CAD/CAPP/CAM INTEGRATION METHODOLOGY FOR THE REMOTE MANUFACTURE OF CYLINDRICAL PARTS THOUGH THE WEB

Alberto José Álvares
Universidade de Brasília, Departamento de Engenharia Mecânica e Mecatrônica, Graco - Grupo de Automação e Controle, CEP 70910-900, Brasília, DF, alvares@AlvaresTech.com.

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

Abstract. *This paper describes a multiagent architecture for CAD/CAPP/CAM integration for the remote manufacturing of cylindrical parts using Internet, especially through the protocols associated with the World Wide Web. This methodology specifies the modules of the system, and how these modules interact with each other. The methodology is conceived starting from the modeling paradigm based on the synthesis of design features, in order to allow the integration of the design activities (CAD), process planning (CAPP) and manufacturing (CAM Planning and Execution). The procedure begins with the modeling of a part by features in a context of remote manufacturing using the Web as the communication means, in a client-server computer model. The system is conceived in a distributed environment of agents of interoperable pieces of software called Community of Manufacturing Agents, whose architecture is stratified in three levels: Design, Process Planning and Manufacturing. The flow of information among the three levels is accomplished through information encapsulation, being adopted the Knowledge Query and Manipulation Language (KQML) as the language pattern of messages among the agents.*

Keywords. *Telemanufacturing, Features, CAPP, CAM, Internet.*

1. Introduction

The integration among the stages of the productive cycle is one of the means to reduce production costs and times. According to Shah and Mantyla (1995), product modelling is the central point for achieving such integration. In an integrated production system, the product model, defined in the CAD module, should be available for other modules (i.e. CAE, CAPP, CAM, CAQ, etc) so that these can accomplish their objectives. These modules should also be capable to send feedback information to the CAD module, so that the necessary alterations in the part can be made in the design stage (for instance, due to problems detected in manufacturing). The use of features as the means to convey information for product modelling is the road to reach this integration (Tönshoff et al., 1994; Salomons et al. 1993).

Information Technology, especially network communication technology and the Internet, is opening a new domain for building CAD/CAPP/CAM environments of the future (Lee et al., 1999), and they can be considered as potential candidates to support the development of integrated systems. This is a new paradigm for computer systems based on a global environment, network-centered and spatially distributed. This will allow that product designers are provided with easier communication means, making it possible design sharing and cooperative design during product development. With the growth in popularity of the navigators based on WWW, it is becoming more evident that the network-oriented environment will turn into a new paradigm for product development.

Erickson (1988) revised more than 127 CAPP systems and associated designs. He observed that the incompatibility in the programs, in the hardware and in the different product representations, resources and process plans, inhibited the development of a general purpose integrated system. Erickson suggests that the future CAPP systems should be modular, easy to integrate, friendly and easy to maintain. Shunmugam et al. (2002) highlighted that few CAPP systems found in the literature take into consideration optimization aspects of the sequence of operations or suggest alternative sequences of operations or process plans. Giusti et al. (1989) present the CAPP system KAPLAN, whose approach is based on knowledge for process planning rotational parts. Cho et al. (1991) describe the development of an integrated process planning and monitoring system for turning operations. Rich et al. (1997) describe a CAPP system for rotational parts based on features and 2D profiles. Kruth and Detand (1992) describe a CAPP system for generating non-linear process plans.

This work describes a multi-agent architecture associated with a methodology for integrating CAD/CAPP/CAM for the remote manufacturing of rotational parts through the Internet, especially the protocols associated with the World Wide Web (WWW). This methodology specifies the modules of the system and how these modules interact amongst themselves, being presented in the form of IDEF0 diagrams.

The proposed architecture for the CAD module is based on the Feature-based Design paradigm. 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 rotational parts 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 Method of Planning dimension, it is a generative CAPP system; in the Depth of Planning 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.

2. Cooperative CAD and related work

In practical engineering design, more and more activities associated with several manufacturing aspects are being considered during the design phase. Part modelling based on features is being considered as a new paradigm for integrating engineering activities, from design to manufacturing. So, the concept of features has been used in a wide range of applications such as part design and assembly, design for manufacturing, process planning and other countless applications. These applications are migrating towards heterogeneous and distributed computer environments to support the design and manufacturing process that will be distributed both in the space and in time. It should be noticed that it is undesirable and frequently unlikely to request that all the participants in the product design and manufacturing activities use the same hardware and software system. Then the components should be modular and communicate with the others through a communication network for effective collaboration.

Many research efforts have been made in the development of design environments for computer networks, usually called network-centered. Shah et al. (1997) developed an architecture for standardization of the communication between the geometric modelling kernel and the applications. Han and Requicha (1998) propose a similar approach that enables the transparent access for several solid modellers. Smith and Wright (2001) describe a distributed manufacturing service called Cybercut, developed at the University of Berkeley (http://cybercut.berkeley.edu) that enables the conception of a prismatic part that will be machined through a CAD/CAM system developed in Java in a context of remote manufacturing (Álvares et al., 2002).

