Virtual Enterprise Process Monitoring: An Approach towards Predictive Industrial Maintenance

Filipe Ferreira, Ahm Shamsuzzoha, Americo Azevedo, and Petri Helo

1   Introduction

Global business dynamics and volatile market environment pushes companies towards collaboration that can ensure business competitiveness with enhanced sustainability. Manufacturing companies, especially small and medium enterprises (SMEs), where there are often resources scarcity are benefited through such business collaboration. There are various forms of business collaboration exists within the manufacturing domain such as business community, industrial cluster, collaborative network organization (CNO) (Camarinha-Matos et al., 2008), virtual organization (VO) (Carneiro et al., 2010), virtual enterprise (VE) Molina et al., 2007), etc. In such business environment, there needs to continuous monitoring and management of the business processes and associated equipment. The equipment as integrated with the collaborative SMEs processes are need to be real time monitoring in order to enhance its predictive maintenance.

With only a few SMEs having capacity to implement innovative manufacturing technologies, the system must be highly adaptable to any equipment with low cost/investment and without the need of doing hard integrations and setups. Using the approach described in this paper, industrial managers have access to the list of machines installed by their equipment suppliers, as well as their status, location, manuals, technical assistance plans, procedures and drawings. The vendor is then notified in real-time and knows exactly which is the equipment that needs service, having access to the machine status and software, all the machine history, manuals, drawings and service records instantaneously. It's a huge advantage in terms of time-to-market.

Moreover, if the equipment is connected to Internet and the problem is about software, vendor can try to solve it remotely and have real-time feedback from the equipment. If the problem can be solved by doing a couple of procedures, depending on each machine, it can be done remotely by client's technicians and monitored by the vendor too. For customers far from vendor's plant, it's common to subcontract 3rd party firms (partners) to assist their clients. One of the advantages of this new approach is that the vendor becomes able to remotely monitor the process by using equipment's control software integration and the mobile applications where the partner updates the service status. The essential issue tackled in this paper and underlying research study refers to the predicative maintenance and the respective maintenance operations management.

2   Literature review

2.1   Predictive Maintenance

In industrial domain two major maintenance management approaches are available namely; failure-driven and time-based maintenance (Moubray, 1997). There are also other maintenance systems such as conditioned-based maintenance (CBM), statistical-based maintenance (SBM), etc., are used to reduce the uncertainty of maintenance according to the needs indicated by any industrial equipment condition (Yam et al., 2001).

Predictive Maintenance (PM) is used as a maintenance methodology to monitor and detect incipient problems and to prevent catastrophic failure. PM solution opens up innovative new possibilities for companies. It does not depend on industry statistics but relies on real signals demonstrated by a single and specific piece of equipment. Any data or signal from specific sensors monitoring machine condition is

F. Ferreira (✉) • A. Azevedo (✉)
Manufacturing Systems Engineering Unit, INESC TEC (formerly INESC Porto), Porto, Portugal
e-mail: filipe.ferreira@inescport.pt; ala@fe.up.pt

A. Shamsuzzoha (✉) • P. Helo (✉)
Department of Production, University of Vaasa, 700 Wolffintie 34, 65200 Vaasa, Finland
e-mail: ahsh@uva.fi; petri.helo@uva.fi

automatically reviewed to pick up any patterns that indicate a possible fault. In addition to early fault detection PM also can be used to avoid unplanned downtimes and both staff and resources can be employed more effectively (Zhou et al., 2007).

One area that many times is overlooked is how to, in an efficient way, transfer the predictive maintenance data to a computerized maintenance management system (CMMS) system so that the equipment condition data is sent to the right equipment object in the CMMS system in order to trigger maintenance planning, execution and reporting. Unless this is achieved, the predictive maintenance solution is of limited value, at least if the predictive maintenance solution is implemented on a medium to large size plant with tens of thousands pieces of equipment.

2.2 OPC (OLE for Process Control)

Object Linking and Embedding (OLE) is a communication standard based on OLE technology from Microsoft that enables interoperability between industry automation and control systems (PLCs, sensors, motor controllers, etc.), field systems (e.g. Manufacturing Execution Systems) and business applications (e.g. ERP).

