

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

**Alberto José Álvares**

Universidade de Brasília, Departamento de Engenharia Mecânica e Mecatrônica, Grupo de Automação e Controle (GRACO), CEP 70910-900, Brasília, DF, [alvares@AlvaresTech.com](mailto: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](mailto: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, em especial os protocolos associados a World Wide Web. Esta metodologia especifica os módulos do sistema e como estes módulos interagem entre si. A arquitetura do sistema CAD/CAPP/CAM consiste de um conjunto de especificações, técnicas e algoritmos utilizados para a definição funcional dos módulos do sistema. 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 de softwares interoperáveis denominado de Comunidade de Agentes de Manufatura, sendo sua arquitetura estratificada em três níveis: Projeto, Planejamento do Processo e Manufatura. 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.

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

### 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 (1995) 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, para possibilitar que as alterações necessárias na peça possam ser efetuadas 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 (Tönshoff et al., 1994; Salomons et al. 1993).

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 (Lee et al., 1999) 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 dos navegadores baseados na WWW está ficando mais evidente que o ambiente de projeto orientado a rede se tornará um novo paradigma para o desenvolvimento de produto.

Erickson (1988) 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. Ham e Lu (1988) sugerem que as futuras pesquisas deverão incluir a integração de projeto e manufatura aplicando técnicas de inteligência artificial. Shunmugam et al. (2002) 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. (1989) apresentam o CAPP KAPLAN descrevendo uma abordagem baseada em conhecimento para planejamento de processo de peças rotacionais. Cho et al. (1991) descrevem o desenvolvimento de um sistema integrado de planejamento de processo e monitoração para operações de torneamento. Rico et al. (1997) descrevem um CAPP para peças rotacionais baseado em *features* e perfis 2D. Kruth e Detand (1992) descrevem um sistema de CAPP para geração de planos de processos não lineares. Cay e Chassapis (1997) fazem um levantamento do estado da arte associado ao CAPP e concluem que sistemas de engenharia distribuídos baseados em agentes irão permitir a integração das atividades de desenvolvimento de produto de forma concorrente. Destacam também a falta de boas metodologias para desenvolvimento de sistemas de engenharia simultânea que incluam planejamento de processo. Sugerem que as metodologias IDEF (Harrington, 1984) devem ser utilizadas para descrever os modelos das atividades de produção.

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.

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 (1995) 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; e 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 a redes de computadores, normalmente, denominados de centrados em rede. Shah et al. (1997) desenvolveram uma arquitetura para padronização da comunicação entre o núcleo de um sistema de modelagem geométrica e as aplicações. Han e Requicha (1998) propõem uma abordagem similar que possibilita o acesso transparente para diversos modeladores sólidos.

Smith e Wright (2001) descrevem um serviço de manufatura distribuído denominado de 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 (Álvares et al., 2002).

O Cybercut é um sistema orientado ao problema de Projeto e Fabricação baseado na web (Smith e Wright, 2001) 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 (Álvares et al., 2002).

Hardwick et al. (1996) 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. (1998) 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 estes trabalhos são conceituais em sua essência e não apresentam uma representação bem estruturada e 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. Logo, é crucial o desenvolvimento de um sistema bem integrado, centrado em rede e com arquitetura baseado em agentes para projeto e manufatura distribuída e cooperativa.

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. (1999) 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 e normalmente denominado *thin clients*. Foi utilizado no NetFeature System um protocolo de comunicação padronizado baseado em CORBA (Common Object Request Broker Architecture).

WebSpiff (Bidarra et al., 2001) 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 (Corney e Lim, 2001); 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. PROJETO POR FEATURES

Segundo Shah e Mantyla (1995) duas metodologias de Projeto por *Features* são comumente utilizadas: *Destruction by Machining Features* e Síntese por *Features* de Projeto. A abordagem destrutiva também é conhecida por *Destructive Solid Geometry* ou *Deforming Solid Geometry* (DSG).

