

WebMachining: UMA METODOLOGIA PARA INTEGRAÇÃO CAD/CAPP/CAM VOLTADA PARA MANUFATURA REMOTA DE PEÇAS ROTACIONAIS VIA WEB

Alberto José Álvares¹ e João Carlos Espíndola Ferreira²

¹ Universidade de Brasília, Departamento de Engenharia Mecânica, GRACO, CEP 70910-900, Brasília, DF, alvares@AlvaresTech.com.

² Universidade Federal de Santa Catarina, Departamento de Engenharia Mecânica, GRIMA, CEP 88040-900, Florianópolis, SC, jcf@grucon.ufsc.br.

Resumo. *Este artigo descreve uma proposta de metodologia para integração CAD/CAPP/CAM voltada para manufatura remota de peças rotacionais utilizando a Internet. A metodologia é concebida a partir do paradigma de modelagem baseada em síntese por features de projeto, a fim de permitir a integração das atividades de CAD/CAPP/CAM. O procedimento inicia-se na modelagem de uma peça por features num contexto de manufatura remota utilizando a Web como meio de comunicação, num modelo computacional cliente-servidor. O sistema é concebido num ambiente distribuído de agentes denominado de Comunidade de Agentes de Manufatura.*

Abstract. *This paper describes a methodology for CAD/CAPP/CAM integration for the remote manufacture of cylindrical parts using Internet, especially through the protocols associated with the World Wide Web. 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/CAPP/CAM. The procedure begins with the modeling of a part by features in a context of remote manufacture 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.*

Palavras-chave: *Telemanufatura, Features, CAPP, CAM, Internet.*

1. Introdução

Na prática de projeto de engenharia, cada vez mais, as atividades associadas aos vários aspectos de manufatura estão sendo considerados durante a fase de projeto. Modelagem baseada em *Features* tem sido considerada como um paradigma para integração das atividades de engenharia, do projeto à fabricação. Aplicações em engenharia estão migrando para ambientes computacionais heterogêneos e distribuídos para suportar o processo de projeto e fabricação que serão distribuídos, tanto na dimensão espacial quanto temporal. Nota-se que é indesejável e frequentemente improvável requerer que todos os participantes nas atividades de manufatura de um produto usem o mesmo sistema de hardware e software. Logo os componentes devem ser modulares e comunicar-se com os demais através de uma rede de comunicação para efetiva colaboração.

Muitos esforços de pesquisa têm sido empregado no desenvolvimento de ambientes de projeto orientados à redes de computadores, normalmente denominados de centrados em rede. Shah et al. [1] desenvolveram uma arquitetura para padronização da comunicação entre o núcleo de um sistema de modelagem geométrica e as aplicações. Han e Requicha [2] propõem uma abordagem similar que possibilita o acesso transparente para diversos modeladores sólidos. Smith e Wright [3] descrevem um serviço de manufatura distribuído denominado Cybercut, desenvolvido na Universidade de Berkeley (<http://cybercut.berkeley.edu>), que possibilita a concepção de uma peça prismática que será usinada utilizando-se de um sistema CAD/CAM desenvolvido em Java, em um contexto de manufatura remota [4].

Hardwick et al. [5] propõem uma infra-estrutura que permite a colaboração entre empresas no projeto e fabricação de novos produtos. Esta arquitetura integra a Web para compartilhamento de informações utilizando o padrão STEP para modelagem de produto. Martino et al. [6] propõem uma abordagem para integrar as atividades de projeto com as demais atividades de fabricação e produção baseada em *features*, suportando Projeto por *Features* e Reconhecimento de *Features*.

Lee et al. [7] apresentam a arquitetura de um sistema de modelagem baseada em *features* centrado em rede, em um ambiente de projeto distribuído, denominado de NetFeature System. Já WebSpiff [8] baseia-se numa arquitetura cliente/servidor consistindo, no lado do servidor, de dois componentes principais: Sistema de Modelagem SPIFF que fornece toda a funcionalidade para modelagem baseada em *features*, utilizando o kernel de modelagem ACIS; e o Gerenciador de Sessão que fornece funcionalidade para iniciar, associar-se, sair e fechar uma sessão de modelagem, bem como gerência todas as comunicações entre o sistema SPIFF e os clientes.