Cybercut is a system that attempts to solve the Design and Manufacturing problem based on the web (Smith and Wright, 2001). It consists of three components: a CAD software written in Java using applets through web pages; a CAPP system that accesses a knowledge base containing information about tools and fixtures; an open architecture CNC that can receive design and planning information in a high level language and to perform machining in a Haas VF-1 milling machine.

Having access to the CAD interface of Cybercut, starting from the Internet, any engineer with a WWW browser becomes a potential user of this tool for on-line rapid prototyping. The Graphic User Interface (GUI) enables the user to load a CAD file in a specific format (of universal data exchange), for the Cybercut server, which will execute the process planning, and will generate the appropriate NC code the CNC milling machine. The part can be machined and then be sent to the designer. The engineer can have a functional product prototype within a few days for a fraction of the cost that the full production would demand. Cybercut works with the milling and drilling processes, however the system does not allow the teleoperation of the CNC machine (Álvares et al., 2002).

Hardwick et al. (1996) propose an infrastructure that allows the collaboration among companies in the design and manufacturing of new products. This architecture integrates WWW for sharing information in the Internet using the STEP standard for product modelling. Martino et al. (1998) propose an approach to integrate the design activities with the other manufacturing and production activities based on features, which supports Design by Features and Feature Recognition. However, although these works are conceptual in essence, they do not present a well structured representation nor detailed algorithms. For instance, these works do not define how to distribute the necessary computational processing among the distributed components, and how to model the communication among the components to minimize the delay of the network. If the actions to exchange data among the applications could not be triggered adequately, this will lead to a critical problem for the distributed computation. Therefore, it is crucial the development of an integrated system, network-centered and with an architecture based on agents for distributed and cooperative design and manufacturing.

Systems for cooperative modelling typically have a client/server architecture, differing in the functionality and data distribution between clients and servers. A problem in the client/server systems is associated with the conflict between the limitation of complexity of the client´s application and the minimization of the network load. In a context of cooperative modelling, the client´s complexity is determined mainly by the ease in modelling and interactivity implemented in the client, while the network load is determined mostly by the type and size of the data model being transferred from/to the clients and on the available band width.

A commitment solution can be conceived between thin and fat clients. A pure thin client architecture typically puts all the functionality in the server, which sends an image of its user interface to be shown in the client.

The other end, a pure fat client offers total ease of interaction and local modelling, maintaining its own model locally. Communication with the server is required when there is the need to synchronize the modifications to the local model data with the other clients.

Lee et al. (1999) present an architecture of a network-centered modelling system based on features, in an distributed design environment, called NetFeature System. This approach combines modelling techniques based on features with communication technology and distributed computing to support activities of product modelling and cooperative design in a computer network. The approach is implemented in an client/server architecture, in which the clients accomplish modelling based on features through the Web. The server creates the model of neutral features and the other applications will communicate with one another using a standard communication protocol to access the remote objects. The system was conceived in order to provide a good balance among the available functionalities in the client side and the available band width in the Internet.

The processing on the client side is important when the application is based on the Web. This means that the server gives the client some responsibility for processing the data, i.e. the client should have more functionalities than being

simply a front-end without local processing and usually called thin clients. A standard communication protocol based on CORBA (Common Object Request Broker Architecture) was used in the NetFeature System.

WebSpiff (Bidarra et al., 2001) is based on a client/server architecture consisting of two main components on the side of the server: the Modelling System SPIFF that supplies all the modelling functionality based on features, using the kernel ACIS (Corney and Lim, 2001); Session Manager that supplies functionality to begin, associate, leave and close a modelling session, as well as management of every communication between the SPIFF system and the clients.

The elements of the WebSpiff portal (http://www.webspiff.org) provide initial access to a WebSpiff session for a new client, that includes a web server where the model data are made available for download by the other clients that are developing the cooperative design. The clients execute operations locally, associated with the visualization, interaction with the model through features, high level semantic messages, specification of modelling operations, as well as the updates to the clients' data are sent through the network. The server coordinates the collaboration session, and it keeps a model of centralized product and it provides all of the functionalities that are not able to be implemented in the client. An important advantage of this architecture is that there is just a centralized model of the product, and so inconsistencies among multiple versions of the same model are avoided.

3. Methodology WebMachining (http://WebMachining.AlvaresTech.com)

The proposed methodology is conceived starting with the modelling paradigm based on synthesis of design features, in order to allow the integration of the design activities (CAD), process planning (CAPP) and manufacturing (CAM Planning and CAM Execution) using as reference the Model of Manufacturing Features defined by AP 224. The procedure begins with the modelling of a part by features in a context of remote manufacturing using the Web as means of communication, in a client/server computer model.

The client is connected to the Neutral Features Modeller through the Web and he begins instancing a new part to be modelled starting with a database, using the library of standardized features made available by the system. After concluding and validating the model, the created part is stored and made available for the other modules of the system.

The system is conceived in a distributed environment of software agents called Community of Interoperable Agents of Manufacturing, and its architecture has three levels: Design, Process Planning and Manufacturing. In the upper level there is a group of Design Agents, which act as a feature-based CAD tool, and it also allows the users to connect to the lower levels.