Object Linking and Embedding (OLE) for Process Control (OPC), is a standard specification developed by a work group of industrial automation in 1996. The standard defines objects, methods and properties to meet the needs of interoperability for process automation applications from different manufacturers in real time. These requirements include a standard technique for accessing field data from plant floor devices, the efficient transfer of data from one device to a business process application and the ability of a client to use multiple servers simultaneously (integration of data from different manufacturer's hardware devices).

Traditionally, every time data access from a device is needed, a custom interface has to be created. The goal of OPC is to define a common interface that is written once and then reused by all business packages, SCADA, HMI, or custom software. There is nothing in the specification to restrict the OPC server to provide access to a device control process. Once an OPC server is written for a specific device, it can be reused by any application that is able to act as an OPC client.

3 Research Methodology

This research is carried out in five steps. First of all, we reviewed the existing literature in areas such as Predictive Maintenance, OLE for Process Control and virtual enterprise for collaborative business. Secondly, we specified a hardware

integration tool to apply to the field equipment, which is then integrated with a Virtual Enterprise Management Platform. Thirdly, a requirement elicitation process is carried out, which includes semi-structured interviews to two different business enterprises, namely; a machinery manufacturing SME located in the north of Portugal (Engineer-to-Order business model) and an electronic and automation SME located in the United Kingdom (Engineer-to-Order business model).
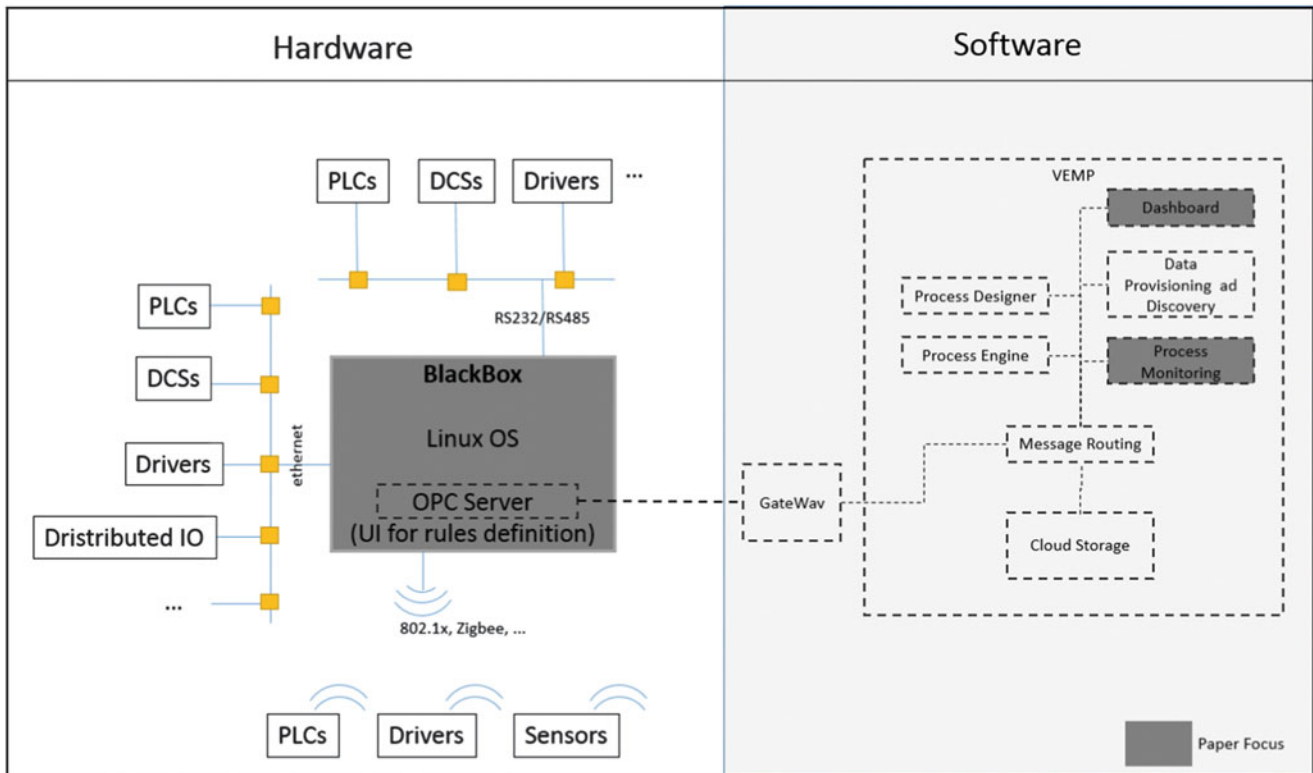
The results of these three steps conclude the requirements elicitation process through collecting the expected requirements list, which answers our first research question and functional specification, answering second research question. Fourthly, a set of discussions were carried out with the platform development team to understand the hardware integration and technical requirements. This fourth step answers to the third and last research questions. Finally, a Virtual Enterprise Management Platform is introduced on the shop floor integration hardware in order to get data from the field systems and controllers.

