



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

ALBERTO JOSÉ ÁLVARES¹, JOÃO CARLOS ESPÍNDOLA FERREIRA²

¹ Universidade de Brasília, Departamento de Engenharia Mecânica e Mecatrônica
CE 70910 -900, Brasília, DF, Brasil

² Universidade Federal de Santa Catarina, Departamento de Engenharia Mecânica
CEP 88040-900, Florianópolis, SC, Brasil

(Recibido 29 de marzo de 2004, para publicación 17 de junio de 2004)

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 projeto (CAD), planejamento do processo (CAPP) e manufatura (CAM Planejamento e CAM Execução). 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, sendo sua arquitetura estratificada em três níveis: Projeto, Planejamento do Processo e Fabricação, adotando-se o Knowledge Query and Manipulation Language (KQML) como padrão de linguagem de mensagens entre os agentes.

1. INTRODUÇÃO

A integração entre as etapas do ciclo produtivo é um dos caminhos que devem ser explorados na busca pela redução de custos e tempos de produção. De acordo com Shah e Mantyla [1] a modelagem do produto é o ponto central para a promoção de tal integração. Num sistema de produção integrado, o modelo do produto, definido no módulo de CAD, deve estar disponível para outros módulos (CAE, CAPP, CAM, CAQ, etc) para que estes possam realizar suas funções. Estes módulos também devem ser capazes de enviar informações de realimentação para o módulo de CAD, possibilitando alterações na peça ainda na etapa de projeto (por problemas detectados na fabricação, por exemplo). A utilização de *features* como base de informação para a modelagem do produto é o caminho para se atingir esta integração [2,3].

A Tecnologia da Informação, em especial, a tecnologia de redes de comunicação e Internet, está abrindo um novo domínio para a construção dos futuros ambientes CAD/CAPP/CAM [4] e se colocam como candidatos em potencial para possibilitar o desenvolvimento de sistemas integrados. Este é um novo paradigma para estes sistemas computacionais baseados em ambiente globalizados, centrados em rede e espacialmente distribuídos. Isto permitirá que os desenvolvedores de produtos, projetistas, tenham maior facilidade de comunicação, possibilitando o compartilhamento e o projeto cooperativo durante o desenvolvimento do produto. Com o crescimento da popularidade da tecnologia Web está ficando mais evidente que o ambiente de projeto orientado a rede se tornará um novo paradigma para o desenvolvimento de produto.

Erickson [5] revisou mais de 127 sistemas CAPP e projetos associados. Ele observou que a incompatibilidade nos softwares, no hardware e nas diferentes representações de produto, recursos e planos de processo inibiram o desenvolvimento de um sistema integrado de propósito geral. O autor sugere que os futuros sistemas CAPP deverão ser modulares, fáceis de integrar, amigáveis e de fácil manutenção. Shunmugam et al. [6] destacam que poucos sistemas CAPP reportados na literatura levam em consideração aspectos de otimização da seqüência das operações ou sugerem seqüências alternativas de operações ou planos de processo. Giusti et al. [7] apresentam o CAPP KAPLAN descrevendo uma

abordagem baseada em conhecimento para planejamento de processo de peças rotacionais. Cho et al. [8] descrevem o desenvolvimento de um sistema integrado de planejamento de processo e monitoração para operações de torneamento. Rico et al. [9] descrevem um CAPP para peças rotacionais baseado em *features* e perfis 2D. Kruth e Detand [10] descrevem um sistema de CAPP para geração de planos de processos não lineares.

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.