The intermediate level consists of Planning Agents that incorporate the characteristics of a generative CAPP using methods of knowledge representation based on artificial intelligence, mainly expert systems, fuzzy logic, neural networks and genetic algorithms. This group of agents interpret the design definitions, geometric modelling accomplished by the user, and it determines how to manufacture the part using the Manufacturing Agent, which is at the lower level.

The Manufacturing Agent also allows teleoperation of the CNC machine through the Internet. The information on manufacturing resources (machines, tools, material information, fixtures, etc) is available through a relational database that is accessed by the several agents in the system. The information flow among the three levels in the architecture is accomplished through information encapsulation, being adopted Knowledge Query and Manipulation Language (KQML) (http://www.cs.umbc.edu/kqml) as a standard language of messages among the agents.

The messages are exchanged using direct socket connections (Álvares and Romariz, 2002) among the agents. The proposed methodology uses the Internet and communication technology to offer a new paradigm for the development of the future integrated CAD/CAPP/CAM environments.

This environment is globalized, network-centered and spatially distributed, having as front-end with the CAD/CAPP/CAM system navigators based on the Web, the Java language and VRML (Virtual Reality Modeling Language), allowing the independence of the user's computer platform.

The originality of this proposal is associated with the methodological contribution for the development of telemanufacturing systems through Internet starting with the conception of a product/part up to its production, using the Internet as a means of connecting among the several advanced manufacturing technologies used.

4. Multi-Agent Architecture

The use of an architecture based on Multi-Agents Systems (MAS) is one the most attractive nowadays, mainly due to the evolution of computer systems, especially of Unix for personal computers, and the use of communication networks based on TCP/IP in an client/server architecture (Shen and Norrie, 1999). Thus, several types of agents working cooperatively can be used to solve several problems associated with CAD/CAPP/CAM integration in a context of an agent community. For instance, a relational database management system such as MySQL (www.mysql.com) can be used to share information on the available manufacturing resources (i.e. machines, tools, materials, fixtures, etc), and also to have the agents to help solve the process planning problem. Agents can be implemented using several approaches in its decision logic: expert systems based on production rules, fuzzy logic, decision tables, etc.

Figure 1 introduces the conceived Manufacturing Agents Community (MAC), which is stratified in three levels: in the upper level there is a group of Design Agents, which act as CAD tools for the users that are connected on the lower levels. The intermediate level consists of Planning Agents, which interpret the design definitions, geometric modelling, accomplished by the user, and it determines how the part will be manufactured using the selected Manufacturing

Agents, which are on the lower level. A transparent interface is used among the design, planning and manufacturing. The information flow among the three levels is accomplished through information encapsulation, being adopted the Knowledge Query and Manipulation Language (KQML) as the pattern language of messages among the agents, that contains information on which agents to send the message, where they are, how to interpret the message received by the addressee. The messages are changed using direct socket connections among the agents.

The JATLite (Java Agent Template Lite - http://java.stanford.edu/index.html) software tool is used. JATLite is a package of programs written in Java that allows users to create software agents that communicate in a robust manner in the Internet. JATLite offers a basic infrastructure in which agents registered to an Agent Message Router (AMR) (usually called facilitator or mediator) use a name and a password, can: be connected or disconnected to the Internet, send and receive messages, transfer files via FTP, and usually exchange information with other agents, across the several computers where they are being executed.

The proposed architecture is constituted by five groups of agents: a facilitator (1), a database manager (2), two design agents (3 and 4), seven process planning agents (5, 6, 7, 8, 9, 10 and 11) and five manufacturing agents (12, 13, 14, 15 and 16), according to following description (figure 1):

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 function. 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 (SQL language) makes a request to DMA, and this sends the answer to the agent that requested the information. The Facilitator Agent accomplishes the routing of the messages among these agents.

3. CAD USER INTERFACE AGENT (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 part and of the finished part based on features. Also, it possesses an embedded CADIA to the interface. This agent (CADIA) will communicate with the community of agents through a connection to AF, and this will perfmorm message routing to the correct agent. Messages are sent to the other modules of the system, communicating the data regarding the design underway (i.e. the product model) containing information such as: user, part name, design name, etc.; this will allow the identification of the product model that the client is creating. The connection with the MySQL database is accomplished directly with the PHP (Personal Home Page Kit) mechanism, in order to improve the functioning of the system, not using GUI for this purpose. In other words, the creation of the part by features and the verification of the feature library, are made through PHP. The 3D visualization of the product model is managed through CADIA, which communicates with the agent for 3D modelling. Figure 2 shows a prototype of the developed GUI, applet Java, and the 3D visualization of a part through VRML.

4. 3D ACIS BASED MODELLING AGENT (ACISA): it is responsible for 3D modelling using the ACIS kernel, transforming the SAT (Standard ACIS Text) format into VRML (Virtual Reality Markup Language). It receives messages from CADIA for building 3D part models based on features.

5. MANUFACTURE FEATURE MAPPING AGENT (MFMA): 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. It receives information from CADIA and consults the database through DMA.