4 Virtual Enterprise Management Platform

The goal of a Virtual Enterprise Management Platform (VEMP) is to simplify the establishment, management, adaptation and monitoring of dynamic manufacturing processes in Virtual Factories. This includes the finding of partners, the design, forecasting and simulation of Smart Processes, and their execution and real-time monitoring.

To establish processes between different companies, data about the partners wishing to collaborate in a virtual factory is needed. Therefore, each Virtual Factory member needs to be able to add data about his company, products, services and processes. To achieve this in a user-friendly way, VEMP has to provide an editor in the scope a Data Provisioning and Discovery component to enter, view, update or delete this data. For reasons of availability, accessibility, access-control and the possibility to have redundant backups if needed, this data should be stored in the cloud. The Cloud Storage component should support several types of data storage like semi-structured data storage (for example for XML or JSON data) used internally by the VEMP, as well as semantic data necessary for semantic company descriptions and also data storage for binary files. Binary files may be used for storing documents such as specifications or even multimedia files.

To design the VEMP, the platform has to provide a Process Designer. To improve and facilitate the usability of all user interfaces should accessible via a single application interface with a single look and feel and a quick learning curve. All the user interfaces should therefore be embedded in the Dashboard, including the process designer.

Fig. 1 Black box integration with the VE Management Platform (VEMP)

The Process Workflow Execution component, executes process models and the Real-time Process Monitoring shows the actual status of the process execution and can additionally query machines interfaces for their current state, collecting information for preventive and predictive analysis.

The Machine sensor interfaces as displayed in Figure 1 will be integrated via the Gateway, with acts as an OPC Client for the Black Box OPC server, the gateway will then communicate with the platform via the Message Routing. External systems like legacy systems etc. should also be able to communicate with the VEMP's Message Routing component making use of the Gateway component, which effectively fulfils the role of a bridge, connecting the external system with the platform. The Gateway as well as the Message Routing may invoke Transformation Services that can be used to translate between external (legacy) technology which will use different messaging protocols (including OPC), interfaces and message formats. The Gateways therefore will be the only components that might need to be expanded or recreated when a new Member wants to connect uncovered legacy systems to the Message Routing. The Message Transformation may be used as a base by the Gateways to transform a variety of data formats, hence allowing a wide support of systems.

4.1 Integration black box at shop floor

An integration black box is designed and developed to interface with the virtual enterprise's equipment to be monitored and managed. As displayed in Figure 1, the black box Runs an OPC Server on top of Linux Operating System, enabling the integration with the overall virtual enterprise platform via a Gateway implemented Services. The black box itself works as a smart object with sensors collects monitored raw data to be used for predictive maintenance from programmable logic controllers (PLCs), Industrial PCs, DCSs, Sensors from different vendors using different communication protocols and physical layers.