A arquitetura proposta para o módulo CAD é baseada no paradigma de Projeto por *Features*. A metodologia CAPP 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 manufatura (CAM Planejamento e CAM execução). O Plano de processo linearizado gerado pelo CAPP é representado utilizando uma estrutura de dados baseada em STEP-NC (ISO 14649 - Part 12), que é um novo padrão para transferência de dados entre sistemas CAD/CAM e máquinas CNC (Xu e He, 2003). A arquitetura proposta para o módulo CAD é baseada no paradigma de Projeto por *Features*. Já os módulos CAPP/CAM idealizados podem ser caracterizados segundo as quatro dimensões definidas por Shah e Mantyla [1] da seguinte forma:

- na dimensão de Planejamento é voltado para geração de planos de processos não lineares para peças rotacionais utilizando tornos CNC, contendo definição de operações, fixação/setup, seqüências de operações, ferramentas, parâmetros de corte, geração do código G e estimativa de custos, ou seja, micro planejamento;
- na dimensão de Planejamento associado ao tempo pode ser aplicado no nível tático e no nível operacional de um sistema de Gestão da Produção;
- na dimensão Método de Planejamento é caracterizado com sendo um CAPP Generativo;
- na dimensão Profundidade de Planejamento é definido como dinâmico, ou seja, planos podem ser mudados dinamicamente durante a manufatura em função das características dinâmicas do sistema de manufatura, trabalhando também de forma *on-line*.

2. CAD COOPERATIVO E SISTEMAS CORRELATOS

Na prática de projeto de engenharia, cada vez mais as atividades associadas aos vários aspectos de manufatura estão sendo consideradas durante a fase de projeto. Modelagem baseada em *Features* tem sido considerada como um novo paradigma para integração das atividades de engenharia, desde o projeto até a manufatura. Assim o conceito de *features* tem sido usado em uma ampla gama de aplicações como projeto de peças e montagem, projeto para manufatura, planejamento de processo e inúmeras outras aplicações. Estas aplicações estão migrando para ambientes computacionais heterogêneos e distribuídos para suportar o processo de projeto e manufatura que serão distribuídos tanto na dimensão espacial quanto temporal. Nota-se que é indesejável e freqüentemente improvável requerer que todos os participantes nas atividades de projeto e manufatura de um produto usem o mesmo sistema de hardware e software. Então 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. [11] desenvolveram uma arquitetura para padronização da comunicação entre o núcleo de um sistema de modelagem geométrica e as aplicações. Han e Requicha [12] propõem uma abordagem similar que possibilita o acesso transparente para diversos modeladores sólidos. Smith e Wright [13] 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 [14].

O Cybercut é um sistema orientado ao problema de projeto e fabricação baseado na web [13] consistindo de três componentes: software de CAD escrito em Java usando applets via páginas web; um sistema de Planejamento de Processo Auxiliado por Computador (CAPP) que acessa uma base de conhecimento contendo informações sobre ferramentas e fixações; um CNC de arquitetura aberta que pode receber informações de planejamento e projeto em linguagem de alto nível e executar a usinagem utilizando uma fresadora Haas VF-1.

Com acesso à interface de CAD (Cybercut), a partir da Internet, qualquer engenheiro com um *browser* WWW torna-se um potencial usuário desta ferramenta de prototipagem rápida on-line. A Interface Gráfica com Usuário (GUI) possibilita que um usuário remoto seja capaz de fazer o carregamento de um arquivo CAD em um formato específico, de troca de dados universal, para o servidor Cybercut, o qual irá executar o planejamento de processo necessário e gerar o Código G apropriado para a fresadora. A peça poderá ser usinada e enviada para o projetista. O engenheiro pode ter um protótipo funcional dentro de poucos dias por uma fração do custo que a fabricação própria exigiria. O Cybercut trabalha com os processos de fresamento e furação, entretanto o sistema não permite a teleoperação da máquina CNC [14].

Hardwick et al. [15] propõem uma infra-estrutura que permite a colaboração entre empresas no projeto e manufatura de novos produtos. Esta arquitetura integra o WWW para compartilhamento de informações na Internet utilizando o padrão STEP para modelagem de produto. Martino et al. [16] propõem uma abordagem para integrar as atividades de projeto com as demais atividades de manufatura e produção baseada em *features*, que suporta Projeto por *Features* e Reconhecimento de *Features*. Entretanto, além destes trabalhos serem conceituais em sua essência, não apresentam uma representação bem estruturada nem algoritmos detalhados. Por exemplo, estes trabalhos não definem como distribuir o processamento computacional necessário entre os componentes distribuídos, e como modelar a comunicação entre os componentes para minimizar o atraso da rede. Se as ações de troca de dados entre as aplicações não puderem ser disparadas apropriadamente, isto acarretará em um problema crítico para a computação distribuída.

