Sensor Networks and Multi-Agents in Industrial Workflows

Emmanuel Sardis, Anastasios Doulamis and Nikolaos Matsatsinis

***Abstract*— Production lines need fast technologies for critical online decisions. Orchestration of different signals and processes from a number of different sensors around the industry, into related workflows are the key signals on workflows tasks recognition. The advantages of sensor networks' technologies outputs, that their outputs are controlled by multi agents systems, in order to orchestrate and execute related production workflows, is the main point of this research work. In specific, the aim of this manuscript is to present a system that manipulates and controls these workflows executions. System architecture and sub modules are presented in more detail through the manuscript, plus a production scenario implementation.**

***Index Terms*— Distributed architecture, Multi agents, Sensor networks, Production environments, Workflows.**

I. Introduction And Problem Statement

Internet technology and related speed on events and products provide competitive pressure that forcing companies to consider strategies to reduce costs and compress time between each stage of the value chain [1]. The concept of product design collaboration over the internet has been widely adopted to reduce time-to-market project's life cycle. Different systems and networks have to collaborate into one end product that will be then added like a 'LEGO' particle in order to create a final product. Integration and collaboration are a common term in the enterprise systems [2]. In collaborative product design, each team member must be responsible for one or more works and contribute what they can in different domains of expertise at various stages to overcome the major weakness of traditional face-to-face communication collaboration. Sensor networks that work either in mobile installations or remotely through the web, have been introduced in production environments that earns the benefits of interoperability and cooperation in multinational and international distributed companies.

From the perspective of supporting collaborative activities, research on the collaborative design can be divided into two main purposes,

(a) To assist users solve complex problems intelligently, and
(b) To mediate cooperative activities intelligently.

Manuscript received March 30, 2011.

Emmanuel Sardis is with the Distributed, Knowledge and Media Systems Laboratory of the National Technical University of Athens-and Adjunct to the Technical University of Crete, 9, Heroon Polytechniou street, 15773 Zografou, Athens, Greece (email: sardism@mail.ntua.gr)

Anastasios Doulamis is with Technical University of Crete, Kounoupidiana, Chania, Crete, 73100, Greece (email: adoulam@ergasya,tuc.gr)

Nikolaos Matsatsinis is with Technical University of Crete, Kounoupidiana, Chania, Crete, 73100, Greece (email: nikos@ergsy.tuc.gr)

These researches are categorized into the domain of computer supported cooperative work (CSCW), where workflow systems play a critical role. These workflow systems constitute the coordinate mechanisms and the 'heart' for a collaborative design environment, with the benefits of flexible process definition, easy tracking of activities, and effective process management.

Coming from 1990s are attracting the scientific community which has started almost from early 2000 to widely use and build computer support for workflows management, using artificial intelligence and multi agents' systems [3]. Researchers apply multi agent technology in the workflow system to achieve the benefits of autonomy, social ability, pro-activeness, and reactivity. Further, using software multi agent technology to control and orchestrate the inputs from sensor networks, provide flexible and autonomous solutions for workflow management for business process coordination.

Current workflow management technologies have difficulties in solving the challenges of collaborative production lines in distributed environments with dynamic nature of product development, distributed knowledge, and resources. The risks of collaboration occur significantly when an organization losses direct control over the product development process, incomplete information is disclosed with lack of transparency [4].

One of the major issues is to deal with the interoperability and communication between different sensor networks that monitor the production chain, cooperate with them, orchestrate in a manner that dynamically be able to adjust and redesign through easy to end user interfaces. Most of the time, adaptive systems have to be implemented and inserted into the production line, for connecting the sensors with the main system that controls the production line. This is acceptable for a couple of sensors but it makes the whole process of monitoring difficult to be extended in an adaptive manner, in big numbers of different types of adaptors. Finally in order to connect the production plants under a same umbrella of production aims, starting from input and finalizing to product output and logistics earns, is a tough task for the logistics departments in every industrial company and organization.

This manuscript presents an infrastructure that can manipulate in production environments the workflows orchestration, their maintenance and finally their reconfiguration through simple user interface that new technologies can provide. Of course, the core platform under the simplicity contains a number of state of the art technologies that cooperate and work for the end product.

Next sections provide the state of the art and the basic scientific infrastructure needed for our proposed platform.

Section three is presenting the main architecture and finally next sections are presenting our implemented system and the related lab tests that with the conclusion section fulfill this scientific work.

II. State of Art on Involved Technologies

In this section, we present a brief description of current research works on sensor networks, and introduce current limitations and our intended breakthrough in multi agent-based workflow management systems for production environments, like industrial plants.

A. Sensor networks

The geospatial technologies of Geographic Information Systems (GIS) and Global Positioning Systems (GPS) have had a transformational effect on both the science and industry of this field, as has the advent of wireless telemetry devices. The introduction of what has become known as ‘computational intelligence’ has added a new dimension to the approaches taken for analyzing geo-coded data and furthermore, to the kind of predictions that can be made from it [5]. Also, test cases that combine GSM and GPS/Galileo technologies, in Wireless Sensor Networks (WSNs) can result into a system architecture, capable of tracking and monitoring of products in the real time [6]. The collaborative nature of industrial wireless sensor networks (IWSNs) brings several advantages over traditional wired industrial monitoring and control systems, including self-organization, rapid deployment, flexibility, and inherent intelligent-processing capability. IWSN plays a vital role in creating a highly reliable and self-healing industrial system that rapidly responds to real-time events with appropriate actions [7].

