ABCM Symposium Series in Mechatronics - Vol. 2 - pp.547-554

DEVELOPMENT OF A SYSTEM INTERNET-BASED COLLABORATIVE CAD/CAPP/CAM IN A CONTEXT OF E-MANUFACTURING

Alberto J. Álvares
Universidade de Brasília, Departamento de Engenharia Mecânica e Mecatrônica, Grupo de Automação e Controle (GRACO), CEP 70910-900, Brasília, DF, Brazil
alvares@AlvaresTech.com

João Carlos E. Ferreira
Universidade Federal de Santa Catarina, Departamento de Engenharia Mecânica, GRIMA-GRUCON, Caixa Postal 476, CEP 88040-900, Florianópolis, SC, Brazil
jcf@grucon.ufsc.br

Abstract. *This work presents a description of the implementation of the WebMachining methodology and system (http://WebMachining.AlvaresTech.com) developed in a context of e-Mfg and Concurrent Engineering, aiming at integrating CAD/CAPP/CAM for the remote manufacturing of cylindrical components through the Internet. The methodology and its implementation are conceived starting from the modelling paradigm based on synthesis of design features, in order to allow the integration among collaborative design (WebCADbyFeatures), generative process planning (WebCAPP) and manufacturing (WebTurning). The system is implemented in a distributed environment of agents (agents' community), and Knowledge Query and Manipulation Language (KQML) is adopted as the language for message exchange among the design, process planning and manufacturing agents.*

***Keywords**: E-Manufacturing, Internet, CAD/CAPP/CAM, Collaborative Design, Features*

1. Introduction

It is in course a new revolution in the work system adopted in the manufacture companies, migrating from the activities aided by the computer (CAD, CAPP, CAM, CAP, etc), to activities based on e-Work (electronic-work), which characterize the beginning of work in the information era, with intensive use of Information Technology (IT).

It, especially the technology of communication networks and the convergence of technologies based on networks and the Internet, is opening a new domain for constructing the future manufacturing environment called e-Mfg (electronic-Manufacturing), using work methods based on Collaborative e-Work, especially for the activities developed during the product development cycle in integrated and collaborative CAD/CAPP/CAM environments. This will allow the product developers and planners to have easier communication, enabling design sharing during the development of the product, as well as the teleoperation and monitoring of the manufacturing devices.

This paper describes the implementation of the WebMachining system and methodology (http://WebMachining.AlvaresTech.com) developed in a context of e-Mfg, aiming at integrating CAD/CAPP/CAM for the remote manufacture of cylindrical components through the Internet. The methodology and its implementation begins with the modelling based on synthesis of design features, in order to allow the integration among collaborative design (WebCADbyFeatures), generative process planning (WebCAPP) and manufacturing (WebTurning). The system is implemented in a distributed environment of agents (agents' community), and Knowledge Query and Manipulation Language (KQML) is adopted as the language for message exchange among the design, process planning and manufacturing agents.

2. Collaborative CAD and Related Systems

In design engineering practice, many activities associated with the several manufacturing aspects are being considered during the design phase. Feature-based modelling has been used in the integration of engineering activities, from design to manufacturing. Thus, the concept of features has been used in a wide range of applications such as component design and assembly, design for manufacturing, process planning and other countless applications. These applications are migrating to heterogeneous and distributed computer environments to give support to the design and manufacturing processes, which will be distributed both in space and in time.

It should be noted that it is undesirable and frequently unlikely to require that all participants in product development activities use the same hardware and software systems. Consequently, the system components should be modular and communicate with one another through a communication network, for effective collaboration.

Many research efforts have been made in the development of design environments oriented to computers networks, usually called network-centred. Shah et al. (1997) developed an architecture for standardization of communication between the kernel of a geometric modelling system and the applications. Han and Requicha (1998) proposed a similar approach that enables transparent access to several solid modelers.

Smith and Wright (2001) described a distributed manufacturing service called Cybercut (http://cybercut.berkeley.edu), which makes it possible the conception of a prismatic component that will be machined using a CAD/CAM system developed in Java in a context of remote manufacturing.

Shao et al. (2004) described a process-oriented intelligent collaborative product design system based on the Analysis-Synthesis-Evaluation (ASE) design paradigm and the parameterization of product design, using agents.

Hardwick et al. (1996) proposed an infrastructure that allows the collaboration among companies in the design and manufacture of new products. This architecture uses the Internet for information sharing, and the STEP standard for product modelling. Martino et al. (1998) proposed an approach to integrate the design activities with the other manufacturing activities based on features, which supports both feature-based design and feature-recognition.

Collaborative modelling systems typically have a client/server architecture, differing in the functionality and data distribution between clients and servers. A common problem in client/server systems is associated with the conflict between the limitation of the complexity of the client application and the minimization of the network load. A commitment solution can be conceived between the two ends, the so-called thin and fat clients. A pure thin client architecture typically places all the functionality in the server, which sends an image of its user interface to be shown on the client.

On the other hand, a pure fat client offers total interaction and local modelling, maintaining its own local model. Communication with the server is required when it is necessary to synchronize the data modifications in the local model with the other clients.

Lee et al. (1999) presented the architecture of a network-centred modelling system based on features, in a distributed design environment, called NetFeature. This approach combines feature-based modelling techniques with communication and distributed computing technologies in order to support product modelling and cooperative design activities in a computer network.

The WebSpiff system (Bidarra et al, 2001) is based on a client/server architecture consisting of two main components on the server side: (i) The SPIFF Modelling System that supplies all the functionality for feature-based modelling, using the ACIS modelling kernel (Corney e Lim, 2001); (ii) The Session Manager that supplies functionality to start, associate, finish and logout a modelling session, as well as the management of all communication between the SPIFF system and the clients.