Este trabalho descreve uma proposta de metodologia para integração CAD/CAPP/CAM voltada para manufatura remota de peças rotacionais utilizando a Internet, em especial os protocolos associados à World Wide Web (WWW). Esta metodologia especifica os módulos do sistema e como estes módulos interagem entre si, sendo apresentado na forma de diagramas IDEF0 e IDEF1X.

2. Metodologia WebMachining (<http://WebMachining.AlvaresTech.com>)

A metodologia proposta é concebida a partir do paradigma de modelagem baseada em síntese por *features* de projeto, a fim de permitir a integração das atividades de projeto (CAD), planejamento do processo (CAPP) e fabricação (CAM Planejamento e CAM execução) utilizando como referência o Modelo de *Features* de Manufatura definido pelo AP 224. O procedimento inicia-se na modelagem de uma peça por *features* num contexto de manufatura remota utilizando a Web como meio de comunicação, num modelo computacional cliente/servidor.

O cliente conecta-se ao Modelador de *Features* Neutro via Web e inicia a instanciação de uma nova peça a ser modelada a partir de uma base de dados, usando a biblioteca de *features* padronizada, disponibilizada pelo sistema. Após a conclusão e validação do modelo, a peça criada é armazenada e disponibilizada para os demais módulos do sistema.

O sistema é concebido num ambiente distribuído de agentes de softwares interoperáveis denominado de Comunidade de Agentes de Manufatura (MAC), sendo sua arquitetura estratificada em três níveis: Projeto, Planejamento do Processo e Fabricação. No nível superior reside um grupo de Agentes de Projeto, os quais atuam como uma ferramenta CAD baseada em *features* e também permite aos usuários que se conectem aos níveis inferiores.

O nível intermediário consiste de Agentes de Planejamento que incorporam as características de um CAPP generativo utilizando métodos de representação do conhecimento baseados em inteligência artificial. Este grupo de agentes interpreta as definições de projeto, modelagem geométrica realizada pelo usuário, e determina como fabricar a peça usando o Agente de Fabricação, que reside no nível inferior. O Agente de Fabricação permite também a teleoperação da máquina de comando numérico via Internet.

As informações sobre recursos de fabricação disponíveis (máquinas, ferramentas, informações de materiais, dispositivos de fixação, etc) estão disponíveis através de uma base de dados relacional que é acessada pelos diversos agentes do sistema. O fluxo de informações entre os três níveis da arquitetura é realizado através do encapsulamento das informações, adotando-se o *Knowledge Query and Manipulation Language* (KQML) como padrão de linguagem de mensagens entre os agentes. As mensagens são trocadas usando conexões *sockets* diretas [4] entre os agentes.

A metodologia proposta utiliza a tecnologia Internet e de comunicação para oferecer um novo paradigma para o desenvolvimento dos futuros ambientes integrados CAD/CAPP/CAM. Este ambiente é globalizado, centrado em rede e espacialmente distribuído, tendo como *front-end* com o

sistema CAD/CAPP/CAM navegadores baseados na Web e a linguagem Java e VRML (*Virtual Reality Modeling Language*), permitindo a independência da plataforma computacional do usuário.

3. Arquitetura Multiagente

A utilização de uma arquitetura baseada em sistemas multiagentes (MAS) é sem dúvida a mais atrativa atualmente, principalmente devido à evolução dos sistemas computacionais, em especial de Unix para computadores pessoais, e a utilização de redes de comunicação baseadas no protocolo TCP/IP em uma arquitetura cliente/servidor. Desta forma pode-se utilizar diversos tipos de agentes trabalhando cooperativamente e de forma distribuída na resolução dos diversos problemas associados à integração CAD/CAPP/CAM em um contexto de uma comunidade de agentes. Por exemplo, pode-se utilizar um sistema de gerenciamento de base de dados relacional (MySQL®) para compartilhar as informações dos recursos disponíveis de manufatura e ter os agentes como os resolvidores das atividades de planejamento do processo.

A figura 1 apresenta a Comunidade de Agentes de Manufatura (MAC) concebida, que é estratificada em três níveis: no nível superior reside um grupo de Agentes de Projeto, os quais atuam como ferramentas CAD (figura 2). O nível intermediário consiste de Agentes de Planejamento, os quais interpretam as definições de projeto, modelagem geométrica realizada pelo usuário, e determina como fabricar a peça usando os Agentes de Fabricação selecionados, que residem no nível inferior. É utilizada uma interface transparente entre o projeto, planejamento e fabricação. É necessário realizar o encapsulamento destas informações, sendo adotado o KQML, que contém informações sobre quais agentes enviam mensagem, onde eles estão e como interpretar a mensagem recebida pelo destinatário. É utilizada a ferramenta computacional JATLite (*Java Agent Template Lite* -<http://java.stanford.edu/index.html>).

