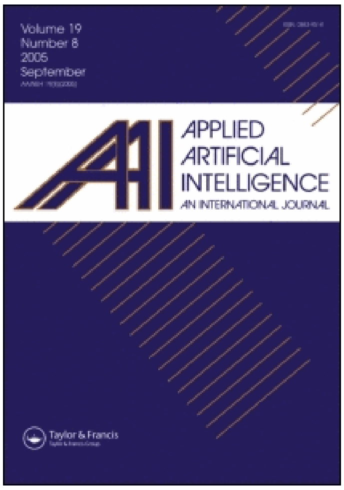

Applied Artificial Intelligence
Publication details, including instructions for authors and subscription information:
http://www.informaworld.com/smpp/title~content=t713191765

A MULTI-AGENT SYSTEM FOR THE MANAGEMENT OF ABNORMAL SITUATIONS IN AN ARTIFICIALLY GAS-LIFTED WELL

José Aguilar [a]; Frank Prato [b]; César Bravo [b]; Francklin Rivas [a]
[a] Facultad de Ingeniería, CEMISID, Universidad de los Andes, Merida, Venezuela [b] PDVSA, West Division, Maracaibo, Venezuela CEMISID (Center of Microelectronics and Distributed Systems),

Online Publication Date: 01 May 2009

To cite this Article Aguilar, José, Prato, Frank, Bravo, César and Rivas, Francklin(2009)'A MULTI-AGENT SYSTEM FOR THE MANAGEMENT OF ABNORMAL SITUATIONS IN AN ARTIFICIALLY GAS-LIFTED WELL',Applied Artificial Intelligence,23:5,406 — 426

To link to this Article: DOI: 10.1080/08839510902872256

URL: http://dx.doi.org/10.1080/08839510902872256

Applied Artificial Intelligence, 23:406–426

ISSN: 0883-9514 print/1087-6545 online
DOI: 10.1080/08839510902872256

A MULTI-AGENT SYSTEM FOR THE MANAGEMENT OF ABNORMAL SITUATIONS IN AN ARTIFICIALLY GAS-LIFTED WELL

José Aguilar[1], Frank Prato[2], César Bravo[2], and Francklin Rivas[1]
[1]*Facultad de Ingeniería, CEMISID, Universidad de los Andes, Merida, Venezuela*
[2]*PDVSA, West Division, Maracaibo, Venezuela CEMISID (Center of Microelectronics and Distributed Systems)*

□ *Concerning industrial automation, the management of abnormal situations becomes more important everyday. The ability to detect, isolate, and handle abnormal situations in industrial installations, could save huge amounts of money which is normally invested in reparations and/or wasted because of unjustified stoppage of processing plants. In this work, a system for the management of abnormal situations in an artificially gas-lifted well based on agents Abnormal Situations Management System (ASMS) is developed, which is part of the architecture of the industrial automation based on multi-agents systems (SADIA) proposed in Bravo, Aguilar, and Rivas (2004). This agent is based on the intelligent distributed control system based on agents (IDCSBA) reference model proposed in Aguilar, Cerrada, Mousalli, Rivas, and Hidrobo (2005). The MASINA methodology (Aguilar, Hidrobo, and Cerrada 2007) is used in matters of analysis, design, and implementation.*

In modern industry, there are key activities closely linked to the importance of the productive process. One of these is the maintenance of the normal operating conditions, through the diagnosis and management of abnormal situations in all plants, so that the expected production levels can be maintained, the delayed production can be reduced by early waste correction, and the unexpected plant stoppage and environmental impact caused by leaking or spilling, among other things, can be diminished.

Carrying out the petrol production process requires a set of industrial devices in charge of hydrocarbons extraction, treatment, distribution,

This work was supported in part by FONACIT under grant 2005000170, CDCHT-ULA under grant I-820-05-02-AA, and PCP Automation Integrated to Processes of Production No. 200500380.

Address correspondence to José Aguilar, Universidad de los Andes, Escueal de Ingenieria de Sistemas, Department of Computacion, Nucleo La Hechicera, Piso 2, Ala Sur, Mérida 5101, Venezuela. E-mail: aguilar@ula.ve

and transport. The artificially gas-lifted petrol wells represent one of the oil production sources in the scope of surface installations. Monitoring, diagnosis, and treatment of abnormal situations in artificially gas-lifted wells is a vital function for the petrol production process.

The use of agents in developing applications offers a whole range of new opportunities, as the agents support the creation of software systems with a higher adaptation capacity (Wagner 2002). The agents can be proactive and autonomous entities, which are crucial characteristics nowadays when process conditions become more and more complex. Using the multi-agents system, we find new possibilities of innovative integration strategies.

THEORETICAL ASPECTS

Multi-Agents Systems (MAS)

The agents constitute a new paradigm in software application programming (Bigus and Jennifer 2001; Jennings 2001; Weiss 1999). The actual meaning of this concept is discussed frequently. Nevertheless, there are some characteristics associated with the agents that have been accepted by most of the researchers in this area. Amongs these characteristics we find autonomy, mobility, rationality, reactivity, and proactivity. So, we can say that a software agent is a program with the capability of acting independently anywhere it finds itself, with the aim of accomplishing the tasks it has been designed for. The agents' characteristic are presented below:

Autonomy: is the ability to be autonomous and self-sufficient without human or other agents' intervention.

Sociability: the agents should interact with other agents and humans through a communication language that has been defined.

Reactivity: an agent should detect its surroundings: the physical world, a user's graphic interface, other agents, etc. The agent should change and adapt itself to the surrounding changes.

Proactivity: an agent should not only respond in consequence to the surrounding changes, it should also take its own initiatives (oriented towards achievements).

Mobility: is the ability an agent has to move through the nodes in a network.