On the above sensor technologies, we have used, for our lab and our use case scenario, camera networks. The camera input is manipulated from computer vision algorithms, aiming to transform and investigate through computer vision recognition and tracking the important events and particles, that control and trigger the production phases, giving the start and stop signals for a workflow.

B. Object recognition though camera sensors

In contrast to traditional sensor networks, our sensors do not read quantities. Our sensors read objects entering or leaving their monitoring region. In other words, the sensors in our case give weakly-semantic events describing which object entered or left the sensed region. These weakly semantic events are our low level events that we will exploit to infer the workflow execution.

We introduce the notion of *trap points*, as a specification of the surveillance area of our sensor. This notion is based on the observation that in computer vision systems pixel values are “what is necessary” in order to detect or recognize objects. Since objects have a finite size, we generalize this concept for our needs. The trap point in its basic construction is an event source and an associated area monitoring fundamental element.

The advantages that nowadays sensor networks provide based on cost, usability, size, installation requirements, performance and operability, provide an add-on in our methodology, since we can build a network of sensors (of cameras) using pan tilt zoom (PTZ) cameras [15], [16] around the surveyed industrial area and manipulating the objects that are participating in the related workflow.

As an event source, a trap point is a kind of sensor that generates notifications about the exit or entrance of objects in its monitoring area. As an area monitoring fundamental element, it maintains knowledge about the number and kind of objects in its surveillance area. The two views are dual but according to context they help discriminate semantically the various object motions in the surveillance areas.

Passing through the boundary of the surveillance area of a trap point generates events. The events come with a timestamp, an object Id and the elementary behavior which is a true/false Boolean value corresponding to an enter or exit behavior.

- The *timestamp* is necessary in order to have precise knowledge of the time the event happened. As we will see later an event is closely related to a micro-task.
- The *object Id* is an identifier of the object that triggered the event.
- Though *elementary*, these sensor readings are enough to keep track of the monitored objects and to generate events.

These signals of events, from the sensor’s side, provide enough signals that can be transferred through gigabit Ethernet networks in a central or in a distributed controller system infrastructure for further manipulation that will trigger the related workflows phases. This controller system is described in the following sections.

C. Multi agents

An agent is a software module that can autonomously perform routine tasks with a degree of intelligence. Multi agents can be applied to filter sensors data, interpret information and merge it, monitor activities from multiple separate and distributed environments, decision support, etc. Multi agents in application domains typically possess four key characteristics that are autonomy, reactivity, communication, and goal driven. They are capable of acting autonomously, cooperatively, and collectively.

Aside from general agent applications in product development, more specific research exists in using agents to configure processes. Specifically, the following research shows that agents can discover and select web services in order to create larger applications or processes using ontologies and system languages:

- Business Process Execution Language (BPEL) can be used to express the initial social order of a multi-agent system. This language can be extended to allow agents to compose adaptable workflows of web services [17].
- Agents can be used to build an application by selecting web service implementations that best match the quality criteria of the application [18].
- Ontologies and semantic web service descriptions can be used to dynamically discover potential workflows to meet system objectives [19].

- Agents can be used to create dynamic workflow for simple design tasks [20].

Multi agent developing frameworks for production execution, most of the times failed due to research based environments that cannot handle the pressure of requirements that a real production and especially industrial environment needs. Characteristics of adaptability and reliability, on their performance, have been recognized in very few cases, where multi agents systems exist and work like (international airports and vessel ports, for the transportation and manipulation, of hand languages and containers).

This system reliability is ensured using latest multi agents platforms like JADE [9], mobile C [10], etc. From the above is clear the lack of multi agents' technologies into production environments like automobile companies and their related product workflows. Our proposal finds such production cases an ideal environment for applicability of our research.

D. Production lines processes

Production line is a complex process involving the coordination of many activities that transform customer and company inputs into fungible products and services. It is inherently an agent-based process with many *human* agents involved in the transformation of inputs to outputs. As a human agent, we simulate the employees activities and participation in the production lines. Most of the times humans and automated tools like robots are the key actors in the related workflows. The human agents provide great flexibility and robustness in the process but they also introduce great variability and inconsistency. For this and other reasons, it is difficult to execute a product development process twice and get the same results.

Many of the tasks in a product development process do not require a high level of flexibility and can be standardized and automated. This case simplifies our work most of the times but this is a small percentage in the overall picture of the production lines inside the company. Other tasks require a great level of flexibility and cannot be standardized. To address the needs for flexibility and standardization, a hybrid product development process is proposed that is composed of human agents and *software* agents.

The new process is constructed so that it can produce a given product and a family of derivative products customized around a predetermined envelope of product variations. The new product development process can therefore be used more than once, thus justifying the investment in capturing and automating it using agent methods. Since the hybrid process involves human agents and software agents, each class of agent must be focused on accomplishing tasks that it is best suited to complete within the overall product development process.

