A CASE-BASED REASONING AGENTS APPROACH TO SUPPORT COLLABORATIVE NONCONFORMITY PROBLEM SOLVING IN THE THERMOPLASTIC INJECTION MOLDING PROCESS DOMAIN

Walter Luís Mikos
Universidade Tecnológica Federal do Paraná, Departamento Acadêmico de Mecânica, Campus Curitiba.
CEP 80230-901, Curitiba, PR – Brazil
mikos@utfpr.edu.br

João Carlos Espíndola Ferreira
Universidade Federal de Santa Catarina, Departamento de Engenharia Mecânica
GRIMA/GRUCON, Caixa Postal 476, CEP 88040-900, Florianópolis, SC, Brazil
jcarlos@emc.ufsc.br

Fabbio Gonzalez Correa Gomes
Universidade Federal de Santa Catarina, Departamento de Engenharia Mecânica
GRIMA/GRUCON, Caixa Postal 476, CEP 88040-900, Florianópolis, SC, Brazil
fabbiogomes@gmail.com

Roman Moura Lorenzo
Universidade Federal de Santa Catarina, Departamento de Engenharia Mecânica
GRIMA/GRUCON, Caixa Postal 476, CEP 88040-900, Florianópolis, SC, Brazil
roman_lorenzo@yahoo.com.br

Abstract. *The injection molding process is one of the most versatile technologies for processing thermoplastic materials used nowadays in industry. In spite of all the technological resources used, the probability of occurring nonconformities in the production stage cannot be completely eliminated, and in general these nonconformities take place due to the complex interrelation among the constructive elements of the molded part, the mold, the rheological characteristics of the resin, and the processing conditions, which present difficult analytical modelling, and frequently cause the task of discovering the source of the nonconformity and its respective solution to become slow and costly. In this context, this paper presents an agent-based architecture that incorporates Case-Based Reasoning Agents to support nonconformity problem solving in manufacturing environments with distributed resources. In this perspective, it is important to emphasize that the underlying premise to Case-Based Reasoning is that, for certain specific domains, the problems to be solved tend to be recurrent and to repeat with small variations in relation to its original version, and in this way previous solutions can also be reapplied with small modifications. This paper also presents the group of descriptors that represent the nonconformity features, as well as the implementation of a prototype based on the Java Agent DEvelopment Framework (JADE Plataform) and the jCOLIBRI CBR Framework as core engine.*

Keywords*: Multi-Agent System, Case Based Reasoning Agents, Nonconformity problem solving, Thermoplastic Injection Molding.*

1. INTRODUCTION

Nowadays, the social and economical environment is characterized by the appearance of new forms of industrial organizations, caused by complex factors such as market globalization, product life cycle reduction, high demand variability, need of high flexibility and reactivity, and fast development of the technologies of information and communication (Zaidat *et al*., 2005).

In this perspective, new forms of organizational structures have been recognized by the scientific community and other professionals of this area, which include: Networked Enterprise, Extended Enterprise, Virtual Enterprise, Virtual Supply Network, Virtual Organization, and Enterprise Clusters (Binder and Clegg, 2006).

The key problem in these new environments consists of integrating the distributed resources, a highly specific core competence asset (*e.g.* new proprietary technology or knowledge) and other less specific assets (*e.g.* shared information or process technology) that contribute to manufacturing, considering that these new organizational structures are geographically distributed, composed by different commercial partners, each of them endowed with specialization and resources for a specific function in the product life cycle (Cecil *et al*., 2006; Binder and Clegg, 2006).

In this scenario, in particular, the plastics converting industry that is the heart of the plastic industry stands out, which nowadays is one the largest manufacturing industries in the United States and in Europe (Society of the Plastic Industry, 2007; Association of Plastics Manufacturers, 2007). In this sense, the plastic converters (also called Processors) are part of a wide range of supply networks (production chains), because they produce semi-finished and finished plastic products, and also execute additional finishing operations such as printing and assembly work directly on several industrial sectors, such as: packing, building and construction, automotive, electrical and electronic, and healthcare.

In this new context, quality continues to be a fundamental requirement for competitiveness, and therefore traditional Quality Management Systems and Quality Improvement Systems developed in the field of operations management, and targeted to traditional companies should be reviewed and expanded for these new forms of organization, in order to consider not only internal processes of a single integrated company, but also to extend outside the boundaries of the traditional company, involving the interconnected companies and the customers (Chin *et al*., 2006; Binder and Clegg, 2006).

From this perspective, the nonconformity problem solving process still remains a great challenge, both from the academic and the industrial points of view, because this process involves knowledge intensive and experience-based activities, which can in many cases extrapolate the knowledge and the experiences of the workers and engineers of a single company. In this sense, in order to solve this problem it is important to share the knowledge and lessons learned, starting with a nonconformity solution, thus proactively improving quality.

On the basis of these issues, this paper presents the development of a Case-Based Reasoning Agents approach to support the collaborative nonconformity problem solving in the thermoplastic injection molding process domain, which aims at sharing and reutilizing current knowledge about the process of investigating the causes of nonconformity about the adopted corrective and preventive actions, as well as the results of its efficacy in support to the activities of management of the organizational knowledge in manufacturing environments with distributed resources.

This paper also presents the implementation issues of a prototype system based on the Java Agent DEvelopment Framework (JADE, 2006) and the jCOLIBRI CBR Framework as core engine (GAIA, 2006), and it discusses the obtained results obtained from the instantiation of the prototype with real cases of nonconformity problem solving in thermoplastic injection molding process related to technical molded parts designed for the electronic industry.