Intelligence: the agents could be provided with reasoning mechanisms that would allow them to manage situations intelligently and evolve by the accumulation of experience through learning mechanisms. In this case, the agents are called "intelligent agents."

An MAS is a set of agents that work together adding their individual capacities to solve complex problems. This strategy is very attractive as it allows the agents in charge of solving these problems to specialize and locate in any position, so that they can assist the search for solutions in a distributed way. The MAS has the possibility to reduce centralization and sequenciality, providing distributed systems.

MASINA

The MAS-CommonKads methodology has been extended in order to include new features for agent modelling (Aguilar et al. 2007). The extended methodology, called MASINA (MAS for integrated automation), is a new approach for MAS modelling in industrial automation processes (Aguilar et al. 2005; Bravo et al. 2004). This methodology completely redefines the experience, coordination, and communication models and extends the agents models and tasks. The model adjustments in MASINA are presented below:

Agent Model: The model describes the general characteristics of the agents in charge of solving a problem; this model is also a link to the other models (Aguilar et al. 2007). Furthermore, in MASINA, this model is allowed to point out which reference models should be used when designing the agent, as well as the position where the MAS is located.

Tasks Model: For achieving the agents objectives, their tasks are described in this model. In the extension MASINA, the macro algorithm is defined as well as the techniques that will be used for reaching the tasks (for example, intelligent techniques).

Intelligence Model: This is a new model that substitutes the MAS-CommonKads experience model. In this model, the reasoning and learning mechanisms are identified, as well as the strategic knowledge about the domain, tasks, and experience, in order to achieve the MAS objectives.

Coordination Model: In MASINA, this model describes the coordination schemes between agents, as well as the conflict resolution strategies, the direct and indirect communication mechanisms, the meta languages, and the ontologies. It describes the different conversations used in the system.

Communication Model: In MASINA, the interactions between agents are described in this model. So this model defines each one of the speaking acts of each one of the conversations in the system.

Intelligent Distributed Automation Systems Based on Agents (IDASBA)

The Intelligent Distributed Automation System Based on Agents (IDASBA) (Bravo et al. 2004) is composed of three abstraction levels (see

Figure 1), each one represented as an MAS. It models at the higher level the elements that take part in the productive process, and at the lower levels the applications that support the mentioned process, such as the control of the process, supervision, or failure management.

In the first level, the trade objects are modelled as agents, that is, the productive process regarded as an MAS where the different productive units are modeled as agents. The agents in this level negotiate between each other so that they can reach an agreement that will help the accomplishment of the established production tasks; such agreements come out of the logical nature of the productive process.

In the second level, each one of the first-level agents is taken as an MAS which is composed of other agents in charge of managing the activities, necessary for accomplishing the goals of the trade objects: process control, engineering and maintenance, abnormal situations management, production factors management, and production planning. The activities previously listed are common for each one of the first-level agents, and as a result, all the agents in that level will be basically constituted by agents that execute each one of these activities. Nevertheless, there are activities that belong exclusively to a trade object in particular, which are modelled through specialized agents, which complement the basic design of each agent in the first level. Finally, because the activities that second-level agents carry out are complex, a third abstraction level is proposed, in which the agents of the second-level are taken as SMAs. Similarly, the tasks concerning the development of every activity are distributed between the agents in that level. A reference mark has been used distributed control system based on agents (IDCSBA) as a reference model for the third

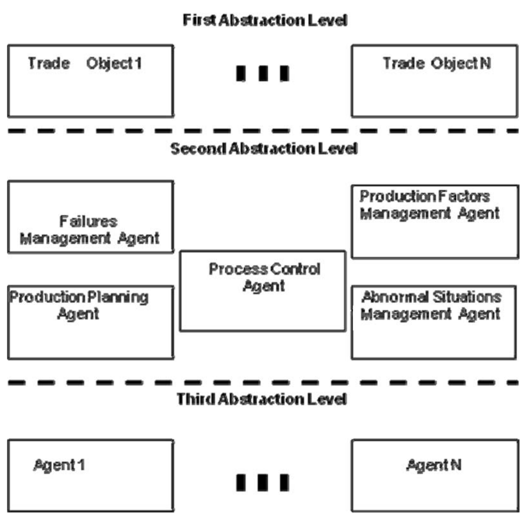

FIGURE 1 SADIA model.

abstraction level, for the agents of the second abstraction level (process control, failures management, abnormal situations management, etc.).

SCDIA Reference Model (Intelligent Distributed Control System Based on Agents)

The SCDIA is a multi-agent reference model, specifically designed for industrial automation systems (Aguilar et al. 2005). The SCDIA proposes a collection of agents that represent the elements of a process control loop, with the intention of establishing a generic frame for the management of control activities in industrial automation. Thus, the following agents are found:

Measuring agent: collects the necessary information to get to know the state of the process.
Controller agent: takes actions based on the observations about the state of the process and the system.
Coordinator agent: modifies or softens the decisions of the controller agent and establishes new objectives and services. Directs other agents in its community.
Actuation agent: carries out the decisions taken by the controller, coordinator, and/or specialized agents.
Specialized agent: carries out special necessary tasks in the control community.

