Using Multi-Agent Systems for Intelligent Plant Maintenance Functionality

Martin Naedele, Patrick Sager, Christian Frei
ABB Corporate Research, CH-5405 Baden-Dättwil, Switzerland
martin.naedele@ch.abb.com

Abstract— **This paper motivates the use of the intelligent SW agent and multi-agent system (MAS) paradigms for asset management applications in manufacturing plants. A proposed layered reference model that has been used with success previously to compare various research approaches and systems in the area of MAS for plant asset management is evaluated with respect to its fit with the recent ISO 13374 standard prescribing a high-level system architecture for condition monitoring systems. It is shown that both reference models are compatible and complement each other well. An example is given for representing ISO 13374 compliant messages in an inter-agent dialog using the FIPA agent communication language.**

I. INTRODUCTION

With today's pressure on cost reduction and quality improvement, management and maintenance of a plant's equipment is a field of growing economic importance for manufacturing enterprizes. In our definition, plant asset management (PAM) is the set of functions and processes needed for the optimal operation of the technical elements of a plant. This comprises both asset condition monitoring for predictive and preventive maintenance, and fault diagnosis for corrective maintenance. PAM is concerned with answering the following important questions with respect to process equipment:

- Has anything failed? (reactive/corrective maintenance)
- What will fail when? (predictive maintenance)
- What failure caused the alarm? (alarm management)
- When should it be repaired/maintained? (maintenance scheduling)

Efforts are ongoing, e.g., by MIMOSA (http://www.mimosa.org), to provide more powerful plant asset management systems by standardizing the interfaces and information exchange semantics between the various subsystems involved, such as data acquisition, diagnosis, and response, especially if the IT systems for these functionalities are provided by different vendors. These activities are intended to result in a series of standards ISO 13374 of which Part 1 "General guidelines" [1] has become an effective standard in spring 2003, while Parts 2 to 4 are still in various stages of development.

ISO 13374 addresses the improvement of plant asset management systems via data exchange and creates the basis for multi-vendor composable systems by defining a common architecture of functional blocks. It does however not address how these blocks are to be best implemented.

II. INTELLIGENT AGENTS FOR PLANT ASSET MANAGEMENT

Plant asset management as defined above is a domain where not just deterministic reactive behavior is required, but where computational "intelligence" can be used profitably to aggregate information about complex situations for human deciders and to optimize available reaction options. Consequently, one possible implementation paradigm for plant asset management systems are intelligent software agents [2]. Academic work in this area has been going on for more than a decade, in projects like ARCHON [3], DIAMOND [4], and MAGIC [5].

Like with the introduction of other new paradigms in SW architecture previously, such as object-orientation, component-based software engineering (CBSE), or recently service oriented architecture (SOA), it is hardly possible to clearly and irrefutably prove superiority of one approach over the other. Thus, the decision whether a certain approach should be used for a problem depends more or less on a subjective weighting of the characteristics of the problem and those of one particular design approach. For the PAM application area, plausible arguments can be made that the multi-agent paradigm is a good fit to the problem domain.

The plant is a distributed system with a large number of different devices, where errors and breakdowns can occur. For each device type, there may be multiple instances in the same plant. Often, the diagnosis of the root cause of a failure is only possible by evaluating the output of various diagnosis algorithms (such as simple rules, expert systems, model-based analysis, neural networks, fuzzy logic, or logic deduction). Fault location and root-cause analysis are obvious candidates for intelligent, non-deterministic, peer-to-peer communication between the individual diagnosis reasoning elements. Often, there is the situation that multiple diagnosis strategies (physical principles) or algorithms (model-based, rule-based) are available for a single asset or plant part that overlap partially or fully in diagnostic coverage, and it is not possible a priori to determine which of these will be the fastest or most accurate for a specific incident and set of symptoms. Perhaps even all of them can work better if they exchange during execution partial results to reduce the hypothesis space they have to search/process (see Fig. 1).

Maintenance management comprises scenarios where different actors have legitimately conflicting goals (see Fig. 2): the maintenance planner application wants to ensure that a

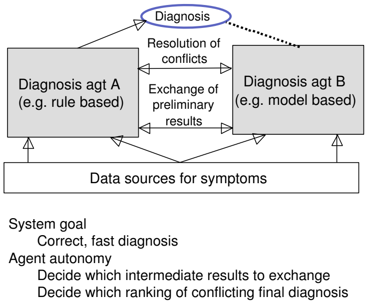

Fig. 1. Cooperation of diagnosis agents with overlapping capabilities

degraded asset is repaired before it breaks and at minimal cost. For this, it has to take into account remaining lifetime as well as availability and cost of spare parts and personnel. The production scheduler has the objectives of archiving high utilization of plant assets and of fulfilling production deadlines promised to customers. Agent-based collaboration using negotiation and auctioning techniques is a suitable approach to automatically handle such conflicting scenarios in a computerized system while achieving some kind of optimum (acceptable compromise) for the system as a whole.