The black also counts with several interfaces such as Ethernet, 802.1x, RS482, RS232, USB as well as direct digital and analog IO which is helpful for old equipment's, where there's no communication at all and all is based on relays and discrete controllers (temperature, pressure, flow, timers, counters, encoders, etc...). Moreover, it counts with an intuitive User interface allowing the definition of alarm trigger rules. Different monitoring events are exposed through the gateway to the workflow engine with retrieves and update the information to the process monitoring.

4.2   Gateways

A gateway will comprise of standard components and custom components with functionality developed or created for connecting to a specific external system type and/or instance. A gateways mission is to communicate with a specific system, meaning that a significant part of a gateway implementation is tailored for specific (e.g. SAP ERP) technology or communication/interface protocol.

Gateways are only about connecting 3rd party systems, having an agnostic view about what is the content exchanged and what the format is. Checking and processing of the information gathered through the gateways will be performed at the destination components (as per example the Process Execution component).

A gateway is based on the following subcomponents:

- Communication: This sub-component is the bridge to the Message Routing component and will send and receive the messages on behalf of the Gateway. It will use a common interface to communicate with the Custom Connector component of the Gateway.
- Custom Connector: Set of customised adaptations that have to be performed in the 3rd Party System side to assure that the data exchanges are performed successfully.
- Black Box Integration: component that allow the black box (sensors, etc.) to be integrated into the platform. It can communicate directly to the Communication subcomponent.
- Legacy/ERP systems: Companies own systems (can also be SaaS systems), that manage company data that should be exposed to Virtual Enterprise Management Platform to assure the functionality envisaged.

4.3   Workflow process engine

The Workflow Process Execution component will be at the heart of the platform, as it will orchestrate all interaction in a virtual factory. Its purpose is to execute Processes, modelled in the Process Designer. This component will deal with Processes, Process Instances and the communication with gateways and logging. From a high level perspective Processes and Instances have the following minimum requirements:

- Processes: Each process has to have a set of attributes, independent of the language in order to capture the process model: Endpoints to enumerate a list of gateways/partners used in a particular process; Data elements (variables) to act as intermediate buffer for capturing results from calls to gateways and providing input for calls to gateways. Description to hold process mode.

- Instances: Each process has to be instantiated, each instance has to be stopped and started. The Name of an instance is intended to hold a description of the circumstances under which the process has been instantiated. The purpose of position is to show a set of activities that are currently executed by the instance. There can be multiple positions because there may be parallel execution of activities. While State will provide a fixed model consisting of a set of conditions and their transition, Status will hold a machine readable semantic/domain specific description of what is currently going on in the process: e.g. "Process is finished, but the results mixed", "Process has an error which is related to machine X", "The results of the process can be found in data element X".
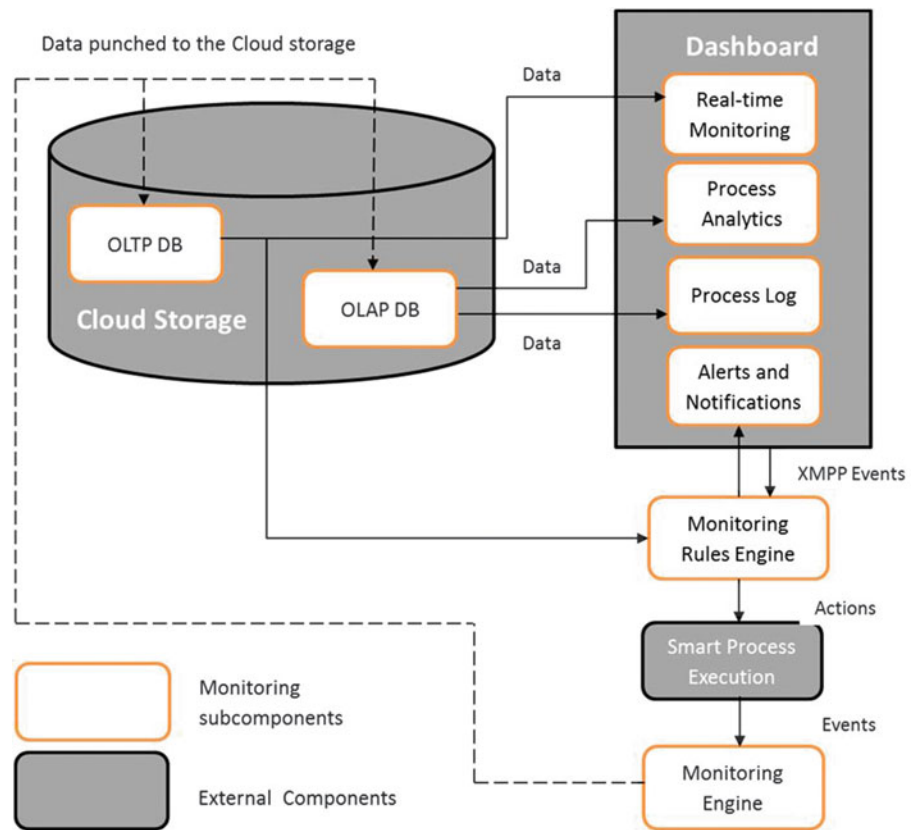
4.4   Process designer

