach to portable ontology specifications. *Knowledge Acquisition*, 5(2), 199–220.
- Horridge, M., Jupp, S., Moulton, G., Rector, A., Stevens, R., & Wroe, C., et al. (2007). *A practical guide to building owl ontologies using protege 4 and co-ODE tools*. Technical report, The University Of Manchester.
- Horrocks, F. D., Broekstra, J., Decker, S., Erdmann, M., Goble, C., van Harmelen, F., et al. (2000). *Ontology inference layer oil*. Available at <http://www.ontoknowledge.org/oil/>

- International Society for Measurement and Control. (1995). *Batch control. Part 1. Models and terminology*. International Society for Measurement and Control.
- International Society for Measurement and Control. (2001). *Data structures and guidelines for languages*. International Society for Measurement and Control.
- International Society for Measurement and Control. (2003). *Batch control. Part 3. General and site recipe models and representation*. International Society for Measurement and Control.
- International Society for Measurement and Control. (2006). *Control batch parte 4 registros de producción de lote*. International Society for Measurement and Control.
- International Society for Measurement and Control. (2007). *Batch control part 5 automated equipment control models & terminology*. International Society for Measurement and Control.
- Klein, M., Fensel, D., Kiryakov, A., & Ognyanov, D. (2002). Ontology versioning and change detection on the web. In *Proceedings of the 13th international conference on knowledge engineering and knowledge management EKAW02* (pp. 197–212).
- Klyne, G., & Carroll, J. J. (2002). *Resource description framework (RDF): Concepts and abstract syntax reference website*. Available at <http://www.w3.org/TR/2002/WD-rdf-concepts-20021108/>
- López, M. F., Gómez-Pérez, A., Sierra, J. P., & Sierra, A. P. (1999). Building a chemical ontology using methontology and the ontology design environment. *IEEE Intelligent Systems*, 14(1), 37–46.
- McQueen, S., & Thompson, H. (2000). *Xml schema reference-website*. Available at <http://www.w3.org/XML/Schema>
- Mendez, C. A., Cerda, J., Grossmann, I. E., Harjunkski, I., & Fahl, M. (2006). State-of-the-art review of optimization methods for short-term scheduling of batch processes. *Computers and Chemical Engineering*, 30, 913–946.
- Missikoff, M., & Taglino, F. (2002). Business and enterprise ontology management with symontox. In S. B. Heidelberg (Ed.), *The semantic web—ISWC 2002*, 2342/2002 (pp. 442–447).
- Morbach, J., Yang, A., & Marquardt, W. (2007). Ontocape: A large-scale ontology for chemical process engineering. *Engineering Applications of Artificial Intelligence*, 20, 147–161.
- Morbach, J., Wiesner, A., & Marquardt, W. (2009). Ontocape 2.0: A (re)usable ontology for computer-aided process engineering. *Computers and Chemical Engineering*, 33, 1546–1556.
- Neches, R., Fikes, R., Finin, T., Gruber, T., Patil, R., Senator, T., et al. (1991). Enabling technology for knowledge sharing. *AI Magazine*, 12(3), 36–56.
- Obrst, L. (2003). Ontologies for semantically interoperable systems. In *CIKM'03: Proceedings of the twelfth international conference on information and knowledge management*, ACM (pp. 366–369).
- Pagels, M. (2006 January). *The darpa agent markup language (DAML) reference website*. Available at <http://www.daml.org/>
- Rippin, D. (1983). Batch process systems—Engineering: A retrospective and prospective review. *Computers and Chemical Engineering*, 17, S1–S13.
- Rodríguez-Martínez, A., López, I., Arévalo, Bañares, R., Alcántara, Aldea, A., et al. (2004). Multi-model knowledge representation in the retrofit of processes. *Computers and Chemical Engineering*, 28, 781–788.
- Shah, N. (1998). Single- and multisite planning and scheduling: Current status and future challenges. In J. F. Pekny, & G. E. Blau (Eds.), *Proceedings of the FOCAPO 1998, CACHE* (pp. 75–90).
- Shim, J. P., Warkentin, M., Courtney, J. F., Power, D. J., Sharda, R., & Carlsson, C. (2002). Past, present, and future of decision support technology. *Decision Support Systems*, 33(2), 111–126.
- Shirasuna, M. (2007). Optimización de la producción en una empresa de cervezas usando el estándar ISA s88 (caso polar). *Rockwell Automatization*.
- Simon, F., & Murray, T. (2007). Decision support systems. *Communications of the ACM*, 50(3), 39–40.
- Subrahmanyam, S., Bassett, M., Pekny, J., & Reklaitis, G. (1995). Issues in solving large scale planning, design and scheduling problems in batch chemical plants. *Computers and Chemical Engineering*, 19, 577–582.
- Sure, Y. & Studer, R., (2002). *On-to-knowledge methodology*. Technical report, University of Karlsruhe.
- Swartout, W., Neches, R., & Patil, R. (1993). Knowledge sharing: Prospects and challenges. In K. Fuchi, & T. Yokoi (Eds.), *Proceedings of the international conference on building and sharing of very large-scale knowledge bases*.
- Venkatasubramanian, V., Zhao, C., Joglekar, G., Jain, A., Hailemariam, L., Suresh, P., et al. (2006). Ontological informatics infrastructure for pharmaceutical product development and manufacturing. *Computers and Chemical Engineering*, 30, 1482–1496.
- Venkatasubramanian, V., Reklaitis, G., Hsu, S. -H., Jain, A., Hailemariam, L., Suresh, P., et al. (2008). *Overview of the ontological informatics infrastructure for pharmaceutical product development*. Technical report, School of Chemical Engineering Purdue University.
- Williams, T. J. (1989). *A reference model for computer integrated manufacturing (CIM)*. Research Triangle Park: ISA.
- XML-Core-Working-Group. (2009 April). *Extensible markup language (xml) reference-website*. Available at <http://www.w3.org/XML/>
- Zhao, C., Jain, A., Hailemariam, L., Suresh, P., Akkisetty, P., Joglekar, G., et al. (2006). Toward intelligent decision support for pharmaceutical product development. *Journal of Pharmaceutical Innovation*, 1(1), 23–35.