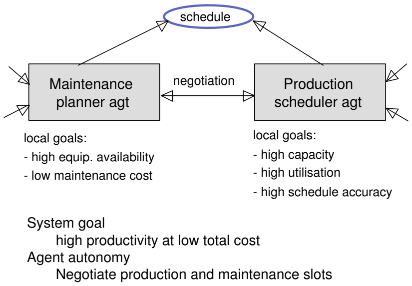

Fig. 2. Conflicting goals in a plant asset management scenario

From an architectural point of view, a diagnosis system has to cope with the fact that the total system configuration and topology change over time, that devices are made by different vendors, that new diagnosis algorithms will become available over time and have to be added to the system without interruption of the fault diagnosis capability or even the production system.

Also, the fault diagnosis system has to be, by the very nature of its task, resilient to failures of both its own elements and the underlying communication/networking structure.

Any newly proposed PAM system or development approach, should take standardization, in this case especially ISO 13374, into account. In [6], we have proposed a multi-layered reference model for PAM applications built as multi-agent systems. In this paper we will show how that layered reference model corresponds to the requirements of ISO 13374.

III. MULTI-AGENTS SYSTEMS

Many definitions and taxonomies of intelligent software agents have been proposed [7]. We see two main characteristics that differentiate agents from other application structuring concepts, and we believe at least one of them needs to be present so that a program can be called ”agent”.

Autonomy: The agent decides on its own, when and on what it (re)acts. ”When” in most cases implies that agents execute in their own thread of control and can act without direct intervention from others (humans or other software processes). ”On what” refers to the fact that a communication with an agent is not a method invocation, but only a request to the agent, which it can follow or ignore depending on its own goals and state.

Goal-orientation: The agent has one or multiple, potentially changing, goals which it tries pro-actively to fulfill/optimize by its actions.

IV. ISO 13374 ARCHITECTURE

The ISO 13374 condition monitoring architecture consist of 8 functional blocks:

The first three blocks are assumed to be specific to the technology of the monitored asset:

- **Data acquisition** Produces digital records with values and descriptive meta-information like time stamps and quality for the sensor/transducer output.
- **Data manipulation** Performs any kind of processing (e.g., signal processing) necessary in order for the sensor values to be meaningful with respect to state and health assessment.
- **State detection** Detects whether the asset is in a normal or abnormal state.

The next three blocks implement the diagnosis and remediation intelligence of the system:

- **Health assessment** Rates the current state as asset health and diagnoses any faults.
- **Prognostic assessment** Predicts the remaining lifetime until the next significant state change.
- **Advisory generation** Provides the user with recommendations on maintenance actions or changes of operational settings to optimize asset performance in the given state.

The last two blocks both extend in parallel to the hierarchical stack formed by the asset management functionality in the first six blocks and provide interfaces to the outside (humans and systems) for all the blocks.

- **Information presentation** Presents the output of any of the previous stages to the user in easily understandable, textual, and graphical form.
- **External systems** Any external system the asset management application is connected to. Specifically mentioned in the standard are archiving, configuration/engineering and history, but connections to computerized maintenance management systems (CMMS), or enterprise resource planning (ERP) for spare parts management, as well as manufacturing execution systems (MES) for replanning production schedules are also imaginable.

V. REFERENCE MODEL

In [6] a layered reference model for multi-agent systems for plant asset management has been proposed in extension of [8], [5], [9], [10] to allow some comparability between the various research projects and proposed MAS architectures for plant asset management. This layered model suggests a certain information flow and information condensation process from bottom to top, but it does not intend to be restricting with respect to possible communication relations between agents. One single agent may very well extend over and provide the functionality of multiple layers.

The reference model is structured into the following major layers:

Data acquisition Provides preprocessed sensor data, where sensor data can be measurements, or self-test results. The latter are either provided by self-test functionality of the device, or by an external agent that wraps stimulus/response testing and provides the result as a self-test state to its communication partners.

Diagnosis Detects whether there are problems with one or multiple assets in the plant, locates the problems, and predicts the behavioral future of device and plant. The various agents in this layer may be spatially and/or semantically distributed [11]. Spatial distribution here means that different agent instances are responsible for different device instances or plant parts, whereas semantic distribution refers to an architecture where different agent instances may concurrently apply different diagnosis algorithms to the same device or plant part.