The module's main object is the Smart Process Model and it provides the required functionality to manage its lifecycle and perform tasks related to the different phases. A new Process Model is created in the Process Designer either as an empty model or based on a ready-to-use template from templates repository. As the broker starts to design (edit) the process, the model enters its 'in design' phase. This is the core phase for the Process Designer and the designer can perform several types of tasks in it:

- Manage process metadata with the goal to enable automation and make the process discoverable
- Design process models, e.g. add/configure/remove process activities or other process model elements, using the notation and semantics supported by the Process model design tool
- Use the simulation module to trigger simulation of the process and verify its qualities before executing it and potentially return to redesign for better results
- Use the optimization module to trigger optimization of the process for optimal business goal results and potentially return to redesign for better results
- Save for further work or share the designed process model
- Load, if the process model has been saved earlier or has been shared by another user

4.5   Process monitoring

The process monitoring data will show the actual status information, log and performance data relating the virtual factory processes. The details sub-components and their interdependencies with each other are presented in Figure 2.

Fig. 2 Process Monitoring sub-components

From Figure 2 it is noticed that process monitoring component consists of the following sub-components:

- Real-time Monitoring
- Process Analytics
- Process Log
- Monitoring Rules Engine
- Monitoring Engine

Details of each of the sub-components can be explained as follows:

Real-time Monitoring: This sub-component shows the actual status of process instances. It uses the same user interface as Process Designer in order to have the same look and feel, allowing identification and tracking of the process instances.

Process Analytics: This sub-component provides an independent service that just queries finished process instances and show them to the user. It collects the key performance indicators associated with the manufacturing processes.

Process Log: It includes a search engine and shows historical data of process instances. It allows users to search for finished process instances and display its data in a graphical interface.

Monitoring Rules Engine: This sub-component mainly responsible to allow the definition of rules and corresponding actions. It throws alerts to the Dashboard, as well as performs action upon the Smart Process Engine. The rules are evaluated based on throwing events and notifications.

Monitoring Engine: It allow the visualization of key performance indicators related to the manufacturing processes and aggregates and analyses data. It provides a graphical display with the objective to track KPIs.

It is observed from Figure 2 that the Monitoring Engine receives data via XMPP events from the Smart Process Execution and stored the relevant event information in the Cloud Storage component, so that all data will be available for predictive maintenance analysis. The events data are stored within the cloud within two separate databases such as OLTP (OnLine Transaction Processing) database and OLAP (OnLine Analytical Processing) data base according to events types after following the XMPP protocol. The Real-time Monitoring sub-component displays a live view of the event data as stored within the OLTP database using the process editor interface. This data visualization helps virtual factory brokers to improve the performance of the manufacturing processes.

