Ontologies for Analysis and Improvement of Business Process Quality in a Virtual Enterprise

Alexandra Galatescu, Taisia Greceanu

National Institute for R&D in Informatics,
8-10 Averescu Avenue, 011455 Bucharest, ROMANIA
E-mail:{agal, gresta}@.ici.ro

Abstract. The paper describes the representation and use of three ontologies in a software aiming at the assistance of a virtual team in business process analysis and improvement (BPI), using total quality management (TQM) technology. In comparison with the existing tools for BPI by TQM, this software has two specific features: (1) the ontology-based integration of the TQM tools (verbal diagrams, statistical charts, data collection sheets, ideas organization tools) and (2) the adaptation of the improvement process to virtual enterprises, where the decisions result from the comparison and integration of ideas issued during the brainstorming in a virtual team. The paper motivates and exemplifies an upper-level ontology with linguistic features for the representation of objects and processes in the BPI, domain and communication ontologies and of the ideas upon them and, also, for the integration of the TQM conceptual tools.

1. Introduction and Motivation

The automation of business process improvement (BPI) should comply nowadays with the requirements of the virtual enterprises regarding the team-based work and decisions. BPI (as a particular case of business process re-engineering) means the analysis and redesign of the team-based workflows and processes within and between organizations [1, 2]. From historical, organizational and technological perspectives, BPI is considered a precursor of the knowledge management in enterprises [3] and the ontologies contribute to this kind of management.

An ontology is a 'specification of a conceptualization' [4] and, practically, 'a vocabulary used to describe a certain reality, plus a set of explicit assumptions regarding the intended meaning of the vocabulary words' [5]. The relations between the concepts in an ontology allow inferences for the information interpretation and for the derivation of new information/ knowledge. The explicit axioms allow the approximation of the term meaning and the validation of ontology specification [16].

Previous applications of the ontologies to business process management already exist. For example, in [1], an IDEF5 ontology is used to describe 'the ontological enterprise'. In [6], an enterprise ontology relies on generalized terms referring to roles, artifacts, interactions between people, norms to create teams, collaborative services etc. In [7], the proposed enterprise ontology is a collection of terms and definitions relevant to business processes. In [8], a PSL (Process Specification Language) twotiered ontology is proposed for the manufacturing integration. In [9], an ontology for business processes is given, together with its formal representation in Loom. In [10], the ontologies are associated with mathematical models for the design of the processes and communication structures in e-commerce. And so on.

Total quality management (TQM) [11, 12] helps for the process (continuous and incremental) improvement. Computer-aided assistance for TQM can be retrieved in existing tools like: Pathmaker [13], Memory Jogger [14], Solutions-PROSPER and PRO-QMS [15], Microsoft Visio, etc. Their main *drawbacks* are: (1) the TQM tools they implement are insufficiently integrated, mainly because of the informal semantics of the symbols, that cannot be transferred between the TQM tools; (2) they miss capabilities specific to virtual enterprises, (3) the members of the team do not share a common vocabulary, (4) the flowcharts and verbal diagrams are exclusively graphical and cannot be compared, (5) the ideas are expressed in natural language (NL) and cannot be automatically compared and integrated.

Brainstorming implementation in these tools does not encourage the definition and use of a common vocabulary between the members (usually with different specializations), because it consists of ideas collection and storing in NL. The ideas mediation and the inference upon ideas are devolved to human members. The ideas are subjective and require many virtual discussions to reach a final decision.

For BPI automation, the ontologies are motivated by several *requirements*: (1) the organization, integration and formalization of the BPI specific knowledge, (2) the representation of the diagrams and charts and the expression of ideas, relying on a common vocabulary and understanding of the concepts exchanged by the members of the team, (3) the inference on BPI and domain specific knowledge, simultaneously; (4) the inference on both objects and processes, simultaneously.

Using ontologies, the BPI assistant referred to in this paper differs from the existing software for BPI by TQM by several features. It provides an *ontology-based integration of the TQM conceptual tools* (see Sect. 5), using the predefined BPI and communication ontologies and the team-defined domain ontology. It is mainly *devoted to the virtual teams*, where the decisions result from the comparison of the members' ideas (see Sect. 4) and where the virtual brainstorming has an important part in almost all steps of the BPI process. It integrates an *ontology agent* (see also [16]) that facilitates: (1) the dynamic creation of the user's interface, based on the BPI ontology, (2) the definition, navigation, extension of the domain ontology, (3) the automatic classification of the domain concepts, according to the working context, (4) the communication between the members of the team, relying on the BPI and domain ontologies, (5) the comparison and inference on the members' ideas expressed using concepts in the domain ontology and using ontological sentences.