6. MACHINING OPERATIONS SELECTION AGENT (MOSA): it selects the machining processes for the features identified by MFMA, and it also considers the constraints associated with the dimensions, tolerances, material of the part, among others. It receives information from MFMA and consults the database through DMA.

7. SETUP AND MACHINING SEQUENCE SELECTION AGENT (SMSSA): it determines the machining sequence and setup for fixturing the part. It receives information from MOSA and checks the database through DMA.

8. TOOL SELECTION AGENT (TSA): it selects the cutting tool, considering the machine-tool, the type of material of the pair part/tool, dimensions and tool geometry, tool life, etc. This agent receives information from SMSSA and checks the database through DMA.

9. CUTTING CONDITIONS AGENT (CCA): 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. It receives information from TSA and checks the database through DMA.

10. NC PROGRAM OUTPUT AGENT (NCPA): it determines the tool path considering the feature-based product

model. It considers the axes of displacement of the tools to avoid collisions, generating the NC program (ISO 6983). It receives information from SMSSA, TSA and DMA.
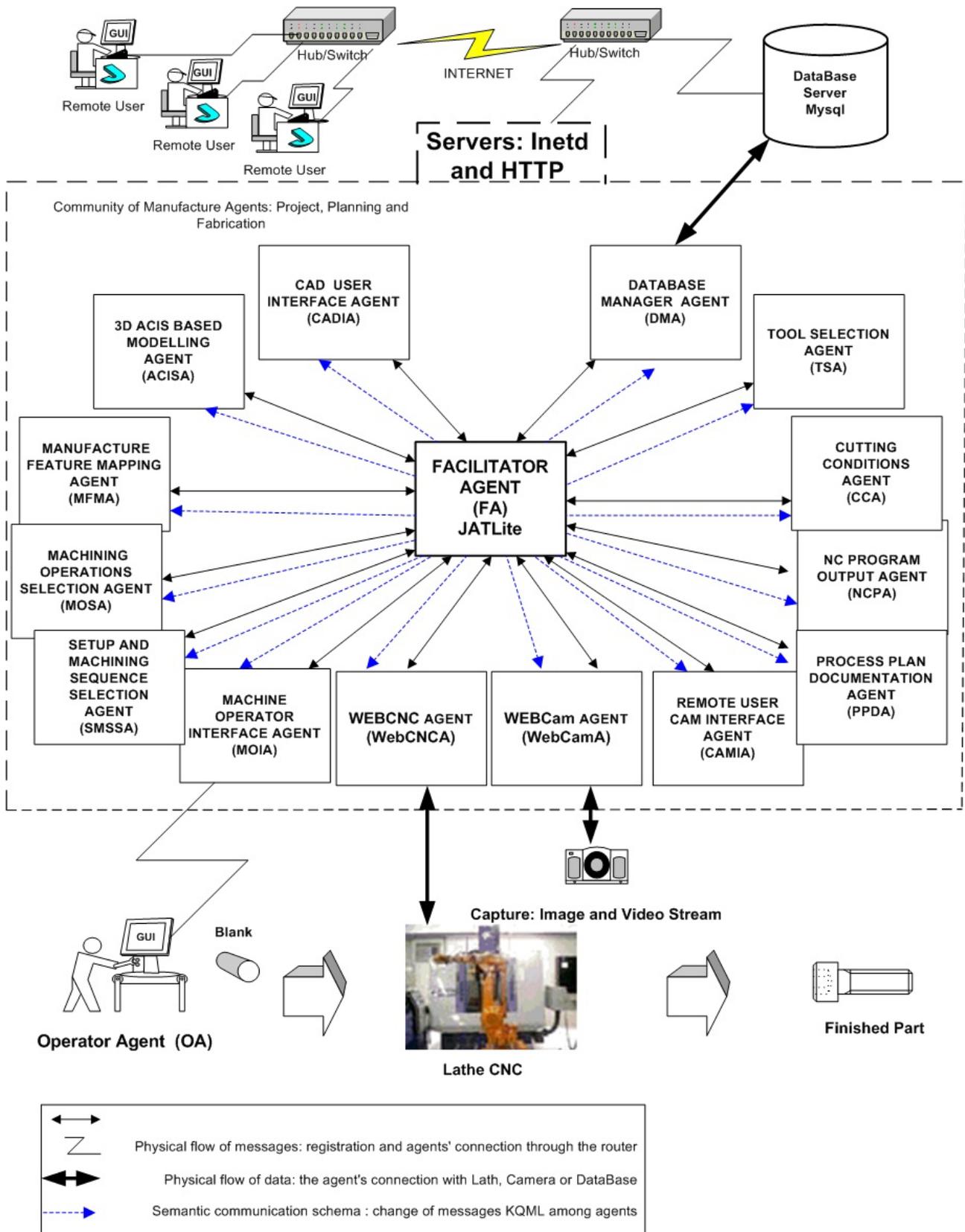

Figure 1. Multi-agent architecture of the WebMachining system.

11. PROCESS PLAN DOCUMENTATION AGENT (PPDA): it sets up the document regarding the process plan including information on alternative plans and cost estimates. PPDA communicates with the other process planning

agents and DMA.