2. NONCONFORMITY AND PROBLEM SOLVING PROCESS

Nonconformity is defined by International Standard ISO 9000:2005 Quality management systems - Fundamentals and vocabulary as "non-fulfillment of a requirement", which in turn is defined as "need or expectation that is stated, generally implied or obligatory", and a qualifier can be used to denote a specific type of requirement "*e.g.* product requirement, quality management requirement, customer requirement" (ISO, 2005). Additionally, the standard defines "defect" as "non-fulfillment of a requirement related to an intended or specified use" of a product. In this sense, it is important to observe the difference between defect and nonconformity in the above definitions, where defect has legal connotations, particularly those associated with product liability issues.

Nonconformity problem solving process is recognized nowadays as an essential condition for the improvement of quality and for the success of any manufacturing organization, and also represents a great challenge to research, both to the academic and industrial communities.

Studies realized in German companies with traditional manufacturing systems in the metal-mechanical and chemical sectors demonstrate that during manufacturing, in average, 60% of the nonconformities are recurrent, *i.e.* they had already occurred the same way or similarly in the past, and consumed an average of 10% of the personnel and machine resources (Klamma, 2000).

And, especially, it can also be noticed the importance of this subject from the efforts of the international community towards including it as a obligatory theme to be considered in the international standards and guidelines, such as: ISO 9001:2000 Quality management systems - Requirements, ISO 9004:2000 Quality management systems - Guidelines for performance improvements (ISO, 2000) and The Effective problem solving Guideline published recently by the Automotive Industry Action Group (AIAG, 2006) as a result of a combined effort of specialists of the AIAG member companies and their supplier community, whose aim was to establish a consensus of the problem solving methodologies and concepts currently used in the automotive industry.

However, the identification of the key elements that contribute to the effectiveness of the nonconformity problem solving process is not a trivial task, and the literature points out the following aspects about the nonconformity problem: reviewing nonconformities (including customer complaints); determining the causes of nonconformities or potential nonconformities; evaluating the need: for actions to ensure that nonconformities do not recur, or for actions to prevent the occurrence of nonconformities; determining and implementing the action needed; recording results of an action taken and review of an action taken.

2.1. Nonconformity problem solving in thermoplastic injection moulding domain

The injection molding process is one of the most versatile technologies for processing thermoplastic materials used in industry, and is responsible nowadays for more than 33% of all polymeric material processed, and in particular it is largely used for the manufacture of technical parts with complex geometry and tight tolerances with high productivity and low costs (Chen and Turng, 2005).

During the injection molding process, the polymeric material is submitted to a complex thermomechanical process that produces significant transformations in its rheological and mechanical properties, which result from the large pressure variations, as well as the temperature variations caused by two heat sources: the heat from heating elements placed along the barrel, and the frictional and viscous heat generation within the polymeric material (Chen and Turng, 2005).

Additionally, the volumetric filling of the mold cavities is determined by factors related to the design of the part and the mold, by the polymeric material used, and by the selected injection parameters, such as: barrel temperature profile (in several zones), nozzle temperature, packing pressure profile, back pressure, injection pressure, fill/pack/hold switchover points, injection speed profile, among others (*op.cit.*).

However, in spite of all the scientific and technological resources applied, the probability of occurring nonconformities in the production stage cannot be completely eliminated, and in general these nonconformities take place due to the complex interrelationship between the molded part and the constructive elements of the mold, the rheological characteristics of the molding compound and the processing conditions (Goodship, 2004).

In this sense, these complex interrelations are difficult to model analytically, which implies the determination of the causes of the nonconformities and their respective solutions by knowledge intensive and experience-based activities (Goodship, 2004).

However, this valuable knowledge generated during the investigation process of the causes of the nonconformities, about the selected corrective or preventive actions, as well as the results of their efficacy, frequently are not kept, and in general are difficult to recover, making it difficult the organizational learning from the solution of the nonconformities.

3. MULTI-AGENT APPROACH AND PROPOSED ARCHITECTURE

In the literature, a software agent is considered as a basic technology that can be used to support knowledge management (van Elst *et al*., 2004; Shen *et al*., 2001). Thus, in this paper, an architecture is proposed for a Case-Based Reasoning Agents approach to support collaborative nonconformity problem solving in the thermoplastic injection molding process domain. The proposed architecture, which is shown in Fig. 1, was developed base on the GAIA methodology (Zambonelli *et al*., 2003), which proposes the conception of a complex multi-agent system from a natural metaphor of a computational organization and the abstraction of organizational roles and interaction models that, in essence, define the social behavior of an agent society.

The proposed architecture also aims at being compatible with the Foundation for Intelligent Physical Agents Specifications (FIPA, 2002), which is currently responsible for disseminating the agent technology and the interoperability of its standards with other technologies. The proposed architecture comprises the following active software entities (agent classes) and their respective roles:

Case Requester Agent (or Applet Agent): it is the agent that is capable of communicating with internal entities (e.g. other agents) and external (e.g. human being), and it acts in the name of the user in the context of the system. In addition, it allows, in an autonomous and transparent manner, the exchange of information and knowledge. Among its roles, the ones below are mentioned:

- Receive the user input from a graphical user interface (GUI);
- Communicate with all the knowledge resource agents active in the agent society through the message transfer in order to complete, in a cooperative way, the necessary common tasks for an efficient knowledge recovery.

Case-Based Reasoning Agents - Knowledge Resource Agents Type (also called ColibriFinalAgents): these are knowledge resource agents in the domain, and are capable of accessing and processing a specific case base structure in order to reply to a request from a user. It is important to observe that each agent, in particular, can be distributed in different collaborating manufacturing organizations (different hosts in an Intranet/Internet), as well as to point to specific processes along a production chain, in a perspective of a geographical and organizational distribution.