Section 2 gives the components of the upper-level ontology with linguistic features, used for BPI automation, along with its advantages in comparison with other upper-level ontologies. Section 3 exemplifies the representation and integration of BPI, domain and communication ontologies by ontological sentences. Section 4 exemplifies the representation, comparison and grouping of the ideas expressed by ontological sentences. Section 5 enumerates the main results on the ontology-based integration of the TQM tools implemented in the new software for BPI by TQM.

2. Ontological Sentences with Linguistic Features

The ontological requirements for BPI, basically the ontology integration, the reasoning with both objects and processes and the need for the representation and integration of ideas motivate the use of an upper-level ontology and, also, of natural language (NL) as an inspiration source for this ontology. NL helps with its universality, its syntactic stability and, implicitly, its integration ability [19, 20]. The linguistic ontologies are preceded by the lexical ontologies, e.g. WordNet [17] and FrameNet [18], that emphasize the relations inside the lexical categories. They are not intended for the composition of sentences in ideas. For the representation of an upper-level linguistic ontology, two research directions and results are important:

- abstraction of *NL semantics*, using a *taxonomy of universal types* of objects, activities, processes (as well as relationships among them), supposed to allow the subsumption of the words belonging to any syntactic category (noun, verb, etc).
- abstraction of NL *syntax*, using rules for building *sentence-like structures*, that stylize the NL sentences and comply with NL syntax. They are supposed to help for the unambiguous description of any type of object, process, activity, as well as for the representation of unambiguous ideas about them.

These two directions are complementary and should be both considered in the definition of a linguistic ontology. From the semantic viewpoint, there are several proposals for taxonomies, compared in [23]. Two linguistically oriented taxonomies are proposed in [24, 25]. From the syntactic viewpoint, the limits of the functional grammar, conceptual dependencies and conceptual graphs (emphasized in [19, 20]), used today for NL translation to object and process models, impose an improvement with respect to the model integration.

The representation proposed in this section and in [21, 22], as an upper-level linguistic ontology, mainly deals with the syntactic aspects of NL translation. Its logical consistencey and its linguistic completeness is proved in [19, 20].

The *basic vocabulary* of this ontology contains the following types of concepts: *active objects* (direct participants in activities), standing for nouns in NL; *object attributes*, standing for adjectives in NL; *activities/ operations*, standing for verbs; *activity attributes*, standing for adverbs; *object and activity determiners/ modifiers/ substitutes*, standing for the noun and verb determiners/ modifiers/ substitutes in NL. The axioms in this ontology are ontological *simple*, *compound and complex sentences* which stylize the corresponding sentences in NL.

Ontological simple sentence unifies the objects with different syntactic roles involved in the description and execution of the main operation (verb) in the sentence. It is a *star graph* [25], where the *nodes* are objects or operations and the *links* are roles, standing for 'object-operation' links or for links between active objects and the attributive objects that describe them. In its linear form, this graph is:

```
(OPERATION)
   AGNT  ∀[AGENT]
   PTNT  ∃/∃?[Object_Type1:C/D{}]
   RCPT  ∃/∃?[Object_Type2:C/D{}]
   <preposition role> ∃/∃?[Object_Type3]
   <adverb role>  ∃/∃?[Object_Type4]
```

where: OPERATION is an atomic operation, standing for the predicate in NL sentence; AGNT stands for the role of the subject(s) in the active voice; PTNT is the role of the direct object(s), i.e. the object(s) upon which OPERATION acts; RCPT is the role of the indirect object(s), i.e. the recipients of the results of OPERATION; 'preposition role' is the role of the prepositional object(s); 'adverb role' is the role of the adverbial modifier(s) (i.e. operation modifier); universal quantifier ∀ replaces the indefinite pronouns 'any', 'all', 'every', 'each' in NL; the two existential quantifiers: ∃, meaning compulsory existence ('must exist') and ∃?, meaning optional existence ('may exist'), replace the definite or indefinite articles in NL; C/ D{} suggest the collective/ distributive plural.

