A Methodology For Web-Based Manufacturing Management and Control

Alberto J. Álvares, José L.N. de Souza Jr., Evandro L.S. Teixeira, João C.E. Ferreira

***Abstract* — The objective of this work is to present a methodology for web-based manufacturing management and control. This methodology is a part of the WebMachining system, which is based on the Electronic-Manufacturing concept. The WebMachining virtual company encompasses three distributed manufacturing systems, all of them located in different cities in Brazil, i.e. Flexible Manufacturing Cell (FMC) at Graco/UnB (Brasília), FMS at SOCIESC (Joinville), and Lathe at UFSC (Florianopolis). The proposed methodology includes planning, scheduling, control, and remote manufacturing of parts. The user (customer) uses the manufacturing services based on the SOA (Service Oriented Application) provided by the WebMachining virtual company through the Internet, in order to execute operations and processes to design and manufacture the parts. The proposed methodology integrates engineering and manufacturing management through an ERP software that will preview which of the three systems will produce the ordered part, and this decision is based on parameters related to each of the three systems. After the decision, the ERP system will generate the production schedule. Also in this work the implementation aspects of a web-based shop floor controller for the FMC at Graco/UnB are presented. The FMC consists of a Romi Galaxy 15M turning center, an ASEA IRB6 robot manipulator, a Mitutoyo LSM-6100 laser micrometer, an AGV, and a pallet to store the blank and finished parts. The functional model, which depicts the modules and their relationships in the web-based shop floor controller, serves as a basic model to implement the real system. After that, the proposed implementation architecture based on the object oriented technology is presented.**

I. Introduction

Production Planning and Control (PPC) is concerned with managing the details of what and how many products to produce and when, and obtaining the raw materials, parts, and resources to produce those products. PPC solves these logistics problems by managing information [1]. PCP aims at guaranteeing that the production occurs efficiently and effectively, and that products are manufactured as required by the customer [2]. This requires that the manufacturing resources are available in an adequate amount, time, and level of quality.

Production Planning and Control systems support the efficient management of material flows, the use of man power and equipment, the coordination of the internal activities with the supply and expediting activities, and communication with the customers for its operational necessities. PPC systems help administrators in the function of decision making [3].

According to MacCarthy and Fernandes [4], there are different systems used for PPC, and some of them are kanban, PBC (Period Batch Control), OPT (Optimized Production Technology), and PERT (Program Evaluation and Review Technique) / CPM (Critical Path Method). Because of this diversity, the choice of which PPC system is the most adequate for different situations is very important. No PPC system can be considered the solution for all cases, since in order to work with different reasoning to meet diverse necessities and demands, many times it is necessary to use more than one PPC system.

In these circumstances, a methodology is proposed in this paper for the web-based manufacturing management and control of the WebMachining virtual company, whose shop floor is composed by three distributed manufacturing systems located in different cities in Brazil: FMC Graco/UnB (Brasília), FMS SOCIESC (Joinville), and Lathe UFSC (Florianópolis). The proposed methodology includes the development of an ERP system and the integration of this ERP system with the engineering module (CAD/CAPP/CAM). In the engineering module, two part development environments are used: WebMachining [5] and Cybercut [6]. The ERP system is developed for the web, thus allowing customers to input its orders of parts anywhere, without having the equipment and software for carrying out the product development cycle. The methodology also allows the company employees to connect remotely to the system and perform activities from any place (figure 1).

For the implementation of the methodology, telemanufacturing is used, which a part of the electronic-manufacturing concept [7]. The customer uses the manufacturing services via web (SOA – Service Oriented Application) to execute the operations and the necessary processes, designing and manufacturing the desired part efficiently and with flexibility, using computational tools for the development of the product life cycle.

This work also presents the implementation aspects of a web-based shop floor controller for the FMC at Graco/UnB. The FMC consists of a Romi Galaxy 15M turning center, an

Manuscript received February 29, 2008.

A.J. Álvares and J.L.N. de Souza Jr. are with the Universidade de Brasília, Departamento de Engenharia Mecânica e Mecatrônica, Grupo de Automação e Controle (GRACO), CEP: 70910-900, Brasília – DF, Brazil (e-mail: álvares@alvarestech.com, leojunior@unb.br).

E.L.S. Teixeira is with the Autotrac Commerce and Telecommunication S/A, Department of Hardware Development (DDH), CEP 70910-901, Brasília, DF, Brazil (e-mail: evandro.teixeira@autotrac.com.br).

J.C.E. Ferreira is with the Universidade Federal de Santa Catarina, Departamento de Engenharia Mecânica, GRIMA/GRUCON, Caixa Postal 476, CEP 88040-900, Florianopolis, SC, Brazil (corresponding author - phone: +(5548) 37219387 ext 212; fax: +(5548) 37217615; e-mail: jcarlos@emc.ufsc.br or jcarlos.ferreira@gmail.com).

ASEA IRB6 robot manipulator, a Mitutoyo LSM-6100 laser micrometer, an AGV, and a pallet to store the blank and finished parts. The functional model, which depicts the modules and their relationships in the web-based shop floor controller, serves as a basic model to implement the real system. After that, the proposed implementation architecture based on the object oriented technology is presented.

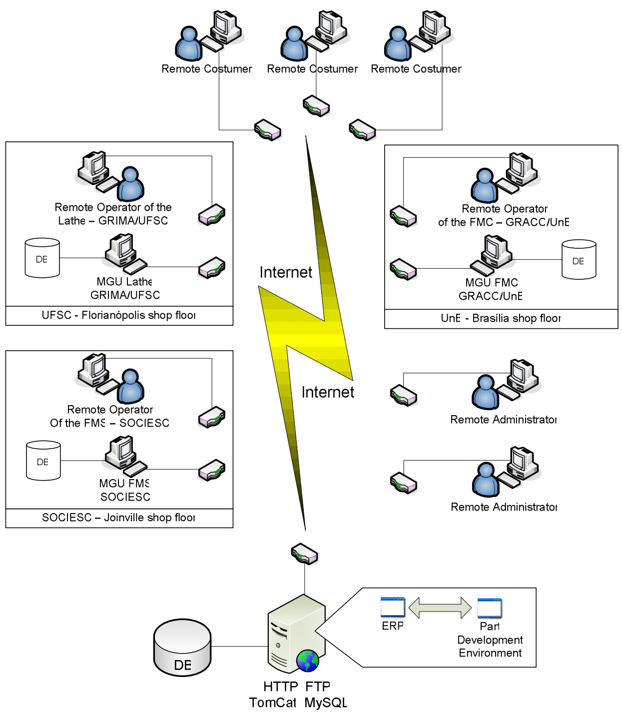

Fig. 1: Remote accesses to the system

II. OVERVIEW

A. *ERP Systems*

With the advances in information technology (IT), the companies started to use computer systems to support its activities. Generally, in each company, some systems were developed to meet specific requirements of the diverse business units, factories, departments and offices. Thus, information was fragmented among different systems. The main problems of this fragmentation are the difficulty in getting the consolidated information, and the inconsistency of stored redundant data in more than one system. ERP systems solve these problems by including, in just one integrated system, functionalities that support the activities of different companies [8].