Case-Based Reasoning Matchmaker Agent - Matchmaker Agent Type: it is the agent capable of locating all the Case-Based Reasoning Agents - Knowledge Resource Agents Type that are able to respond to a specific request, and it is based on the matchmaking interaction pattern (Jha *et al.,* 1998), since it is a dynamic agent society, where agents can enter or leave the society any time. The steps of this interaction are as follows: (a) the Case-Based Reasoning - Knowledge Resource Agents register/advertise their capabilities with the Matchmaker Agent, which stores them in the form of pieces of meta-information; (b) an Case Requester Agent – Applet Agent queries the Matchmaker for agents with a desired capability to start a direct interaction later with these agents using the portions of meta-information to control the interactions among the agents.

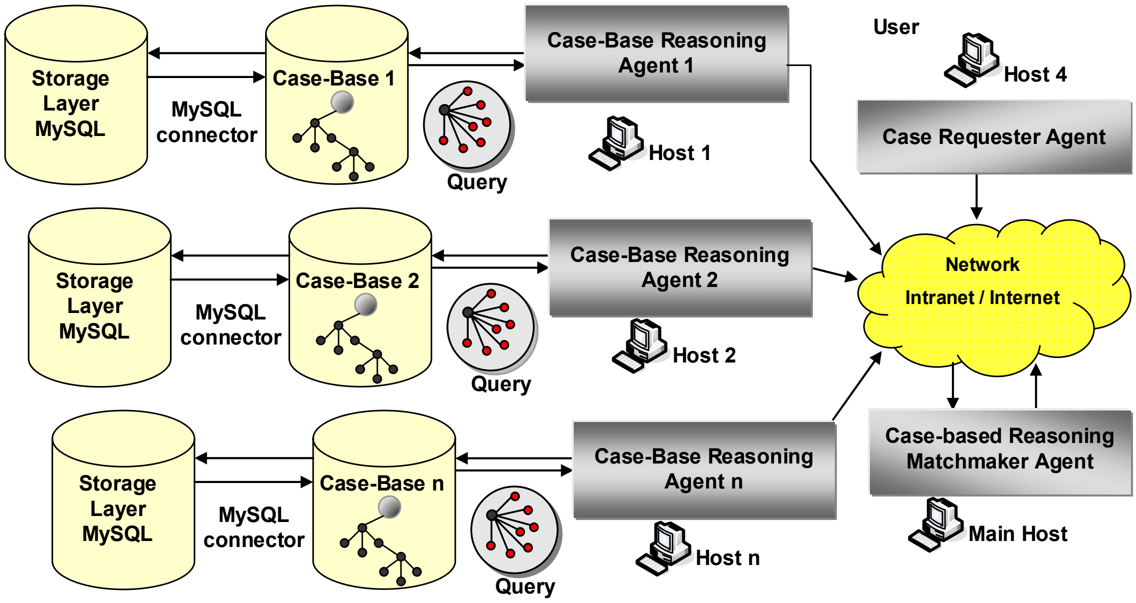

Figure 1. Architecture of the proposed multi-agent system organization

3.1. Case-based reasoning and case as formal knowledge representation

According to Aamodt and Plaza (1994), case-based reasoning is a problem solving paradigm that is able to utilize the specific knowledge of previously experienced problem situations (also called cases), instead of using solely general knowledge of a problem domain, or making associations along generalized relationships between problem descriptors and conclusions.

In this sense, a new problem is solved by finding a relevant similar previous case, and reusing it in the new case, considering the assumptions that underpin case-based reasoning, which in certain domains, in particular, the problems to be solved tend to be recurrent and to repeat with small variations compared with its initial version. In this way, previous solutions can be reapplied, with small modifications. It is important to observe that case-based reasoning is also an incremental and sustained learning approach, since a new case (experience) is retained each time a new problem has been solved (Aamodt and Plaza, 1994).

On the scientific point of view, case-based reasoning is based on the models of the cognitive science, in particular the theory of dynamic memory and on the packages of memory organization (PMO's) proposed by Schank (1982).

It can be understood as a dynamic model in a process-oriented vision, whose objective is to emphasize the idea of cycle in sequential stages in a model composed of the six processes - CBR-Cycle (Watson, 2003): through the Retrieve process, the more relevant similar case (or cases) is retrieved and selected from the case library; through the Reuse process, the retrieved knowledge of the previous case can be combined and applied to solve the new problem; through the Revise process this solution is tested, *e.g.* being applied to the real world environment or evaluated by a specialist or by software simulation; through the Review process, once a new solution has been generated, the outcome should be critically reviewed; through the Retain process, useful experience is retained for future reuse, and the case-base is updated by a new learned case; through the Refine process, also called case-base maintenance, errors, conflicts, contradictions and duplicates in the case-base can be eliminated, in order to optimize retrieval performance over time, or reacting to new customer requirements.

The case-based reasoning application is influenced significantly by the content and by the form of representation of cases stored in the knowledge base or memory of cases (Watson, 2003; von Wangenheim and von Wangenheim, 2003).

According to Kolodner (1993), “a case is a contextualized piece of knowledge representing an experience that teaches a lesson fundamental to achieving the goal of the reasoner”. In this sense, a case can be considered as a record of experiences that contain explicit or tacit knowledge, which is not only a summary of a solution to a previously problem, but also can be reused in the solving of new similar problems and in general a case includes:

- the problem/situation description: which describes the state of the world at the time the case was happing;
- the solution: which presents the stated or derived solution to the problem specified in the problem description, and it could be an action, a plan or a useful information to the user;
- the outcome: which describes the state of the world after implementing the proposed solution, or, still, how well the solution solved the problem.