Sistemas de modelagem cooperativa tipicamente têm uma arquitetura cliente/servidor, diferindo na distribuição de funcionalidade e de dados entre clientes e servidores. Um problema recorrente nos sistemas cliente/servidor está associado ao conflito entre a limitação da complexidade da aplicação cliente e a minimização do carregamento da rede. Em um contexto de modelagem cooperativa, a complexidade do cliente é determinada principalmente pelas facilidades de modelagem e interatividade implementadas no cliente, enquanto que o carregamento da rede é determinado, principalmente, pelo tipo e tamanho do modelo de dados sendo transferido de/para os clientes e da largura de banda disponível.

Uma solução de compromisso pode ser concebida entre os dois extremos, os chamados *thin clients* e *fat clients*. Uma arquitetura *thin client* pura tipicamente coloca toda a funcionalidade no servidor, o qual envia uma imagem de sua interface de usuário para ser mostrada no cliente.

O outro extremo, um puro *fat client* oferece total facilidade de interação e modelagem local, mantendo seu próprio modelo local. Comunicação com o servidor é requerida quando houver necessidade de sincronizar as modificações de dados do modelo local com os outros clientes.

Lee et al. [4] apresentam a arquitetura de um sistema de modelagem baseada em *features* centrado em rede, em um ambiente de projeto distribuído, denominado de NetFeature System. Esta abordagem combina técnicas de modelagem baseada em *features* com tecnologia de comunicação e de computação distribuída para suportar atividades de modelagem de produto e projeto cooperativo em uma rede de computadores. A abordagem é implementada em uma arquitetura cliente/servidor, na qual os clientes realizam a modelagem baseada em *features* através da Web. O servidor cria o modelo de *features* neutras e as outras aplicações se comunicarão umas com as outras usando um protocolo de comunicação padrão para acessar os objetos remotos. O sistema foi concebido a fim de ter um bom balanceamento entre as funcionalidades disponíveis no lado do cliente e a largura de banda disponível na Internet.

O processamento no lado do cliente é importante quando a aplicação é baseada na Web. Isto significa que o servidor dá ao cliente alguma responsabilidade pelo processamento dos dados, ou seja, o cliente deve ter mais funcionalidades do que simplesmente um *front-end* desprovido de processamento local, *thin clients*. Foi utilizado no NetFeature System um protocolo de comunicação padronizado baseado em CORBA (*Common Object Request Broker Architecture*).

WebSpiff [17] 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 feature, utilizando o kernel de modelagem ACIS [18]; 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.

Os componentes do portal WebSpiff (<http://www.webspiff.org>) fornecem o acesso inicial a uma sessão WebSpiff para um novo cliente, que inclui um servidor web onde os dados do modelo são disponibilizados para *download* pelos demais clientes que estão desenvolvendo o projeto cooperativo. Os clientes executam operações localmente, associadas com a visualização, interação com o modelo em *features*, mensagens semânticas de alto nível, especificação de operações de modelagem, bem como as atualizações de dados dos clientes são enviados via rede. O servidor coordena a sessão de colaboração, mantém um modelo de produto centralizado e provê todas as funcionalidades que não podem, ou não devem, ser implementados no cliente. Uma importante vantagem desta arquitetura é que existe apenas um modelo de produto centralizado, então evita-se inconsistências entre múltiplas versões do mesmo modelo.

3. METODOLOGIA WebMachining (<http://WebMachining.AlvaresTech.com>)

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 a metodologia CAPP gerar o plano de processo com alternativas para a peça, sua linearização e a representação do plano de processo linearizado baseado em STEP-NC ISO 14649 - Part 12 [19], bem como, a geração do programa NC para um torno CNC específico, no caso o centro de torneamento Galaxy 15M da Romi (<http://video.graco.unb.br>).

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 manufatura (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 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 (Fig. 1): 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, destacando-se os sistemas especialistas e algoritmos genéticos. Este grupo de agentes interpreta as definições de projeto, modelagem geométrica realizada pelo usuário, e determina como fabricar a peça, que será usinada pelo 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 [14] entre os agentes.