Preposition and adverb roles are abstracted by acronyms like: RSLT (result of activity), INST (instrument to achieve the activity), LOC (location of activity), SRC (source of activity), DEST (destination of activity), and so on (a detailed list is in [20]). Each acronym has a preposition, conjunction or adverb as linguistic synonym, e.g. 'by' for AGNT, 'upon' for PTNT, 'for/ to' for RCPT, 'into' for RSLT, 'with' for INST, etc. These roles allow the domain independent processing of the operations (the code uses only these roles instead of domain specific types of objects/ attributes). Also, with their disjunctive semantics, they eliminate the ambiguities in NL.

Object determiners are object type, quantifier, plural, cardinality. *Operation* is described by *determiners* that can be direct, indirect or prepositional objects.

Special types of simple sentences represent *generic operators* for (1) semantic relations between objects or operations (e.g. holonymy, hypernymy, synonymy, antonymy etc) and (2) the dynamic qualification of objects or operations (see Sect.3).

Ontological compound and complex sentence uses inter-operation connectors (as intersentential relations) for the correlation of the operations (verbs in NL) and, implicitly, of the ontological simple sentences that describe them. These relations correlate the ontological simple sentences into compound or complex sentences. As in NL, the *compound sentence* joins independent simple sentences and the *complex sentence* is composed of dependent (subordinated) sentences correlated to a main sentence. Examples of intersentential relations for ontological compound sentences are MUST, MAY, AND, OR, NOT, GROUP, REPEAT, etc. In a complex sentence, the activities are correlated by subordinating relations abstracted by: IF-THEN-ELSE, DSCR (description), GOAL, EVENT, DO, WHILE, subordinating CAUSE or RESULT, THEN, CASE, SPEC (specialization), BEFORE, AFTER, BUT etc.

Brief comparison with other ontologies. In comparison with the taxonomies proposed for other upper-level ontologies [23], the *primary semantics* of the first level in this ontology borrows from the semantics of the basic syntactic categories (nouns become objects, verbs become activities, adjectives or adverbs become object and operation's attributes). Any type of concept could be further subsumed to the concepts on this level. The *contextual semantics* of the concepts and the relationships between them, in domain-specific ontologies, come from the syntactic roles of the objects/ attributes in object/ operation description and from the inter-operation connectors.

Both the primary and the contextual semantics in this ontology are outside the code. Consequently, the main benefit from this representation is the conceptual *integration of object and process models, outside the code and in the early phases of* *the enterprise system life cycle.* This advantage for model integration has been detailed in [19, 20].

The existing ontology editors (e.g. Protege, OilEd) do not separate the object and activity-like concepts. In most enterprise ontologies today, the processes are represented by object-oriented representations and the object and process integration is mostly encoded, using object-oriented programing languages. This limit makes difficult the ontology use in process-centric applications and in the ideas (or queries) expression, comparison and interpretation.

Instead, the proposed representation can be implemented in any language, including in relational databases (as in the implementation of the new software for BPI by TQM referred to in this paper).

3. Representation of BPI, Domain and Communication Ontologies

BPI, domain and communication ontologies used for BPI assistance have different vocabularies and different axiomatizations that must be correlated in the automatic reasoning. They need the same conceptual representation means. Two alternative solutions could be used for the integration of the three ontologies: (1) by an *upper-level ontology*, able to represent both objects and processes in the domain and BPI ontologies, the communicative acts in the communication ontology, as well as the correlations between them; (2) by a *translation and correlation algorithm* between the concepts and rules in the three ontologies. This algorithm has the disadvantage that is mostly encoded. Consequently, the first alternative is a better solution and was implemented in the new software.

The basic concepts in BPI ontology are organized according to following aggregation hierarchy for the description of the improvement process:

```
Improvement Process
   General Scenario
      Improvement Step
          Complex/ Compound Operation
             Atomic operation
             Connector between atomic operations
          Atomic Operation
             Object
                Characteristic/ attribute of the object
          Connector between atomic or complex/ compound operations
          Pre-condition for the execution of the operation
      Connectors between improvement steps
      Pre-condition for the execution of the Step
```

The process is described by a general scenario composed of steps. The steps are composed of complex or atomic operations. Each atomic operation is described by objects. The objects are described by attributes. The operations or steps are pre-conditioned and are correlated by connectors.