Software agents are able to perform repetitive tasks consistently and accurately. Human agents are able to make decisions and react to changing environments. To take advantage of these agent strengths, a hybrid process is created that consists of a workflow made up of software agents and human interaction agents. The workflow can then be executed to take the human agents through all the tasks, calculations, models, etc. needed to create all product deliverables for a specific member of a family of product derivatives.

The human agents operate this framework as though it were a product derivative generator. They are then able to explore a greater number of product variants, connect optimization engines to the generator, and generally explore product design space more readily. The methods in this paper focus on decomposing an existing "virtual" product development process as though it existed in some real form. The decomposition allows the identification of the software agents, which in our case are the signals of the objects recognition through the multi sensors (multi cameras inside the surveyed industrial plant) and the creation of the workflows.

However, these workflows are not static. Mapping agents are introduced that are not part of the product development process but organize process agents into run-time workflows and therefore allow for the workflow structure to be derived at execution based upon a library of available task agents.

This approach aligns with an architecture, where a company might establish a registry of product development task agents that can be connected dynamically, based on dependencies, into a workflow so that a more flexible structure results. This allows exploration of a wider envelope of derivative products including possibilities for topological product optimization as well as the execution of alternative process paths to support dynamic process objectives. These methods have been successfully applied to existing processes to demonstrate feasibility and effectiveness of the strategy.

E. Interoperability & Workflows

What is important into the keyword 'workflow', and especially into 'workflow management systems'? The workflow management coalition (WfMC) defines workflow management as: "*The automation of a business process, in whole or part, during which documents, information or tasks are passed from one participant to another for action, according to a set of procedural rules [11].*" Workflow management system (WfMS) is a software application that supports specifications and executions of workflow. Workflow design is very much related to the formulation and analysis of the value chain activities (Lau et al., 2003). WfMS provides the "*. . . procedural automation of a business process by management of sequence of work activities and the invocation of appropriate human and/or IT resources associated with various activity steps [12].*"

It contains a set of tools providing supports for the necessary services of workflow enactment services, process definition, administrative and monitoring tasks and events, workflow client applications and other invoked applications.

As we stated in the above sections the inputs in our system and in the workflow, are coming from the multiple sensor networks. The manipulation of different inputs, formats, timing and synchronization, events and triggers that support the start and end point of the timing in a workflow, have to be controlled and orchestrated under a same protocol.

For this reason, we define a generic workflow metadata model (i.e. data schema) for the development of collaboration functionalities and for the interoperability between multi agents' platforms that will control the sensors inputs. This drives us to design a standardized flow logic by using

XML/RDF ontology schema. The multi agent-based workflow architecture using the workflow data schema to perform the desired functions into a common representation of workflow logics and elements to enable design chains cooperation, design knowledge reuse and coordination in real time. The essential features of the multi agent-based workflow systems are autonomous and dynamic collaboration and communication.

Multi agents must cooperate, communicate and negotiate with other agents for coordinating and controlling the flow of works and executing any tasks of the workflow from various locations. In order for agents to communicate effectively and enhance interoperability, they need to have mutually understandable and standard semantic constructs. Using semantic web to design intelligent agents has the following advantages, e.g. easy to understand, easy resource integration, and resource reuse.

III. System Architecture and Main Modules

Our proposal uses the RDF standard in order to provide the interoperability background on the sensors information. RDF is a semantic construct framework describing and interchanging metadata for the web application. It is developed by the World Wide Web Consortium (W3C)[13] and provides the foundation for metadata interoperability across different resource description communities. The objective of RDF is to enhance the interoperability of metadata.

Extensible markup language (XML) provides structural and semantic information and offers a standard approach for describing, capturing, processing information on web. By integrating the RDF/XML concept into agent-based workflows, their messages can be organized and modeled by RDF/XML schema standard, making easier for software agents to access, understand, and share sensors data.

The agent architecture overview diagram is presented in "Fig. 1". The core modules are the following: The Workflow Management Controller (WMC), the Agent Communication Mechanism (ACM), and the User Interface (UI). The (WMC) module includes the following sub modules:

(1)system administration,
(2)Rule engine,
(3)Workflow monitoring and control module
(4)Workflow maintenance module, and
(5)Workflow execution module.

The (WMC) uses the FIPA [14] protocols to communicate with the related agents per sensor network. In each sensor network a group of related Sensor Agents is constructed that is responsible to manipulate the information per sensor or per sensor network.

The Workflow monitoring and control module executes and cashes the signals from the Sensor Agents and investigates if the related signals per sensor that construct the workflow under monitoring have been executed according to the administrator rules or based on the Rule Engine module ordering of commands.