Li et al. (2004) and Fuh and Li (2004) mention several distributed and integrated collaborative design systems and Concurrent Engineering, and none of such systems implements collaborative design activities integrated with process planning and remote manufacturing systems via Web for the cylindrical components domain, with symmetrical and asymmetrical features. Most of those systems consider prismatic components, like WebCAD 2000 of the Cybercut system, which does not implement collaborative design.

The development of the WebCADbyFeatures collaborative design system differs from the above systems because it models cylindrical components, based on synthesis of design features (symmetrical and asymmetrical), having as motivation the development of an integrated CAD/CAPP/CAM system that allows the collaborative design through the web, in a context of Concurrent Engineering.

3. Methodology of the WebMachining System

Figure 1 presents part of the IDEF0 model of the proposed system, called WebMachining. The proposed methodology is divided into three basic activities: collaborative product modelling (WebCADbyFeatures), Generative CAPP (WebCAPP) and CAM (WebTurning).

The client, WebCADbyFeatures interface agent (fig. 2) is connected to the neutral feature modeler via Web, and it begins the instantiation of a new component to be modeled from a database, using a library of standardized form features.

Then, the data about the component are sent to the server. Since the component is cylindrical, the user models the component in two dimensions, and may visualize it in 3D through VRML. A database was implemented in MySQL that stores the information on the product modeled by features, containing information associated with the form features, material features, tolerance features and technological features (which include surface treatment, thermal treatment and production data). This combined information allows the mapping of design features into machining features, which is fundamental for process planning.

After completing and validating the model, the designed component is stored and made available to the CAPP module to generate the process plan, and the representation of the linearized process plan is based on STEP-NC (ISO 14649 - Part 12) (ISO, 2003). Then, the NC program is generated for a specific CNC lathe, in the case the Romi Galaxy 15M turning centre (http://video.graco.unb.br).

The communication with the Romi Galaxy turning centre, with CNC Fanuc 18i-Ta (fig. 3) is accomplished through an Ethernet connection (physical and data link layers of the ISO/OSI standard), using the TCP/IP protocols (network and transport layers of the ISO/OSI standard) associated with the application protocol Focas1/Ethernet libraries of Fanuc. Focas1 is an API for the development of applications using a standardized data structure to access 300 CNC functions (http://webdnc.graco. unb.br).

Then, the work order is sent through the Internet to the remote manufacturing system, which provides the necessary resources for producing the component (i.e. NC programs, cutting tools, fixtures and blank).

The teleoperation system of the Romi Galaxy turning centre, called WebTurning, is based on a client/server architecture, composed by two modules (figures 1 and 3): WebCam and WebDNC Servers, interfaced with the programs in a computer under the Linux operating system, connected logically via TCP-IP sockets and Ethernet to the machine tool and the clients, being responsible for capturing images (http://video.graco.unb.br) and supervisory control of the CNC lathe. The clients are interfaced through Java applets and HTML pages.

The WebTurning teleoperation servers are composed by the video and teleoperation servers of the machine, which provide command services, program execution, download and upload of programs, troubleshooting and other functions associated with the DNC1 communication protocol, available in the CNC Fanuc 18i-TA, accomplishing the remote supervision of the machine. All control action is executed locally, as a function of the delay of the TCP/IP protocol.

The video server is responsible for video and image capture (with four cameras), and for its distribution through the TCP/IP protocol. The other servers associated with the teleoperation services work in a bi-directional way, receiving commands through the Internet and sending status data about the machine.

The implementation of the three computer modules of the WebMachining system is presented in IDEF0 and UML diagrams, and for system development the Java, JavaScript, HTML, C and C++ programming languages were used. A case study is presented in this paper, showing the collaborative modelling of a component, the generation of the process plan with alternatives and its linearization associated with a NC program, and finally the teleoperation and monitoring of the CNC turning centre.