Correction After diagnosing and locating a problem, corrective actions need to be initiated. These actions can either be initiated by a human operator based on context-specific recommendations prepared by the plant asset management system, or even fully automatically without human intervention.

Meta-control The rationale for using the multi-agent paradigm is to give autonomy to the individual agents to let them act and collaborate (cooperate and compete), according to flexible optimization goals, and not fixed rules. This, however, requires that for optimal performance of the system as a whole, the behavior of the multi-agent system has to be monitored, and that the objectives and other decision parameters may need to be adjusted during runtime. This can be done by an operator or automatically. In this sense, the meta-control layer reflects the correction layer, but with respect to the plant asset management MAS, not the manufacturing process.

Human machine interface Provides clear, understandable, and condensed information about the plant asset state and the asset management system state to the users of different groups, such as operators, operations engineers, reliability analysts, and managers [1]. This layer is separated out, as on the one hand in a fully automated system no HMI may be present, and on the other hand the presentation to a human requires additional and different types of intelligent behavior than the diagnostic and corrective reasoning.

Table I summarizes this reference model and contrasts it with the structuring prescribed by ISO 13374. Note that ISO 13374 ”external systems” are not mapped, as they are supposed to be systems outside the scope considered here. Of course, any of theses external systems can also be agent-based or have an agent interface. From the table it can be clearly seen that a system developed in the proposed multi-agent reference model is compliant to the standard, but potentially would go in functionality beyond what is defined there, e.g., for MAS specific functionality such as conflict resolution or meta-control.

ISO 13374 also discusses - without being normative - various communication options between the functional blocks, such as file exchange, MMS [12], CORBA, and XML. The MIMOSA Common Relational Information Schema (CRIS) version 1 and 2, and the corresponding XML-representation is specifically mentioned to be standard compliant. With regard to the use of multi-agent systems for plant asset management we suggest that an agent communication language (ACL), such as the ACL proposed by FIPA [13], should be added to the list of communication options in the next revision of the standard. The above mentioned MIMOSA XML representation for data could then be used (as part of) the domain specific content language (see Section VI) of the ACL.

The standard also makes concrete suggestions on how the user interface should be structured. User interface agents can easily comply with these suggestions.

VI. EXAMPLE

The following example describes how a communication between a health assessment block for a single device (”equipment module” [14]) and a prognostic assessment block for a plant subsystem (”unit” [14]) according to ISO 13374 can be mapped to a FIPA ACL [15] communication between a single device diagnosis agent and a multi device diagnosis agent according to the MAS reference model.

The standard ISO 13374-3, which covers the communication between different condition monitoring and diagnostic blocks, is not yet available. However, since MIMOSA [16] is the driver behind ISO 13374 and MIMOSA suggests using Tech-XML [17] for CRIS [18] related information exchange, we assume here that the communication format of ISO 13374-3 will be based on or similar to Tech-XML. The Tech-XML specification contains XML schemas for different application areas, such as DIAG (diagnostics, health) and TREND (trends, alarms). These schemas can be considered as some kind of (sub)domain ontologies. The messages of the example are based on the schemas of the DIAG package.

In the example scenario, the prognostic assessment block ”027” in site ”003” wants to predict the future health of valve ”042” in site ”004”. Therefore it requests the health status of valve ”042” from the responsible health assessment block ”007” by sending the following Tech-XML message:

```
<mim_v2_2_6002
  xmlns=
    "http://www.mimosa.org/TechXMLV2-2"
  xmlns:xsi=
    "http://www.w3.org/2001/
      XMLSchema-instance"
```

MAS reference model		ISO 13374
Primary layers	Secondary layers	
HMI	Notification and display	Information presentation
Meta-control	MAS performance tuning	(Advisory generation)
Correction	Automatically executing corrective measures	-
	Decision support / proposing corrective measures	Advisory generation
Diagnosis	Diagnosis conflict resolution	-
	Multi device input diagnosis	Prognostic assessment
		Health Assessment
	Simulation for verification or forecast	-
	Single device input diagnosis	Prognostic assessment
		Health Assessment
Data acquisition	Device self-test	State detection
	Processing values	Data manipulation
	Measuring raw values	Data acquisition

TABLE I

Correspondence between the proposed multi-agent reference model and the ISO 13374 reference architecture.

```
  xsi:schemaLocation=
    "http://www.mimosa.org/TechXMLV2-2
      V2-2-6002-01QueryAsHealth.xsd">
  <query_as_health_req>
    <header session_id="003_027_367"/>
    <param asset_org_site="004"
           asset_id="042"/>
  </query_as_health_req>
</mim_v2_2_6002>
```

Every communication is uniquely identified by a session ID. In the example the session ID is composed of the site ID ”003”, the ID of the prognostic assessment block ”027”, and a sequence number (”367”).