12. REMOTE USER CAM INTERFACE AGENT (CAMIA): every GUI associated with CAM that executed by a remote client and used to teleoperate the CNC machine, possesses CAMIA embedded in the interface. This agent communicates with the community of agents through a connection to FA, performing message routing for the corresponding agent (Figure 1).
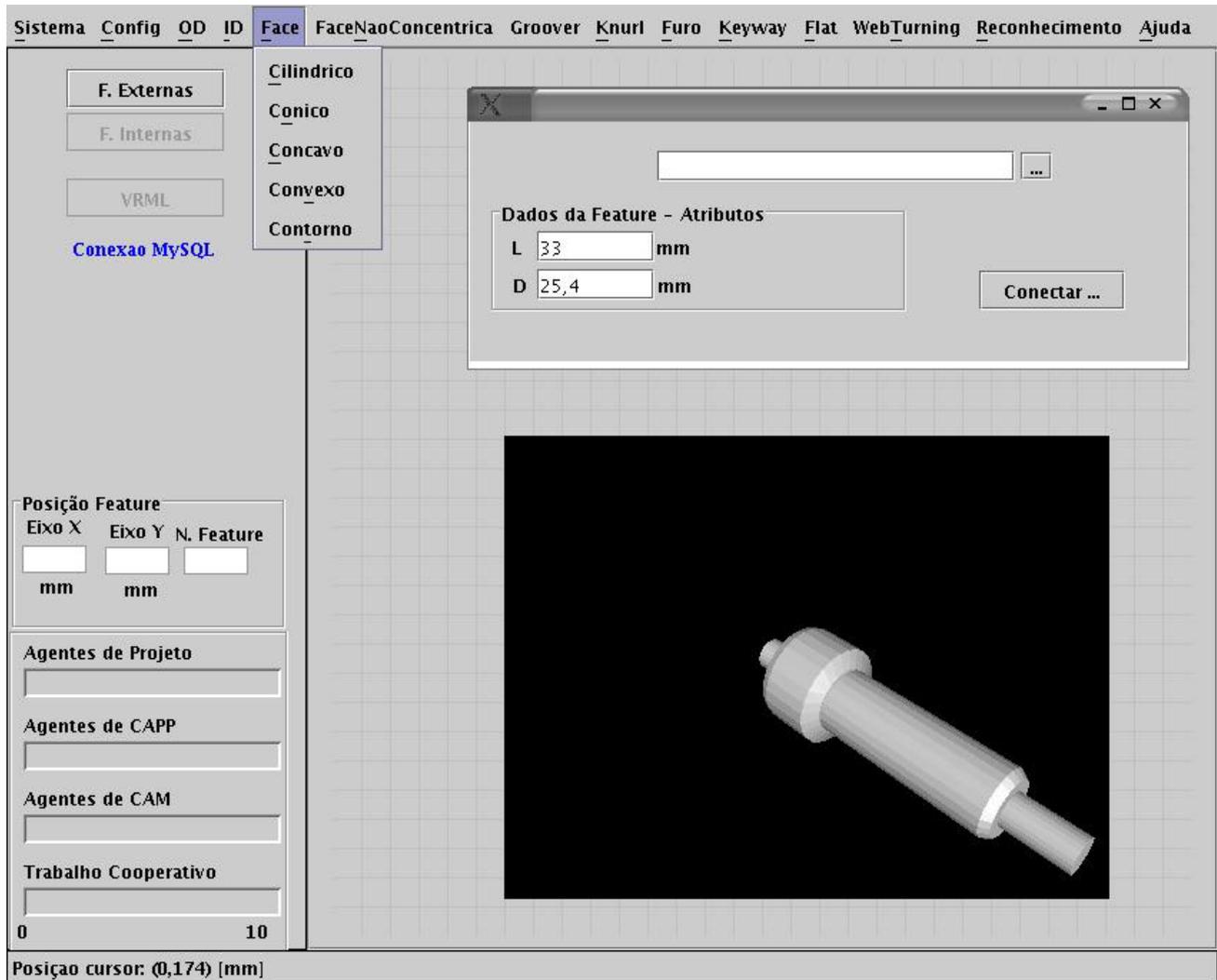

Figure 2. CAD User Interface Agent (CADIA): Java GUI and part visualization in VRML.

13. WebCam AGENT (WebCamA): 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.

14. WebCNC AGENT (WebCNCA): it is responsible for the remote control of the CNC machine, receiving commands and sends the machine status to the GUI associated with CAM. It receives messages from AF regarding the user's identification, login and password, name of the file with the NC program and process planning data (fixtures, tools and raw part), being responsible for the implementation of the Distributed Numeric Control (DNC) protocol through the Web.

15. MACHINE OPERATOR INTERFACE AGENT (MOIA): this graphic interface (GUI) instructs the operator on the shop-floor, being implemented through a Java applet. This operator interface agent (MOIA) gives the instructions to the operator about fixturing the raw part, tools setup, machine preparation, production scheduling, among others.

16. OPERATOR AGENT (OA): this agent corresponds to the machine-tool operator, who receives instructions for fixturing the raw part, tool setup, machine setup, production scheduling of a part and other data associated with process planning and that can only be treated by a human operator.