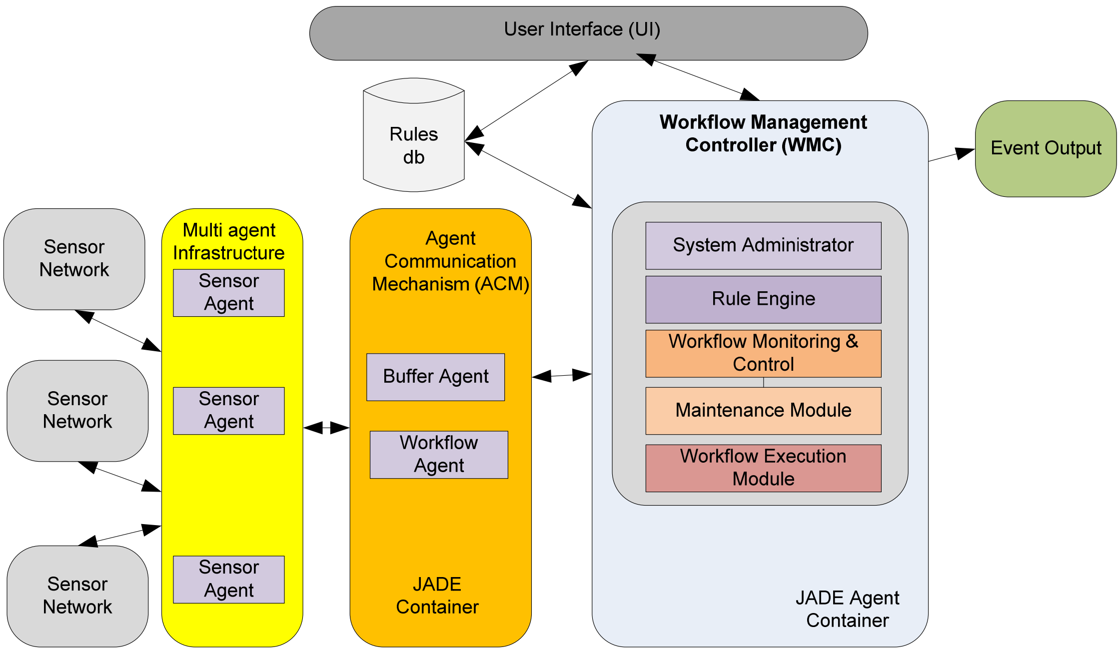

Fig 1. System Architecture, analysis of the involved agents and system modules

The Sensor Agents are controlling the signals either from the human agents, recognized through the sensors identification and recognition algorithms, or from the objects agents recognized again from related object recognition and tracking algorithms of the sensor networks of cameras sensors.

The Workflow Monitoring and Control module is performed by workflow monitoring and control agent (WfM&CA). This agent WfM&CA tracks the real-time status of flow progress, e.g. start, suspend, resume, abort, and query based on the predefined basic workflow rules that are provided from the Rules db.

The workflow maintenance module (agent) can add, modify, delete, and query any workflow template and its associating tasks. It includes three main functions, (i) design task, (ii) design workflow template, and (iii) download workflow RDF definition. The "design task" function can add, modify, and delete a task. The "design workflow template" function can add, modify, change version, and delete a workflow template by putting associating tasks into the flow template. The "download workflow RDF" function supports the RDF document management. The user module (agent) can modify personal data, show personal work list waiting for execution, and highlight the delayed tasks to corresponding user.

Finally, the Workflow Management Controller contains another module called Workflow Execution module which uses the workflow execution agent (WfEA) to pass the attributes, which outputted by tasks, to the buffer agent (BA) and workflow agent (WfA) at ACM communication mechanism for sub-sequential tasks.

The system can be connected with an output module (Event Output, based on "Fig. 1"), that related events from the (WMC) module can drive through XML messages, related alarming conditions through alarming modules. Generally speaking, the system architecture is based on a distributed organization per sensor network. This distribution is enabled according to the agents' protocols that can cooperate and work in such distributed environments. This is a benefit of our system as many case studies and scenarios need such architectures in production environments.

Also the module (UI) helps the system administrator to first arrange and configure the sensors signals that construct a workflow sequence. In simple words, it orchestrates these signals into a sequence of events, and finally monitors the workflows execution. The (UI) is capable to provide simulation functionality as the administrator can watch and evaluate the related actions and events and provide feedback, or rearrangement, or changing of some of them, based on economical and production time criteria. Of course, the decisions each time are based on the production environment case study that the system manipulates and works on. The graphical interface that in our case is a mix of sensors information and logic calculations that are provided on screen helps the end user to bear the load of data manipulation and logic calculation. In our lab test cases and also in the industrial scenario that we provided the system, the usage of sensor networks (cameras) helped the user interface to have the background of an image plus the graphics of the data manipulation on screen.

These graphics are recognized human agents (related signaling for human actors) and object agents (related bounding boxes, or dots on screen characterizing the type of the recognized object and its attributes). The Rules db with the help of the (WMC) module provide the end user information on screen, in parallel with signals like "workflow execution xx number done", or "workflow execution yy number under processing", etc. This interface is a useful image for the whole process, whole workflow, performance and monitoring by the production line administrator, or the logistics administrator and handler. As they both can investigate and control the whole sequence of events, if they are under the correct timing, and under the correct sequence of workflow sub tasks. Alarming signals in cases of abnormal conditions save a lot of production costs and delays.