The health assessment block ”007” of site ”004” receives the messages and determines the values for rod positioning range of the valve (code ”13”) and the flow through the valve in open position (code ”14”) at the current time (2003-11-15 13:15). The values are assessed by the block and normalized to a scale from 0 (completely broken) to 10 (perfect health). The health assessment block then sends the results of the assessment back to the prognostic assessment block:

```
<mim_v2_2_6002
  xmlns=
    "http://www.mimosa.org/TechXMLV2-2"
  xmlns:xsi=
    "http://www.w3.org/2001/
      XMLSchema-instance"
  xsi:schemaLocation=
    "http://www.mimosa.org/TechXMLV2-2
      V2-2-6002-01QueryAsHealth.xsd">
  <query_as_health_ack>
    <header session_id="003_027_367"/>
    <status success="true"/>
    <row>
      <asset_health
        asset_org_site="004"
        asset_id="042"
        gmt_assessment=
          "2003-11-15T13:15:00"
        by_agent_site="004"
        by_agent_id="007"
        hgrade_type_code="13"
        hgrade_float="9.1"/>
    </row>
    <row>
      <asset_health
        asset_org_site="004"
        asset_id="042"
        gmt_assessment=
          "2003-11-15T13:15:00"
        by_agent_site="004"
        by_agent_id="007"
        hgrade_type_code="14"
        hgrade_float="4.9"/>
    </row>
  </query_as_health_ack>
</mim_v2_2_6002>
```

Note that the session ID of the response is identical to the session ID of the request and that the mandatory parameter ”status” indicates that the query has been processed successfully and that the message contains its results.

In the MAS reference model, a health assessment block corresponds to an input diagnosis agent. A prognostic assessment block is realized as part of an agent on the multi device diagnosis layer. In the example scenario, the health assessment block ”007” of site ”004” is mapped to the input diagnosis agent ”_004_007” and the prognostic assessment block ”027” of site ”003” is mapped to the multi device diagnosis agent ”_003_027”.

The agents are talking to each other using FIPA Query [19] as interaction protocol, FIPA ACL [15] as agent communication language, and MIMOSA Tech XML as content language. The following FIPA ACL query corresponds to the request for the health of asset ”042” in site ”004”:

```
(query-ref
  :sender   _003_027
  :receiver _004_007
  :content (
    <mim_v2_2_6002 ... >
      <query_as_health_req>
        <header
          session_id="003_027_367"/>
        <param asset_org_site="004"
               asset_id="042"/>
      </query_as_health_req>
    </mim_v2_2_6002> )
  :language MIMOSA-TECH-XML
  :ontology DIAG
  :protocol fipa-query
  :conversation-id _003_027_367 )
```

The content field of the request contains the same message as exchanged between the two ISO 13374 blocks. The conversation ID is identical to the session ID. In the example, the XML message headers (namespace and schema reference) are omitted for clarity. As value for the language parameter we suggest to use ”MIMOSA-TECH-XML” (or perhaps ”ISO-13374”, once the corresponding part of this standard exists). On one hand, this value is not a reserved value by FIPA, on the other hand, the FIPA specification [20] containing the reserved values for content languages is marked deprecated, so we assumed that we could take some liberties here. The ontology parameter specifies, which Tech XML package was used.

After assessing the health of asset ”042” the single device diagnosis agent ”_004_007” sends the following FIPA-ACL inform message back to the multi device diagnosis agent ”_003_027”:

```
(inform
  :sender   _004_007
  :receiver _003_027
  :content (
    <mim_v2_2_6002 ... >
      <query_as_health_ack>
        <header
          session_id="003_027_367"/>
        <status success="true"/>
        <row>
          <asset_health
            asset_org_site="004"
            asset_id="042"
            gmt_assessment=
              "2003-11-15T13:15:00"
            by_agent_site="004"
            by_agent_id="007"
            hgrade_type_code="13"
            hgrade_float="9.1"/>
        </row>
        <row>
          <asset_health
            asset_org_site="004"
            asset_id="042"
            gmt_assessment=
              "2003-11-15T13:15:00"
            by_agent_site="004"
            by_agent_id="007"
            hgrade_type_code="14"
            hgrade_float="4.9"/>
        </row>
      </query_as_health_ack>
    </mim_v2_2_6002> )
    :language MIMOSA-TECH-XML
    :ontology DIAG
    :protocol fipa-query
    :conversation-id _003_027_367 )
```

The two FIPA messages above are expressed in FIPA ACL string representation [21]. However, they could also be similarly expressed using FIPA ACL XML representation [22].