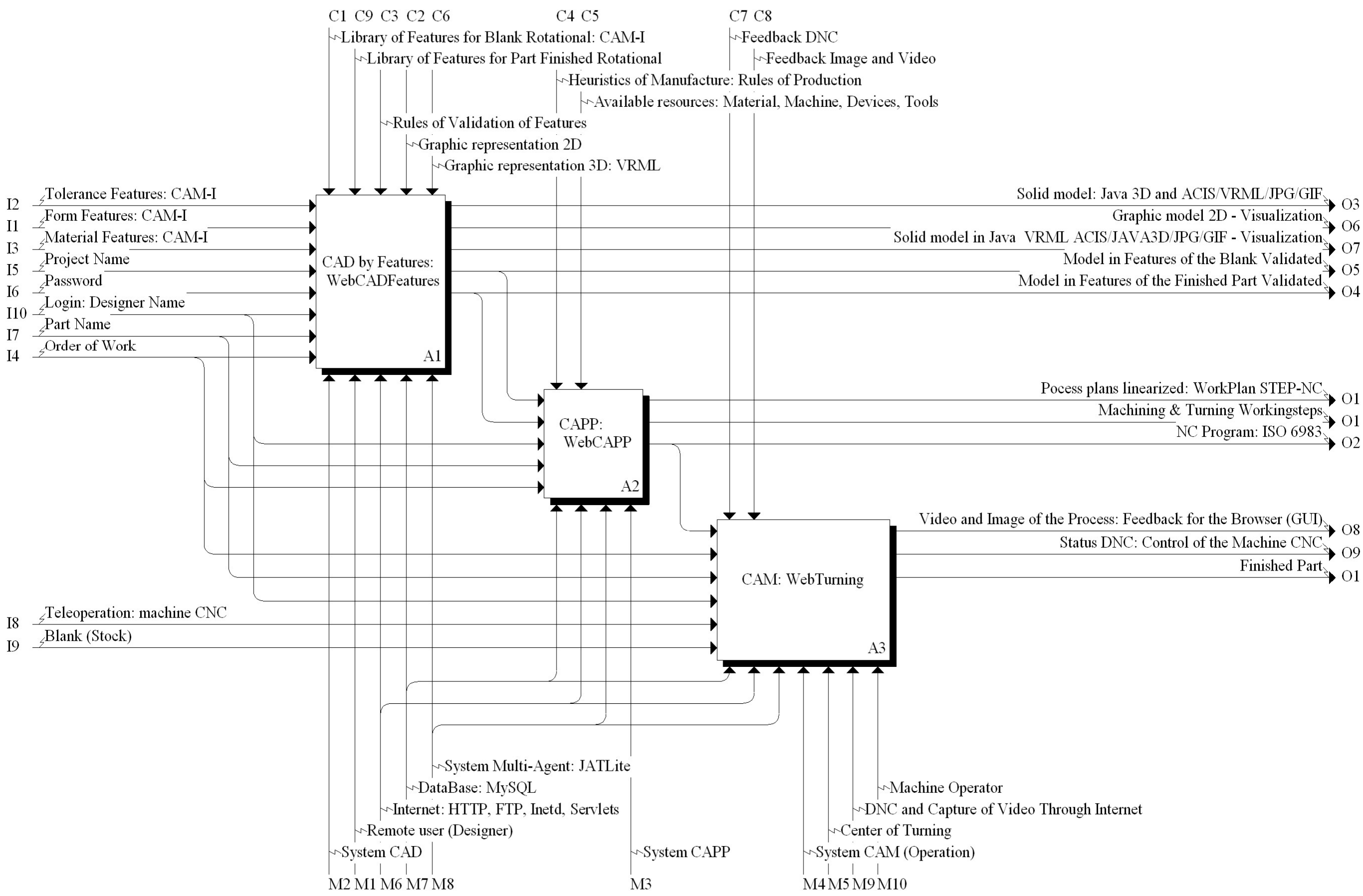

Figure 1. Modeling of the WebMachining system, using the IDEF0 methodology.

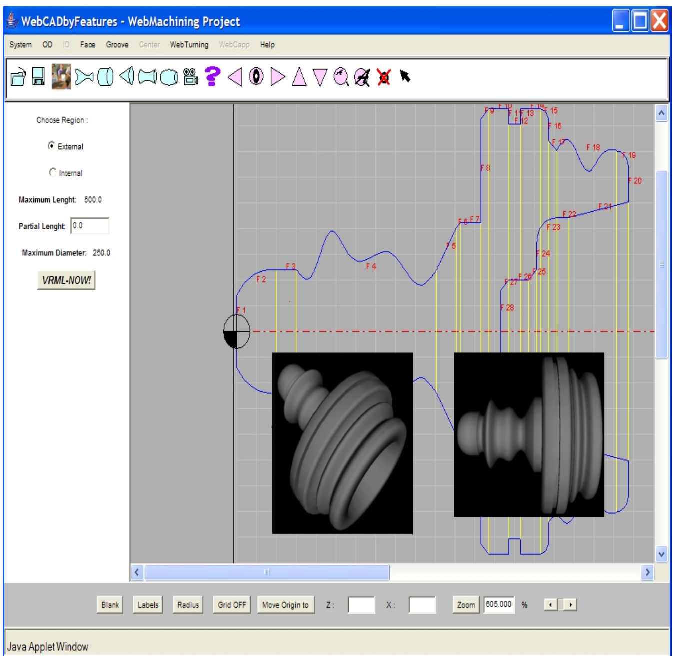

Figure 2. Main window of the WebCADbyFeatures system, showing the profile of a cylindrical part.

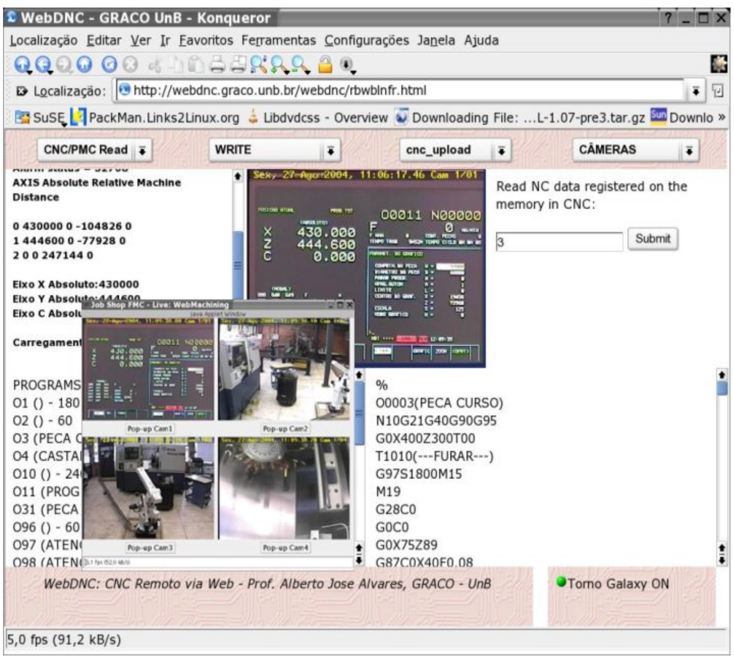

Figure 3. WebTurning: teleoperation and remote monitoring of the turning center.

4. Multi-Agent Architecture for the WebMachining Methodology

The proposed architecture for the multi-agent system (MAS) can be characterized by the agents' behavior as being Deliberative, in the internal organization as being of the Blackboard type, and in the architecture itself as being of the Federated type using the Facilitator approach (Shen et al., 2001).