A metodologia proposta utiliza a tecnologia Web (Internet) e de comunicação para oferecer um novo paradigma para o desenvolvimento dos futuros ambientes integrados CAD/CAPP/CAM. Este ambiente é

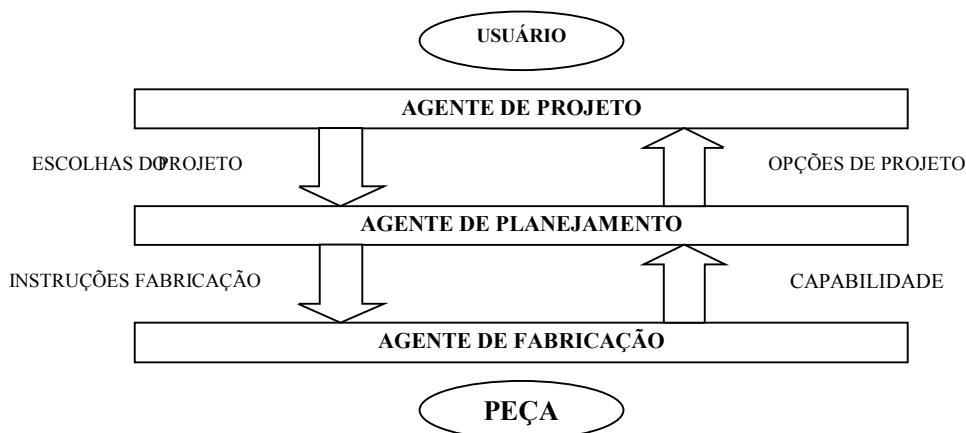


Fig. 1. Comunidade de agentes de manufatura.

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. A originalidade desta proposta está associada à contribuição metodológica para o desenvolvimento de sistemas de TeleManufatura via Web a partir da concepção de um produto/peça até a sua fabricação, usando a Internet como meio de conexão entre as várias tecnologias avançadas de manufatura utilizadas.

4. ARQUITETURA MULTIAGENTE

A arquitetura proposta para o sistema multiagentes (MAS) pode ser caracterizada no comportamento dos agentes como sendo Deliberativa, na organização interna como sendo do tipo *Blackboard* e na arquitetura do sistema multiagentes como sendo do tipo Federativa utilizando a abordagem do Facilitador [20].

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 [21]. 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 resolvedores das atividades de planejamento do processo.

É utilizada a ferramenta computacional JATLite (*Java Agent Template Lite*). JATLite (<http://java.stanford.edu/index.html>) é um pacote de programas escritos em Java que permite aos usuários a criação de agentes de softwares que comunicam-se de forma robusta na Internet. JATLite oferece uma infra-estrutura básica na qual agentes registrados em um *Agent Message Router* (AMR), normalmente denominado de facilitador ou mediador, usam um nome e uma senha, conectando-se e desconectando-se da Internet, enviando e recebendo mensagens, transferindo arquivos com FTP e geralmente trocando informações com outros agentes, através dos vários computadores onde eles estão sendo executados.

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 (Fig. 2):

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. O AMR é muito importante no ambiente de desenvolvimento JATLite, pois os agentes sempre se comunicam uns com os outros 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 estes 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 permiti a identificação do modelo de produto que o cliente está elaborando. A conexão com a base de dados MySQL é realizada diretamente via mecanismo PHP (*Personal Home Page Kit*) para introdução de alguns dados. A visualização 3D do modelo de produto é gerenciada via AICAD, que se comunica com o agente de modelagem 3D. A Fig. 3 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 Fig. 3 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 SEQÜÊNCIA DE USINAGEM E SETUP (ASUS): faz a determinação da seqüência de usinagem e *setup* para fixação da peça. Ele recebe informações do ADOU e consulta a base de dados através do ADB.
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.