The example has shown how blocks of ISO 13374 are mapped to agents of the MAS reference model and how Tech-XML communication between ISO 13374 blocks is expressed in FIPA-ACL.

VII. CONCLUSION

This paper has motivated the use of the multi-agent system (MAS) paradigm for asset management applications in manufacturing plants. Examples have been given for situations where collaboration of intelligent SW agents may give better results compared to fully deterministic systems.

A layered reference model which was previously used to compare various research approaches and systems in the area of MAS for plant asset management has been mapped to the recent ISO 13374 standard prescribing a high-level system architecture for condition monitoring systems. It has been shown that both reference models are compatible and complement each other well, and it has been suggested to add an agent communication language to the (not normative) list of communication mechanisms between the different function blocks in the standard.

REFERENCES

[1] ISO TC108 SC5, “Condition monitoring and diagnostics of machines - data processing, communication, and presentation,” Standard ISO 13374-1:2003, March 2003.

[2] M. Wooldridge, *An Introduction to Multi-Agent Systems*. Wiley, 2002.

[3] D. Cockburn and N. Jennings, *Foundations of Distributed Artificial Intelligence*. Wiley, 1996, ch. ARCHON: A Distributed Artificial Intelligence System For Industrial Applications.

[4] M. Albert, “DIAMOND Reference Model,” Karlsruhe University, Esprit project 28735, Tech. Rep., 2001.

[5] H. Wörn, T. Längle, and M. Albert, “Multi-agent architecture for monitoring and diagnosing complex systems,” in *Proceedings of the 4rd International Workshop on Computer Science and Information technologies*, Sept 2002.

[6] M. Naedele, C. Frei, and P. Sager, “Multi-agent systems for plant automation, version 3.0,” ABB Corporate Research,” Technical Report, 2003.

[7] S. Franklin and A. Graesser, “Is it an Agent, or just a Program? A Taxonomy for Autonomous Agents,” in *Proceedings of the Third International Workshop on Agent Theories, Architectures, and Languages, 1996, published as Intelligent Agents III*. Springer-Verlag, 1997.

[8] W. Nejdl and M. Werner, “Distributed Intelligent Agents for Control, Diagnosis and Repair,” RWTH Aachen, Informatik, Tech. Rep., 1994.

[9] M. Albert, T. Längle, and H. Wörn, “Development tool for distributed monitoring and diagnosis system,” in *Proceedings of Thirteenth International Workshop on Principles of Diagnosis*, 2002.

[10] D. H. Zhang, J. B. Zhang, M. Luo, Y. Tang, and L. Q. Zhuang, “A reference architecture and functional model for monitoring and diagnosis of large automated systems,” in *Int. Conf. on Emerging Technologies for Factory Automation*, 2003.

[11] P. Fröhlich and W. Nejdl, “Resolving conflicts in distributed diagnosis,” in *12th European Conf. on Artificial Intelligence*, 1996.

[12] International Standardization Organisation / International Electrotechnical Commission, “ISO/IEC 9506, Industrial automation systems - Manufacturing Message Specification,” Tech. Rep., 1990.

[13] “FIPA - Foundation for Intelligent Physical Agents,” http://fipa.org/.

[14] ANSI/ISA, “S88 Batch Control Standard, Part 1: Models and Terminology,” ANSI/ISA-S88.01-1995, Oct 1995.

[15] “FIPA ACL Message Structure Specification,” http://www.fipa.org/specs/fipa00061/.

[16] “MIMOSA - Machinery Information Management Open Systems Alliance,” http://www.mimosa.org/.

[17] “Open System Architecture for Enterprise Application Integration,” http://www.mimosa.org/osa-eai/techinfo.htm.

[18] “Common Relational Information Schema - CRIS Version 2.2,” http://www.mimosa.org/osa-eai/documents/CRISn-2003].pdf.

[19] “FIPA Query Interaction Protocol Specification,” http://www.fipa.org/specs/fipa00027/.

[20] “FIPA Content Languages Specification,” http://www.fipa.org/specs/fipa00007/.

[21] “FIPA ACL Message Representation in String Specification,” http://www.fipa.org/specs/fipa00070/.

[22] “FIPA ACL Message Representation in XML Specification,” http://www.fipa.org/specs/fipa00071/.