IV. Computing Performance

From the computing performance, the sensor networks provide a heavy load on each machine that manipulates the sensor agents signaling. The distributed architecture has the benefit to arrange based on load the power computing in some cases, or the light computing performance in some others. For example, the algorithms that are doing image recognition in a camera signal and these results are controlled by a Sensor Agent, need power computing performance. This means more system resources. The following diagram provides the resources we need based on Sensor Agents.

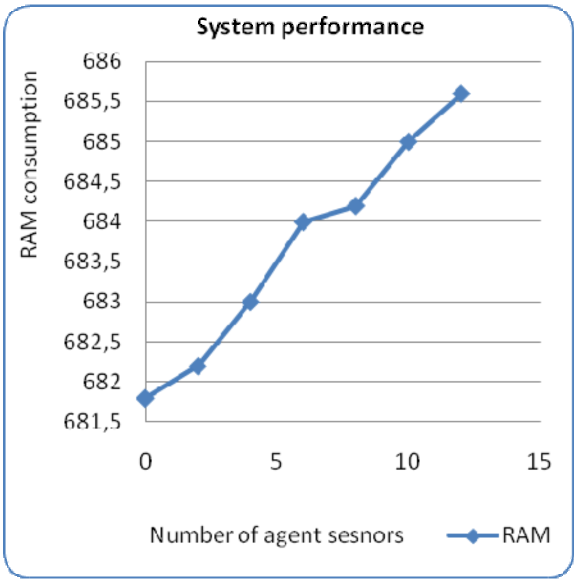

Fig. 2. System performance and scalability

V. Implementation and Lab Tests

This proposed architecture can provide workflow execution and related simulations on different sites, of the production line, either locally or remotely. A more detailed presentation is given in this section for the sub modules of our system.

A. The Ontology

As workflow is an automatically running process among multi-participants to transfer information, or tasks according to some previous defined rules or sequences. The integration and cascade of tasks formulate a typical workflow model, and thus, form the basis of WfMS. In the following "Fig. 3", we present the related ontology for the workflow model. The involved objects in this model are Task, Tasks instance, Rules, Workflows, Workflow instance, Events, and Triggers.

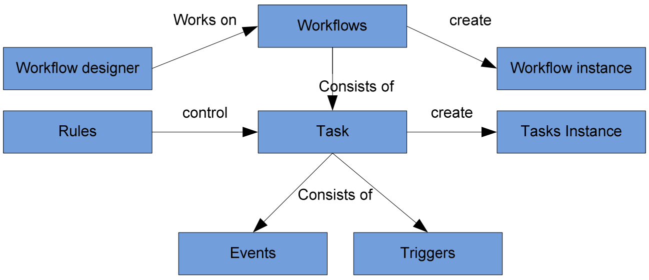

Fig. 3.Workflow ontology model

The workflow metadata model can essentially and effectively describe the information relevant to a workflow, such as pre/post conditions for the operation of a task or event, data structure, control parameters, operational modes, or extra information per sensor data. This information is manipulated by related agent models. The workflow RDF metadata schema is shown in "Fig. 4".

<!ELEMENT Rule (Rule_id, WF_id, step, time, timeout)>
<!ELEMENT Workflow (WF_id, Task_id, WF_data, step, time, timeout)>
<!ELEMENT Task (Task_id, input, output, task_data, step, time, timeout)>
<!ELEMENT Event (Event_id, input, output, event_data, step, time, timeout)>
<!ELEMENT Trigger (Trigger_id, Trigger_data, step, time, timeout)>
…..

Fig. 4. RDF schema for a workflow description

B. The sub modules

The (UI) module is responsible for the presentation of data, receiving user events, controlling the UI, and interactive with the user. For interoperability consideration, this research was developed using a JAVA-based graphic user interface (GUI). The client site can communicate with the applications without any additional hardware/software installation.

On the application middleware layer, we handle the business objects with pre-defined functional purposes in order to proceed the data manipulation and logic operation. Functional objects are encapsulated with methods and data logics for the usage of user presentation layer. The cooperation between the (WMC) and the (ACM) module provides the data storage, including workflow rules, workflow and system execution data, and administrator custom workflow rules. The (WMC) includes the related algorithms for working the workflow engine, process definition tools, administrative and monitoring tools, and communication infrastructure with the end user interface.

C. Multi agents hierarchy

The agents' framework, on our system is constructed with intelligent multi-agent support in a distributed collaborative design environment. The system has been implemented and constructed using the JADE agent platform, respectively. The proposed system consists of a number of agents, which operate autonomously and cooperate with each other to accomplish and manipulate the sensors networks information. The agent hierarchy of the workflows manipulation is presented in "Fig. 5".

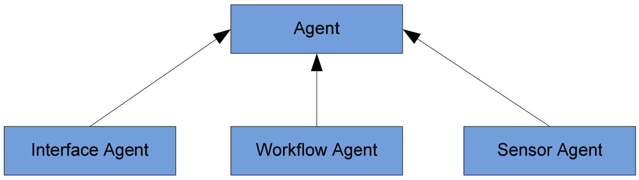

Fig. 5. Agents Hierarchy

The agents can be classified into three groups according to their functionality: (1) Interface Agent is associated with a user in an organization, (2) Workflow Agent handles process definition, controls a process instance, maintains control data of workflow and workflow execution, and finally (3) Sensor Agent is associated with the other resource from the sensors plugged into the workflows.