3.2. Case structure and case-base

The proposed case structure, which can be accessed by Case-Based Reasoning Agents to support nonconformity problem solving in the thermoplastic injection molding process domain, consists of a collection of 127 descriptors organized in three main blocks of descriptors: problem/situation, solution and outcome. In this context, a descriptor is an attribute-value pair used in the description of each case.

The problem/situation descriptors consist of 114 descriptors organized in nine categories: classification and boundary conditions of nonconformity; molding material, part design, mold design, temperature profiles, pressure profiles, time profiles, sequence and motion profiles, and machine information according to variables involved in the process (Manrich, 2005; Chen and Turng, 2005; Rees, 2002; Rauwendaal, 2000, Mok *et al*., 2000; Malek *et al*., 1998).

The solution consists of a collection of 10 descriptors: team responsible for the suggested solution, root cause, available image *URL*, interim containment action, permanent corrective action, other potential causes of failure, reasoning steps used to solve the problem by team, acceptable corrective actions that were not chosen (reasoning and justifications), unacceptable corrective actions that were ruled out (reasoning and justifications), expectation of what will result upon deployment of the suggested solution.

The outcome consists of a collection of 3 descriptors: outcome itself that describes what happened as a result of carrying out the solution, whether the outcome was a success or failure, and actions to prevent recurrence.

Thus, a case was formally defined as:

$$Ci = < Di, Si, Oi > \tag{1}$$

where D_i is the problem/situation descriptor, S_i the solution and O_i the outcome.

3.3. Case-Based Reasoning Agents Type Responsibilities

The role model of the proposed Case-Based Reasoning Agent Type developed on the basis of GAIA methodology (Zambonelli *et al*., 2003) identifies the role that an agent has to play within the multi-agent system, which consists of four attributes: responsibilities, permissions, activities, and protocols. In this paper, in particular, the responsibilities (liveness properties) that describe the tasks that the agent must fulfill will be highlighted.

In this sense, given a new problem to be solved, a Case-Based Reasoning Agent Type receives a message from the Case Requester Agent (an Applet Agent) with the descriptors of the new case to be solved, and starts the recovery process of the relevant cases that are more similar, and then it sends a message to the Case Requester Agent with the descriptors of the k-most similar recovered cases: problem/situation, solution and outcome descriptors.

In this perspective, the Case-Based Reasoning Agent Type calculates the similarity between the new case to be solved and the cases stored in the case-base using the *Nearest-Neighbor Algorithm*, as shown in Eq. (2).

$$sim(C_{new}, C_{base}) = \frac{\sum_{i=1}^{n} sim(f_i^{new}, f_i^{base}) \times w_i}{\sum_{i=1}^{n} w_i} \tag{2}$$

where w_i is the weight expressing the importance of the *i* attribute (descriptor), and f_i^{new}, f_i^{base} are the values of the *i* attribute in the new case and case-base respectively, and $sim(f_i^{new}, f_i^{base})$ is the similarity between these values, and for numeric values Eq. (3) is used.

$$sim(f_i^{new}, f_i^{base}) = 1 - (| f_i^{new} - f_i^{base} | \div l_i) \tag{3}$$

where l_i is the predetermined range values of the *i* attribute.

3.4. Instantiation and Implementation of the architecture

In order to demonstrate the proof of concept of Case-Based Reasoning Agents approach to support collaborative nonconformity problem solving in the thermoplastic injection molding process domain, the architecture was instantiated and a prototype was implemented based on the Java Agent DEvelopment Framework (JADE, 2006). The jCOLIBRI CBR Framework as core engine (GAIA, 2006) was used as the case-based reasoning core engine for processing the knowledge case-bases, which was instanced with real cases of nonconformity problem solving in thermoplastic injection molding process related to technical molded parts designed for the electronic industry, and tested in a private network.

JADE is an agent-oriented development environment and an open source software written in Java language for the development of agent applications in compliance with FIPA standards.

JADE is composed of two main components: the FIPA-compliant Agent Platform, which included functionalities that are independent of the application, such as: Transport of Agent Communications Language (ACL) messages, asynchronous in relation to the operation of the agents, encoding and parsing and agent life-cycle among others, besides a package to development JAVA agents.

On the other hand, jCOLIBRI is an object-oriented framework written in Java for building Case-Based Reasoning applications developed by the Group for Artificial Intelligence Applications (GAIA) at Complutense University of Madrid, and allows software reuse by integrating the application of software engineering techniques with a knowledge level description that separates the Problem Solving Methods, which define the reasoning processes, from the domain model, which describes the domain knowledge (GAIA, 2006).

3.5. Prototype Implementation Issues and Results

In the prototype, the knowledge retrieval strategy was implemented as a behavior of the Case Requester Agent (Applet Agent), and it is based on the dynamic configuration of a search standard, that are presented in the many panels and menus in the graphical user interfaces (GUI).

Figure 2 shows the interaction of a user with the Applet Agent, which aims in this case to recover the more relevant cases that are similar to the description of the nonconformity considered, which is available in the distributed case-bases from a geographical and organizational perspective.