Because of the evolution from MRP systems to ERP systems, today it is possible to include and to control all the company processes, without the redundancies found in the previous systems. Information is displayed in a clearer way, immediately and safely, providing a greater control of the business, which includes its vulnerable points, such as: costs, financial control and supplies.

B. *Electronic Manufacturing (e-Mfg)*

IT, especially the network communication technology and the convergence of wireless and Internet, is opening a new domain for building the future manufacture environments called e-Mfg (electronic-Manufacturing), using labor methods based on collaborative e-Work, especially the activities developed during product development in integrated and collaborative CAD/CAPP/CAM environments. E-Work (Electronic Work) is defined as any activity that is collaborative, supported for computer and by communication in highly distributed organizations of robots and/or people or independent systems [9]. In essence, e-Work is composed by e-activities (electronic-activities), i.e., activities based and executed by the use of information technology. These e-activities include v-Design (v for virtual), e-Business, e-Commerce, e-Manufacturing, v-Factories, v-Enterprises, e-Logistics, and similarly, intelligent robotics, intelligent transport, etc.

E-Mfg can be considered as a new paradigm for these computer systems based on global environments, network-centered and spatially distributed, enabling the development of activities using e-Work. This will allow product designers to have easier communication, making it possible the sharing and collaborative design during product development, as well as the teleoperation and monitoring of the manufacturing equipment [5].

C. *WebMachining Methodology*

The design portion of the WebMachining methodology is based on the synthesis of design features, i.e. union of features for turning operations and subtraction of features for milling operations [5]. The methodology has the purpose of allowing the integration of the collaborative design activities (CAD), process planning (CAPP) and manufacturing (CAM Planning and CAM execution). In order to achieve this, it uses as design reference the manufacturing features model defined by the part 224 of STEP - Standard Exchange of Product Model Data (ISO 10303) [10], and more specifically the taxonomy of form features for rotational parts defined by CAM-I [11].

The procedure begins in the collaborative modeling of a part using features in a context of remote manufacturing via web, in a client-server computer model. Some of the data that are generated by the system include: the geometric and feature-based model of the part (detailed design), the process planning with alternatives (WebCAPP module), and the NC program. Then, the teleoperation of CNC lathe is carried out (WebTurning module). The methodology can be applied for the manufacture of rotational parts, as well as prismatic parts.

D. *CyberCut*

CyberCut is a web-based design-to-manufacturing system, developed by Brown and Wright [6], consisting of the following major components:

- Computer-aided design software written in Java and embedded in a web page. This CAD software is based on the concept of destructive solid geometry (DSG); that is, by constraining the user to remove entities from a regularly shaped workpiece, the downstream manufacturing process for the part is inherently incorporated into the design,

- A computer-aided process planning system with access to a knowledge base containing the available tools and fixtures,
- An open-architecture machine tool controller that can receive the high-level design and planning information and carry out sensor-based machining on a Haas VF-1 machine tool.

According to Brown and Wright [6], by providing access to the CyberCut CAD interface over the Internet, any engineer with a web browser becomes a potential user of this on-line rapid prototyping tool. A remote user would be able to download a CAD file in some specified universal exchange format to the CyberCut server, which would in turn execute the necessary process planning and generate the appropriate NC code for milling. The part could then be manufactured and shipped to the designer. The engineer could have a fully functional prototype within a matter of days at a fraction of the cost of in-house manufacturing.

III. THE PROMME METHODOLOGY

PROMME is a methodology to provide the means to manufacture and control management in a distributed manufacturing environment, and it is applied to the virtual WebMachining Company, whose shop floor is formed by three distributed manufacturing systems. An ERP software was developed in order to carry out manufacturing management, enabling the receipt of customer orders, management functions, CAD/CAPP/CAM integration, and part manufacture in one of the three manufacturing systems.

A. *Distributed Shop Floor*

The shop floor of the virtual WebMachining Company, as described previously, is formed by three distributed manufacturing systems: FMC Graco/UnB, FMS SOCIESC and Lathe UFSC. The UnB (Brasilia - DF) system is a FMC (Flexible Manufacturing Cell) (http://video.graco.unb.br) composed by a CNC turning center, an industrial robot, an AGV, pallets of parts, laser micrometer, management unit (MGU) and an audio and video monitoring system, as shown figure 2. In the FMC-UnB system, the parts with non-concentric features can be manufactured in the 3-axis Romi Galaxy 15M CNC turning center.

The SOCIESC system (Joinvile), shown in figure 3, is a FMS (Flexible Manufacturing System) composed by a Feeler CNC lathe, a Feeler CNC machining center, an ABB 2400 robot, and an automated storage and retrieval system (AS/RS).

The third system, located at UFSC (Florianópolis), is composed only by a Romi CNC lathe. This system makes only parts with concentric features, and in this case the presence of an operator to feed the lathe is necessary.

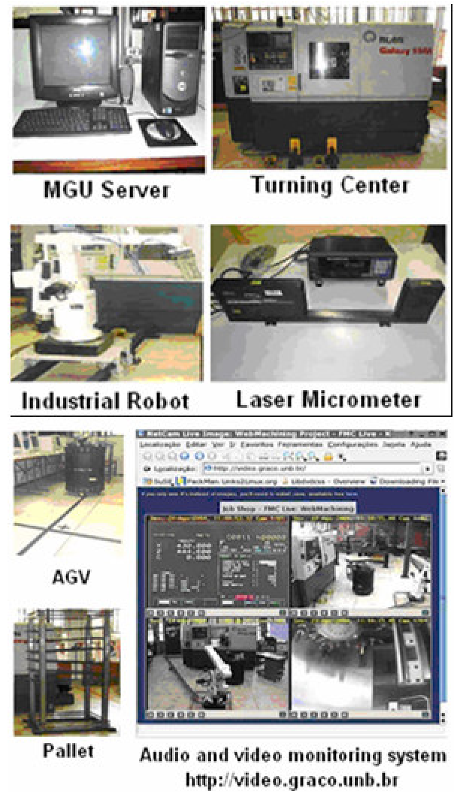

Fig. 2: FMC GRACO/UnB in Brasilia (http://www.graco.unb.br)

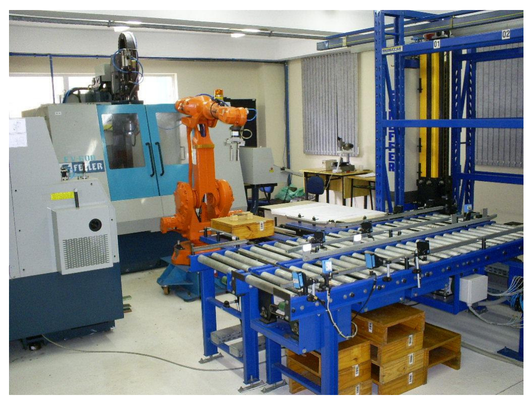

Fig. 3 : FMS SOCIESC (http://www.grima.ufsc.br/sociesc/fms2/fms2.htm)

B. *ERP Manufacturing*

The ERP Manufacturing is a web-based system, written in Java (http://java.sun.com) and JavaServerPages (JSP - http://java.sun.com/products/jsp), which enables the management of the virtual company. The management is composed by: the control of user accesses through the internet, the integration with CAD/CAPP/CAM modules, the computer-aided production module (CAP), the integration with the management units of the distributed shop floor, and the management activities of the company. All these modules are described in the next sections.