A arquitetura proposta é constituída por cinco grupos de agentes: um facilitador (1), um gerenciador de base de dados (2), dois de projeto (3 e 4), sete de planejamento do processo (5, 6, 7, 8, 9, 10 e 11) e cinco de fabricação (12, 13, 14, 15 e 16), conforme descrição a seguir (figura 1):

1. AGENTE FACILITADOR (AF): realiza o gerenciamento da comunicação entre os agentes, administrando o roteamento das mensagens entre os agentes, segurança do sistema e o registro de agentes, por exemplo. É implementada através do *Agent Message Router* (AMR) da arquitetura JATLite. É necessário mais de um AF em função da quantidade de agentes presentes no sistema, a fim de melhorar o desempenho do mesmo. Os agentes se comunicam via AMR.
2. AGENTE GERENCIADOR DE BASE DE DADOS (ADB): este agente faz a interação com a base de dados MySQL. Qualquer agente que deseja alguma informação disponibilizada pela base de dados (linguagem SQL) faz uma solicitação para o ADB, e este envia a resposta para o agente que solicitou a informação. O Agente Facilitador realiza o roteamento de mensagens entre os agentes.
3. AGENTE DE INTERFACE CAD COM O USUÁRIO (AICAD): a GUI para projeto por *features* é implementada através de um *applet* Java. Assim toda GUI executada por um cliente remoto, para definir o modelo de *features* e geométrico da peça bruta e da peça acabada (modelagem do produto), possui um AICAD incorporada à interface. Este agente (AICAD) irá se comunicar com a comunidade de agentes através de uma conexão ao AF, e este fará o roteamento da mensagem para o agente em questão. São enviadas mensagens para os demais módulos do sistema comunicando os dados referentes ao projeto em desenvolvimento (modelo de produto) como: usuário, nome da peça, nome do projeto, entre outros; isto permitiu a identificação do modelo de produto que o cliente está elaborando. A visualização 3D do modelo de produto é gerenciada via AICAD, que se comunica com o agente de modelagem 3D. A figura 2 apresenta um protótipo de GUI desenvolvida em JAVA e a visualização de um eixo em VRML.
4. AGENTE DE MODELAGEM 3D BASEADO EM ACIS (AACIS): é responsável pela modelagem 3D utilizando o kernel ACIS, transformando o formato .sat em .vrml. Ele recebe mensagens do AICAD para construção de modelos 3D das peças modeladas por *features*. A figura 2 apresenta a visualização de um eixo em VRML.

5. AGENTE DE MAPEAMENTO DE FEATURES DE MANUFATURA (AMFM): realiza o mapeamento de *features* de projeto em *features* de manufatura, no caso, usinagem para operações de torneamento cilíndrico interno e externo, faceamento, furação, rosqueamento, entre outras. Ele recebe informações do AICAD e consulta a base de dados através do ADB.
6. AGENTE DE DETERMINAÇÃO DAS OPERAÇÕES DE USINAGEM (ADOU): faz a seleção dos processos de usinagem tendo como entrada as *features* de usinagem determinadas pelo AMFM e as restrições são associadas às dimensões, tolerâncias, material da peça, entre outras. Ele recebe informações do AMFM e consulta a base de dados através do ADB.
7. AGENTE DE DETERMINAÇÃO DA SEQUÊNCIA DE USINAGEM E SETUP (ASUS): faz a determinação da sequência de usinagem e *setup* para fixação da peça. Ele recebe informações do ADOU e consulta a base de dados através do ADB.