As seen in Figure 2, the SCDIA can be divided into two levels: level of interaction with the environment, where the measuring and actuation

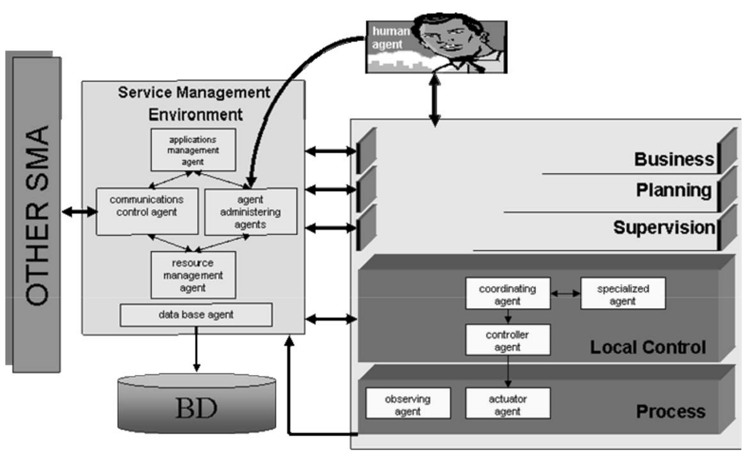

FIGURE 2 SCDIA model.

agents are found; and a decision level where the other community agents are found. In addition, the SCDIA proposes an agent's community in charge of the jobs organization within the SCDIA (Aguilar et al. 2004). This community has been called Services Management System (SGS), they are basically the agents established by FIPA to give support to Agents Communities, such as (Aguilar, Bravo, Rivas, and Cerrada 2004; Foundation for Intelligent Physical Agents 1997): an agent's administrator agent, a communication control agent, and three resource, data, and applications management agents.

Abnormal Situations Management in Industrial Processes

Frequently in technological plants, processes are forced in order to maximize production. When forced, the probability of technologic or staff failures increases. So, the equipment is not following the predesigned boundaries and these are challenged; the process operates beyond its original possibilities and the staff is asked to manage a complex operation (Abnormal Situation Management System 2001; Barberii 2000; Optegrity Brochure 2007).

An abnormal situation is a condition, an event, or a circumstance that involves the incapacity of a mechanism to carry out functions associated with it. This causes operations to deviate from their normal operational state. The consequences could be minimal or catastrophic. These situations get extended and change over time, adding complexity to the required intervention. For this reason, it is important to understand the generating factors. These can occur in isolated situations, but in most cases those kinds of situations occur as a result of the interaction between many variables.

The particular effects or symptoms of an abnormal situation will allow its identification. Apart from these effects, the consequences of abnormal situations affect the productive process (Abnormal Situation Management System 2001; Barberii 2000; Bravo et al. 2004). The goal of abnormal situation management in industry is to maintain the operational performance, maintain the production assets availability, reduce operators' duties, and minimize operation costs. For this, the abnormal situations should be identified early and steps should be taken before the operation gets affected.

For managing abnormal situations in production plants (multiple artificially gas-lifted wells) in an oil field exploitation unit, Bravo et al. (2004) shows a generic model based on agents named abnormal situations manager agent (AMSA). The design of this agent is based on the SCDIA reference model. The MAS is composed by the five SCDIA agents, each one with specific duties related to abnormal situations management.

DESIGN OF THE SYSTEM OF MANAGEMENT OF ABNORMAL SITUATIONS BASED ON AGENTS FOR WELLS LAG

In this section the design of the agents within the abnormal situations management agent (ASMS) in artificially gas-lifted wells appear, following the MASINA methodology. The agent models are developed as well as the tasks, communication, and intelligence. These models take out the principal characteristics of the system about to be built.

Oil Field Exploitation Unit (OFEU) as a Multi-Agents System

In Bravo et al. (2004) the designs of five agents are developed. These represent the most important installations in a UEY, users of the industrial automation design SADIA. These five agents are representative of the typical petrol exploitation plants using artificial gas-lifting. So, the design's first level is formed by the following agents:

Well agent: responsible for all the activities necessary for the functioning of a petroleum well.
Flux station agent: this agent models the functioning of the flux stations.
Compressor plant agent: manages and controls the activities associated with the stations related to the gas compression in the exploitation process.
MILAG agent: this agent manages activities related to distribution of the injection gas, carried out at the different LAGs.
Tanks yard agent: this agent manages and controls the activities related to the tank yards of an UEY.

In the Second level, the following agent's collection was proposed as the common shape for all first-level agents: process control agent, production planning management agent, production factors management agent, maintenance engineering agent, and abnormal situations management agent, which are described in Bravo et al. (2004). In the third level, the SCDIA reference mark was used as an agent's model (Aguilar et al. 2005).

Artificially Gas-Lifted Well Agent

This instance comes out when using the industrial automation design SADIA, developed in Bravo et al. (2004), in an UEY composed of LAG wells. In Figure 3 the LAG well agent is represented; this agent belongs to the first abstraction level of the design. According to SADIA, the well agent is composed of the following agents: process control agent, production factors management agent, production planning agent, failures management agent, and abnormal situations management agent.

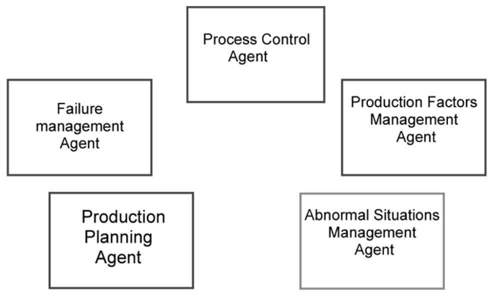

FIGURE 3 LAG well agent.

These agents belong to the second abstraction level and they are in charge of making sure that the well agent carries out his tasks. Finally, each one of the second-level agents are taken as MAS, which are modelled according to the IDCSBA reference model (Aguilar et al. 2005).

Abnormal Situations Management Agent for a LAG Well

The abnormal situations management agent for an artificially gas-lifted well (see Figure 4), manages the abnormal situations that could take place in a monitored gas-lifted well. This agent holds a database which contains the experience of the operators, engineers, and experts on abnormal situations management. He is responsible for detecting possible abnormal situations, shut alarms, and recommends possible ways of solving them collectively.