1) Institutional module

The institutional module is where the employees of the WebMachining Company perform the administrative and operational activities of the company. The managers are responsible for registering new employees, excluding or modifying a register of an employee, modifying values of the production cost calculation of each shop floor, visualizing monthly profits and expenses, and visualizing the systems' production by using Gantt graphs.

Each shop floor has operators who have some functions, such as: registering suppliers, updating supply of tools, requesting the purchase of materials, registering monthly expenses, getting the daily production of the manufacturing system.

2) Commercial module

After having access to the site of the virtual WebMachining company, the customer enters the commercial module, registers, performs system log-in, and then the page with the customer menu is available. In this page the customer can input a new work order, see the work order status, modify or cancel a work order and modify its registered data. This is the first stage in the production process of the company, and one of most important. It is in this stage that the customer registers information of the priority, the part type and the batch size.

The customer priority can be determined based on the production time (e.g. if the batch must be manufactured in the shortest possible time, thus the work order becomes more expensive), or on the production cost (e.g. the time is not the most important factor, but the final batch price).

In the proposed methodology, there are two types of parts that the customer must inform the system: prismatic or rotational. This definition is the first information that is used by the system for decision making, since only the FMS - SOCIESC is capable to produce prismatic parts.

3) Integration with the Part Development Environment

The part development environment is composed by the WebMachining and CyberCut systems. In this work WebMachining is used for the design of rotational parts, whereas CyberCut is used for prismatic parts.

After the preliminary pieces of information about the work order are registered, the ERP shows to the customer, via a servlet, a CAD interface with one of the two tools, depending on the part type. In this interface the customer designs the part, which is then sent to the process planning module (WebCAPP) [12], which is responsible for including in data base the information about the machining operations, the time of each machining operation, the list of tools to be used in the manufacturing process, and the NC program to be sent to one of the three management units. These pieces of information are crucial in the decision making about where the part will be manufactured.

4) CAP – Computer Aided Production

Production planning in PROMME is divided into two parts: decision making, which determines which shop floor will be responsible for manufacturing the part, and the production scheduling, which is composed by the daily production plan, and by the part families formation.

Decision Making

Decision making is carried out considering basically the production capacity of the shop floor and the customers priorities. The algorithm bellow describes how decision is made.

Decision Making Algorithm

1. If (Part Type = prismatic)
    - 1.1. start CyberCut
    - 1.2. If (Priority = cost)
        - 1.2.1. calculate the production cost and the maximum production time
    - 1.3. Else, calculate the production cost and the minimum production time
2. Else If (Part Type = rotational)
    - 2.1. start WebMachining
    - 2.2. compare the required work order tools with the manufacturing systems tools
    - 2.3. If (Priority = cost)
        - 2.3.1. verify the sending type
        - 2.3.2. calculate the sending cost + production cost for each system and gets the lower
        - 2.3.3. calculate the maximum production time
    - 2.4. Else
        - 2.4.1. verify which system has the lesser number of work orders to be produced
        - 2.4.2. calculate the production cost and the minimum production time

The total production time of the work orders is calculated based on times of machining operations, available in the data base, and on the tools setup time. It is assumed that the shop floors are available sixteen hours a day, seven days a week. According to the production times of the parts, scheduling is performed each day.

The calculation of the minimum production time is made by analyzing the work orders in the data base. All work orders have a completion date, i.e., the date in which the manufacture will be concluded. Every time the system has to preview when the work order will be completed, a search in the database is made to know the next date available to manufacturing the batch. The result of this search is shown to the customer, and he/she decides if it confirms or not the manufacture of the batch. The batch will only be scheduled if the customer confirms the production of the work order.

The maximum manufacturing time, which is the same for all shop floors, is one month after the date the work order entered the system. As in the previous case, the batch will only be scheduled if the customer confirms the manufacture of the work order.

Production Scheduling

The master production schedule groups the work orders, which were created by the customers and included in the database, to their respective destination shop floor. The master schedule provides the knowledge about which shop floor the work order is.

With regard to the formation of the part families, the algorithm of rank order proposed by King [13] was applied. In this case, the parts that have the same tool characteristics are grouped into the same family. This prevents that a new setup of tools is made every time that a new part is processed. The parts are grouped using the list of tools included in data base by the part development environment.

Production scheduling is performed in the decision making process because the foreseen date for manufacturing completion must be shown to the customer for him/her to confirm. The work order will only be included in the master production schedule if the customer accepts the foreseen date. The sequence in which the part families are produced in a day does not matter. The important thing is that the work orders are manufactured before the date that was shown to the customer.

5) Integration with the Management Units of the distributed shop floors

The integration with the management units (MGUs) is made remotely, via database. All the work orders to be produced are into the master production schedule. The operators in each shop floor connect via the Internet with the web server and get the information about the work orders to be done in the day. The MGUs are responsible for the success of the production. They are responsible for controlling the pieces of equipment, and also for the production scheduling in the shop floor.

IV. SYSTEM MODELING

A. IDEF0

According to the IDEF0 standard (http://www.idef.com), IDEF0 may be used to model a wide variety of automated and non-automated systems. For new systems, IDEF0 may be used first to define the requirements and specify the functions, and then to design an implementation that meets the requirements and performs the functions. For existing systems, IDEF0 can be used to analyze the functions the system performs and to record the mechanisms (means) by which these are done. The IDEF0 methodology also prescribes procedures and techniques for developing and interpreting models, including ones for data gathering, diagram construction, review cycles and documentation. Figure 4 shows the IDEF0 modeling of the PROMME methodology.

B. UML

In the field of software engineering, the Unified Modeling Language (UML - http://www.uml.org) is a standardized specification language for object modeling. UML is a general-purpose modeling language that includes a graphical notation used to create an abstract model of a system, referred to as a UML model. There are lots of diagrams to model a system.

In the Unified Modeling Language, a package diagram depicts how a system is split up into logical groupings by showing the dependencies among these groupings. As a package is typically thought of as a directory, package diagrams provide a logical hierarchical decomposition of a system. Figure 5 shows the package diagram of the PROMME methodology.