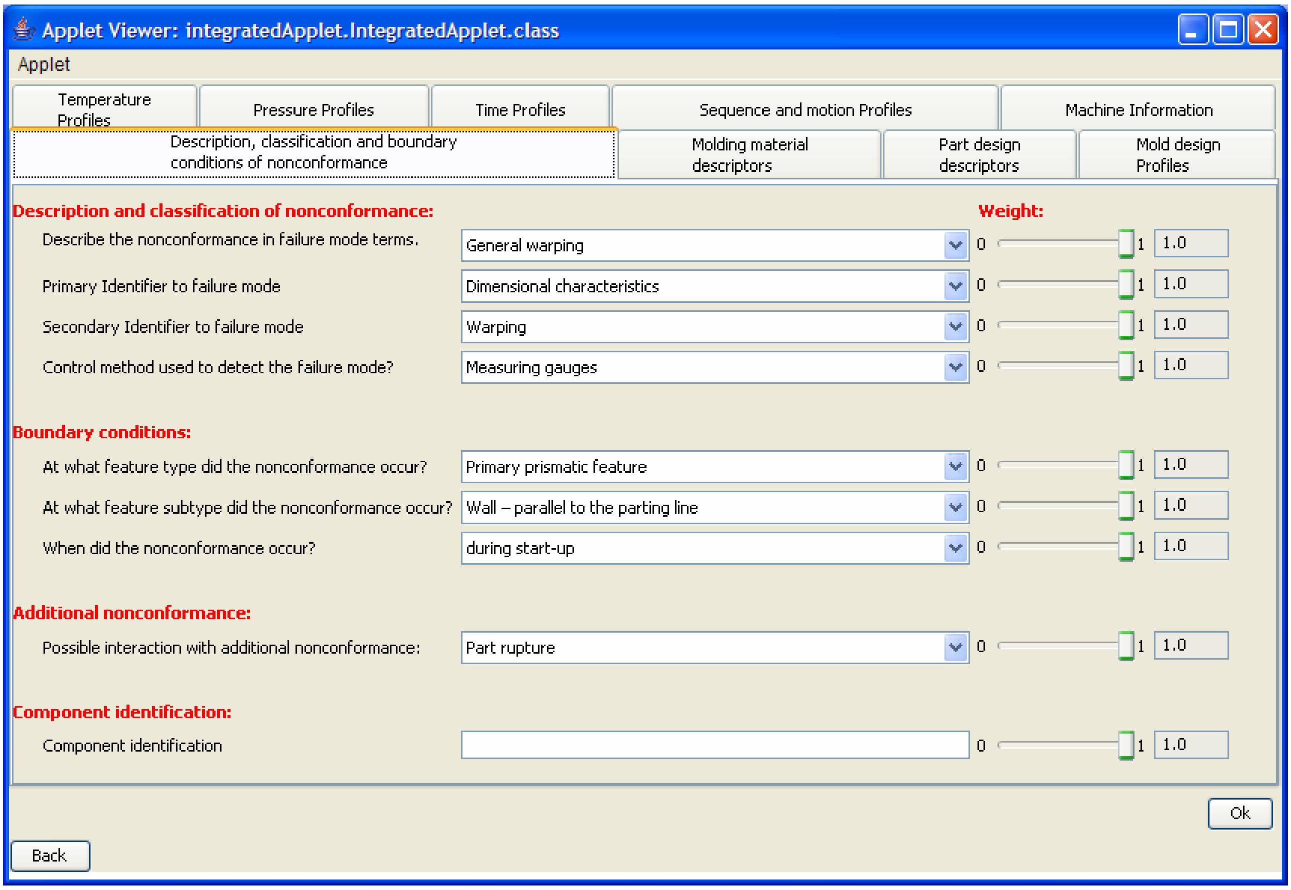

Figure 2. Case Requester Agent (Applet Agent) snapshot

In this way, the agent must in an autonomous and transparent manner identify all Case-Based Reasoning Agents - Knowledge Resource Agents Type registered in the Case-Based Reasoning Matchmaker Agent implemented in the JADE Platform through the Directory Facilitator Agent (df), and then start the exchange of messages in order to finish the task.

The panel shown in Fig. 2 illustrates the descriptors related to the classification and boundary conditions of the nonconformity. It is important to note that in the first item (*i.e.* description and classification of nonconformance) the user should write the nonconformity in terms of failure mode, and in this sense in the dropdown menu the different failure modes found in the literature are presented (Basf, 2007; Bayer, 2007; Ge Plastic, 2007; Goodship, 2004).

Additionally, are presented the descriptors of primary and secondary identifiers to failure mode, as well as the allocation of the failure with regard to the taxonomy of features related to moldability presented by Canciglieri *et al.* (2006), whose main objective is to reduce the possible ambiguity between the instances of the failure mode and to increase the expressiveness of the representation of the knowledge and the capacity of the retrieval tasks.

The concept of the "primary identifier" represents the generic aspects related to the failure manifestation and failure inducing agent, which involve: the main characteristic of the failure mode, or the characteristic of the environment under which the failure mode occurred, or the kind of solicitation. The "secondary identifier" represents the aspects related to: types of materials involved, characteristics of failure or presence of other factors or specific means (Tumer *et al.,* 2003).

It is also important to observe that the user can adjust freely the weight that expresses the importance of the attribute (descriptor), in order to personalize the calculation of the global similarity measure as a function of its necessities, optimizing the recovery process, although a default weight for each attribute can be assigned by the Case Requester Agent if the user does not perform the adjustment. The proposed recovery strategy also allows the recovery of cases even when many of the descriptors of the problem/situation related to the new case are not known or even considered relevant by the user for the considered problem.

Figure 3 shows a sequence of exchange of messages among instantiated agents in different JADE containers (Main-Container, Container-1 and 4, representing different hosts here). The exchange of messages starts with the REQUEST performative, which indicates that the Case Requester Agent (Applet Agent) waits that all the Case-Based Reasoning Agents - Knowledge Resource Agents Type (ColibriFinalAgents) perform an action that involves the retrieval of the relevant cases that are more similar to the ones found in the respective knowledge case-base.

As part of the strategy, the Case Requester Agent (Applet Agent) must encapsulate the content of the messages in a JAVA object type Hashmap, through which the sender and receiver agents must be able to encode/parse the communicative intentions.