Specifications of the Model for an Abnormal Situations Management Agent for a LAG Well

The abnormal situations management agent for a LAG well is based on the SCDIA model. Each one of the agents, belonging to the abnormal situations management agent, is described below:

Measuring agent: collects the data coming from the SCADA system and from the data bases that could bring in information about the state

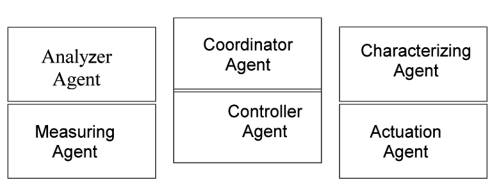

FIGURE 4 Abnormal situations management agent.

of the processes taking place in a monitored well. As well, he could process and validate data, calculate rates, etc. if needed. The agent passes on the information of the state to the respective AMSA agents.

Controller agent: receives information about the state emitted by the measuring agent and compares the current conditions with the wanted conditions of the process. If these conditions drift apart from a certain tolerance zone, he executes the rules contained into an inference motor, which can activate alarms, orders of diagnosis applications, characterizing of operation conditions, etc. He changes the established values for operation conditions (for example, nominal values for process variables).

Actuation agent: depending on the decisions taken by the controller agent, activates alarms and makes them visible by all actors involved in the solution of the problem (SCADA operators, optimization engineers).

Coordinator agent: sends the service requests to the specialized agents. Also contacts the failures management agent to request corrective maintenance planning, and the actuation agent to indicate the following actions.

Analyzer agent: there are some rules inside the inference motor that require a detailed analysis based on mathematical models of the wells (generated by the optimization engineers) about a possible abnormal situation that could have taken place. The analyzer agent will do that job.

Characterizing agent: He is in charge of characterizing the behavior of the different variables that affect the well: head pressure (CHP), pipe pressure (THP), gas injection rate (QGI), gas lift multiple pressure (GLMANP), as continuous, intermittent, and erratic variables. The characterization of variable behavior is useful data for the analyzer agent.

All the MAS agents should make use of the services provided by the service management system (SGS), so that the communication between agents can be granted as well as an efficient management of resources and services. In this article, a data management agent, particular for this domain, will be developed, incorporated to the SGS in order to support the tasks accomplishment related to the management of information coming from the different pertinent repositories. For developing this abnormal situation management agent, the MASINA methodology has been used (Aguilar et al. 2007). Based on each one of its models, the principal characteristics of the MAS are described. Here, the general model shown in Bravo et al. (2004) is made exclusive for a LAG well. Below, the developed models for the particular case of the controller agent are exposed; the rest can be referenced in Prato, Aguilar, Bravo, and Rivas (2007).

Agent Model

Table 1 provides the description of one of the agents within the abnormal situation management agent for a LAG well (the controller agent). In that model, its objectives, characteristics, and offered services are specified. Only one of them is exposed. For more information see Prato et al. (2007).

TABLE 1 Agent Model for the Controller Agent of the ASMS Agent for a LAG Well

Controller agent	
Name	Controller
Position	Third level
Components	NA
Reference Mark	SCDIA
Description	Evaluates the well's monitored behavior, comparing it to the desired behavior and it is able to make recommendations for managing the abnormal situation.
Tasks	
Name	Evaluate condition
Description	Its task is to evaluate the well's condition in order to identify abnormal situations and in that case recommend possible corrective actions that will maintain the well in the wanted conditions.
Entrance Parameters	Information about the state, countersigns
Exit Parameters	State of the well, corrective actions, plant stoppage alarms, gas and fire protection alarms.
Activating Condition	As the agent is created
Ending Condition	As the agent is eliminated
Success Condition	The well functions as there are no abnormal conditions
Failure Condition	There are abnormal conditions that cannot be solved.
Ontology	Abnormal situations management in a well and communication ontology.
Services	
Name	Recommend corrective actions
Description	Using the information about the state and the established countersigns, the agent is able to recommend possible actions to take in order to solve the possible abnormal situations.
Entrance Parameters	Information about the state, countersigns
Exit Parameters	Recommendations of possible corrective actions
Name	Submit alarm activation order for fire and gas protection
Description	When and abnormal situation happens involving a disaster, the agent can send an alarm activation order so that the process of protection against fire and gas can get started.
Entrance Parameters	Information about the state
Exit Parameters	Fire and gas alarm activation order
Name	Send secure stoppage alarm activation order
Description	When an abnormal situation happens implying danger for the monitored well, the staff, or the environment, the agent can send an alarm activation order to initiate the secure stoppage procedure.
Entrance Parameters	Information about the state
Exit Parameters	Alarm activation order to initiate the secure stoppage

TABLE 2 Agent-Tasks Relationship

Agents	Tasks
Abnormal Situations Management Agents for a LAG Well	
Observation agent	T1. Obtain information about the state T2. Request data to the data management agent (AGD) T3. Preprocessing of data coming from the process
Controller agent	T1. Diagnosis T2. Identification of causes T3. Identification of consequences T4. Damage degree evaluation T5. Changes in intonation parameters
Coordinator agent	T1. Request for the execution of characterization service T2. Request for the execution of detailed analysis service
Actuation agent	T1. Execution of corrective actions T2. Alarm status register T3. Changes in the state of the alarms
Analyzer agent	T1. Abnormal situation analysis
Characterizer agent	T1. Behavior characterization of the CHP variable T2. Behavior characterization of the THP variable T3. Behavior characterization of the QGI variable T4. Behavior characterization of the GLMNP variable

Tasks Model

Table 2 describes the tasks carried out by each agent in order to accomplish his objectives and specific services.

In addition, the tasks for the agent manager of data belonging to the SGS specialized for AMSA are listed in Table 3.

Some of the tasks for the controller agent are described in Table 4. See additional tasks in Prato et al. (2007).