The use of an architecture based on a multi-agent approach is the most attractive currently, mainly due to the evolution of operating systems, especially Unix for personal computers, and the use of network communication based on TCP/IP in a client/server architecture (Shen e Norrie, 1999). In this way several types of agents working cooperatively and in a distributed way can be used in order to solve many problems associated with CAD/CAPP/CAM integration in a context of a community of agents.

The JATLite (Java Agent Template Lite) software tool is used for implementing the collaborative product design system. JATLite (http://java.stanford.edu/index.html) is a software written in Java that allows the users to create agents that communicate in a robust way over the Internet. JATLite offers a basic infrastructure in which agents that are registered in an Agent Message Router (AMR), usually called facilitator or mediator, use a name and a password, being connected and disconnected through the Internet, sending and receiving messages, transferring files via FTP and usually exchanging information with other agents, through the many computers in which they run. These means are used for developing the management system of the collaborative design sessions, where a design agent interface provides its design for the other participant agents of the product modelling session.

The proposed architecture is composed by six groups of agents (fig. 4): facilitator (1), database manager (2), collaborative design (3 and 4), VRML manager (5), process planning (6) and manufacturing (7, 8, 9 and 10):

1. FACILITATOR AGENT (FA): it performs communication management among the agents, managing the routing of the messages among the agents, system safety and agents registration. It is implemented through the Message Router Agent, which is part of the JATLite architecture. It will be necessary more than one FA due to the amount of agents present in the system, in order to improve the its performance. The AMR is very important in the JATLite environment, because the agents always communicate with other agents through AMR.
2. DATABASE MANAGER AGENT (DMA): this agent performs the interaction with the MySQL database. Any agent that wants some information from the database (through the SQL language) makes a request to DMA, and it sends the answer to the agent that requested the information. The Facilitator Agent accomplishes the routing of the messages among these agents.

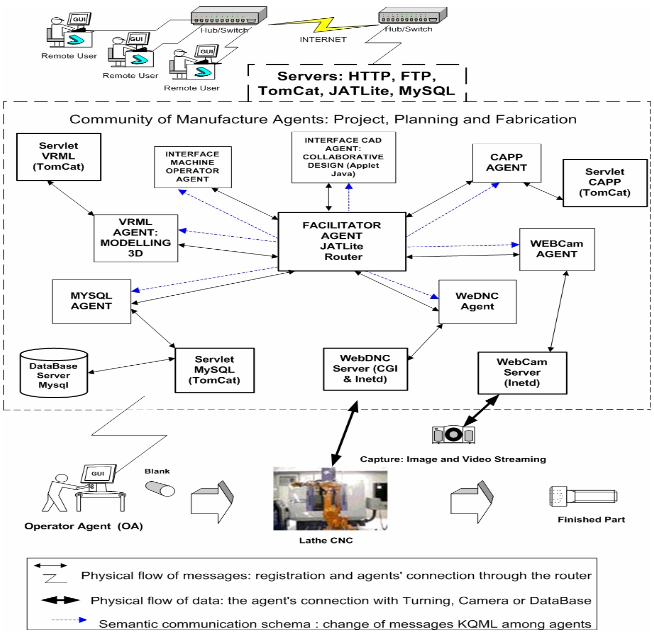

Figure 4. Multi-agent architecture of the WebMachining system.

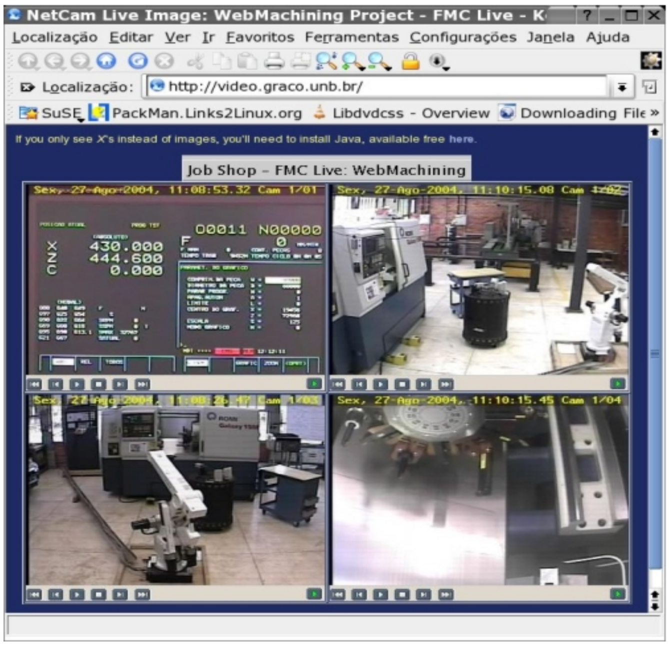

Figure 5. WebCam GUI : capture video (http://video.graco.unb.br).

3. COLLABORATIVE DESIGN (CADIA): this is a GUI for feature-based design, implemented through a Java applet. This GUI is executed by a remote client aiming at defining the model and geometry of the raw material and of the finished component based on features. This agent will communicate with the community of agents through a connection to FA, and this will perform message routing to the correct agent. Messages are sent to the other modules of the system and users, containing the data regarding the design underway (i.e. the product model), which includes information such as: user, component name, design name, etc.; this will allow the identification of the product model that the client is creating. The creation of the component by features and the verification of the feature library, for which it is necessary a connection with the MySQL database, are accomplished directly through PHP (Personal Home Page), in order to improve the execution of the system. The 3D visualization of the product model is managed through this GUI, which communicates with the agent for 3D modelling. Figure 2 shows a prototype of the developed GUI, which is a Java applet, and the 3D visualization of a component through

VRML.