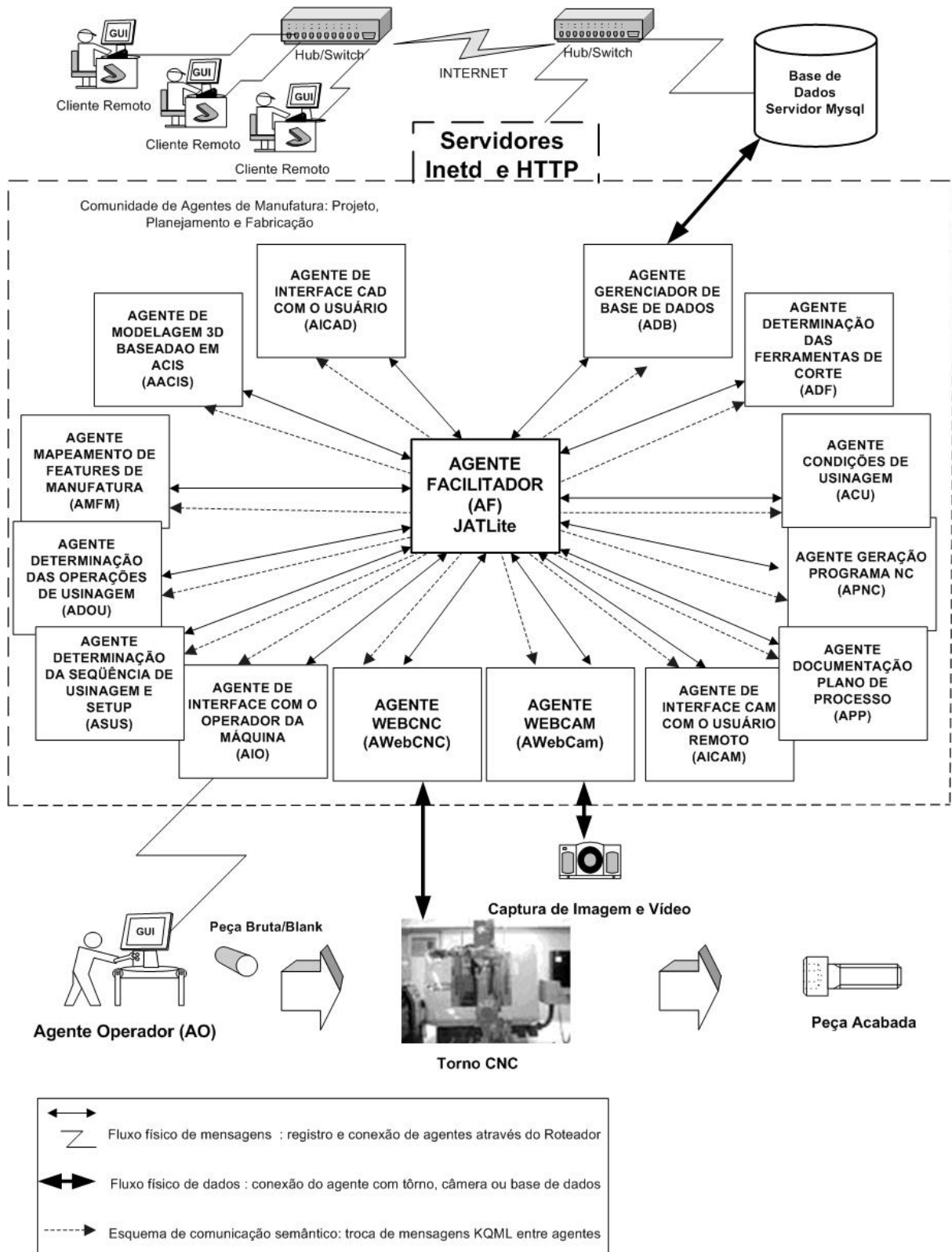


Fig. 2. Arquitetura multiagente do sistema WebMachining.