Coordination Model

Table 5 the strategies that the AMSA agent's community will use for achieving group objectives. Particularly, the conversations between agents are described as well as the ontologies used by the MAS, the meta languages, and each one of the used coordination schemes (Aguilar et al. 2007). Figure 5 is the description of a conversation of the controller agent.

TABLE 3 Agent-Tasks-Data Manager Agent Relationship

Data manager agent	T1. Information search T2. Information updates T3. Data and communication errors management

TABLE 4 Example of the Task Model: Diagnosis Task of the Controller Agent

Task	
Name	Diagnosis
Objective	Diagnose abnormal situation after realizing symptoms
Description	Abnormal situations, possible causes, and possible consequences are identified after the detection of symptoms
Precondition	Information about the state of the process
Subtasks	Identification of causes and consequences
Name	Identification of causes
Objective	Identify causes of abnormal situation
Description	Once the abnormal situation has been identified, it is necessary to find out the causes
Precondition	Abnormal situation identified
Subtasks	NA
Name	Identification of consequences
Objective	Identify consequences of an abnormal situation
Description	Once the abnormal situation has been identified, it is necessary to find out the possible consequences
Precondition	Abnormal situation identified
Subtasks	NA

TABLE 5 Example of the Coordination Model

Conversation: Control Getting	
Objectives	Determination of control actions
Agents	Observation, coordination, actuation
Initiator	Observer
Conversation acts	Obtaining information about state of the well, request for diagnosis agents services, execution order
Precondition	State of the process
Termination condition	Action or error execution order
Description	The object of the controller agent is to execute the necessary actions before an abnormal situation, for this, it interacts with the observer, coordinator and actuation agents.
Coordination Scheme	
Following objective	To plan the interactions between the observation, coordinator, and actuation agents
Type	Predefined
Default coordination	Centralized through message exchange mechanisms, in order to relate the agents present in the conversation
Planning	
Type	Predefined
Technique	There are request protocols between the observer and the coordinator agents
Communication Mechanism	
Type	Direct
Technique	Message exchange
Meta language	Agents communication language (ACL-FIPA)
Ontology	Abnormal situations management in a LAG well, of communication

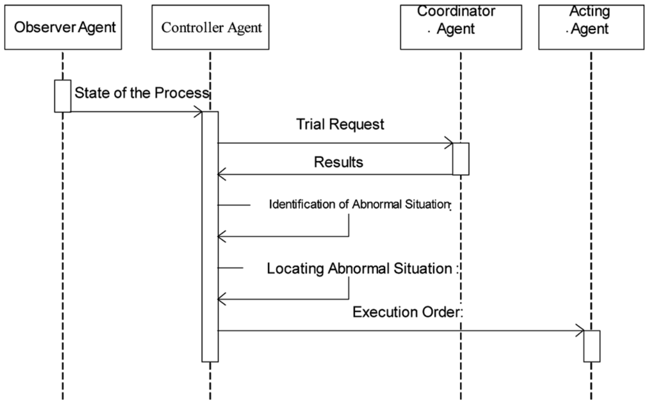

FIGURE 5 Diagram of the sequences of the control holding conversation.

Communication Model

The communication model describes the interventions (or communicating acts) of a conversation. Table 6 shows some of the communicating acts of the conversation described in the previous section.

TABLE 6 Example of the Communication Model

Communicating Act: Trial Request	
Name	Trial request
Type	Request
Objective	Trial request
Agents	Controller, coordinator agent
Initiator	Controller
Precondition	Request for analysis of abnormal situation
Termination condition	Acceptance of request
Conversations	Control holding
Description	The controller agent requests the coordinator agent to contact the analyzer agent in order to revise in detail a given abnormal situation.
Communicating Act: Execution Order	
Name	Execution order
Type	Inform
Objective	Send an execution order for an action
Agents	Controller, actuation
Initiator	Controller
Precondition	Action to be executed
Termination condition	Sending of action to be executed
Conversations	Control holding, executing actions
Description	The controller agent sends an execution order of certain action to the actuation agent.

TABLE 7 Example of the Intelligence Model

Reasoning Mechanism	
Information source	Agente de observación, Agente Coordinador, SGS
Activation source	Intervention
Inference type	Deductive
Reasoning strategy	Expert system
Learning Mechanism	
Type	Adaptive
Representation technique	Rules
Learning source	Failures or success
Updating mechanism	Previous experiences are used for updating knowledge.

Intelligence Model

Table 7 is the controller agent's intelligence model, describing the reasoning, learning, and knowledge representation mechanisms.

The inference motor of the controller agent is formed by production rules. Each rule consists of an IF and a THEN part (also called as a

TABLE 8 Composing Rules of the Inference Motor of the Controller Agent

Rule Number	Rule Description
1	Determines whether the web is actually producing by the use of a flux sensor.
2	Determines when there is a difference between the sign well producing (obtained by rule 1) and the sentry stored well category.
3	Determines when there is a break in the production line while the web is operating.
4	Determines changes on the artificial lifting through comparison of the artificially designed method with the operation method.
5	Determines when gas is being injected to a GL well.
6	Determines when a GL well is closed without authorization.
7	Determines when the gas-lifting flux is obstaculized and the well is not being supplied.
8	Determines a rupture in the connecting line between the multiple GL and the GL well while it is receiving GL gas.
9	Determines when the production line of a GL well is blocked or closed.
10	Determines a possible communication between the pipe and the coating in continuous GL wells.
11	Determines possible communication between the pipe and the coating in continuous GL wells.
12	Determines a major pipe-coating communication. A major communication is defined as the cause of a drastic fall in production.
13	Determines possible blocking or erosion of the merla valve.
14	Determines when there are problems with the control loop or the merla valve.
15	Determines infiltrations of the merla valve.
16	Determines problems with the gas flux at the control loop or at the actuator of the valve decrease.

condition and an action). The IF parts are a set of conditions combined in a logical way. The rules are described in Table 8.