4. REMOTE USER CAM INTERFACE AGENT (WebDNC): every GUI associated with CAM that is executed by a remote client and used to teleoperate the CNC machine, has WebDNC embedded in the interface. This agent communicates with the community of agents through a connection to FA, performing message routing to the corresponding agent.
5. 3D VRML BASED MODELLING AGENT (VRML): it is responsible for 3D modelling using VRML (Virtual Reality Markup Language). It receives messages from CADIA for building 3D component models based on features.
6. PROCESS PLANNING AGENT (WebCAPP): it is responsible for process planning.
7. WebCam AGENT (WebCam): it is responsible for the video and image capture of the teleoperation system, sending the captured images directly to the GUI associated with CAM. It receives messages from FA regarding the user's identification, login and password, to allow the execution of the WebCam server (fig. 5).
8. WebCNC AGENT (WebCNC): it is responsible for the remote control of the CNC machine, receiving commands and sending the machine status to the GUI associated with CAM, i.e. WebDNC. It receives messages from FA regarding the user identification, login and password, filename with the NC program and process planning data (fixtures, tools and raw material), being responsible for implementating the Distributed Numeric Control (DNC) protocol through the Web (fig. 3) via Focas1/DNC1.
9. MACHINE OPERATOR INTERFACE AGENT (MOIA): This agent gives the instructions to the shop-floor operator about fixturing the raw material, tools setup, machine setup, production scheduling, among others.
10. OPERATOR AGENT (OA): this agent corresponds to the machine-tool operator, which receives instructions for fixturing the raw material, tool setup, machine setup, production scheduling of a component and other data associated with process planning and that can only be treated by a human operator.

5. WebCADbyFeatures: Implementation

The inputs for the WebCADbyFeatures system (fig. 5) are the feature model and other necessary information, and it outputs the feature model of the raw material and the finished component, which becomes an input to the CAPP module, which in turn is responsible for generating the process plan and the NC program for the corresponding component (figure 1). The computer system WebMachining is based on the client/server architecture, and the Java, C and C++ programming languages were used.

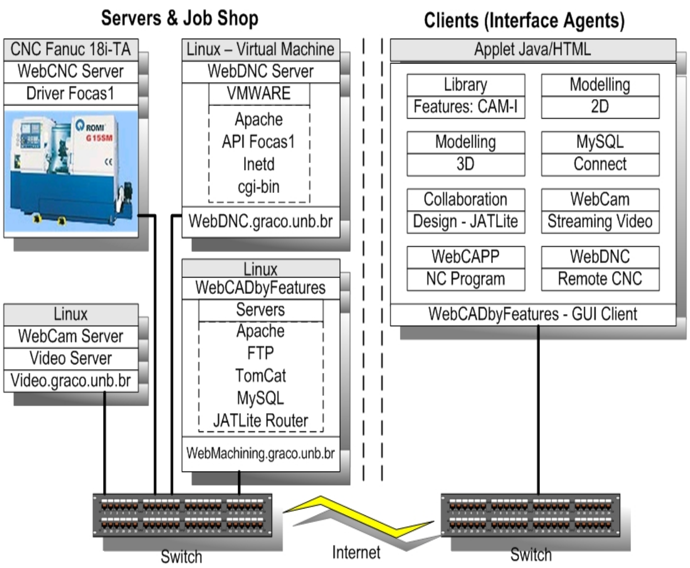

Figure 6. Detailed architecture: WebCADbyFeatures system modules.

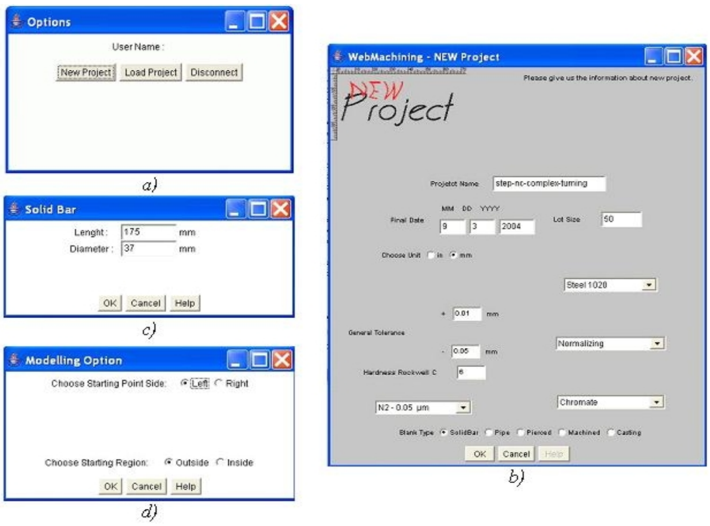

Figure 7. Stages in the design of a part: (a) Options window; (b) Window with data about a new project; (c) Window of data about the raw material (Solid Bar); (d) Window with modeling option

On the client side the GUI is represented by applets, and two servers (VRML and JATLite) are implemented (fig. 6):

- VRML server based on servlets (TomCat): used for generating the 3D model of the component in VRML, from the design feature model of the component;
- JATLite Router/Facilitator server: allows the management of many sessions of collaborative product modelling, performing the coordination of communication between the many WebCADbyFeatures interface agents, as well as the other agents in the system, managing the routing of messages between the agents, system security and agent registration. It is implemented through the Agent Message Router (AMR) of the JATLite architecture. The AMR is very important in the JATLite development environment, because the agents always communicate between each other via AMR, peforming activities such as sending an e-mail message (e.g. via SMTP) or a file with the feature model in 2D (e.g. via FTP).