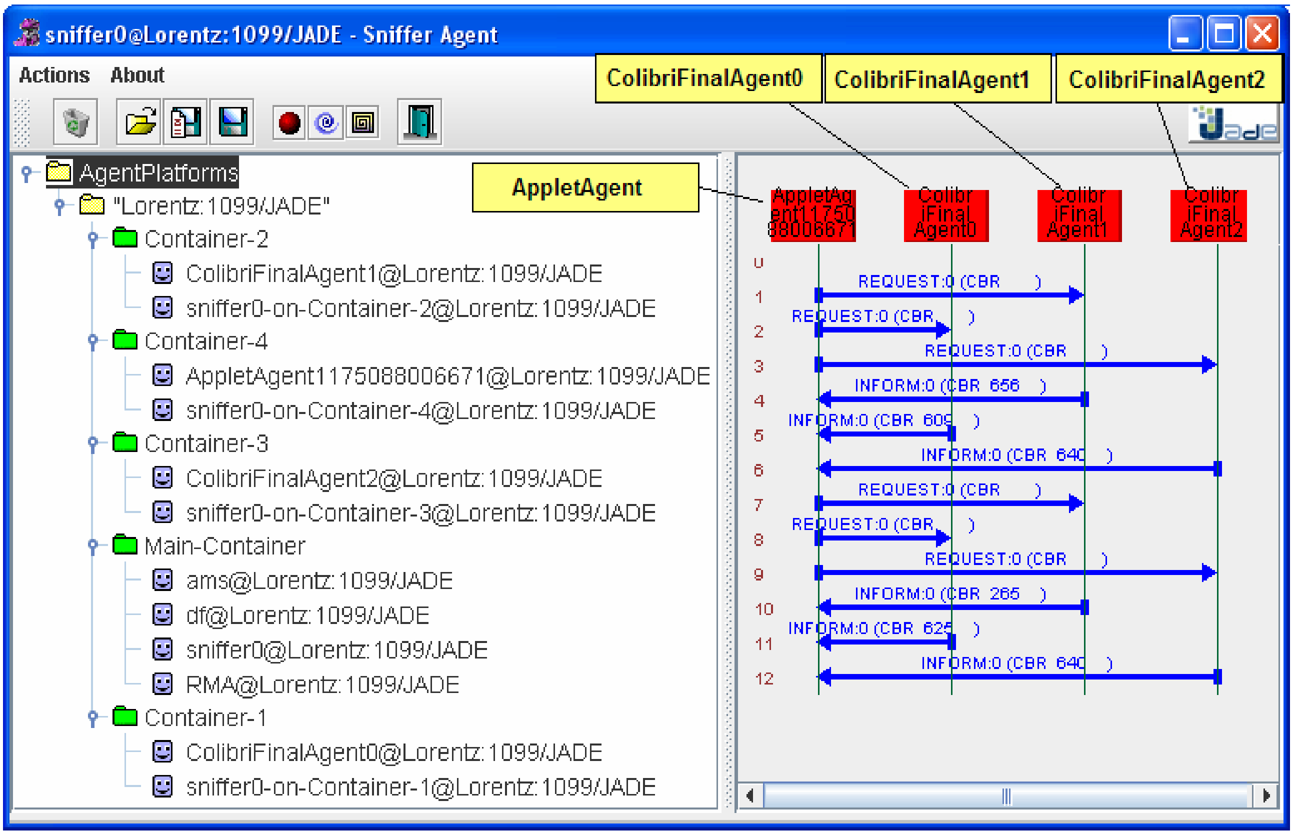

Figure 3. Exchange of messages among agents - Snifer Agent snapshop

On the other hand, each one of the Case-Based Reasoning Agents (ColibriFinalAgents), through the INFORM performative, communicate the results of this action, which are presented thereafter in the suggested solutions and outcome panels. The case based reasoning core engine or agent intentions are implemented in JADE Platform through the use of behaviors. Behaviors are logical execution threads that can be composed in various ways to achieve complex execution patterns and can be initialized, suspended and spawned at any given time (Bellifemine *et al*., 2006)

Figure 4 illustrates the solution descriptors, and it is important to point out that the knowledge retrieval strategy implemented in the Case Requester Agent involves receiving the k-most similar cases send by the many Case-Based Reasoning - Knowledge Resource Agents Types registered in the Case-Based Reasoning Matchmaker Agent, compare and organize them in an ordered list according to the degrees of similarity. In this way, in the knowledge reuse phase of the CBR Cycle, this list will serve as a guideline that will support the engineers and workers involves in the nonconformity problem solving process to find an appropriate solution for the new case. On the other hand, the new cases and their respective solutions and outcomes, after being validated by the responsible team, can be included in one of the distributed case-bases, ensuring the intellectual property related to the new solution.