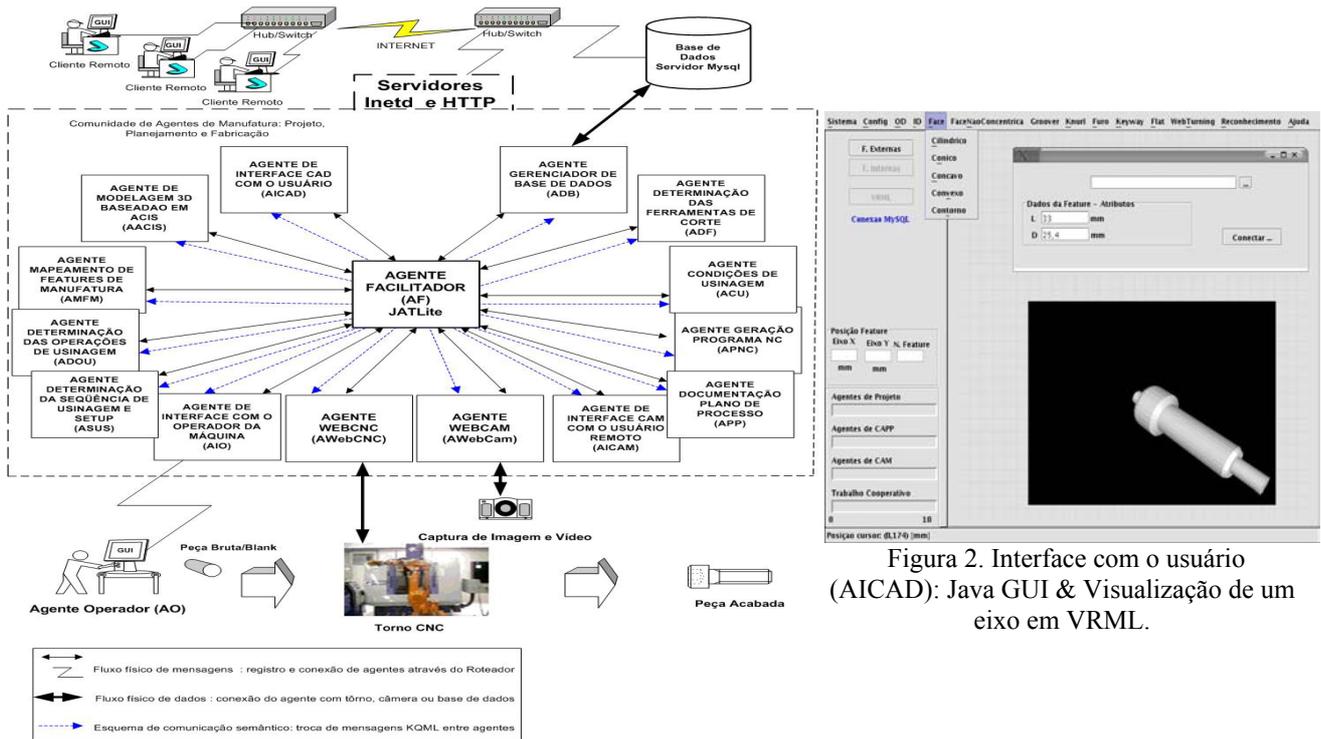


Figura 2. Interface com o usuário (AICAD): Java GUI & Visualização de um eixo em VRML.

Figura 1. Arquitetura multiagente do sistema WebMachining.

8. AGENTE DE DETERMINAÇÃO DAS FERRAMENTAS DE CORTE (ADF): faz a seleção da ferramenta de corte tendo como entrada as máquinas-ferramenta, e as restrições são o tipo de material do par peça/ferramenta, dimensões e geometria das ferramentas, vida da ferramenta, entre outras. Ele recebe informações do ASUS e consulta a base de dados através do ADB.
9. AGENTE DE CONDIÇÕES DE USINAGEM (ACU): determina as condições de corte tendo como entrada os parâmetros das ferramentas e material e as restrições são os critérios utilizados como vida de ferramenta, econômicos, potência da máquina, capacidade da máquina, entre outros. Ele recebe informações do ADF e consulta a base de dados através do ADB.
10. AGENTE DE GERAÇÃO PROGRAMA NC (APNC): determina a trajetória das ferramentas tendo como entrada o modelo do produto baseado em *features* e as restrições baseada nas próprias *features* e eixos de deslocamento das ferramentas para evitar colisões, gerando ao final do processo o programa NC para a máquina-ferramenta a ser utilizada. Ele recebe informações do ASUS, ADF e do ADB.
11. AGENTE DE DOCUMENTAÇÃO PLANO DE PROCESSO (APP): monta o documento referente ao plano de processo incluindo informações de planos alternativos e de estimativa de custos. Comunica-se com os demais agentes de planejamento de processo e ADB.
12. AGENTE DE INTERFACE CAM COM O USUÁRIO REMOTO (AICAM): toda GUI associada ao CAM, executada por um cliente remoto e utilizada para fazer a teleoperação da máquina-ferramenta CNC, possui um AICAM incorporada à interface. Este agente irá se

comunicar com a comunidade de agentes através de uma conexão ao AF, fazendo o roteamento da mensagem para o agente em questão (figura 2).