WebCADbyFeatures allows the creation and manipulation of the feature model for the raw material and finished component in a collaborative way, the storage of that information in a MySQL database, the validation of the model and the visualization of the geometric model in 2D and 3D (via VRML).

It is composed by a GUI that has menus, visualization options, error messages, feature manipulation, communication with the JATLite session manager for collaborative modelling, communication with the database server, communication with VRML server, monitoring of the shop floor (WebCam), teleoperation of the CNC turning centre, among other functions.

The main components of WebCADbyFeatures are: GUI as a Java applet, the feature library, 2D Graphical Interface, Collaborative Design IPLayer Router Client, components for 2D visualization (graphical primitives such as straight lines and arcs) and components for 3D visualization (VRML). The information regarding the features is handled through a database management system.

The modelling of the component begins with the access by the client to the web page for running the CAD Java applet. If the user is registered, an access to the database is made in order to verify the user login and password. Then, the applet is called, downloaded via web, and automatically the local Java machine runs the applet. The AWT standard (Abstract Windowing Toolkit) is used in order to allow a better performance and compatibility with any Java machine version 1.1, which is implemented in a native way in most browsers (Netscape, Iexplorer, among others), without need of a specific plug-in for a certain Java version, facilitating its execution by the user.

The first window to appear shows the initial options (figure 7a). For a non-registered user it is only possible to create a new project. Then, a new window opens up (figure 7b) that gathers the design information. The system guides the user, asking for the relevant information for component modelling and process planning. If the solid bar raw material is chosen, a new window appears (figure 7c), requesting the geometric information about a solid bar, which are its diameter and length.

The last window in this stage (figure 7d) provides the options for selecting the floating zero (left or right) and if the user prefers to begin modeling with the external or internal portion of the part. The default is the modeling from the left-hand side, and beginning with the external features, which is the most common procedure among designers.

Proceeding with collaborative modelling, a drawing window opens up (figures 8 and 9), where the desired component is modeled, and the available form features in the feature library are selected. Initially, the component is modeled using the feature union method, in which features are used as blocks for building the component geometry (like bricks), based on the CAM-I taxonomy (CAM-I, 1986). After finishing the union phase, features are subtracted from the component, including the features associated with the C-axis of the CNC turning center, which include keyways, eccentric holes, radial holes, etc.

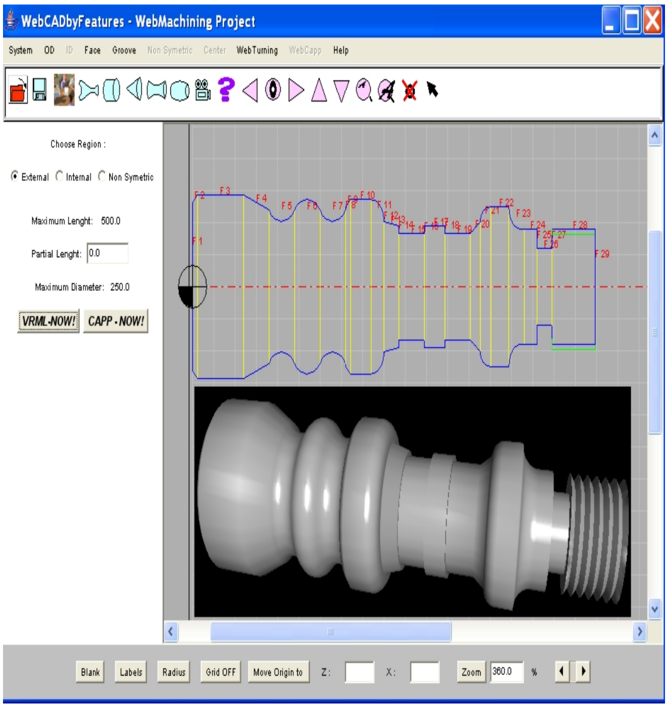

Figure 8. WebCADbyFeatures GUI: 2D and VRML.

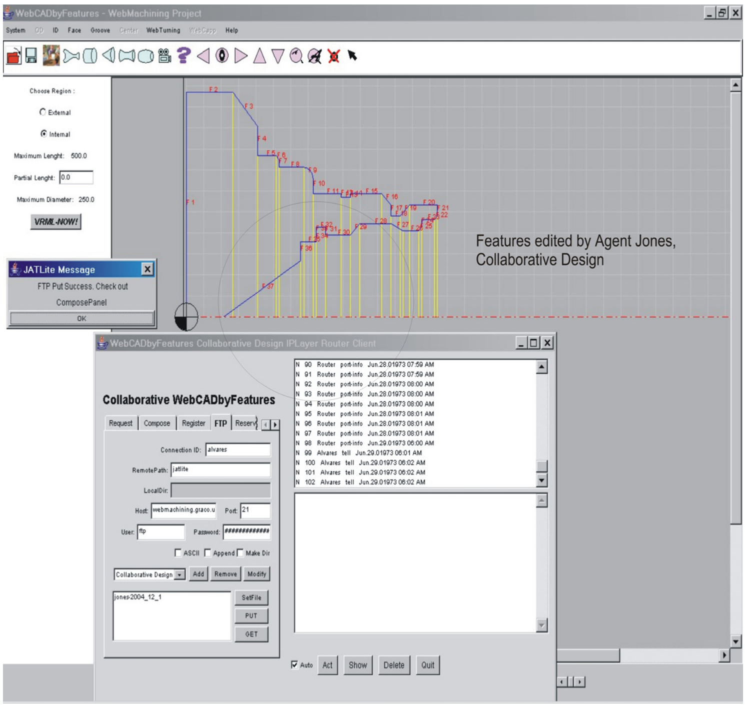

Figure 9. Collaborative modeling between two design agents: Alvares and Jones.