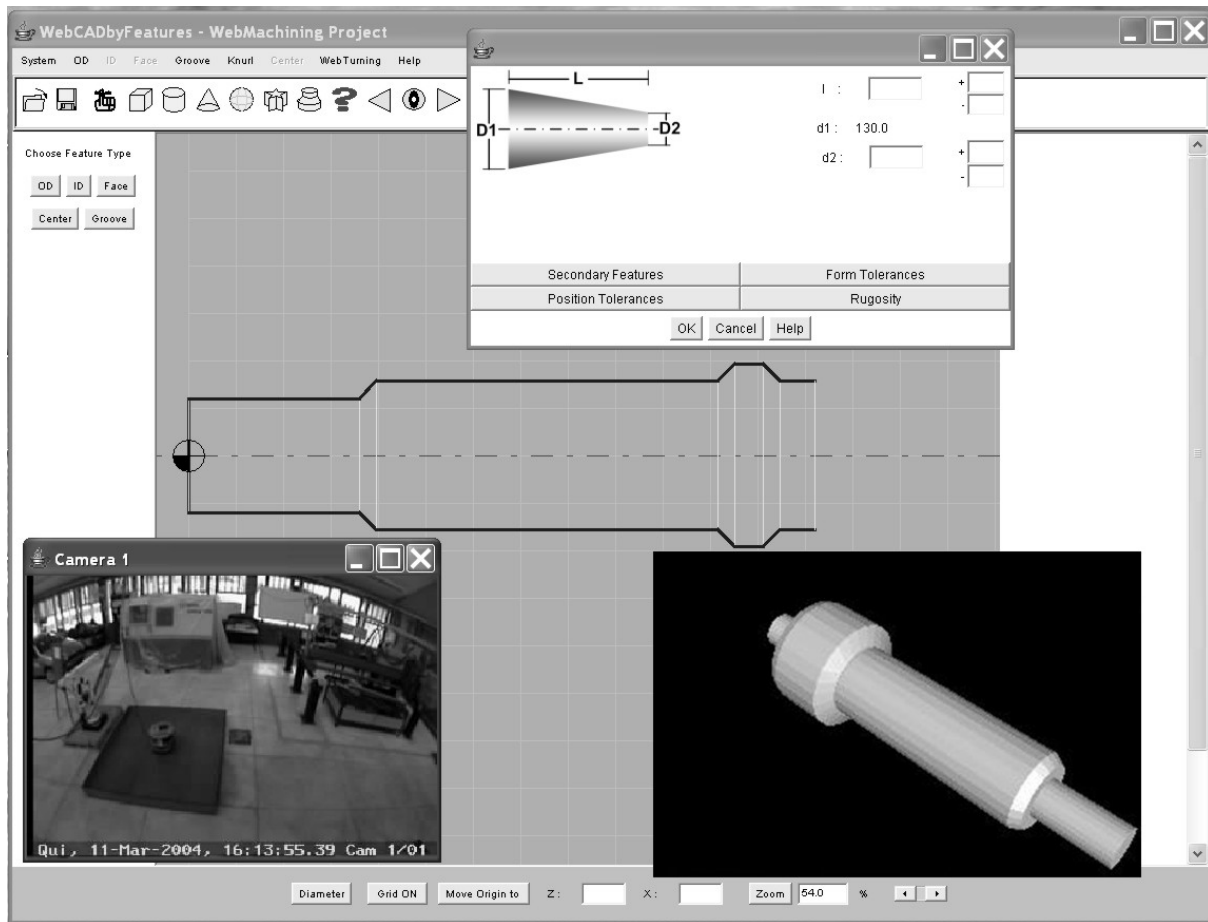


Fig. 3. Agente de interface com o usuário: Java GUI, WebCam e visualização eixo em VRML.

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 (Fig. 3).
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 (Fig. 3).
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 (DNC1/FOCAS1/FANUC) através da Web.
15. AGENTE DE INTERFACE COM O OPERADOR DA MÁQUINA (AIO): esta interface gráfica (GUI) instrui o operador no chão-de-fábrica, sendo implementada através de um *applet*