The concepts in the domain ontology are user-defined instances of the concepts in the following hierarchy for the description of the analysed process:

```
Process in domain
   Complex/ Compound Operation in Process
      Atomic operation in Process
```

```
   Connector between atomic operations
Atomic Operation in Process
   Object in Process
      Characteristic/ attribute of the object
Connector between atomic or complex/ compound operations
Pre-condition for the execution of the operation in Process
```

The objects and operations in either ontology are represented by sentences with linguistic features, as this section will exemplify (see details in [21, 22]) .

Ontological sentences in BPI ontology. *Ontological simple sentences* are mainly used for object and operation definition (that unifies objects that uniquely identify the object/ operation) and description (that dynamically unifies objects that qualify another object or determine the execution of an operation). For example, the object MEMBER is defined and qualified by the first two sentences below. The execution of BRAINSTORMING operation is described by the third sentence.

(Object IDENTIFICATION)	**(Object QUALIFICATION)**	**(BRAINSTORMING)**
RCPT ∀**[Member]** NAME ∃[Member_Name] LOC ∃[Department]….	RCPT ∀[Member] GOAL ∃[Responsibility]	AGNT ∃[Member:C{}] TIME ∃[DateTime] DUR ∃[Period] SUBJ ∃[Subject] RSLT ∃[IdeasList]..

The simple sentences are also used for the representation of generic operators that semantically correlate objects or operations, similarly to the relationships provided in WordNet (noun holonymy hyponymy, synonymy, antonymy, etc). The following examples are for object holonymy and operation entailment (implication):

(Object_HOLONYMY)	**(Operation_ENTAILMENT)**
DEST ∀[FlowChart] *- whole* PART1 ∃[StartPoint] *-component* PART2 ∃[Activity:C{}] PART3 ∃[DecisionPoint:C{}]..	RCPT ∀(Diagram_INTERPRETATION)*- entailed* PTNT ∃(Diagram_CREATION) *- entailing* *operation*

Ontological complex sentences describe the scenarios for BPI methodology, for its steps and for its complex/ compound operations. Figure 1 exemplifies few atomic operations from the scenario for the brainstorming session. Each atomic operation is further described by an ontological simple sentence.

MUST	**Prepare Brainstorming Meeting**
MAY	**Express ideas**
MAY	**Export / Import Structures**
MAY	**Prepare Send Ideas/ Diagram (create Excel file)**
MAY	**Send/ Receive structures with ideas (call OutLook Express)**
MAY	**Collect and view ideas (from Excel files)**
MAY	**Multivote for collected ideas**
CASE	**Multivote per idea**
	Multivote per sequence of idea
	View structure with multivote result
MAY	**Operations on Affinity Diagram**
CASE	**Create Affinity Diagram**
	Change Group Name in Affinity Diagram
	Order Ideas by Group in Affinity Diagram

Fig. 1 Part of the scenario (complex sentence in BPI ontology) for brainstorming session

Ontological sentences in the domain ontology. An example of process to improve in the healthcare domain is ‘Medication administration’ and an improvement objective (proposed in [22]) is ‘Reduction of medication errors'. Each object and operation in this process, as well as the inter-object and inter-operation semantic relationships, are represented by ontological simple sentences. These sentences force the members of the team to select the most relevant elements that describe the process and to analyse them. E.g., the sentences that identify and dynamically describe the object ‘Patient’ or describe the atomic operation Med Order:

(Object IDENTIFICATION)	**(Object_QUALIFICATION**	**(Med ORDER)**
RCPT ∀**[Patient]**	RCPT ∀[Patient]	RCPT ∀[Patient]
ID ∃[PersonSSN]	POSS ∃[Medicament]	AGNT ∃[Physician]
NAME ∃[PersonName]	STAT ∃[HealthState]	PTNT ∃[Med:C{}]
DATE ∃[BirthDate] …		QTY ∃[Med Dose]

In order to match different vocabularies (e.g. a scientific and a popular one), one may find necessary to explicitly represent synonymy relationships like:

(Object SYNONYMY)	**(Operation SYNONYMY)**
PTNT1 ∃[Medicament]	PTNT1 ∃[Med ORDER]
PTNT2 ∃[Drug]	PTNT2 ∃[Med PRESCRIBE]
PTNT3 ∃[Med]	

The process to improve is described in a flowchart as a complex sentence (Figure 2). The intersentential relations and all elements for the activity description can be seen only in the linear form of the flowchart.