The user has the option of zooming the drawing in 2D, move it on the screen, and also generate the VRML representation at any moment for 3D visualization. When selecting the "VRML-NOW!" button, the component model is sent to the WebMachining server through servlets, which saves the file in the server, and it is sent to the client's browser via FTP, which calls the available VRML plug-in.

The visualization in 3D is made through a plug-in for VRML, previously installed by the user in the browser. When selecting the button "VRML-NOW!" (fig. 8), the component model is sent to the WebMachining server through servlets, which saves the file in the server, and it is sent to the client's browser via FTP. Then, it calls the available VRML plug-in, allowing the visualization of the component. There are the options for saving the geometric model locally in 2D and 3D (.wrl extension) or through features (.ftr extension), and this can be done because the security policy of the local Java machine is changed, allowing reading and writing files to the client computer. The Java machine is configured in a safe way, preventing applets from having access to the local resources of the machine. In figures 2, 8 and 9 an example component is shown in 2D and the 3D solid in VRML.

6. CAPP/CAM System: Implementation

The CAPP/CAM modules can be characterized according to the four dimensions defined by Shah and Mantyla (1995): in the Planning dimension it generates non-linear process plans for cylindrical components machined in CNC lathes, containing definition of operations, fixtures, setup, sequences of operations, cutting tools, cutting parameters, NC code generation and cost estimation, in other words, micro-planning; in the Planning dimension related to time, it can be applied both on the tactical level and on the operational level of a Manufacturing Management System; in the Planning Method dimension, it is a generative CAPP system; in the Planning Depth dimension, it is defined as dynamic, in other words, plans can be changed dynamically during manufacturing depending on the dynamic characteristics of the manufacturing system.

The CAPP system, called WebCAPP (http://WebMachining.AlvaresTech.com), is composed by ten activities, based on STEP-NC (figs. 8, 10 and 11):

1. mapping of design features into manufacturing features (machining): it accomplishes the mapping of design features into manufacturing features, including machining operations such as internal and external cylindrical turning, facing, boring, parting-off, threading, etc.
2. determination of the machining operations with alternatives, associated with the machining features (workingsteps): it selects the machining processes for the identified features, and it also considers the constraints associated with the dimensions, tolerances, material of the component, among others.

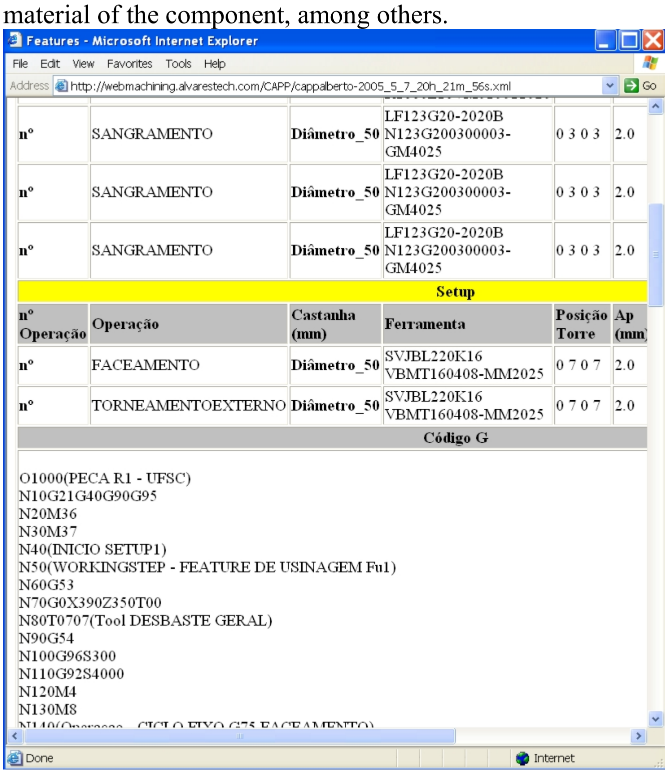

Figure 10. WebCAPP: Process planning and NC program.

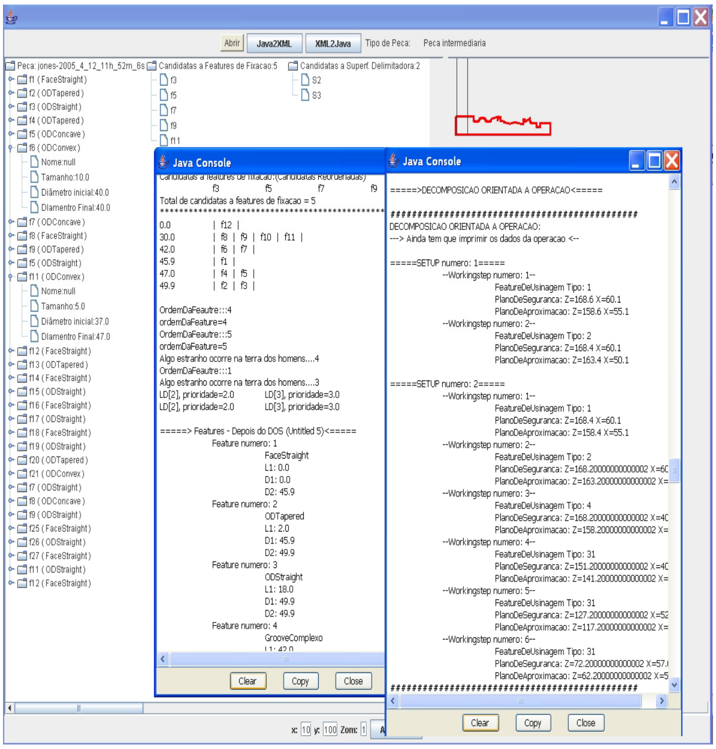

Figure 11. WebCAPP Activities and workingsteps.