A abordagem *Destruction by Machining Features* inicia-se com um modelo da peça bruta que será usinada. O modelo da peça é criado pela subtração na peça bruta de *features* que correspondem ao material removido por operações de usinagem. O sistema Cybercut utiliza esta abordagem, obrigando o usuário a remover entidades geométricas de uma peça de formato regular através da usinagem (p.ex. fresamento) da peça bruta, de tal forma que o processo de manufatura da peça é incorporado, de forma inerente, ao projeto. A vantagem deste método é que as *features* de usinagem estão diretamente disponíveis no modelo da peça não sendo necessário o reconhecimento ou mapeamento de *features*. Uma desvantagem consiste no fato de que o projetista deve ter um amplo conhecimento de manufatura e força o projetista a pensar em termos de *features* de manufatura. Normalmente, o projetista está interessado, inicialmente, na forma da peça e nos aspectos funcionais.

Na segunda abordagem, Síntese por *Features* de Projeto, o modelo pode ser construído tanto pela adição quanto pela subtração de *features*, não sendo necessário iniciar com um modelo de uma peça bruta.

Nas abordagens de Projeto por *Features*, as peças são criadas diretamente usando *features* e o modelo geométrico é gerado do modelo de *features*. Isto requer que o sistema de projeto (CAD) tenha definições genéricas de *features* disponibilizadas pela Biblioteca de *Features*, permitindo a instanciação das *features* pela especificação de dimensões, parâmetros de localização, a feature/face/aresta sobre a qual está localizada e vários outros atributos (propriedades), restrições e relacionamentos.

#### 3.1. ISO10303-224: Definição de Produto para Planejamento de Processo

O AP 224 é um Protocolo de Aplicação desenvolvido para definição de produtos mecânicos voltado para o planejamento de processo computadorizado usando *features* de forma. O AP 224 inclui tolerâncias, propriedades do material e de superfícies e dados de controle da produção e administrativos. Uma *feature* de manufatura é definida como uma informação de produto necessária no planejamento de processo automatizado. Existem três categorias de *features* de manufatura, segundo o AP 224 (Shah e Mantyla, 1995): *features* de usinagem, *features* de transição e *features* repetidas.

Uma *feature* de usinagem é definida como uma forma que representa volumes para serem removidos por usinagem (p.ex. um furo). Uma *feature* de transição especifica uma área de transição entre duas superfícies; diferindo das *features* de usinagem em que nenhuma orientação de posicionamento é necessária (p.ex. chanfro). Uma *feature* repetida é especificada por uma feature base e um arranjo geométrico de cópias da feature base.

O AP 224 especifica onze categorias de Unidades de Funcionalidade que são necessárias para a atividade de planejamento de processo, como "*machining\_features*" e "*manufacturing\_part\_properties*". Uma "*machining\_feature*" (nomenclatura STEP - *Standard Exchange of Product Model Data*) é uma "unidade de funcionalidade (UoF) que contém as informações necessárias para identificar as formas que representam os volumes de material que devem ser removidos da peça por um processo de usinagem".

#### 3.2. ISO10303-48: Features de Forma

Segundo Shah e Mantyla (1995) existem vários tipos de *features* geométricas como: *Features* de Forma (descrevem porções da geometria nominal/idealizada da peça); *Features* de Tolerância (descrevem variação geométrica em relação a forma nominal); *Features* de Montagem (descrevem relacionamentos entre peças em uma montagem mecânica). Além das *Features* Geométricas temos as *Features* Funcionais e *Features* de Materiais.

Modelos de *features* são dependentes do domínio, ou seja da aplicação. Quando uma peça é projetada por *features*, o modelo resultante nem sempre é conveniente para uma dada aplicação, como para o planejamento de processo. Assim é necessário a transformação de *features* de uma visão para outra, sendo

este processo denominado de mapeamento de *features*.

O AP 48 define um Modelo de *Features* de Forma de Propósito Geral tendo por objetivo fornecer uma definição de *features* independente do domínio, ao contrário do AP 224 que é limitado ao escopo para produzir *features* pelos processos de torneamento e fresamento .

Na metodologia desenvolvida no presente trabalho, é utilizado como referência a definição de *features* associada ao AP 224, utilizando sempre que possível os conceitos associados ao AP 48.

#### 4. METODOLOGIA WEBMACHINING

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 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* padronizadas 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, sendo sua arquitetura estratificada em três níveis: Projeto, Planejamento do Processo e Manufatura.

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, lógica difusa, redes neurais e algoritmos genéticos. Este grupo de agentes interpreta as definições de projeto, modelagem geométrica realizada pelo usuário, e determina como fabricar a peça usando o Agente de Manufatura, que reside no nível inferior.