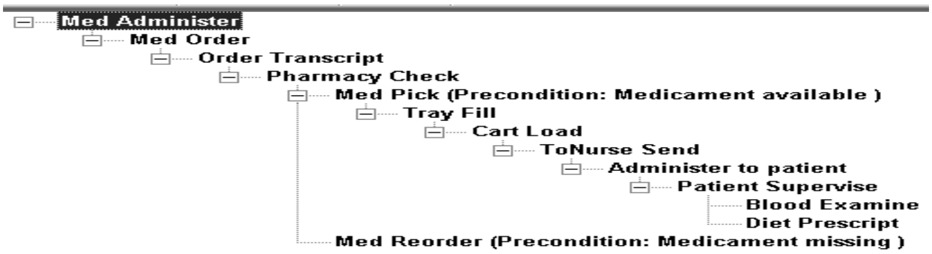

Fig. 2 Complex sentence in domain ontology representing the analysed process

Ontological sentences in the communication ontology describe the communication acts ‘query’ and ‘reply’ for structures (diagrams, data collection sheets, structures with ideas) by two basic operations: 'Collect' for the reception of structures and their import in the BPI database and 'Send' for the export and transmission of structures to other members. E.g., the mediator's query for data sheets from a collector and the collector's reply are instances of the following sentences:

(**Collect)**	(**Send)**
AGNT ∀[Mediator]	AGNT ∀[Collector]
RCPT ∃[Collector:D{}]	RCPT ∃[Mediator]
GOAL ∃[ObjectType]	GOAL ∃[ObjectType]
PTNT ∃[DataSheet:D{}]	PTNT ∃[Data_Sheet:D{}]
SRC ∃[Mediator_Email]	SRC ∃[Collector_Email]
DEST ∃[Collector_Email:D{}]	DEST ∃[Mediator_Email]

These operations don't exclude the communication by messages in NL, the only type of communication provided in the existing tools for BPI by TQM.

4. Ontological Sentences for Ideas Expression and Comparison

The user can express any idea using simple, compound or complex ontological sentences. He is guided to create and then to use the concepts in the domain ontology. The expression of ideas using concepts in this ontology seems restrictive. But, it forces the members to use the same vocabulary, so it saves many virtual discussions needed to reach a common understanding on the concepts they use. Also, it forces the members to focus on the most relevant concepts and problems, to understand and deeper analyse them. Another advantage is the automated comparison and grouping of ideas, that saves the mediator's time.

This section exemplifies the representation and grouping of the ideas expressed by ontological sentences, collected (in this example) in a cause-effect diagram for the identification of the causes of the process instability. This diagram can be represented either (1) with causes defined in natural language (NL) (as in the existing tools for TQM) or (2) with causes defined using the concepts in the domain ontology and ontological sentences, as in Figure 3. The second variant allows the automatic comparison of the causes according to syntactic criteria. In both variants, the causes and subcauses are correlated by logical operators (AND, OR, NOT). Figure 3 represents the causes for the problems (negative effects) 'High medication cost', 'Too many days in hospital', Too many complaints', relative to the quality characteristics 'Medicine_cost' and 'Number of complaints' for 'Patient'.