D. System lifecycle analysis

The system administrator firstly has to define the workflows sub tasks that is expecting to happen during the workflow. These sub tasks are characterized by related events, positions of objects or humans in the industrial plant. Positioning is similar to know approximately humans or objects related (x,y) position in the sensor image. The camera that manipulates the related event have to send in parallel with the video frames the detections and tracking form the recognized human or objects agents. This is the main step that provides the main input to our (WMC) module through the (ACM) module. For performance the sensors agents and some of the (WMC) submodules (db Rules, Workflow agent, and workflow execution module) can be installed remotely in a distributed type architecture, together with the sensors agents, giving an enhance in computing performance as ready for usage events are coming directly to the (WMC) central modules (system administrator, rule engine, and workflow monitoring and control) for further evaluation in order to drive the outputs for the (UI) module.

During the system runtime the administrator is monitoring through the (UI) the alarming and the video frames per camera (per sensor). This procedure can be exactly the same during the offline mode, for simulation and better system configuration. The provided system alarms drive the output connected modules and inform either the production line admins or the involved employees for the status of their work and performance. The last can be useful also for security and safety procedures inside the industrial plant.

VI. Industrial Environment Case Study

In this section, we present a real case of workflows collaborative scenario using information from camera sensors, distributed in an automobile company. The screenshot presents the (UI) application with related controls of workflows recognition and indications for the end user. The workers have to perform a sequence of task using spare parts for constructing an automobile. The sensors informs the system for the tasks signals and the workflow rule either provides

- an alarm when something is happening in a wrong way,
- or a warning, in order to protect the whole sequence of the workflow,
- or a successful and 'next' event in order to know the employees what workflow have to be perform next.

System performance is critical in such environments and related integration mechanisms need a lot of effort in order to be tested in such cases. Every sensor machine has its own JADE agent container built on its Java Virtual Machine (JVM) in order to manage the agents for specific behavioral mechanisms.

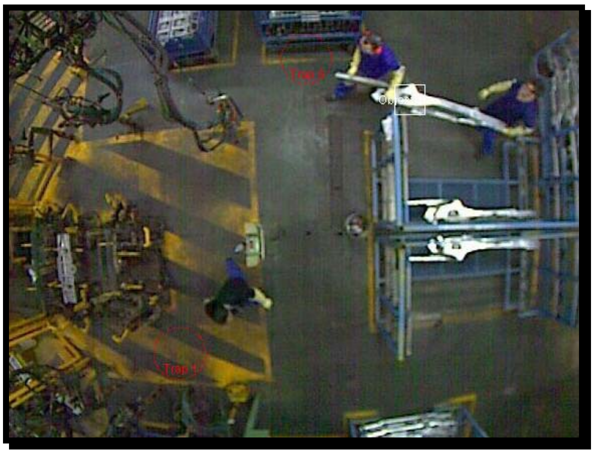

Fig. 6. User interface - GUI for workflows monitoring and manipulation

The JVM provides a complete run time environment for agent execution and allows agents to execute on the same host concurrently. Every JADE agent container will register itself to the Remote Method Invocation in JADE main container. Therefore, an agent finds any other agent, in this agent platform, through the DF in the JADE main container.

The picture in "Fig. 6" has been provided from a panoramic camera through a gigabit Ethernet network into the main workstation that is performing the execution of the (WMC) module software. In the same figure the 'red' markers on screen are the positions of the triggers that characterize the phases of the workflow performed from the three employees in the industrial plant.

Finally the "Fig. 7" presents the whole architecture [21] and its modules applied in an industrial environment for surveying and manipulating separate workflow processes in a distributed architecture but through a user friendly and only one monitor for the end user.

VII. Conclusions

This work has presented the availability of new technologies to support production environments. Their workflows manipulation on distributed networks of sensors can be handled in a manner way, providing real time orchestration and even recognition signals.

The basic architecture modules of our work, based on camera sensors and multi agents' software modules, have been presented and explained. Of course, related research on specific aspects of this research like simulation and sensors industrial case studies are our aim to be investigated in the future, providing a real time end to end system available to work under the required performance and accuracy industrial environments needs.