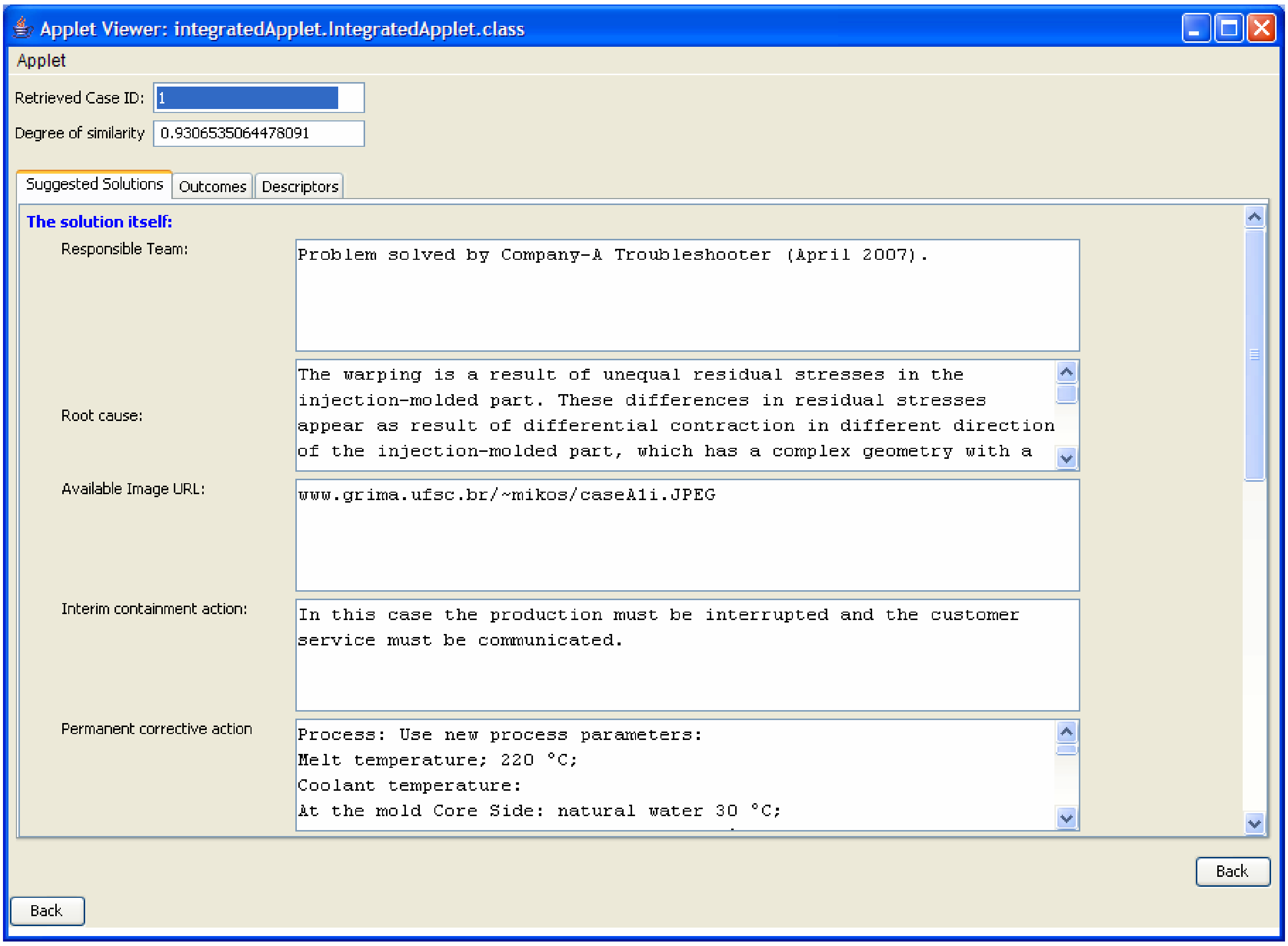

Figure 4. Solution descriptors snapshot

4. CONCLUSIONS

This paper described the development of a Case-Based Reasoning Agents approach to support collaborative nonconformity problem solving in the thermoplastic injection molding process domain.

In this paper has been proposed an architecture organization for the Case-Based Reasoning Agents approach that aims at solving the inference problems and knowledge retrieval, transparent to the user, through an Applet Agent that converts the requested queries by the user and requests the services of the other agents, each of them being responsible for connecting and processing one of the different knowledge bases that can be distributed over the Intranet/Internet.

Additionally, a case in a formal knowledge representation in the domain was presented. In this sense, the case structure has been proposed, which consists of a collection of 127 descriptors organized in three main blocks of descriptors: problem/situation, solution and outcome.

A prototype was developed using the Java Agent DEvelopment Framework (JADE) and the Case-Based Reasoning Agent Type using the jCOLIBRI Framework as a core engine for processing knowledge case-bases. Those knowledge case-bases were instanced with real experiences on the nonconformity problem solving in thermoplastic injection molding process related to technical molded parts designed for the electronic industry.

In particular, from a feasibility and conceptual perspective, the approach is adequate and promising to support collaborative nonconformity problem solving in the thermoplastic injection molding process domain, allowing knowledge inference and retrieval between human or computer agents, as well as the support to the activities of management of organizational knowledge on collaborative manufacturing environments with distributed resources.

Finally, the proposed architecture organization of the multi-agent system was developed to allow, in future works, the addition of other knowledge resources having specific knowledge bases about other complex manufacturing processes.

5. REFERENCES

Aamodt, A. and Plaza, E., 1994, "Case-Based Reasoning: Foundational Issues, Methodological Variations, and System Approaches", AI Communications - IOS Press, Vol. 7, No. 1, pp. 39-59.

Association of Plastics Manufacturers – PlasticsEurope. 20 February 2007, Plastic and Economy. 10 April 2007 <http://www.plasticseurope.org/>.

AIAG - Automotive Industry Action Group, 2006, "Quality Manual: CQI-10 Effective Problem Solving Guideline", Southfield, Michigan.

Basf Plasticsportal Europe - Technical Resource. 2007, Injection Moulding Troubleshooter. 10 April 2007 <http://worldaccount.basf.com>.

Bayer Material Science – Resources. 2007, Troubleshooting - Injection Molding. 10 April 2007 <http://bayermaterialsciencenafta.com/resources/>.

Bellifemine, F., Caire, G., Trucco, T. and Rimassa. G., 2006, "Jade Programmer's Guide", 20 September 2006 <http: //sharon.cselt.it/projects/jade/>.