V. THE WEB-BASE SHOP FLOOR CONTROLLER (WSFC) FOR THE FMC AT GRACO/UNB

A web-based shop floor controller for the Flexible Manufacturing Cell (FMC) at Graco/Unb (in Brasilia) was also implemented, and it uses WWW resources to perform the remote manufacture of parts. The FMC receives instructions from the controller and converts them onto the operations necessary to manufacture the parts. The web-based shop floor controller (WSFC), as a computer system, should meet the following requirements: (a) it support production planning, (b) it should have functions to verify the availability of the production resources allowing the instruction loading on the workstations, (c) it should control and monitor the production process reacting on abnormal condition that can hinder the fulfillment of the activities established previously on the production planning.

A. COMMUNICATION WITHIN THE FLEXIBLE MANUFACTURING CELL

In order to describe the implementation of the control for the FMC at Graco/UnB in Brasilia, it is important to visualize the FMC communication structure, describing the method used by the human operator to access the FMC resources (figure 6).

The Turning Center (Romi Galaxy 15 M) communication is established by an Ethernet interface, using the TCP/IP protocol, linked to the programming library (FOCAS1). The API FOCAS1 drivers and programming library provides the communication and programmable access to PC based on CNC System [14]. This programming library supplies about 300 function calls that can be implemented by the customer applications.

The Ethernet/Radio system is used to establish the AGV communication. This system possesses a Proxin RangeLan2 interface to connect the robot on the computer network and to communicate with the bridge server. This server connects the robot on the local network using the TCP/IP protocol [15]. This configuration mode provides access to the main network mechanisms and patterns (ftp, telnet, TCP/IP, sockets) and the robot can be operated as a network workstation.

The micrometer communication is established by means of a RS232C interface. The communication process is restricted to 23 programmed commands defined by the micrometer manufacturer (Mitutoyo). These commands provide both remote programming of geometric parameters (diameter and tolerances) of a part feature and the conditions in which the inspection will be performed.

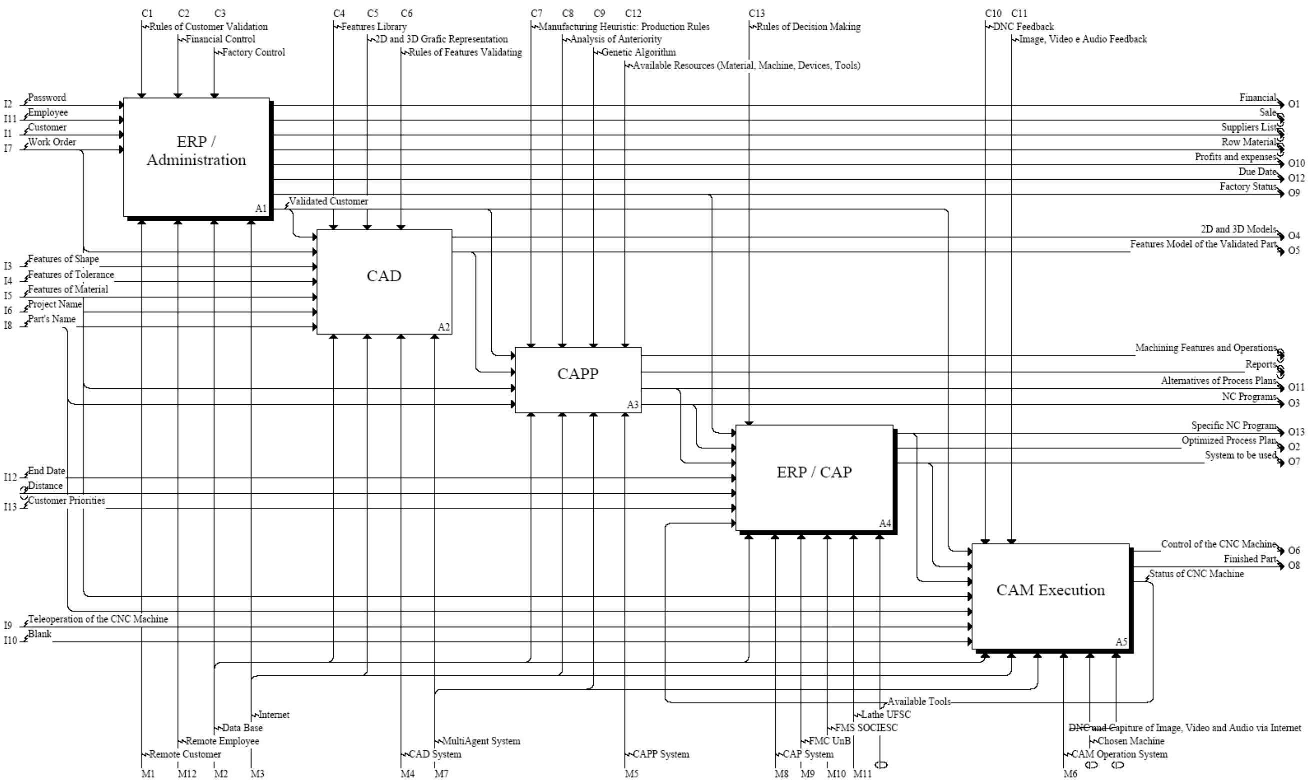

Fig. 4: IDEF0 modeling of the PROMME methodology

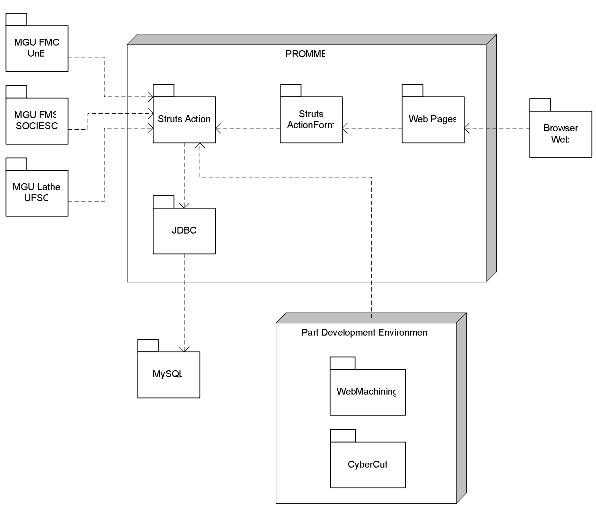

Fig. 5: UML Package Diagram - PROMME Architecture

The material and handling communication is limited to 13 digital I/O's (7 inputs and 6 outputs). This constraint resulted in the design of a dedicated interface to establish the indirect communication with the robot based on a CNC/PMC/Robot controller architecture. This was carried out in a partnership with the manufacturer of the CNC turning center.

A. WEB SHOP FLOOR CONTROLLER IMPLEMENTATION

The implementation architecture of the controller should encompass the main functionality that the real system should offer (from the human operator to the workstations). The implementation architecture was built based on object oriented technology.