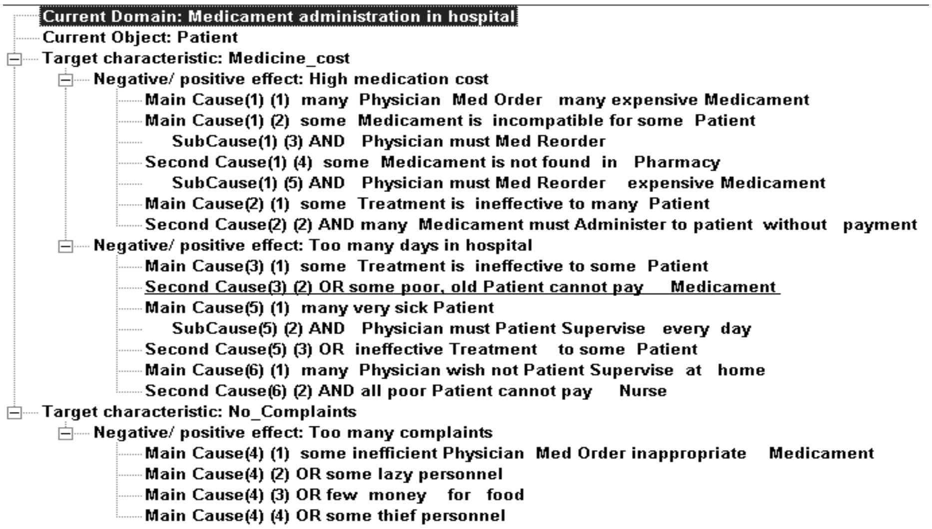

Fig. 3 Cause-effect diagram using ontological sentences

The ideas expressed by ontological sentences are automatically compared and grouped according to their syntactic components. The result is an *affinity diagram* (as in Figure 4). The members express their vote on the final list of ideas and the mediator calls the multivote function, that automatically calculates the vote per idea (usually, complex sentence) or sequence of idea (simple sentence).

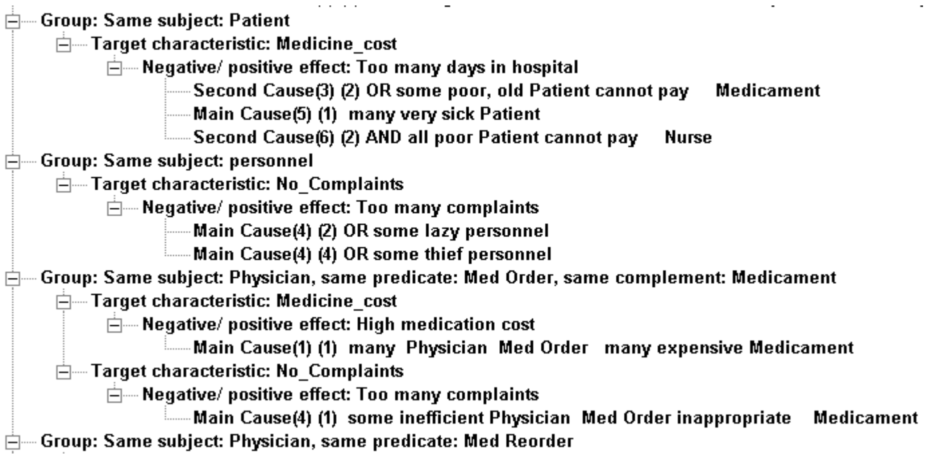

Fig. 4 Part of the affinity diagram that groups the ideas in the cause-effect diagram (Figure 3) according to a multi-level syntactic algorithm

5. Ontology-based Integration of the TQM Tools

In the existing products, the integration of the TQM tools basically consists in the integration of the data collection sheets with the graphical charts (run chart, control charts, Pareto charts, histograms etc) for data statistical analysis. Their integration with the so-called 'verbal diagrams' (flowcharts, cause-effect diagrams, affinity diagrams, etc), as well as with the members' ideas is manual and devolved to the users. In a virtual team for BPI, this integration facilitates and standardizes the communication between members, increasing the performance of the BPI process.

Integration of BPI steps, operations and objects. Dynamic creation of the interface. All BPI steps, operations and objects are uniformly represented by means of ontological sentences in BPI and communication ontologies. The interface of the software is dynamically created, at the user's request (only for the required steps, operations and objects), using the concepts in BPI and communication ontologies.

Integration of TQM tools. After the creation of the domain ontology (at the beginning of BPI process), the process flowcharts, data collection sheets, verbal diagrams, statistical charts, as well as the structures with ideas are all built using concepts in this ontology. Few integration examples are given below:

Integration of the domain ontology with the verbal TQM structures (process flowcharts, cause-effect diagrams, structures with ideas, affinity diagrams). These structures unify concepts that represent operations, objects, characteristics in the domain ontology, or synonyms of these concepts. The concepts in the domain ontology can be named in any language.