IMPLEMENTATION OF AMSA WITHIN THE SCDIA

JADE was used as a developing tool, as it has already been used to develop the service management system of the SCDIA (Aguilar et al. 2004; Bellifemine, Poggi, and Rimassi 1999), as well as the SCDIA agents community for code generation (Zayas, Aguilar, Rivas, and Cerrada 2005) and the maintenance engineering agent (Cerrada, Faneite, Aguilar, and Cardillo 2007).

Implementation Process of the AMSA Based on Agents

For developing AMSA, the following package structure was created in JAVA. It contains the different classes of elements (agents and ontologies) that form part of AMSA (see Figure 6).

The package AMSA agent contains all the classes that represent the agents taking part in AMSA which are: measuring, controller, actuation, coordinator, characterizer, and analyzer. The subpackage AMSA agent GUI contains the classes that make use of the graphic interface of the actuation agent.

The package AMSA ontology is integrated by three JAVA subpackages which contain the JAVA classes representing the concepts, predicates, and actions of the agents, that is, the AMSA ontology. The package AMSA Util, provides a group of classes that allow the management of different databases, as well as other utilitarian classes.

A class's hierarchy was designed that implements the AMSA design (see Figure 7). This hierarchy has an agent as a basic class. The component agents of AMSA, extending this class are: observer agent, controller agent, actuation agent, coordinator agent, and the specialized agents characterizer and analyzer agents. The basic interaction mechanisms between agents are inherited from the basic class as well as a set of basic methods that could be called in order to implement the personal behaviors

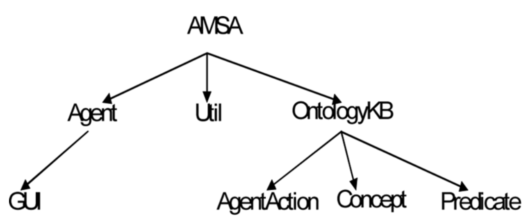

FIGURE 6 Structure of the AMSA JAVA packages.

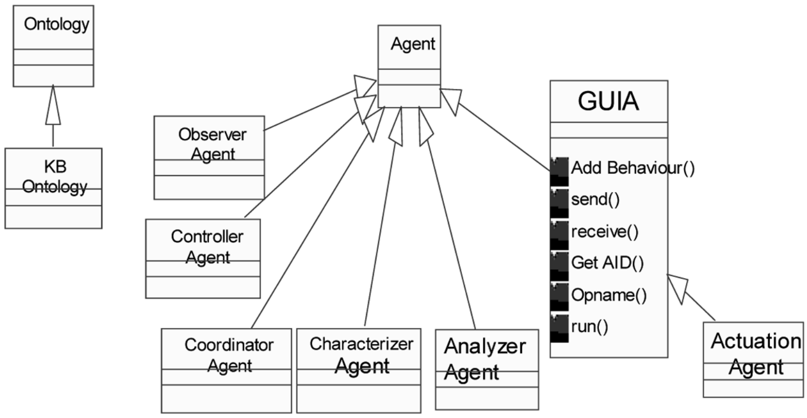

FIGURE 7 AMSA classes diagram.

of each of the agents (for example, sending and receiving messages or registering in other domains). Each AMSA agent is associated to one or more behaviors, which allows the implementation of the tasks and services of the agent. The behaviors are units of logical activities of different types in order to reach complex execution patterns. An agent can have many behaviors that are executed as the initiation conditions are completed; these are generally events or conditions related to a specific environment. The actuation agent has a graphic interface in order to interact with the user. For this, the actuation agent extends the GUI agent class. On the other hand, the knowledge ontology has been defined as an extension of the ontology class.

The classes hierarchy was made in order to accomplish the models developed using the MASINA methodology.

STUDY CASE

The following study case has the aim of demonstrating the AMSA functioning in a well, so that when there is an abnormal situation in a given well, the AMSA generates an alarm that tells the user about it as well as recommendations that would help to get over it.

Petrol Exploitation Process

In order to carry out the petrol production process, a group of installations is required in order to facilitate the extraction, treatment, and hydrocarbons distribution.

The petrol exploitation process is described in artificially gas-lifted wells. This exploitation process is located at the oil field exploitation unit and the main components are the wells, flux stations, artificial lifting multiples, and compressor plants and tank yards. The process begins and ends at the well and involves many different processes as extraction, separation, pretreatment, oil storing and distribution, and gas distribution and compression.

Abnormal Situations in LAG Wells

For the hydrocarbons extraction from the bottom to the surface, the LAG wells have a series of installed elements and field devices. They are also associated to lifting multiples and flux stations (see Figure 8). In the daily operation of the LAG well, the different undesired factors and circumstances produce abnormal situations that affect the productivity of the well and its components.

In this article, a group of abnormal situations typical in LAG wells has been selected. These have been used by the AMSA in order to define the IF-THEN rules so that the abnormal situations can be corrected. In our case, they were used for nourishing the knowledge base of the controller agent (see Table 8), which will allow the MAS the implementation of such situations.