13. AGENTE WebCam (AWebCam): é responsável pela captura de vídeo e imagem do sistema de teleoperação, enviando as imagens capturadas diretamente para a GUI associada ao CAM. Recebe mensagens do AF referente à identificação do usuário, login e senha, para permitir a execução do servidor WebCam.
14. AGENTE WebCNC (AWebCNC): é responsável pelo controle remoto da máquina CNC, recebendo comandos e enviando o *status* da máquina para a GUI associada ao CAM. Recebe mensagens do AF referente à identificação do usuário, login e senha, nome do arquivo com o programa NC e dados de planejamento do processo (fixação, ferramentas e peça bruta), sendo responsável pela implementação do protocolo de Comando Numérico Distribuído (DNC) através da Web.
15. AGENTE DE INTERFACE COM O OPERADOR DA MÁQUINA (AIO): esta interface gráfica (GUI) instruí o operador no chão-de-fábrica, sendo implementada através de um *applet* Java. Este agente de interface com o operador (AIO) fornece as instruções para o operador sobre fixação da peça bruta, *setup* de ferramentas, preparação da máquina, agendamento da fabricação de uma peça, entre outras.
16. AGENTE OPERADOR DA MÁQUINA (AO): este agente (AO) é o operador da máquina-ferramenta, que recebe instruções de fixação da peça bruta, *setup* de ferramentas, preparação da máquina, agendamento da fabricação de uma peça e outros dados associados ao planejamento de processo e que só podem ser tratados por uma operador humano.

4. Sistema WebMachining: Modelagem IDEF0 e IDEF1X

O sistema integrado CAD/CAPP/CAM proposto, denominado de WebMachining, foi modelado através da metodologia IDEF0. A figura 3 apresenta a estratificação do sistema nos três níveis associados aos agentes de softwares interoperáveis denominado de Comunidade de Agente de Manufatura, sendo as atividades: CAD by Features, CAPP e CAM.

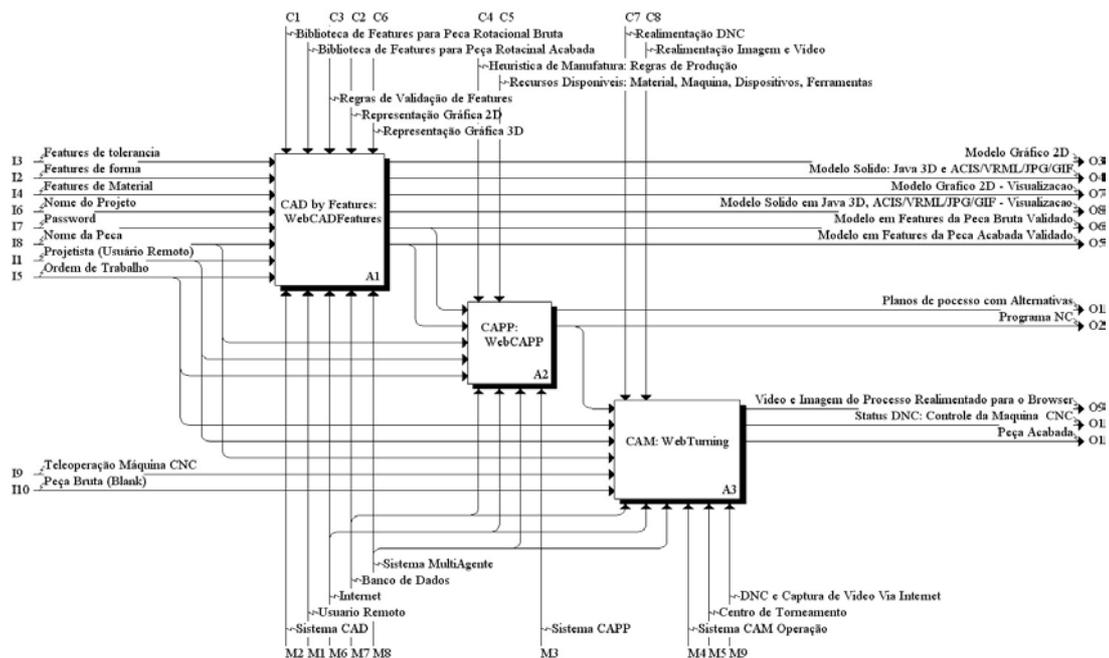


Figura 3. Modelo IDEF0 com as principais atividades.