Comparison and integration of the flowcharts. The flowchart reflects the hierarchical sequence of operations in the process, the decision points (operation preconditions), the redundant operations, the cycles in the process, the type of operation (value or cost added), the operations where data must be collected. The team builds the flowchart for the existing (AS-IS) process. Each member can contribute with modifications on it, resulting in a new flowchart of the same process. The differences between two flowcharts (including, those for AS-IS and TO-BE process) are automatically identified and can be graphically visualized as in Figure 5:

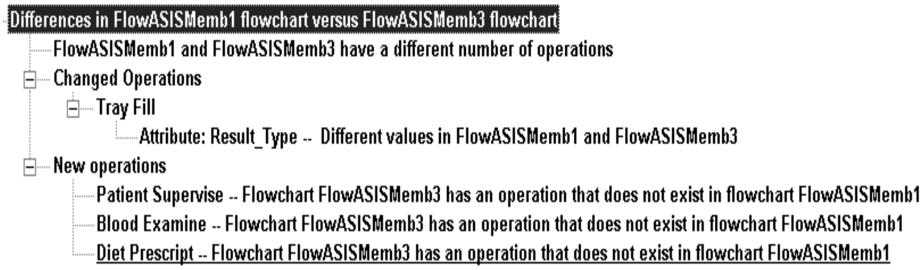

Fig. 5 Results from the comparison of two flowcarts for the same AS-IS process

The members' changes on an initial flowchart are automatically merged, resulting into the final flowchart of the process, subject to the vote of the members of the team.

Integration of the flowcharts with the data collection sheets results from the dynamic creation of the data collection sheets (on user's demand), relying on concepts in the domain ontology that define (in flowchart) the analysed process. According to the team's decision during the flowchart definition, for certain operations, data are collected on quality characteristics for certain objects involved in the operation execution.The schema (definition) of the data collection sheet is dynamically created. It is composed of previously selected quality characteristics for the analysed process.

Integration of the statistical charts with data collection sheets and flowcharts. Process stability and its improvement ability are checked by statistical charts (run charts, control charts, histograms), built using the data collected in previously created sheets. These sheets describe the evolution of the characteristics for objects associated to AS-IS or TO-BE processes, previously analysed in the corresponding flowcharts. Figure 6 is an example of (control) X-Bar and R charts, that analyse the quality characteristic 'Medicine_cost' for the object 'Patient'. LCL (lower control limit) and UCL (upper control limit) are located at three standard deviations from the centerline. Any stable characteristic must have values only between these limits.

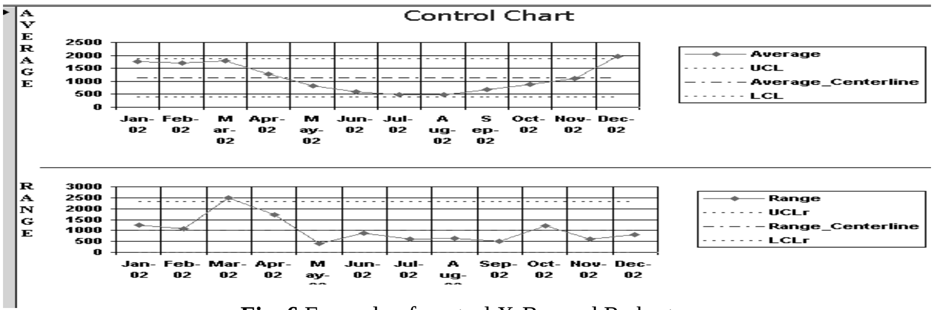

Fig. 6 Example of control X-Bar and R charts

Integration of the cause-effect diagrams and Pareto charts with the flowcharts and data collection sheets. The causes for the process instability are identified in the proposed software using two TQM tools: Pareto chart and cause-effect diagram. Both diagrams refer to quality characteristics that describe (in the domain ontology) an object in the process previously described in a flowchart.

Integration of the affinity diagram and multi-vote with the cause-effect diagram and other structures with ideas. For either operation (creation of affinity diagram or multi-vote), the user only specifies the structure with the ideas he wants to compare and group (e.g. the diagram in Figure 3). The groups automatically built in the affinity diagram (as in Figure 4) can be further grouped by the user, by filling the automatically created *super-affinity diagram*.

5. Conclusions

The paper motivates and describes the results from the automation of BPI by TQM, using an ontological infrastructure and providing two specific features of the new software: the ontology-based integration of the TQM tools and the adaptation of the improvement process to virtual enterprises. The main benefits from the use of ontologies for the team-based work in BPI are: a common vocabulary for the team; the automatic comparison and integration of verbal diagrams and structures with ideas; the communication (including import and export) with structures, not only with messages in NL; an extensible user interface, relying on the BPI ontology.