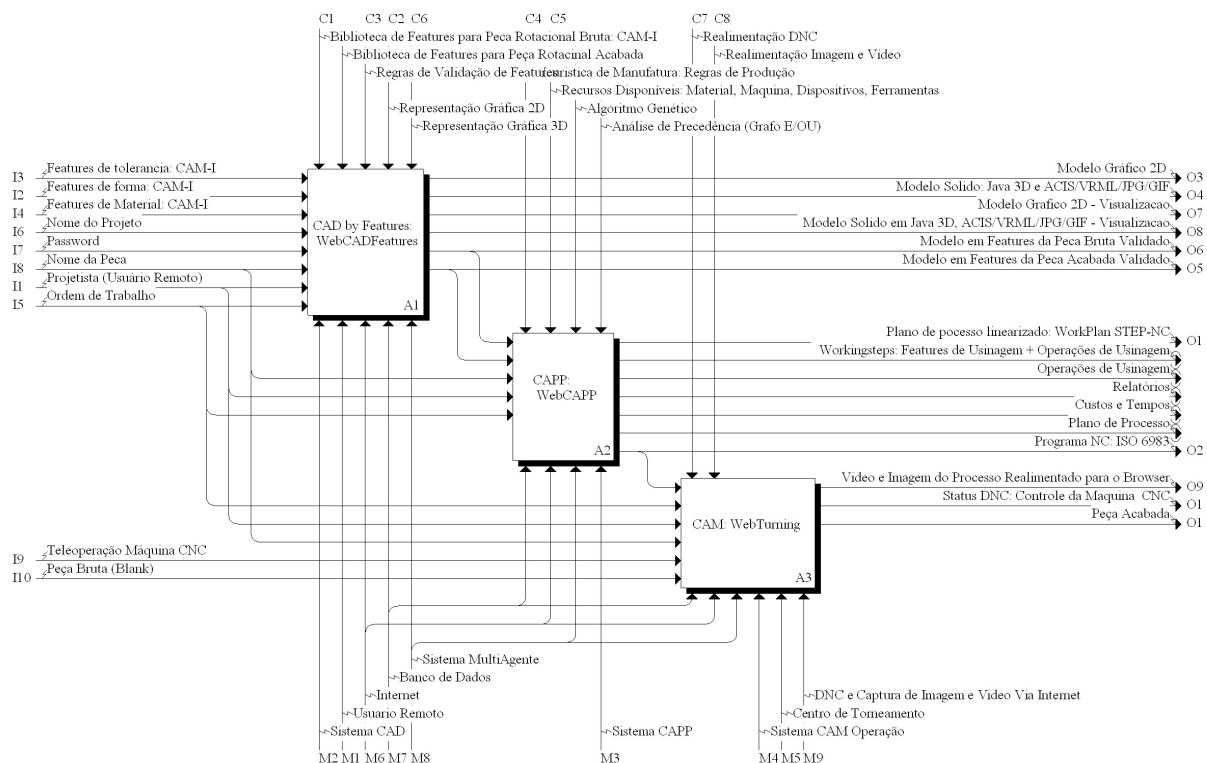


Fig. 4. Modelo IDEF0 com as principais atividades do sistema WebMachining.

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 um operador humano.

5. SISTEMA WEBMACHINING: MODELAGEM IDEF0 E IDEF1X

O sistema integrado CAD/CAPP/CAM proposto, denominado de WebMachining, foi modelado através da metodologia IDEF0 [22]. A Fig 4 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.

O sistema de CAPP denominado por WebCAPP (<http://WebMachining.AlvaresTech.com>) é constituído por dez atividades. A Fig. 5 apresenta as atividades associadas ao diagrama IDEF0, sendo: mapeamento de features de projeto para features de manufatura (usinagem); determinação das operações de usinagem com alternativa, associadas às features de usinagem (workingsteps); determinação da seqüência de usinagem com alternativas (workplan não linear); estratégias para geração de trajetórias de ferramentas; determinação das ferramentas de corte (insertos e suportes); determinação do modelo de tempos e cálculo dos tempos padrão e custo padrão para cada workingstep; determinação das condições tecnológicas de usinagem otimizadas utilizando algoritmos genéticos; linearização do plano de processos não linear baseado algoritmos genéticos; geração do programa NC (ISO 6983); geração de relatórios e plano de processo.