Managing abnormal situations in an artificially gas-lifted well implies a constant observation of the state of the well, that is, observe the functional variables of the well and verify that the values are within the permitted range. In case the state breaks the permitted operation range, an abnormal situation is taking place that should be diagnosed. For this its causes and consequences should be identified and an action course should be designed for solving the abnormal situation. In case the well experiences

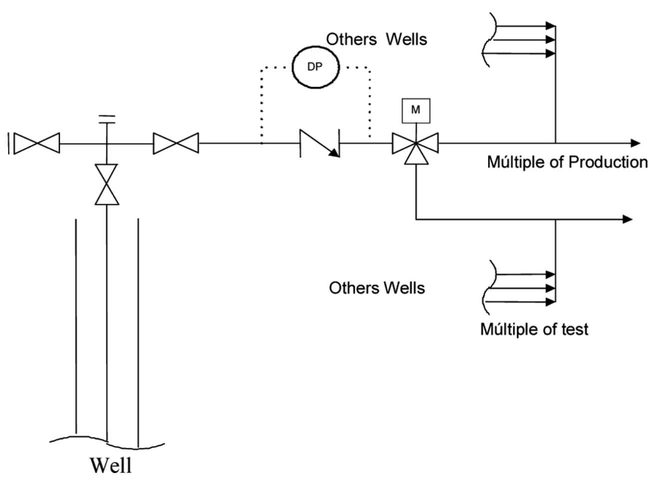

FIGURE 8 Process flow chart.

abnormal situations that have not been regarded before at the knowledge base of the AMSA, the optimization engineer is in charge of the solution.

Use of the AMSA in the Abnormal Situations Management for LAG Wells

For the management of abnormal situations in LAG wells, using the AMSA prototype, the JADE platform, should be executed in an application's server. The SCDIA service management system should be located in the main container of the platform together with the actuation agent belonging to the AMSA. Secondary containers of the JADE platform should be created as well. These contain the analyzer, characterizer, controller, and coordinator agents of AMSA (see Figure 9).

Once AMSA is initiated, a graphic interface will appear before the user (see Figure 10), which contains the supervised wells. Each observation agent associated to a well will request of the data manager agent of the service management system a view of the state of the well. In this study case, the data management agent should obtain data about the intonation parameters and the functional variables associated with the well from the SCADA and the databases. If necessary, the obtained data will be treated (rates, standard deviation, etc.) by the observer agent. The observation of the state of the well is permanent. Once the observing agent knows about the state of the well, he sends it to the controller agent who is in charge of detecting abnormal situations. For this, the controller agent uses his reasoning mechanism based on rules for the purpose of making a diagnosis of the abnormal situation, that is, identify its causes, consequences, and send an alarm later on followed by recommendations to the actuation agent.

Finally, the actuation agent shows the diagnosis to the operator agent perceived as a visible alarm icon, which could be recognized or not. If the user clicks this icon, the actuation agent will show a screen (see Figure 11)

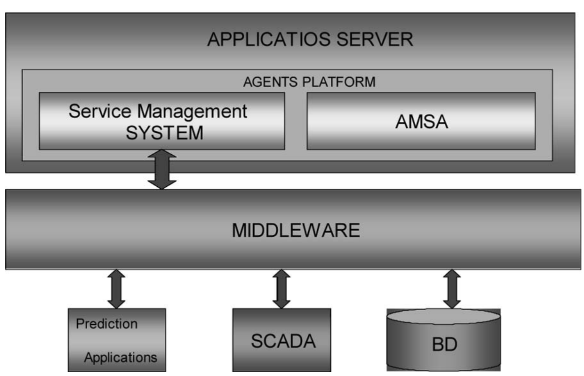

FIGURE 9 Multi-agents platform in the applications server.

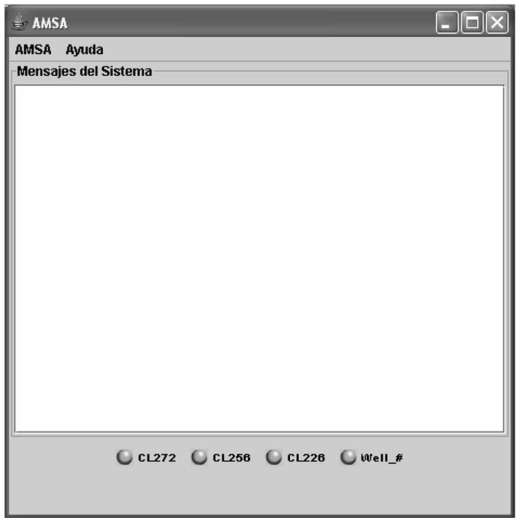

FIGURE 10 Actuation agent screen.

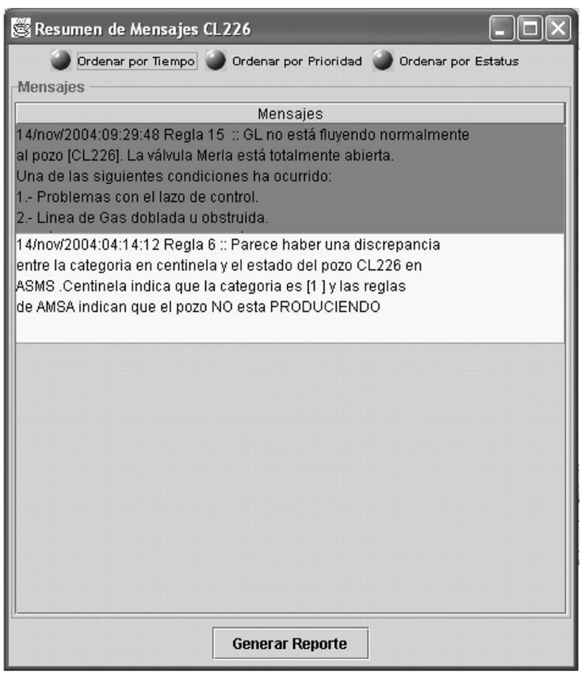

FIGURE 11 Alarms summary of the well CL226.

in which the descriptions of the abnormal situations are presented, as well as the recommendations for treating these abnormal situations and a time stamp. For recognizing the alarm, you should click on the description of the abnormal situation that will be blinking. Then a dialog screen will be seen in order to confirm if you wish to recognize the alarm. Once the alarm is recognized, the actuation agent registers it and has taken actions so that the knowledge base of the AMSA controller agent can get updated with the aim of nourishing the experience of the system.