1) Implementation architecture – Package and component diagram

The package and the component diagram of the Unified Modeling Language (UML) were used to design and to document the implementation architecture. The WSFC modules have their responsibilities distributed on packages. One package is a basic mechanism used to organize and classify the elements of a group [16]. Class, interfaces and components that possess similar functionality were grouped on package. Figure 7 shows the WSFC package diagram, where the initialization package has only one class (the Initialization class). This class has only one method (the main method) invoked every time that the WSFC is initialized. The main method is responsible to instance the WSFC Navigator Controller.

The package Controller groups all the Controller classes, which encapsulate the logical approach of the system. It can be classified as FrameController or LayerController class. One FrameController class listens

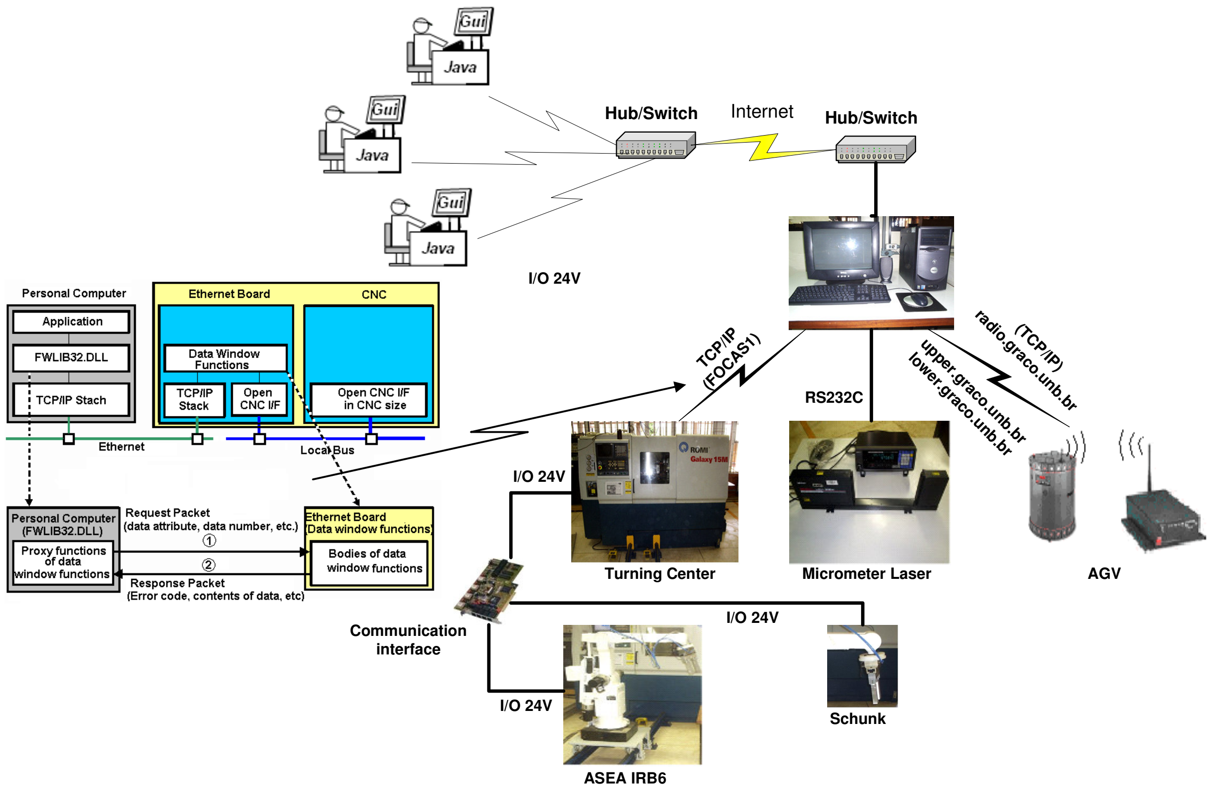

Fig. 6. Communication structure in the FMC at Graco/UnB in Brasilia

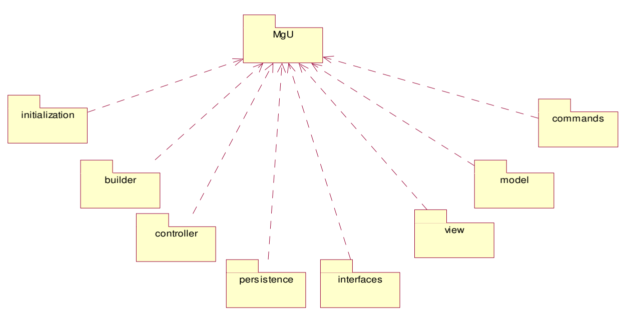

Fig. 7. Web Shop Floor Controller package diagram

to every user interaction with the GUI's, formatting and encapsulating user information to be processed, while one LayerController class manages the system navigability and the service changes among the software layers.

BuilderScheduler, BuilderDispatcher and BuilderMonitor are the main classes stored in the Builder package. These classes are responsible for building the WSFC modules and their interconnections. The WSFC build process consists of instancing all the FrameController and LayerController objects that compose the WSFC modules, and connects them by means of relationships.

The Interface package groups the entire interfaces used in the WFSC. One interface is a mechanism used to reduce the coupling degree among objects. When a software layer is connected by interfaces, the modification of the one layer does not expand to the others. Thus, this mechanism provides easier expansibility and maintenance of the system.

The Command package groups the entire Command class. The instance of a Command class encapsulates any request as an object, and consequently the object that invokes the operation does not need to know how the request should be executed. For example, when the SQLInsertWorkOrder (one Command object) is initialized, it receives one WorkOrderData object, which encapsulates the WorkOrder attributes such as due date, priority, etc. After the execute method is invoked, a String containing the SQL instruction is created and used to insert one new WorkOrder on the database.

The View package aggregates the entire GUIs developed for the WSFC. These GUIs provide the user a means to interact with the workstations. Some customer objects, such as DateChooseButton, are built considering the principle to modularize the information (encapsulates all the common functionalities), besides preventing code repetition.

The Persistence package contains the entire classes that establish the communication with the external devices. The instances of DBConnection, MicrometerConnection, TurningCenterConnection, RobotConnection and AGVConnection class establish the connection with the database, with the micrometer, with the robot, with the Turning Center and with the AGV respectively.