5. WebMachining System: IDEF0 and IDEF1X Modelling

The proposed integrated CAD/CAPP/CAM system, called WebMachining, was modeled through the IDEF0 methodology (Álvares and Ferreira, 2003). Figure 3 shows the three levels of the system associated with the software agents referred to as Community of Interoperable Manufacturing Agents, with the following activities: CAD by Features, CAPP and CAM.
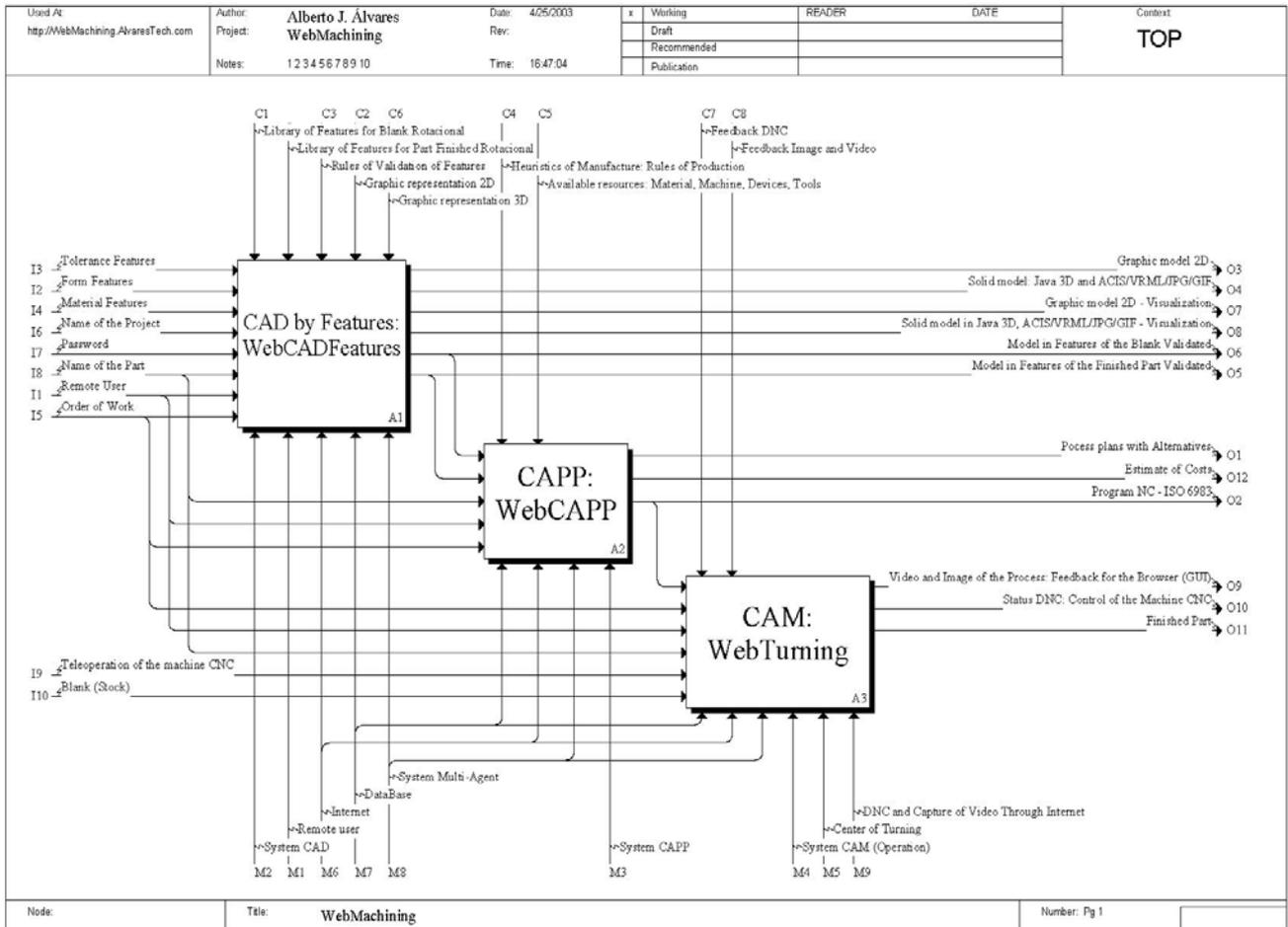

Figure 3. Basic architecture of the system (IDEF0).

The CAD by Features has as data input the feature-based model and other important information, and as output the feature model of the raw part and of the finished part, which is an input for CAPP (figure 4). The output of CAPP is the process plan and the NC program. CAM has as input the data regarding the teleoperation of the CNC machine and the NC program (DNC and video/image of the process), and as output machine control and part manufacture.

The information model was conceived through the IDEF1X methodological approach (relational database). All the data regarding the information model are contained in this diagram. The information model describes each subsystem in terms of entities/objects, attributes and relationships. Starting with this IDEF1X model, the physical model of the database was generated for the database management system MySQL (http://WebMachining.AlvaresTech.com/qualificacao/qualificacao_terceira_versao.pdf).