Binder, M. and Clegg, B., 2007, "Enterprise management: A new frontier for organizations", International Journal of Production Economics, Vol. 106, issue 2, pp. 409-430.

Cancieglire Junior, O. and Young, R.I.M., 2006, "Information structure as support in multiviewpoint of product in the design for manufacture", Produção, Vol. 16, No. 2, pp. 354-374.

Cecil, J., Davidson, S. and Muthayan, A., 2006, "A distributed internet-based framework for manufacturing planning", International Journal Advanced Manufacturing Technology, Vol. 27, No. 5, pp. 619-624.

Chen, Z. and Turng, L-S, 2005, "A review of current developments in process and quality control for injection molding", Advances in Polymer Technology, Vol. 24, No. 3, pp. 165-182.

Chin, K.-S., Duan, G. and Tang, X.-Q, 2006, "A computer-integrated framework for quality chain management", International Journal of Advanced Manufacturing Technology, Vol. 27, No. 5, pp. 547-560.

FIPA - Foundation for Intelligent Physical Agents – FIPA, 2002, "FIPA Abstract Architecture Specification" 10 September 2006 <http://www.fipa.org/>.

Ge Plastics South America - Pesquisa Técnica. 2007, Troubleshooting guide. 10 January 2007 <http://www.geplastics.com.br/resins/techsolution/troubleshooting.html >.

Goodship, V., 2004, "Troubleshooting in Injection Moulding: An Arburg Guide", Rapra Review Reports, Vol. 15, No 16. pp.1-50.

GAIA - Group for Artificial Intelligence Applications. 7 July 2006, jCOLIBRI CBR Framework 1.1.0 version. 18 July 2006 <http://sourceforge.net/projects/jcolibri-cbr/>.

ISO. ISO 9000:2005: Quality management systems -- Fundamentals and vocabulary.

ISO. ISO 9001:2000: Quality management systems -- Requirements.

ISO. ISO 9004:2000: Quality management systems -- Guidelines for performance improvements.

JADE - Java Agent Development Framework. 17 November 2006, JADE 3.4.1 version. 20 November 2006 <http://jade.tilab.com/>.

Jha, S., Chalasani, P., Shehory, O. and Sycara, K., 1998, "A formal treatment of distributed matchmaking", Proceedings of 2th International Conference on Autonomous Agents, pp. 457-428.

Klamma, R., 2000, "Vernetztes Verbesserungsmanagement mit einem Unternehmens-gedächtnis – Repository". 259 f. Doktors der Naturwissenschaften. Fakultät für Mathematik, Informatik und Naturwissenschaften der Rheinisch-Westfälischen Technischen Hochschule Aachen. Aachen.

Kolodner, J.L., 1993, "Case-Based Reasoning", Morgan Kaufmann publishers, Inc., USA.

Malek, M., Toitgans, M-P., Wybo, J-L. and Vincent, M., 1998, "An operator support system based on case-based reasoning for the plastic moulding injection process", . Proceedins of the 4th European Workshop on Case–Based Reasoning, Lecture Notes in Computer Science, Vol. 1488, pp. 402-413.

Manrich, S., 2005, "Processamento de Termoplásticos: rosca única, extrusão e matrizes, injeção e moldes", Artliber Editora Ltda., S.Paulo, Brazil.

Mok, S.L., Kwong, C.K. and Lau, W.S., 2000, "An intelligent hybrid system for initial process parameter setting of injection moulding", International Journal of Production Research, Vol. 38, No 17, pp. 4565-4576.

Rauwendaal, C., 2000, "SPC – statistical process control in injection molding and extrusion", Carl Hanser Verlag, Munich.

Rees, H., 2002, “Mold Engineering”, 2nd Edition, Carl Hanser Verlag, New York.
Shank, R., 1982, “Dynamic memory: A theory of learning in computers and people”, Cambridge Univ. Press, New York.
Shen, W., Norrie, D.H. and Barthès. J-P., 2001, “Multi-agent systems for concurrent intelligent design and manufacturing”, Taylor & Francis Inc., New York.
Society of the Plastics Industry. 20 March 2007, Economic statistics - Global business trend**s.** 10 April 2007 <http://www.socplas.org/industry/global.htm>.
Tumer, I.Y., Stone, R.B. and Bell, D.G., 2003, “Requirements for a failure mode taxonomy for use in conceptual design”, Proceedings of International Conference on Engineering Design, Stockholm, Sweden, paper No.1612.
van Elst, L., Dignum, V. and Abecker, A., 2004, “Towards Agent-Mediated Knowledge Management”, In: van Elst, L., Dignum, V. and Abecker, A. (Eds.) Agent Mediated Knowledge Management, LNAI 2926, Heidelberg: Springer, pp. 1-31.
von Wangenheim, C.G. and von Wangenheim, A., 2003, “Raciocíno Baseado em Casos”, Editora Manole Ltda., S.Paulo, Brazil.
Watson, I., 2003, “Applyig Knowledge Management: Techniques for building Organizational Memories”, Morgan Kaufmann Publishers Inc., San Francisco, CA.
Zaidat, A., Boucher, X. and Vincent, L., 2005, “A framework for organization network engineering and Integration”, Robotics and Computer-Integrated Manufacturing, Vol. 21, issue 3, pp. 259-271.
Zambonelli, F., Jennings, N. and Wooldridge, M., 2003, “Developing multiagent systems: the GAIA methodology”, ACM Transactions on Software Engineering and Methodology, Vol. 12, No. 3, pp. 317-370.

6. RESPONSIBILITY NOTICE

The authors are the only responsible for the printed material included in this paper.