Figure 8 shows the component diagram of the WSFC based on the client-server architecture. The GUIs and the upper level functionalities are available on the client module,

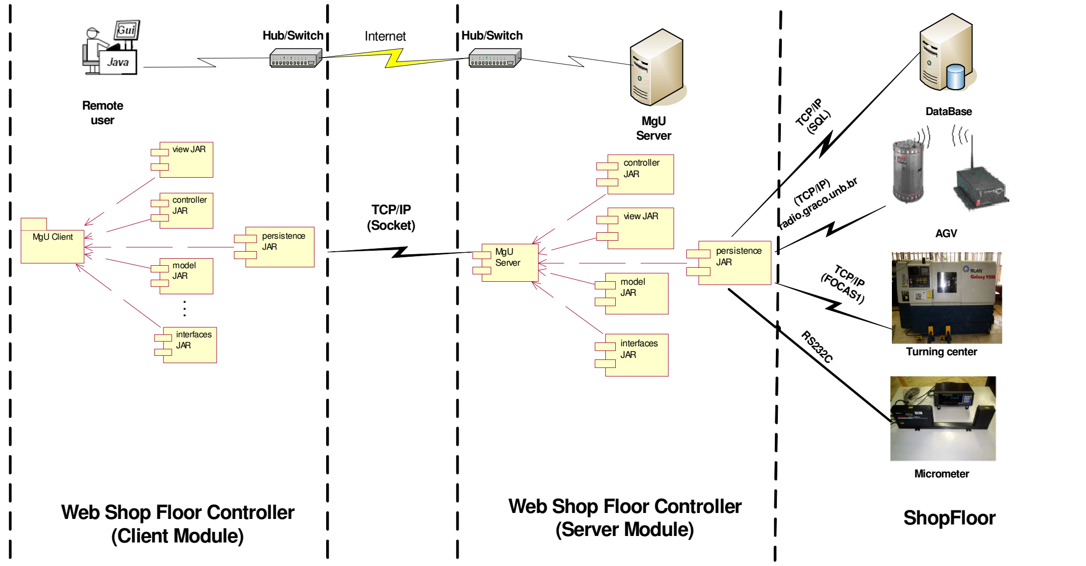

Fig. 8. The component diagram of the web-based shop floor controller

while the lower level functionality (e.g. direct connection with the workstations) are available on the server module. The communication between these modules is established by sockets, using the TCP/IP communication protocol.

The client (any remote user) connects to the FMC server through the following URL: http://webfmc.graco.unb.br/mgu/mgu.jnlp. This link points to the JNLP archive, one XML document (Extended Markup Language) that specifies which JAR archive file from the client module should be downloaded to the remote computer user. When all the files are download (specified on the XML document), the WSFC is ready to be executed.

Besides incorporating all the advantages offered by applets (i.e. to execute the application via web without installing it, restriction policy, etc), the use of the Java Network Launch Protocol (JNLP - http://java.sun.com/products/javawebstart/download-spec.html) technology allows the incremental download of the application. It means that every time the application is to be executed on the client computer, only the modified JAR archives from the web server will be downloaded.

1) Implementation under the distributed architecture

To provide remote access to the workstations (via web), maintaining the portability that the system should offer, the web-based shop floor controller was implemented in a client-server architecture. So, the system was designed in two modules: the client and server module. Figure 9 shows the component diagram of WSFC designed under the client-server architecture.

The client module encapsulates the upper level functionality that does not access directly the operating system resources, providing system portability, and communicates with the logical controller of the server module. In this module, the instances of SchedulerController, DispatcherController and MonitorController class encapsulate the upper level functionality (such as *requestMasterPlanningScheduling()*, *sendWorkstationCommands()*, *verifyWokstationStates()*, etc), and establishes the communication with the workstation's logical controller by means of their respective interfaces.

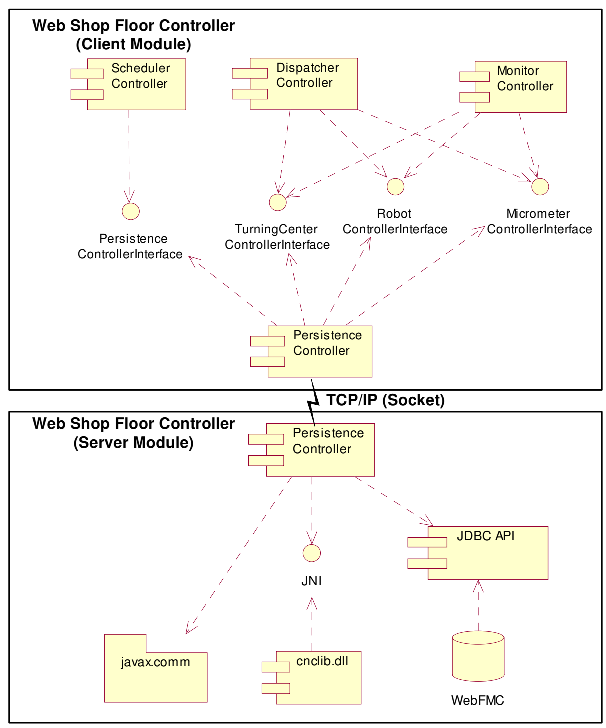

Fig. 9. Component diagram under the client-server architecture

The server module, designed under the multithread environment, implements the upper level functionalities offered to the client module and establishes the direct communication with the workstation controllers. In this module, the communication with the CNC turning center is established using the JNI technology (Java Native Interface - http://java.sun.com/j2se/1.5.0/docs/guide/jni/index.html). The JNI allows the code executed into the JVM (Java Virtual Machine) to interact with another applications and libraries written in other programming languages, such as C, C++, etc.

Afterwards, by means of these interfaces, the WSFC (written in Java technology) was able to communicate with the library developed in C++ (cnclib.dll) that encapsulates some programming functions offered by API FOCAS1 (used on CNC communication).

The Java Communication is the Application Programming Interface (API) used to establish the communication with the micrometer. This API provides serial port access (via RS232 interface) and parallel port access (IEE-1284). To access the WebFMC database, the API Java Database Connectivity (JDBC - http://java.sun.com/products/jdbc/overview.html) was used. This API is an industrial standard established to connect the Java technology and some databases, using the Structured Query Language (SQL). It is necessary to maintain the Java portability and permits to change the database without modify the server module from the WSFC.

B. RESULTS

1) Inspection and production planning

The inspection plan is included to the information used to plan, control and monitor the inspection of the parts. This plan is composed by a set of inspection programs previously recorded on the database. Each program has the part geometry information (diameter and tolerance of each feature that will be inspected), as well as the inspection conditions (unit system, reference, scale, etc).

The GUI InspectionPlan (figure 10) was implemented in order to provide the possibility to add, to edit and to delete the inspection programs of the inspection plan. Each inspection program can be used to group up to 10 feature geometric information, and consequently the same production program can be used to inspect several parts without modifying the current micrometer program, since the inspection conditions are the same.

The Master Production Scheduling (MPS) is added to the work orders recorded on the database. One work order has attributes such as priority, due date, process time, etc., which will be used by other WSFC modules. The GUI ProductionPlan (figure 10) to provide the possibility to add, to edit and to delete the work orders from the database. Each work order has an attribute called “status”, which informs the system about the situation of the work order (i.e., “to produce”, “in production”, and “produced”). Therefore, in order to schedule the production, the operator should select the work orders that will be produced, setting the work order status attribute as “in production”.