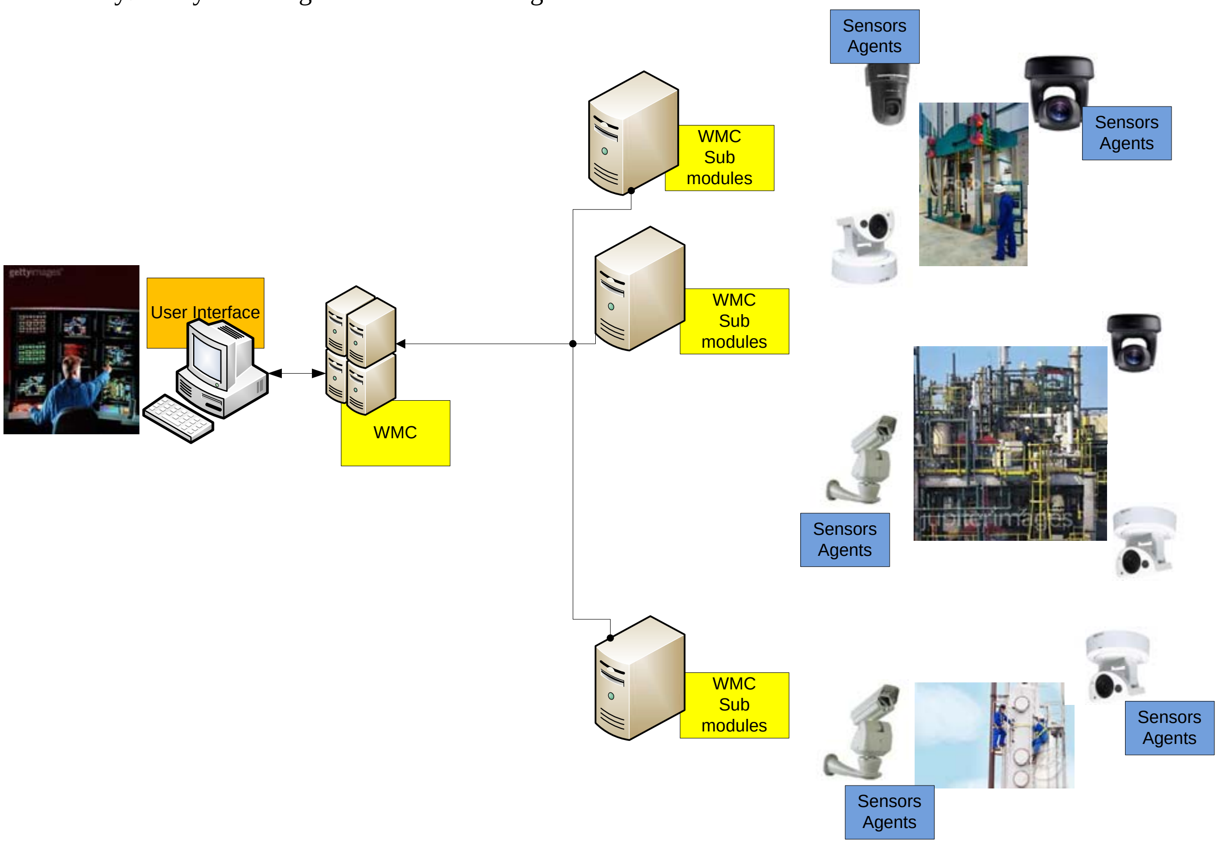

Fig. 7. Applying the whole system software architecture and hardware in an industrial plant.References

[1] P. Humphreys, G. Huang, T. Cadden, "A web-based supplier evaluation tool for the product development process", Industrial Management & Data Systems, Vol. 105 No. 2, 2005, pp. 147-63.

[2] T. Gulledge, "What is integration?", Industrial Management & Data Systems, Vol. 106, No. 1, 2006, pp. 5-20.

[3] P. Buhler, J.M. Vidal, "Towards Adaptive Workflow Enactment Using Multiagent Systems", Journal of Information Technology and Management, Volume 6, Issue 1, January 2005, Kluwer Academic Publishers Hingham, MA, USA.

[4] H. Parker, "Interfirm collaboration and the new product development process", Industrial Management and Data Systems, Vol. 100 No. 6, 2000, pp. 255-60.

[5] S. Shanmuganathan, P. Sallis, L. Pavesi, M.C.J. Muñoz, "Computational intelligence and geo-informatics in viticulture", IEEE Second Asia International Conference on Modeling & Simulation, May 2008, AICMS 08, pp. 480-485.

[6] S. Mahlknecht, S.A. Madani, "On Architecture of Low Power Wireless Sensor Networks for Container Tracking and Monitoring Applications", 5th IEEE International Conference on Industrial Informatics, 2007, pp. 353- 358, Vienna.

[7] V.C. Gungor, G.P. Hancke, "Industrial Wireless Sensor Networks: Challenges, Design Principles, and Technical Approaches", IEEE Transactions on Industrial Electronics, Vol. 56, No. 10, October 2009, pp. 4258-4265.

[8] WfMC (1999), "Workflow management coalition terminology & glossary", document numberWfMC-tc-1011, issue 3.0.

[9] JADE Development Framework, http://jade.tilab.com/

[10] Mobile C platform, http://www.mobilec.org/

[11] WfMC (1999), "Workflow management coalition terminology & glossary", document number WfMC-tc-1011, issue 3.0.

[12] Hollingsworth, D. (1995), "The workflow reference model", Workflow Management Coalition.

[13] W3C (2005), "Resource description framework (RDF)", available at: www.w3.org/RDF/

[14] FIPA, http://www.fipa.org

[15] Desai, P., Rattan, K.S., "System Level Approach for Surveillance Using Wireless Sensor Networks and PTZ Camera". IEEE National Aerospace and Electronics Conference, 2008. NAECON 2008.