O Agente de Manufatura permite também a teleoperação da máquina de comando numérico via Internet. As informações sobre recursos de manufatura 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 (Álvares e Romariz, 2002) 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 distribuídos, 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 Internet 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.

A seguir são apresentadas algumas técnicas a serem utilizadas nos diversos módulos do sistemas, em especial nos módulos associados ao CAPP.

##### 4.1. Interface Gráfica com o Usuário (GUI)

A funcionalidade necessária para suportar operações baseadas em *features* numa interface para sistema CAD está associado a menus, opções de visualização, prompts e mensagens de erro, interações com o usuário de forma gráfica e textual, seleção de *features* e de bibliotecas de *features*, instanciação de *features*, manipulação das *features* entre outros (Bliznakov et al., 1995).

Os principais componentes do módulo GUI proposto são: Interface Gráfica com o Usuário (GUI), biblioteca de *features*, Interface Gráfica 2D, componentes de visualização 2D (primitivas gráficas) e 3D (Java 3D) processadas localmente na própria interface, componentes de visualização 2D (arquivos jpg e gif) e 3D (Kernel ACIS e VRML) com processamento remoto, Browser, plug-ins Java, plug-ins VRML, Applets Java e sockets TCP/IP.

## 4.2. Mapeamento de Features

O módulo de projeto WebMachining fornece uma entrada para o sistema de planejamento de processo. Para evitar dificuldades para o projetista em termos da entrada de dados a ser realizada a partir de menu, faz-se uso do conceito de *features* no nível do usuário e *features* no nível do modelo. Assim o usuário projeta a peça com um simples conjunto de *features* no nível do usuário disponibilizadas pela applet GUI. Estas *features* são convertidas para *features* no nível do modelo com maior detalhamento.

O ambiente de projeto baseia-se em *features* de alto nível como chanfro, furo, rasgo de chaveta, etc. que irão corresponder a operações de usinagem. O mapeamento de *features* de alto nível para *features* de baixo nível, que correspondem mais diretamente às operações de usinagem, será implementado por meio do procedimento de refinamento de *features* (Shah e Mantyla, 1995). Por exemplo, um eixo pode ser tratado como uma seqüência de segmentos, pois o processo de torneamento pode ser adequadamente modelado por um sistema gráfico 2D, sem nenhuma ambigüidade na sua representação.

Uma seqüência de segmentos é uma feature composta que consiste de uma coleção de segmentos cônicos e cilíndricos. Os segmentos têm a mesma linha de centro. Assim as *features* de forma, associadas à definição de instanciação da peça podem ser mapeadas para *features* de seqüência de segmentos e as operações de torneamento com seus diversos passes podem ser determinadas para operações de desbaste, semi-acabamento e acabamento. As *features* de transição podem também ser mapeadas, determinando-se os chanfros associados às *features* de seqüência de segmentos.

Uma vez que estes volumes a serem usinados foram identificados, é necessário determinar as operações de usinagem que podem produzi-los. Operações de torneamento podem ser modeladas em termos do tipo de arestas, ou seja, de um desenho que representa o perfil 2D da peça modelada por *features* de forma, segundo o seu contorno. A remoção de material necessária é obtida em função da forma da peça bruta especificada.

Uma abordagem complementar (Ferreira, 1996) é a utilização de tabelas que já apresentam roteamentos típicos de usinagem para superfícies cilíndricas externas, internas e planas. Um exemplo de roteamento de usinagem para uma determinada superfície poderia ser: torneamento (desbaste) => torneamento (semi-acabamento) => retificação (desbaste) => retificação (acabamento) => super-acabamento (Tolerância IT de 3-5 e Rugosidade de 0,008-0,16 µm).

## 4.3. Operações de Usinagem

As operações consideradas são torneamento, faceamento, mandrilamento, furação, sangramento e rosqueamento em um torno CNC. A metodologia de planejamento utilizada é uma combinação de dois métodos: uma árvore de decisão e um método heurístico usando regras de produção. Uma estrutura em árvore é construída através do seguinte procedimento: mapeamento de *features*; geração do volume a ser usinado e determinação da operação; modificação da geometria da peça.

As *features* normalmente presentes em peças torneadas são: cilindro, face, rosca, cônico, chanfro, arco, rasgo e entalhe. As *features* consideradas podem ser classificadas de forma geral em quatro categorias: rosca, chanfro e rasgo, perfil externo e perfil interno.