3. determination of the machining sequence with alternatives (nonlinear workplan): it determines the machining sequence with alternatives and setup for fixturing the component.
4. strategies for generatinon tool paths: it determines the strategies for generating tool paths based on STEP-NC.
5. determination of the cutting tools (inserts and supports): it selects the cutting tool considering the machine-tool, the component material and geometry, tool life, etc.
6. determination of the time model and calculation of the time and cost standards for each workingstep.
7. determination of the technological conditions of optimized machining using genetic algorithms: it determines the cutting conditions considering the tool parameters and material, and subject to the following constraints: the tool life criteria used, machine power, machine capacity, among others.
8. linearization of the nonlinear process plan using genetic algorithms.

9. generation of the NC program (ISO 6983): the NC program (ISO 6983) is generated in such a way that collisions are avoided.
10. generation of reports and process plan: it sets up the document regarding the process plan, including information on alternative plans and cost estimates (fig. 10).

7. Conclusion

In this article the implementation of a system for of collaborative product development was described, based on modelling for synthesis of design features for cylindrical components (symmetrical and asymmetrical features) through Internet. This software, called WebCADbyFeatures, is one of the modules of the WebMachining system, which proposes a framework for CAD/CAPP/CAM integration.

A multi-agent system that enables collaborative design was developed, being implemented in a client/server architecture, composed by servers, HTML pages and Java applets, which allow the remote user to carry out the collaborative modelling of the component in 2D and its visualization in 2D and 3D, through VRML.

Some of the characteristics of the WebCADbyFeatures system are pointed out below:

- ✓ It uses multi-platform servers based on Servlets, JATLite, HTTP, MySQL and FTP; implemented in Java, HTML, Javascript and PHP. The servers were developed in the Linux platform;
- ✓ The client is based on Java applets, using AWT, not being necessary the Java plug-in, enabling total compatibility with the browsers; the user will just have to install a plug-in for visualization of the component in VRML;
- ✓ Multi-user and multi-task system, based on threads, both on the server side and on the client side;
- ✓ Remote communication among people, eliminating the geographical and temporal barriers for product development, allowing the implementation of Concurrent Engineering;
- ✓ The user can model parts with splines for "general_revolution" type features, and he/she can use eccentric features (C-axis) in the design, moving beyond STEP NC-Part 12 (ISO 14649, 2003).
- ✓ Speed and safety in the communication among the agents;

The WebMachining system can be accessed via web through the following URL: http://WebMachining.AlvaresTech.com. Many of its modules are available for use, thus providing a remote laboratory and a machining rapid prototyping system, in a context of e-Mfg (TeleManufacturing), allowing collaborative modeling, process planning and remote machining through web

8. References

Bidarra, R., Van den Berg, E., Bronsvoort, W. F., 2001, "Collaborative Modeling with Features", Proceedings of DET'01, ASME Design Engineering Technical Conferences, Pittsburgh, USA.

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

Corney J., Lim T., 2001, "3D Modeling with ACIS", Saxe-Coburg Publications, 2a Edition.

Fuh, J.Y.H., Li. W. D., 2004, “Advances in collaborative CAD: the-state-of-the art”, Computer-Aided Design xx 1–11.

Han, J. H., Requicha, A. A. G., 1998, "Modeler-independent Feature Recognition in a Distributed Environment". Computer-Aided Design, Vol 30, No. 6, pp 453-463.

Hardwick, M., Spooner, D. L., Rando, T, Morrir, K. C., 1996, "Sharing Manufavturing Information in Virtual Enterprises", Communications of the ACM, Vol 39, No. 2, pp 46-54.

ISO 14649, 2003, Data model for Computerized Numerical Controlers - Part 12: Proces Data for Turning, Draft International Standad, V09.

Lee, J. Y., Han, S. B., Kim, H., Park, S. B., 1999, "Network-centric Feature-based Modeling", Pacific Graphics.

Li WD, Ong SK, Fuh JYH, Wong YS, Lu YQ, Nee AYC., 2004, “Featurebased design in a collaborative and distributed environment”, Computer-Aided Design,Vol 36, No. 9, pp 775–97.

Martino, T. D., Falcidieno, B., Hasinger, S., 1998 "Design and Engineering Process Integration Through a Multiple View Intermediate Modeler in a Distributed Object-oriented System Environment", Computer-Aided Design, Vol 30, No. 6, pp 437-452.

Shah, J. J., Dedhia H., Pherwani V., Solkhan S., 1997, "Dynamic Interfacing of Applications to Geometric Modeling Services Via Modeler Neutral Protocol", Computer-Aided Design, Vol 29, pp 811-824.

Shah, J. J., Mäntylä, M., 1995, "Parametric and Feature-Based CAD/CAM: Concepts, Techniques, and Applications", John Wiley & Sons, New York.

Shao X., Li Y., Li P., Liu Q, 2004, “Design and implementation of a process-oriented intelligent collaborative product design system”, Computers in Industry, Vol 53, No. 2, February, pp 205-229.

Shen, W., Norrie D.N., Barthés J.P.A, 2001, “Multi-Agent Systems for Concurrent Intelligent Design and Manufacturing”, Taylor & Francis, New York.

Shen W. , Norrie D.H., 1999, “Agent-Based Systems for Intelligent Manufacturing: A State-of-the-Art Survey” Knowledge and Information Systems, an International Journal, Vol 1, No. 2, pp 129-156.

Smith, C. S., Wright, P. K., 2001, "Cybercut: An Internet-based CAD/CAM System", ASME Journal of Computing and Information Science in Engineering, Vol. 1, No. 1, pp 1-33.