Sometimes the controller agent requests the coordinator agent to contact the specialized analyzer agent for a detailed analysis of a certain abnormal situation detected by it. For this, the coordinator agent should request the characterizer agent to classify the well variables CHP, THP, QGI, and GLMANP. Furthermore, the coordinator agent should request other data from the observer agent and then be able to request the special analyzing service of the diagnosis specialized agent. Once the diagnosis agent makes the analysis, the results are sent to the coordinator agent who finally sends it to the actuation agent who will show it to the user.

CONCLUSIONS

In this work, the general model proposed in Bravo et al. (2004) was developed for the management of abnormal situations in the case of an artificially gas-lifted well. For this, the SCDIA agents were specified, apart from two particular specialized agents for the LAG well, the characterizer, and analyzer agents. The prototype was also implemented.

By following the MASINA methodology agent's models, tasks, coordination, communication, and intelligence for each agent were designed. To provide intelligence to the MAS, a knowledge base was incorporated based on production rules IF-THEN to the controller agent, which allow him to reason about abnormal situations presented in the LAG wells. For system implementation, JADE was used together with the SCDIA code generation community developed in Zayas et al. (2005), using the advantages provided by these tools.

In order to demonstrate the functions of the prototype, a group of wells were selected in which to use the prototype for managing abnormal situations. During this phase, the prototype detected abnormal situations in the wells and their actual existence was tried in order to prove their validity. In addition, the abnormal situations detected by the prototype were compared to the ones detected by the ASMS, showing coherence.

One of the advantages of using the prototype is that its distribution can manage abnormal situations in more wells with no risk of overworking the application's server where the system is being executed, a situation that happens with the use of ASMS due to its centralized focus. Another advantage of the abnormal situations management agent in LAG wells is its easy adaptation to the exploitation units. For this, a multi-agent platform should be installed in a system associated with the respective exploitation unit and execute it in each of them—service that cannot be provided by the current system due to its license limitations.

One disadvantage of the prototype is the absence of a component that would allow the incorporation of new rules in a graphic way, service that is currently provided by the abnormal situations management system in PDVSA-Occidente (ASMS). Developing future works is proposed with the purpose of incorporating intelligent behaviors that provide the controller agent with better reasoning mechanisms, as well as the development of another component that would allow the incorporation of new rules in a graphic way to the ASMS. Within the MASINAS intelligence model for the controller agent (the one requiring this ability), the introduction of this characteristic (learning) to the system is foretold.

REFERENCES

Abnormal Situation Management System (ASMS). 2001. *User's Guide.* Houston, TX: ABB Inc.

Aguilar, J., V. Bravo, F. Rivas, and M. Cerrada. 2004. Diseño de un modelo de gestión de servicios para sistemas multiagentes. In *Proc. XXX Latin American Conference on Computer Science*, Arequipa, Perú, pp. 440–450.

Aguilar, J., M. Cerrada, G. Mousalli, F. Rivas, and F. Hidrobo. 2005. A multiagent model for intelligent distributed control systems. *Lecture Notes in Artificial Intelligence* 3681:191–197.

Aguilar, J., F. Hidrobo, and M. Cerrada. 2007. A methodology to specify multiagent systems. *Lecture Notes in Artificial Intelligence* 4496:92–101.

Barberii, E. 2000. *Pozo Ilustrado.* Fourth Edition. PDVSA.

Bellifemine, F., A. Poggi, and G. Rimassi. 1999. JADE: A FIPA-compliant agent framework. In *Proc. Practical Applications of Intelligent Agents and Multi-Agents*, London, England, pp. 97–108.

Bigus, J. and B. Jennifer. 2001. *Constructing Intelligent Agents Using Java.* 2nd ed. Canadá: Wiley Computer Publishing, Toronto.

Bravo, C., J. Aguilar, and F. Rivas. 2004. Diseño de una Arquitectura de Automatización Industrial Basada en Sistemas Multiagentes. *Revista Ciencia e Ingeniería, Facultad de Ingeniería* 25(2):77–88.

Cerrada, M., R. Faneite, J. Aguilar, and J. Cardillo. 2007. Agents-based design for fault management systems in industrial processes. *Computer in Industry* 58:313–328.

Gómez-Pérez, A., M. Fernández, and A. De Vicente. 1996. Towards a method to conceptualize domain ontologies. In *Proc. Workshop on Ontological Eng.* European Coordinating Committee for Artificial Intelligence, pp. 41–52.

Jennings, N. 2001. An agent-based approach for building complex software systems. *Communications of the ACM* 44(4):35–41.

Optegrity Brochure. 2007. Abnormal condition management for process manufacturing. *Technical Report*, Gensym Corporation, Austin, Texas, USA. www.gensym.com. Last accessed 2/2/09.

Prato, F., J. Aguilar, C. Bravo, and F. Rivas. 2007. Desarrollo del Agente de Manejo de Situaciones Anormales para un Pozo de Levantamiento Artificial por Gas. *Technical Report 05-2007*, CEMISID, Universidad de los Andes. Mérida, Venezuela.

Wagner, T. 2002. An agent-oriented approach to industrial automation systems. *Technical Report*, Institute of Industrial Automation and Software Engineering, University of Stuttgart, Stuttgart, Germany.

Weiss, G. 1999. *Multiagent Systems.* Massachusetts: The MIT Press.

Zayas, W., J. Aguilar, F. Rivas, and M. Cerrada. 2005. Sistema de Generación de Agentes de Control. In *Proc. IV Latin American Conference on Systems*, USA, pp. 166–177.