The ontologies and ideas are stored in a relational database and the users need only Windows 2000 and Microsoft Office.

The existing product is currently extended and integrated with functions for the control and optimization of the process quality using Taguchi method. With this method, the quality characteristics will be deeper analysed, depending on (controllable or uncontrollable) factors that impact on them.

References

1. Grover V., Kettinger W. (Ed.): Business Process Change – Re-engineering Concepts, Methods and Technologies, Idea Group Publishing (1995)
2. Malhotra Y.: Business Processs Redesign: An Overview. IEEE Engineering Management Review, 26(3) (1998)
3. Davenport T.: Business Process Reengineering: Where It's Been, Where It's Going. Business Process Change-Reengineering Concepts, Methods and Technologies, Idea Group Publishing (1995)
4. Gruber T.: Toward Principles for Design of Ontologies Used for Knowledge Sharing. In Formal Ontology in Conceptual Analysis and Knowledge Representation, Kluwer Academic Publishers (1993)
5. Guarino N. (Ed): Formal Ontology and Information Systems, IOS Press (1998)
6. Hawryszkiewycz I.: Supporting Teams in Virtual Organizations. Conf. on Database and Expert Systems Applications, Italy (1999)
7. Uschold M., King M., Moralee S., Zorgios Y.: The Enterprise Ontology, Knowledge Engineering Review, vol. 13 (1998)
8. Schlenoff. C, Ivester R., Knutilla A.: A robust process ontology for manufacturing systems integration, CommerceNet ECO Framework, http://www.ontology.org/main/papers/psl.html (1998)
9. Scacchi W., Valente A..: Developing a Knowledege Web for Business Process Redesign. http://www1.ics.uci.edu/~wscacchi/Papers/KnowledgeWeb (1999)
10. Kim H., Tham D.: Designing Business Processes and Communication Infrastructures for e-Business Using Ontology-based Enterprise Models with Mathematical Models. Enterprise Information Systems IV, Kluwer Academic Publishers (2002)
11. MIT: TQM Toolkit. http://web.mit.edu/tqm (1996)
12. DON (Department Of Navy): Handbook for Basic Process Improvement, USA (1996)
13. SKYMARK: PathMaker 3.0. Software, http://www.skymark.com/pathmaker/ (2002)
14. GOALQPC: Memory Jogger Software. http://www.goalqpc.com (1999)
15. DSS Infotech: Solutions Prosper (Software for Implementation of Continual & Break-through Improvement) and PROQMS (Process Quality Management System), DSS Infotech, India (2000)
16. FIPA.: Ontology Service Specification. Geneva, http://www.fipa.org (2000)
17. Miller G.: WordNet: A Lexical Database for English. Comm. of ACM 38(11) (1995)
18. Fillmore, C., Hiroaki, S. Transparency and Building Lexical Dependency Graphs. 28th Annual Meeting of the Berkeley Linguistics Society (2002)
19. Galatescu A: A Unifying Translation of Natural Language Patterns To Object and Process Modeling. Information Modeling in the New Millenium, Idea Group Publ. (2001)
20. Galatescu A.: Reifying Model Integration Abilities from Natural Language. Journal of Conceptual Modeling, InConcept, http://www.inconcept.com, March (2002)
21. Galatescu A., Greceanu T.: Using Ontologies in Virtual Branstorming for Business Process Reengineering. Collaborative Business Ecosystems and Virtual Enterprises, Kluwer Academic Publishers (2002)
22. Galatescu A., Greceanu T.: Ontologies Supporting Business Process Reengineering. Enterprise Information Systems IV, Kluwer Academic Publishers (2002)
23. Bateman, J.: Linguistic and non-linguistic ontologies: can we find the line between them?. Computational Linguistics Colloquium, Univ. of Saarland (2003)
24. Oltramari, A., Gangemi, A., Guarino, N., Masolo, C. DOLCE: a Descriptive Ontology for Linguistic and Cognitive Engineering. Proc. 13th Intl. Conf. on Knowledge Engineering and Knowledge Management, Spain.(2002).
25. Sowa J.F.: Knowledge Representation – Logical, Philosophical and Computational Foundation. Brooks Cole Publishing Co., Pacific Grove, CA (2000)