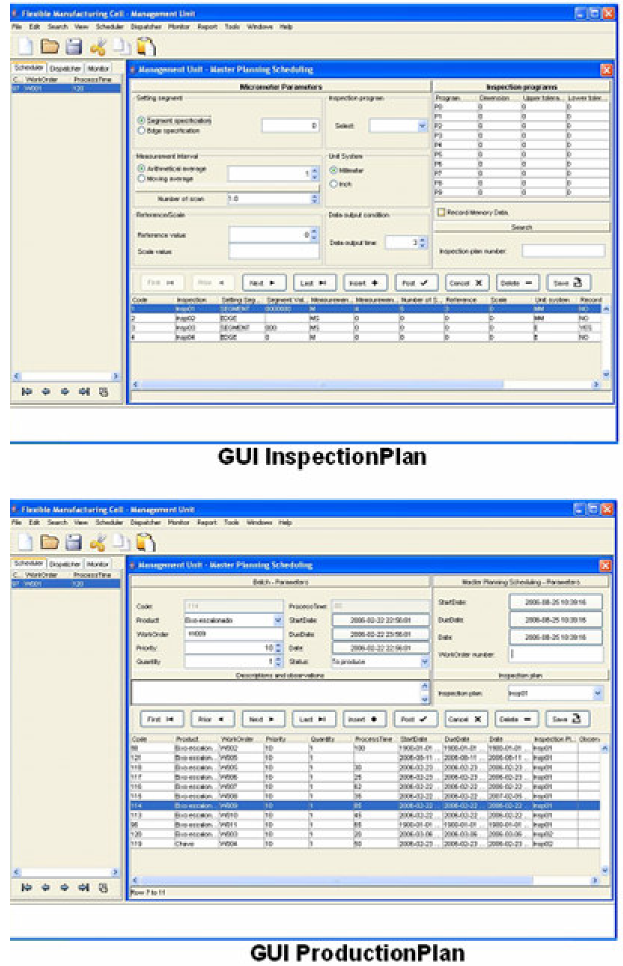

Fig. 10. GUI InspectionPlan and ProductionPlan

2) Scheduling production and dispatching

After concluding the production plan, the next step consists of establishing the sequence in which the work orders will be manufactured. The scheduling method adopted in this work is based on priority rules [17]. Figure 11 shows the GUI GanttGraph used to provide the necessary support to generate the operation sequence for the work orders selected from the production plan.

When the operator selects the scheduling manual mode, a JDialog shows the planned work orders, and the human operator can schedule the work orders manually. On the other hand, if the automatic mode is selected, the automatic scheduler algorithm verifies the program method (Forward or Backward) and the sequence rule (Priority, Earliest Due Date, First In First Out or Shortest Process Time) chosen to schedule the work orders.

After determining the task list (i.e. the work order operations sequence), the human operator should have to dispatch the task list to the workstation. This action is composed by two phases: (a) verification of the workstation status and (b) loading of the task list on the workstations. The GUI VerifyWorkstationStatus (figure 11) was implemented to provide the necessary operation interaction to verify the workstation status.

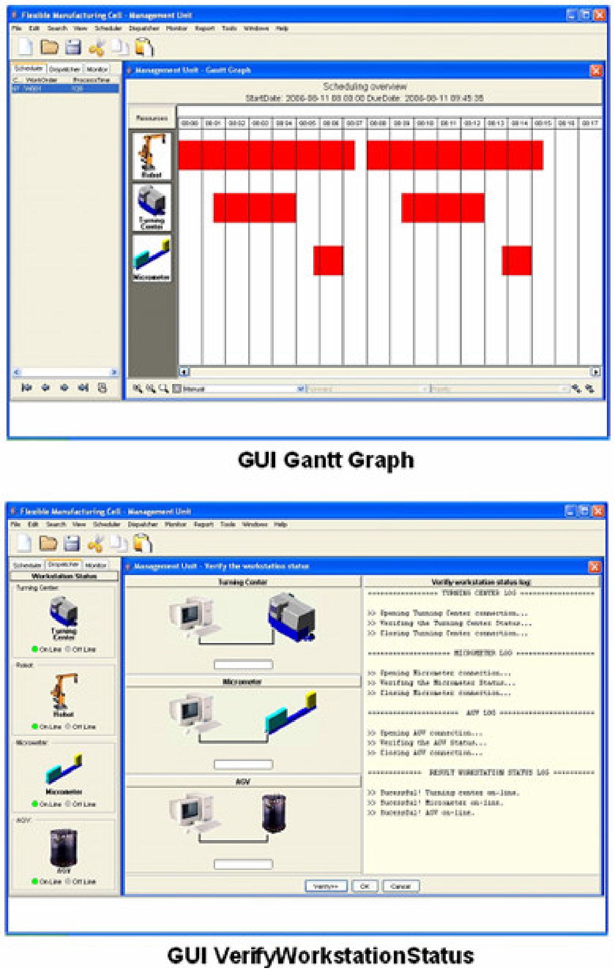

Fig. 11. GUI GanttGhaph and VerifyWorkstationStatus

The WSFC communicates with each workstation in order to verify whether it is available to receive a task upload. While the WSFC is checking the workstation status, a report (log) of the executed command is shown on right side of the GUI. If any fault occurs during the verification process, the human operator cannot dispatch the task list to the workstations.

3) Production monitor and quality control

The VirtualMonitorFrame (figure 12) provides the virtual monitoring of the workstations. The tab monitor (on the left-hand side) shows the PMC tags from the CNC turning center allocated to the workstation integration. In the center the virtual images from the workstations manufacturing a part are presented. On the right-hand side, a JPanel shows the report (log) of each occurred event on the shop floor.

The GUI QualitControl provides the human operation interaction with the quality control process. The statistical method selected to control the process is the pre-control [18]. This method is composed of three steps: to qualify the process, operation and sample frequency. The inspection of a part starts with the positioning of the manufactured part on the micrometer's read unit. After positioning the part on the micrometer and the programmed inspection time (DataOutputTimer) expires, the micrometer sends the inspection result to the WFSC via the RS232 interface.

The program number is the geometric information from the inspected feature (diameter and tolerance). If the judgment criteria was activated, the micrometer process unit evaluates the inspection result and checks if the measured value is inside on the tolerance limits (previously defined). The result can be –NG (if the measurement value is lower than the lower tolerance limit), G0 (if the measured is within tolerance limits) and +NG (if the measured value is larger than the upper tolerance limit). Figure 12 shows the virtual monitoring of the real time activity.