O CAD by Features (WebCADbyFeatures) 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 (Fig. 5). O CAPP tem como saída a geração do plano de processos com alternativas e o programa NC, baseado em STEP NC [19, 23]. 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 82 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* (Fig. 3) 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. Um usuário pode ter vários projetos concebidos e armazenados no sistema de base de dados. Cada usuário tem uma identificação única no sistema.

6. 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.

Esta proposta é inovadora no que concerne a abordagem de projeto por síntese de *features* e no processo de fabricação utilizado, sendo voltados para fabricação de peças rotacionais. Inova também na incorporação de funcionalidades associadas à teleoperação via Web da máquina-ferramenta CNC permitindo a integração CAD/CAPP/CAM. A partir desta metodologia está sendo implementado um protótipo que deverá estar concluído até 2005. Maiores informações sobre o projeto podem ser obtidas em <http://WebMachining.AlvaresTech.com>.

REFERÊNCIAS

- [1] J. J. Shah, M. Mäntylä, 1995, "Parametric and Feature-Based CAD/CAM: Concepts, Techniques, and Applications", John Wiley & Sons, New York.
- [2] H. K. Tönshoff, J.C. Aurich, TH. Baum, 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.
- [3] O.W. Salomons, F.J.A.M. van Houten, H.J.J. Kals, 1993, "Review of research in feature-based design, Journal of Manufacturing Systems", Vol.12, no.2, págs 113-132
- [4] J. Y. Lee, S. B. Han, H. Kim, S. B. Park, 1999, "Network-centric Feature-based Modeling", Pacific Graphics.
- [5] R. E. Erickson, 1988, "The state of the art in computer aided process planning", CAM-I report.
- [6] M. S. Shunmugam, P. Mahesh e S. V. B. Reddy, 2002, "A method of preliminary planning for rotational components with C-axis features using genetic algorithm", Computers in Industry, 1605, págs 1-19.

- [7] F. Giusti, M. Santochi, G. Dini, 1989, "KAPLAN: a Knowledge-Based Approach to Process Planning of Rotational Parts", *Annals CIRP*, Vol. 38, págs 481-484.
- [8] K. K. Cho, S. H. Lee, J. H. Ahn., 1991, "Development of Integrated Process Planning and Monitoring System for Turning Operation", *CIRP*, Vol. 40, págs 423-427.
- [9] C. S. Rico, S. Mateos, E. Cuesta, A. Duarte, 1997, "An Automatic CAPP System for Rotational Parts", 0-7803-4192-9/97, IEEE.
- [10] J. P. Kruth e J. Detand, 1992, "A CAPP System for Nonlinear Process Plans", *Anais CIRP Vol. 41/1*, págs 489-492.
- [11] 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.
- [12] 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.
- [13] 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.
- [14] 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.
- [15] M. Hardwick, D. L. Spooner, T Rando, e K. C. Morrir, 1996, "Sharing Manufacturing Information in Virtual Enterprises", *Communications of the ACM*, 39(2), págs 46-54.
- [16] 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.
- [17] 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.
- [18] J. Corney e T. Lim, 2001, "3D Modeling with ACIS", *Saxe-Coburg Publications*, 2ª Edição.
- [19] ISO 14649 Data model for Computerized Numerical Controllers - Part 12: Proces Data for Turning, *Draft International Standad*, V09 April 2003.
- [20] W. Shen, D.H. Norrie e J.P.A. Barthés, 2001, "Multi-Agent Systems for Concurrent Intelligent Design and Manufacturing", *Taylor & Francis*, New York.
- [21] W., Shen, D.H. Norrie, 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.
- [22] A. J. Álvares, J. C. E. Ferreira, 2003, "Uma Metodologia para Integração CAD/CAPP/CAM Voltada para Manufatura Remota de Peças Rotacionais Via Web", *II COBEF*, Uberlândia, MG.
- [23] X.W. Xu, Q. He, Striving for a total integration of CAD, CAPP, CAM and CNC, *Robotics and Computer Integrated Manufacturing*, Article in Press, accepted 7 August 2003.

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

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), 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 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, whose architecture is stratified in three levels: Design, Process Planning and Manufacturing.