[16] Micheloni, C., Rinner, B., Foresti, G.L., "Video Analysis in Pan-Tilt-Zoom Camera Networks", IEEE Signal Processing Magazine, Volume: 27 , Issue: 5, Publication Year: 2010 , Page(s): 78 – 90

[17] Buhler, P., Vidal, J. N., "Adaptive Workflow = Web Services + Agents", Proceedings of the International Conference on Web Services, Las Vegas, NV, 131-17, CSREA Press, 2003.

[18] Maximilien, E. M., Singh, M. P., "Agent-based Architecture for Autonomic Web Service Selection", In Proc. of the 1st International Workshop on Web Services and Agent Based Engineering, Sydney, Australia, July 2003.

[19] Laukkanen, M., Helin, H., "Composing Workflows of Semantic Web Services", In Proc. of the 1st International Workshop on Web Services and Agent Based Engineering, Sydney, Australia, July 2003.

[20] Lander, S. E. "Issues in Multiagent Design Systems", IEEE Expert 12(2), 1997, 18-26.

[21] Sardis, E.S., "Applying Multi-Agents Technologies in Industrial Plants", 16th International Conference on Systems, Signals and Image Processing, 2009. IWSSIP 2009. Publication Year: 2009, Page(s): 1–4.

Dr. Emmanouel Sardis received the Bachelor degree in Electrical and Computer Engineering from the National Technical University of Athens in 1996. He received his Ph.D. degree from the same University in 2001 in the area of E-Commerce and Security. He is currently wor [1] king as Senior Research Engineer in National Technical University of Athens-NTUA with the Distributed, Knowledge and Media Systems Laboratory, and visiting assistant professor in Technical University of Crete at the department of Production Engineering and Management. Dr. Sardis is leading and coordinating numerous National and European projects in the areas of Internet Technologies, Embedded Systems, and Distributed architectures with the team of Distributed, Knowledge and Media Systems Laboratory of NTUA. Dr. Sardis has been also involved in numerous National and European Union projects, in the private sector as an employee of (Intrasoft International, European Dynamics, Dibomar) with more than 10 years working experience in sectors of computer and software engineering and in maritime information technology systems. His research interests are in the field of Internet technologies, artificial intelligence, multi-agent systems and architectures, e-commerce, media processing techniques and data fusion.

Prof. Anastasios D. Doulamis received the Diploma degree in Electrical and Computer Engineering from the National Technical University of Athens (NTUA) in 1995 with the highest honor. In 2000, he has received the PhD degree in electrical and computer engineering from the NTUA. From 1996-2000, he was with the Image, Video and Multimedia Lab of the NTUA as research assistant. From 2001 to 2002, he serves his mandatory duty in the Greek army in the computer center department of the Hellenic Air Force, while in 2002, he join the NTUA as senior researcher. His PhD thesis was supported by the Bodosakis Foundation Scholarship. In 2006, he is Assistant professor in the Technical University of Crete in the area of multimedia systems. Dr. Doulamis has received several awards and prizes during his studies, including the Best Greek Student in the field of engineering in national level in 1995, the Best Graduate Thesis Award in the area of electrical engineering with A. Doulamis in 1996 and several prizes from the National Technical University of Athens, the National Scholarship Foundation and the Technical Chamber of Greece. In 1997, he was given the NTUA Medal as Best Young Engineer. In 2000, he received the best Phd thesis award by the Thomaidion Foundation in conjunction with N. Doulamis. .In 2001, he served as technical program chairman of the VLBV'01. He has also served as program committee in several international conferences and workshops. He is reviewer of IEEE journals and conferences as well as and other leading international journals. He is author of more than 200 papers in the above areas, in leading international journals and conferences. His research interests include, non-linear analysis, neural networks, multimedia content description, intelligent techniques for video processing.

Prof. Nikolaos Matsatsinis received his B.A. in Physics from Aristotle University of Thessaloniki (1980) and his PhD in Intelligence Decision Support Systems from Technical University of Crete (1995).

He also has over twenty eight years of experience in Information and Decision Systems Development. He is a full Professor of Information and Intelligent Decision Support Systems at the Department of Production Engineering and Management of the Technical University of Crete, Greece. He is Chairman of the Department, Director of Decision Support Systems Lab and President of the Hellenic Operational Research Society (HELORS). He has contributed as scientific or project coordinator on over of forty scientific projects. Professor Matsatsinis is member of the international advisory board of three scientific journals, belongs to several scientific groups and has a rich teaching experience. He is the author or co-author of fifteen books and over of sixty eight articles in international scientific journals and books. His research interests fall into the areas of Decision Support Systems, Artificial Intelligent and Multi-Agent Systems, e-Business, e-Marketing, Multicriteria Decision Analysis, Group Decision Support Systems, Workflow Management Systems.

Prof. Matsatsinis is member of the Institute of Electrical and Electronics Engineers (IEEE); member of the American Association for Artificial Intelligence (AAAI); Member of IFIP, CEPIS; International Consortium for Electronic Business; member of INFORMS, EURO, Euro Working Group on Multicriteria Decision Aid; Euro Working Group on Decision Support Systems; European Federation for Information Technology in Agriculture, Food and the Environment – EFITA.