The Process Analytics sub-component displays the key performance indicators related to the manufacturing process over the Dashboard user interface, which were stored in the OLAP data base within the Cloud Storage. The Process Log also receives finished process instances from the OLAP database and visualizes in a graphical interface over the Dashboard as displayed in Figure 2. The Monitoring Rules Engine evaluates rules based on the events stored within the OLAP database in the Cloud Storage and triggers events and
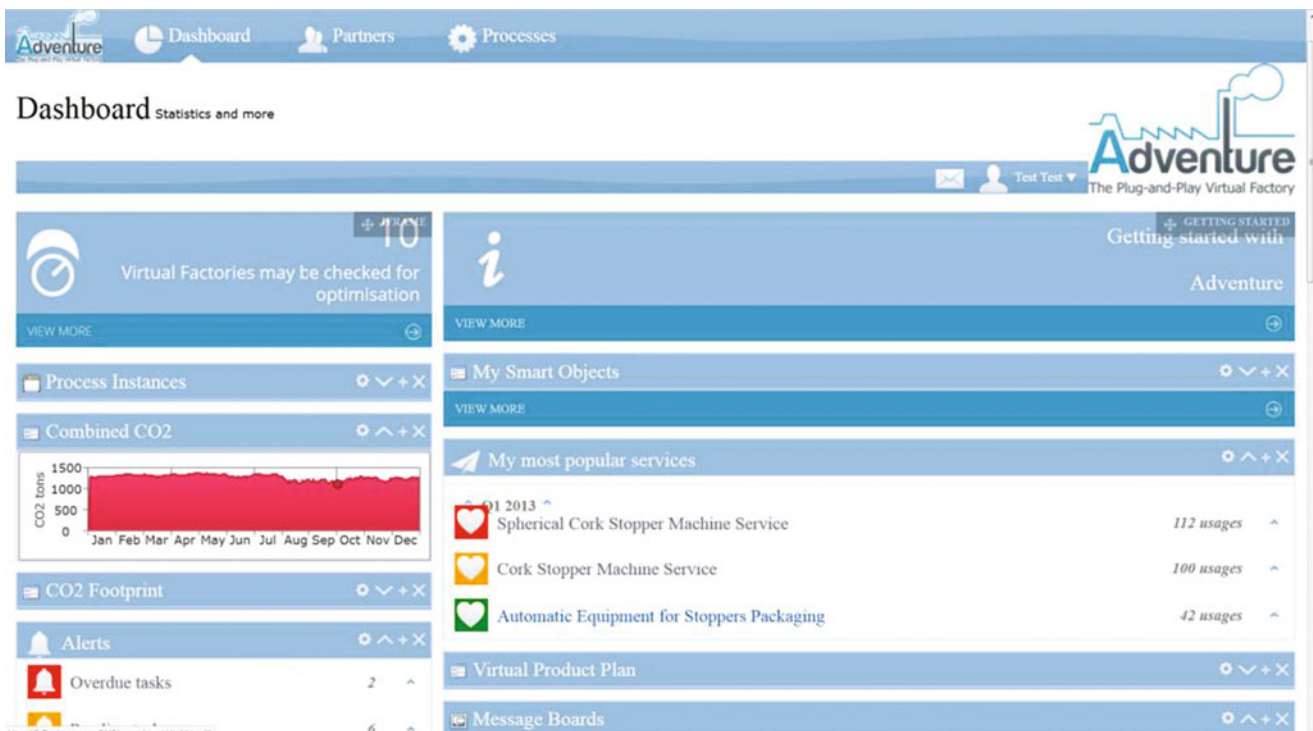
notifications. Finally, the Alerts and Notifications sub-component provides the definition of the rules based on process execution delays which are evaluated by the Monitoring Rules Engine and throws alerts to the Dashboard.

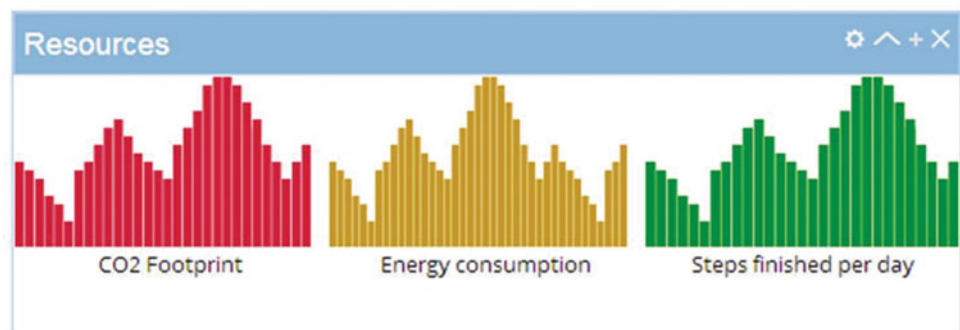## 5 ADVENTURE Dashboard: tool to VE process monitoring