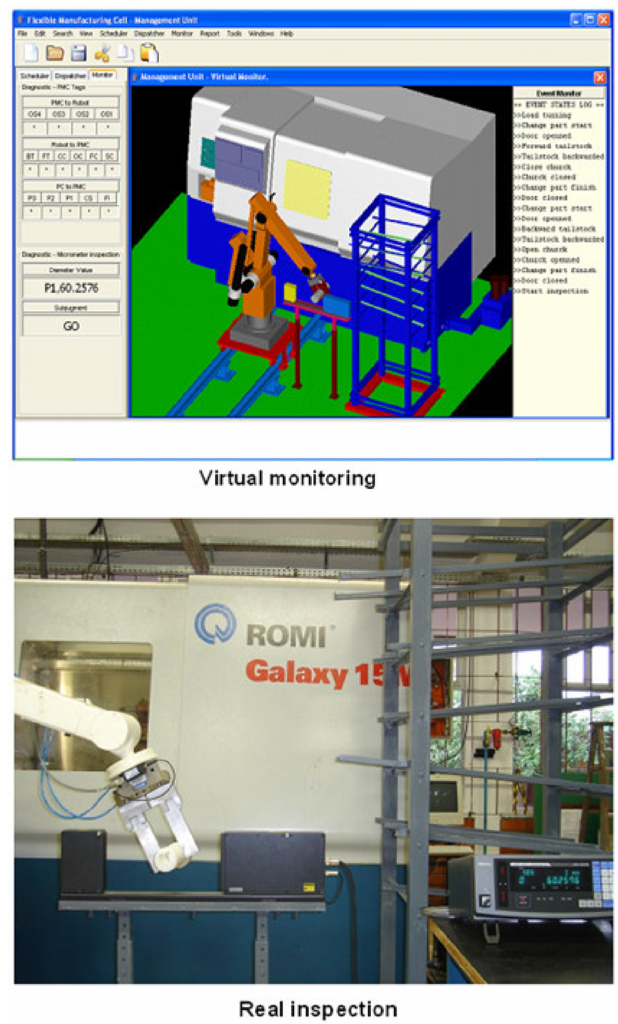

Fig. 12. GUI Virtual Monitor and the real inspection

VI. CONCLUSIONS

The proposed PROMME methodology contains a concept for web-based manufacturing management that encompasses a web based system, an ERP software written in the Java language, and the presence of distributed manufacturing systems located in different places.

Control is performed using e-manufacturing concept that integrates the remote accesses by users (customers and employees), the customers orders, engineering (CAD/CAPP/CAM), the distributed shop floor, and sales. This integration allows the customer to execute operations and the required processes to design and produce the parts with a high amount of efficiency and flexibility, without possessing the necessary pieces of equipment. The integration also allows the employees to carry out company activities remotely.

With regard to the web-based shop floor controller (WSFC), it is a computer system that uses Internet resources to promote the remote manufacturing of parts. Besides of the portability and the remote access via Internet, the WSFC schedules, controls and monitors the activities on the shop floor. The first computational version of the WSFC can be executed at http://webfmc.graco.unb.br/mgu/mgu.jnlp. Although many functions are not implemented yet, the proposed algorithms and the supplied UML diagram are an important documentation to accomplish their further implementation in the future.

These UML diagrams were designed based on object oriented technology, and therefore can be applied to any object oriented programming language. The package and component diagrams serve as a model in order to facilitate future changes.

The implementation based on Java technology enables the WSFC to be executed over the Internet without the need for the user to install the application. However, the client must have the Java Runtime Environment (JRE) installed, which supports the Java 2 platform.

The implementation based on a client-server architecture encapsulates services that call the operating system functions (e.g. CNC communication and micrometer communication) on the server side. Therefore, the portability inherited from the Java technology is maintained, and the WSFC can be executed on different operating systems (client side).

VII. REFERENCES

[1] M. K. Groover, *Automation, Production Systems, and Computer-Intergated Manufacturing*, 2nd ed. New Jersey: Prentice-Hall, 2001

[2] N. Slack, S. Chambers and R. Johnston, *Operations Management*. 4th Edition, Finantial Times/Prentice-Hall, 2004.

[3] T. E. Vollman, W. L. Berry, D. C. Whybark, *Manufacturing Planning and Control Systems*. 4th ed. New York: McGraw Hill, 1997.

[4] B. L. MacCarthy and F. C. Fernandes, *A multidimensional Classification of Production Systems for the Design and Selection of Production Planning and Control Systems*. Production Planning & Control, vol.11, no. 5, 2000.

[5] A. J. Álvares and J. C. E. Ferreira, *A System for the Design and Manufacture of Feature-Based Parts Through the Internet*. International Journal of Advanced Manufacturing Technology, Springer, UK, 2006.

[6] S. M. Brown and P. K. Wright, *A Progress Report on the Manufacturing Analysis Service, an Internet-Based Reference Tool*. Journal of Manufacturing Systems, vol. 17, no. 5, 1998, pp.389-401.

[7] L. A. Malek, C. Wolf and P.D. Guyot, *Telemanufacturing: A Flexible Manufacturing Solution*. International Journal of Production Economics, vol. 56-57, 1998, pp. 1-12.

[8] T. H. Davenport, *Putting the enterprise into the enterprise system*. Harvard Business Review. Vol. 16, no. 4, July-August, 1998, pp.121-131.

[9] S. Y. Nof, *Collaborative E-Work And E-Mfg.: The State Of The Art And Challenges For Production And Logistics Managers*. International Federation of Automatic Control 11th IFAC Symposium on Information Control Problems in Manufacturing (INCOM), Salvador, Brazil, April 5-7th, 2004.

[10] National Institute of Standards and Technology. *Part 224 - Mechanical Product Definition for Process Planning*. Document ISO TC184/WG3 N264(T7), USA, 1996

[11] CAM-I, Deere & Company, *Part Features for Process Planning*, Moline Illinois, CAM-I, 1986.

[12] A. J. Álvares, J. C. E. Ferreira and R. M. Lorenzo, *An Integrated Web-Based CAD/CAPP/CAM System for the Remote Design and Manufacture of Feature-Based Cylindrical Parts*. Journal of Intelligent Manufacturing - Special Issue on Advanced Technologies for Collaborative Manufacturing, Springer, UK, 2007 – accepted for publication.

[13] J. R. King, *Machine-component grouping in production flow analysis: An approach using a rank order clustering algorithm*. International Journal of Production Research, vol. 18, 1980, pp. 213-232.

[14] GE Fanuc, *Application Development - FOCAS1 (Drivers & Programming Libraries)*. 2006, <http://www.geindustrial.com/cwc/products?pnlid=2&id=cnc_mec_39>

[15] A. J. Álvares, G. F. Andriolli, P. R. C. Dutra, M. M. Sousa and J. C. E. Ferreira, *A navigation and path planning system for the Nomad XR4000 mobile robot with remote web monitoring*. ABCM Symposium Series in Mechatronics, vol. 1, 2004, pp. 18-24.

[16] B. Booch, J. Rumbaugh and I. Jacobson, *The Unified Modeling Language User Guide*, Addison-Wesley Object Technology Series, USA, 1998, 512p.

[17] M. Starbek, J. Kušar and A. Brezovar, *Optimal Scheduling of Jobs in FMS*. CIRP Journal of Manufacturing Systems, vol. 32, no. 5, 2003.

[18] S. H. Steiner, *Pre-control and Some Simple Alternatives*. Quality Engineering, vol. 10, 1998, pp. 65-74.