The information model is divided in domains (Chep et al., 1998) associated with the feature database (Form Features, Tolerance Features, Manufacturing Features and Materials Features) and the machining technology database (Machine-tool Library, Cutting Tools Library, Machining Library and Fixture Library). The feature database links with the product model, while the machining database links with the resource model. These databases were physically implemented through a single database called WebMachining, in MySQL, being composed of eighty two tables.

The remote user connects to the system creating a new design that will be composed of data regarding the feature modelling of a finished part and of a raw part, besides technological information, dimensional and geometric tolerances, surface finish, production data and the user data. Every design will be composed of this group of data for raw and finished part, and they are defined in several domains. A user may have several conceived and stored designs in the system database. Each user has a single identification in the system. More details about the IDEF models can be obtained in Álvares and Ferreira (2003).

C1
Heuristics of Manufacture: Rules of Production

C2
Available resources: Material, Machine, Devices, Tools

I5 Order of Work
I4 Remote User
I3 Name of the Part
I2 Model in Features of the Finished Part Validated
I1 Model in Features of the Blank Validated

Mapping of Features of Project in Features of Manufacture (Machining) — A21

Machining Features

Data of the Blank (Stock) and Finished Part: Name, User, Dimensions, Tolerances and Material

Determination of the Machining Operations — A22

Machining Operations

Determination of the Operations with Alternatives — A23

Sequence Of Machining

Determination System of Fixture and Reference - SETUP — A24

Reference System

Fixture System (Part and Tools)

Selection of Tools — A25

Machining Tools

Determination Technological Conditions of Machining — A26

Cutting Conditions

Generation of the G Code- it Programs NC (ISO 6983) — A27

Program NC - ISO 6983 → O3

Reports: Determination of the Estimate of Time and Costs and Generation of the Process Plan with Alternatives — A28

Pocess plans with Alternatives → O1
Estimate of Costs → O2

System CAPP
System Multi- Agent
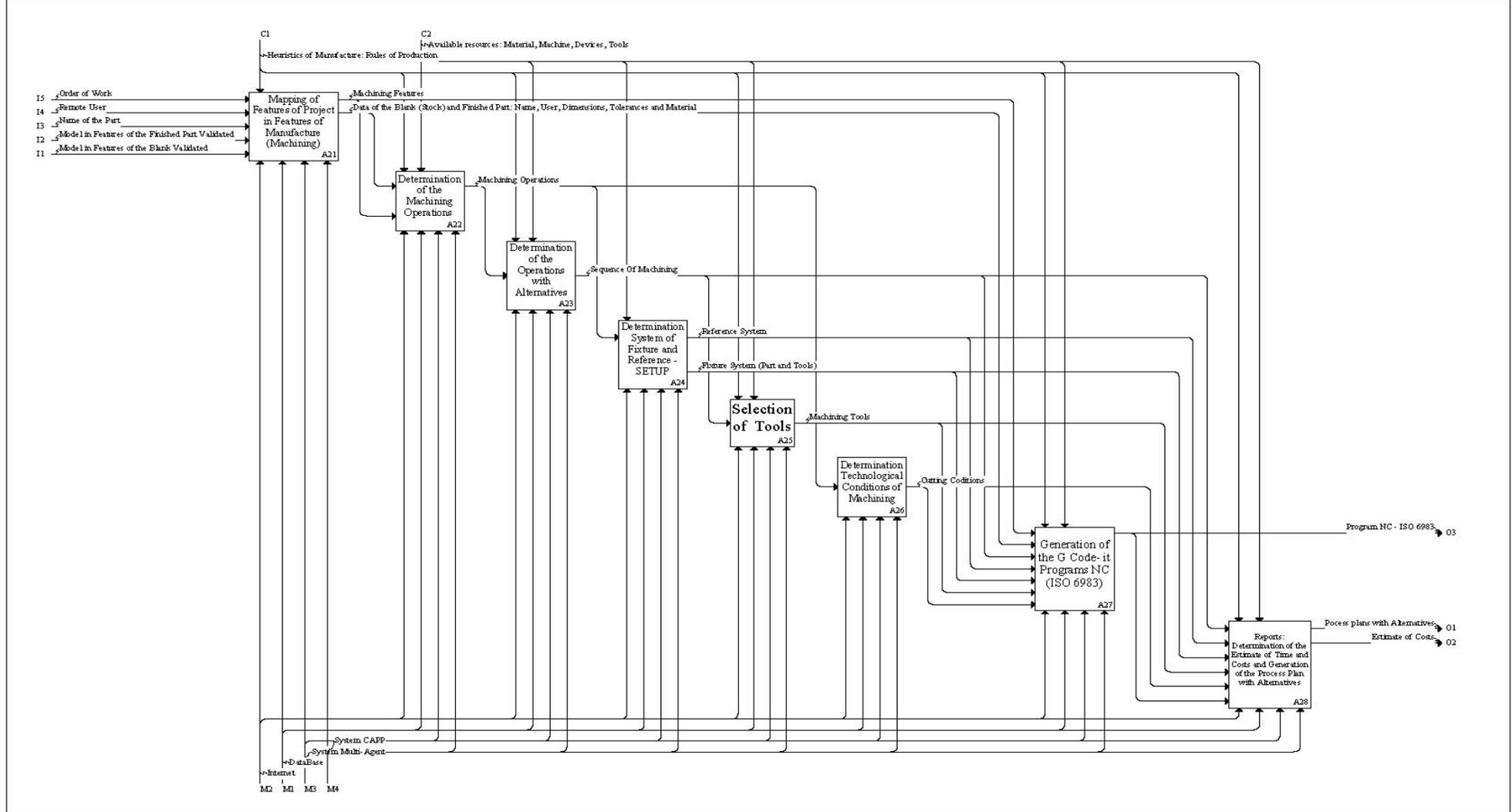DataBase
Internet