In order to visualize the data as are collected from various sources (e.g. smart objects, sensors), an interactive user interface layer is designed and developed within the scope of this research. This user interface layer termed as 'Dashboard' displays the different processes monitored data in various formats (e.g. tables, graphs, texts). This data

visualization offers real-time information update of the VE processes and resources which directly influence to overall decision making processes within the VE partners. A snap shot of the dashboard interface as developed within the scope of ADVENTURE project [ADVENTURE, 2011] is displayed in Figure 3.

From Figure 3, it is seen that the dashboard interface contains several porlets or widgets, such as, process instances list, my smart objects, resources, etc. These widgets are responsible for visualizing individual data or information according to the widget type. For instance, 'Resource' widgets as displayed in Figure 4 visualizes the corresponding resource status such as CO_2 footprint, energy consumption and steps finished per day. Often this

Fig. 3 A snap shot of the ADVENTURE Dashboard homepage

Fig. 4 Snap shot of ADVENTURE dashboard's Resources widget

widget also displays other relevant resource information such as resource ready, breakdown, downtime, shortage, etc.

6 Conclusions

The main focus of this research is to highlight a complete loosely coupled virtual enterprise management tool applied to the Predictive Virtual Enterprise Maintenance Processes. This tool composed of with nine modules: (i) The integration Black Box with OPC server, (ii) Gateway with OPC Client, (iii) Process Execution Engine, (iv) Message Routing, (v) Process Designer, (vi) Process Monitoring,(vii) Data Provisioning and Discovery, (viii) Cloud Storage and (ix) Dashboard. This study mainly highlights two components such as Process Monitoring and Dashboard that are directly interfaced with VE business process monitoring and management. All other components are the supporting ones and are responsible to execute the virtual enterprise management platform successfully. The integration 'black box' as highlighted in this research collects resources or equipment data and visualizes over the dashboard through gateway services. The collected data from an individual equipment or resource by the integration black box acts as the source of predictive maintenance of the specific equipment.

User interactive component Dashboard as presented in this study enables the overall data visualization that can be used as critical decision making process. The information visibility also ensures the current condition of the equipment and/processes which to be used as the necessary corrective actions in case of abnormality or failure.

References

ADVENTURE (2011), Adaptive Virtual Enterprise Manufacturing Environment, European RTD project, Grant agreement no: 285220, Duration 01.9.2011-31.08.2014.

Camarinha-Matos, L.M., Afsarmanesh, H,, Galeano, N. & Molina, A. (2008) "Collaborative Networked Organizations – Concepts and practice in manufacturing enterprise", Computers & Industrial Engineering.

Carneiro, L., et al.: An innovative framework supporting SME networks for complex product manufacturing. Collaborative networks for a sustainable world, Volume 336, p. 204-211 (2010).

Molina, A., Velandia, M. & Galeano, N. (2007) "Virtual Enterprise Brokerage: A Structure-driven Strategy to Achieve Build to Order Supply Chains", International Journal of Production Research, Vol. 45, No. 17, pp. 3853- 3880.

Moubray, J. (1997), Reliability-Centered Maintenance, 2nd edn, Industrial Press, New York.

Yam, R.C.M, Tse, P.W., Li, L. and Tu, P. (2002), "Intelligent predictive decision support system for condition-based maintenance", International Journal of Advance Manufacturing Technology, Vol. 17, pp. 383-391.

Zhou, X., Xi, L. & Lee, J. (2007) "Reliability-centered predictive maintenance scheduling for a continuously monitored system subject to degradation", Reliability Engineering & System Safety, Vol. 92, No. 4, pp. 530-534.