O CAD by Features tem como entrada de dados a modelagem por *features* e outras informações pertinentes, e como saída o modelo de *features* da peça bruta e da peça acabada, que serve como entrada de dados para o CAPP. O CAPP tem como saída a geração do plano de processos com

alternativas e o programa NC. O CAM tem como entrada dados referentes à teleoperação da máquina CNC e o programa NC (DNC e vídeo/imagem do processo), e como saída o controle da máquina e a fabricação da peça.

O modelo de informação foi concebido através da abordagem metodológica IDEF1X (base de dados relacional). O modelo de informação é dividido em domínios associados a base de dados de *features* (*Features* de Forma, *Features* de Tolerância, *Features* de Produção e *Features* de Materiais) e base de dados de tecnologia de usinagem (Biblioteca de Máquinas-ferramenta, Biblioteca de Ferramentas de corte, Biblioteca de Usinabilidade e Biblioteca de Fixação). A base de dados de *features* relaciona-se com o modelo do produto e a base de dados de tecnologia de usinagem relaciona-se com o modelo de recursos. Estas bases de dados foram implementadas fisicamente através de uma única base de dados denominada de WebMachining, em MySQL, sendo constituída por oitenta e duas tabelas (http://WebMachining.AlvaresTech.com/qualificacao/qualificacao_terceira_versao.pdf).

O usuário remoto conecta-se ao sistema criando um novo projeto que será constituído pelos dados referentes à modelagem por *features* (figura 2) de uma peça acabada e de uma peça bruta, além de informações tecnológicas, tolerâncias dimensionais e geométricas, acabamento superficial, dados de produção e dados do usuário. Todo projeto será constituído por este conjunto de dados para peça bruta e acabada, que estão definidos nos vários domínios descritos.

5. Conclusão

A arquitetura e metodologia proposta utiliza a Internet e sistemas de comunicação para oferecer um novo paradigma para o desenvolvimento dos futuros ambientes integrados CAD/CAPP/CAM. O sistema proposto apresenta muitas contribuições para o desenvolvimento de sistemas de telemanufatura baseado na Web, integrando atividades de projeto, planejamento de processos e de fabricação através da modelagem por *features* e utilizando os protocolos TCP/IP em uma arquitetura multiagente denominada de Comunidade de Agentes de Manufatura. Maiores informações sobre o projeto podem ser obtidas em <http://WebMachining.AlvaresTech.com>.

6. Referências

- [1] J. J. Shah, H. Dedhia, V. Pherwani e S. Solkhan, 1997, "Dynamic Interfacing of Applications to Geometric Modeling Services Via Modeler Neutral Protocol", *Computer-Aided Design*, 29, págs 811-824.
- [2] J. H. Han, e A. A. G. Requicha, 1998, "Modeler-independent Feature Recognition in a Distributed Environment". *Computer-Aided Design*, 30(6), págs 453-463.
- [3] C. S. Smith, P. K. Wright, 2001, "Cybercut: An Internet-based CAD/CAM System", *ASME Journal of Computing and Information Science in Engineering*, Vol. 1, No. 1, págs 1-33.
- [4] A. J. Álvares e L. J. Romariz, 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.
- [5] M. Hardwick, D. L. Spooner, T. Rando, e K. C. Morriri, 1996, "Sharing Manufacturing Information in Virtual Enterprises", *Communications of the ACM*, 39(2), págs 46-54.
- [6] T. D. Martino, B. Falcidieno e S. Hasinger, 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.
- [7] J. Y. Lee, Han, S. B., H. Kim, S. B. Park, 1999, "Network-centric Feature-based Modeling", *Pacific Graphics* 1999.
- [8] R. Bidarra, E. Van den Berg e W. F. Bronsvort, 2001, "Collaborative Modeling with Features", *Proceedings of DET'01, 2001 ASME Design Engineering Technical Conferences*, Pittsburgh, USA.