M2 M1 M3 M4

Figure 4. IDEF0: CAPP System WebMachining.

6. Conclusion

The architecture and proposed methodology uses the Internet and communication systems to offer a new paradigm for the development of the future integrated CAD/CAPP/CAM environments. These environments will be global, network-centered and spatially distributed, having as front-end with the CAD/CAPP/CAM system navigators based on the Web and the Java language, allowing the independence of the user's computer platform.

The proposed system presents some contributions for the development of telemanufacturing systems based on the Web, integrating design activities, process planning and manufacturing through feature modelling and using the TCP/IP protocols. This proposal is innovative in what concerns the design approach based on feature synthesis and in the production process used, aiming at manufacturing rotational parts. It also innovates in the incorporation of functionalities associated with teleoperation of the CNC machine-tool through the Web, allowing CAD/CAPP/CAM integration. A prototype is being implemented with this methodology, and it should be concluded until 2005.

7. References

Álvares, A. J, Ferreira, J. C. E., 2003, "Uma Metodologia para Integração CAD/CAPP/CAM Voltada para Manufatura Remota de Peças Rotacionais Via Web", II COBEF, Uberlândia, MG.

Álvares, A. J. e Romariz, L. J., 2002, "Telerobotics: Methodology for the Development of a Through-the-Internet Robotic Teleoperated System", Journal of the Brazilian Society of Mechanical Sciences, Vol. XXIV, No. 2, págs 122-126.

Álvares, A. J, Ferreira, J. C. E., Shimabukuro, L. e Fernandes, C. A., 2002, "Um sistema de telemanufatura baseado na web orientado ao processo de oxicorte", XXII ENEGEP, Curitiba.

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

Chep, A., Tricarico, L. Bourdet, P., Galantucci, L., Desiggn of Object-oriented database for the definition of Machining Operation Sequences of 3D Workpieces, Computers In. Engng, Vol. 34, No. 2, 257-279, 1998.

Cho, K. K., Lee, S. H., Ahn, J. H., 1991, "Development of Integrated Process Planning and Monitoring System for Turning Operation", Anais do CIRP, Vol. 40, págs 423-427.

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

Erickson, R. E., 1988, "The state of the art in computer aided process planning", CAM-I report.

Giusti, F., Santochi M., Dini, G., 1989, "KAPLAN: a Knowledge-Based Approach to Process Planning of Rotational Parts", Annals CIRP, Vol. 38, págs 481-484.

Han, J. H. e Requicha, A. A. G., 1998, "Modeler-independent Feature Recognition in a Distributed Environment". Computer-Aided Design, 30(6), págs 453-463.

Hardwick, M., Spooner, D. L., Rando, T, e Morrir, K. C., 1996, "Sharing Manufacturing Information in Virtual Enterprises", Communications of the ACM, 39(2), págs 46-54.

Kruth, J. P. e Detand, J., 1992, "A CAPP System for Nonlinear Process Plans", Anais CIRP Vol. 41/1, págs 489-492.

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

Martino, T. D., Falcidieno, B. e Hasinger, S., 1998 "Design and Engineering Process Integration Through a Multiple View Intermediate Modeller in a Distributed Object-oriented System Environment", Computer-Aided Design, 30(6), págs 437-452.

Rico, C. S., Mateos, S., Cuesta, E., Duarte, A., 1997, "An Automatic CAPP System for Rotational Parts", 0-7803-4192-9/97, IEEE.

Salomons O.W., Houten F.J.A.M. van, Kals H.J.J., 1993, "Review of research in feature-based design, Journal of Manufacturing Systems", Vol.12, no.2, págs 113-132.

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

Shah, J. J., Mäntylä, M., 1995, "Parametric and Feature-Based CAD/CAM: Concepts, Techniques, and Applications", John Wiley & Sons, 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, 1(2), págs. 129-156.

Shunmugam, M. S., Mahesh, P. e Reddy, S. V. B., 2002, "A method of preliminary planning for rotational components with C-axis features using genetic algorithm", Computers in Industry, 1605, págs 1-19.

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, págs 1-33.

Tönshoff, H.K., Aurich, J.C., Baum, Th., 1994, "Configurable Feature-Based CAD/CAPP System". Proceedings of the IFIP International Conference on Feature Modeling and Recognition in Advanced CAD/CAM Systems. Valenciennes, France, págs 757-769.