A determinação do volume de material a ser removido e a operação baseia-se no tipo de feature reconhecido pelo sistema, conforme descrito no item mapeamento de *features*. A cada feature reconhecida o perfil de usinagem e a operação podem ser determinados.

## 4.4. Seqüência de Operações de Usinagem

A determinação da seqüência de operações de usinagem baseia-se na análise de restrições, as quais podem ser de quatro tipos: restrições de operações, restrições geométricas, restrições de ferramentas e restrições de tolerâncias geométricas. Sundaram (1986) apresenta um método e descreve a solução através de manipulação de matrizes, o que é facilmente implementado em um computador.

Nesta técnica, as operações elementares de usinagem podem ser agrupadas numa seqüência e o grupo de operações elementares pode ser processado numa única máquina. A partir da especificação de operações de usinagem monta-se uma tabela de relacionamento de precedência de operações.

Do ponto de vista da manufatura existem algumas regras que devem ser seguidas visando determinar o seqüenciamento das operações: as duas faces planas externas e furos de centro devem ser usinados em primeiro lugar; operações de desbaste devem ser executadas antes das operações de acabamento; determinar as superfícies de sujeição; se um furo está presente em uma peça ele deve ser usinado primeiro.

## 4.5. Determinação das Condições de Usinagem

As condições de usinagem são obtidas através da consulta a uma base de dados que disponibiliza as informações de condições de usinagem e ferramentas baseado em dados de fabricantes de ferramentas. Faz-

se a seleção das condições de corte para uma dada combinação de materiais de peça e ferramenta, bem como, da geometria da ferramenta (Ferreira, 1996).

As restrições associadas à velocidade de corte, avanço, profundidade de corte, força de corte e potência de corte devem ser levadas em consideração (Huang, 1988) utilizando-se modelos de processos de usinagem para calculá-los. Trabalha-se com várias restrições para se determinar os parâmetros de corte conforme descrito em Halevi e Weill (1995) e Álvares (2001).

#### 4.6. Sistema de Fixação da Peça no Torno

É feita uma análise dos requisitos tecnológicos para seleção do sistema de fixação da peça (Huang, 1988) baseando-se nos mesmos procedimentos adotados por um processista na definição da fixação da peça: análise do desenho da peça (tolerâncias dimensional, geométrica e acabamento superficial); define quais os elementos geométricos a serem usinados no mesmo setup para garantir os requisitos tecnológicos; define o sistema de fixação para os diversos *setups*. Existem muitos métodos de fixação de peças simétricas em um torno, e alguns deles são apresentados em Halevi & Weill (1995) e Álvares (2001).

#### 4.7. Arquitetura Multi-Agente

A utilização de uma arquitetura baseada em sistemas multi-agentes (MAS) é 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 ao planejamento do processo. Por exemplo, pode-se utilizar um sistema de gerenciamento de base de dados relacional (MySQL® ou SQL®) para compartilhar as informações dos recursos disponíveis de manufatura (máquinas, ferramentas, informações de materiais, dispositivos de fixação, etc) e ter os agentes como os resolvidores das atividades de planejamento do processo. Os agentes podem ser implementados utilizando diversas abordagens na sua lógica de decisão: sistemas especialistas, lógica difusa, redes neurais, tabelas de decisão, entre outros.

Este ambiente em rede de agentes de softwares interoperáveis é denominado de Comunidade de Agentes de Manufatura (MAC). A figura 1 apresenta a arquitetura MAC que pode ser estratificada em três partes. No nível superior reside um grupo de Agentes de Projeto, o quais atuam como ferramentas CAD e também permite aos usuários que se conectem aos níveis inferiores. O nível intermediário consiste de Agentes de Planejamento. 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 selecionado, o qual reside no nível inferior. É utilizada uma interface transparente entre o projeto, planejamento e fabricação.

A partir deste fluxo de informações entre os três níveis da arquitetura é necessário realizar o encapsulamento destas informações, sendo adotado o KQML (<http://www.cs.umbc.edu/kqml>) como padrão de linguagem de mensagens entre os agentes. KQML contém informações sobre quais agentes enviam mensagem, onde eles estão, como interpretar a mensagem recebida pelo destinatário. As mensagens são trocadas usando conexões sockets diretas entre os agentes.

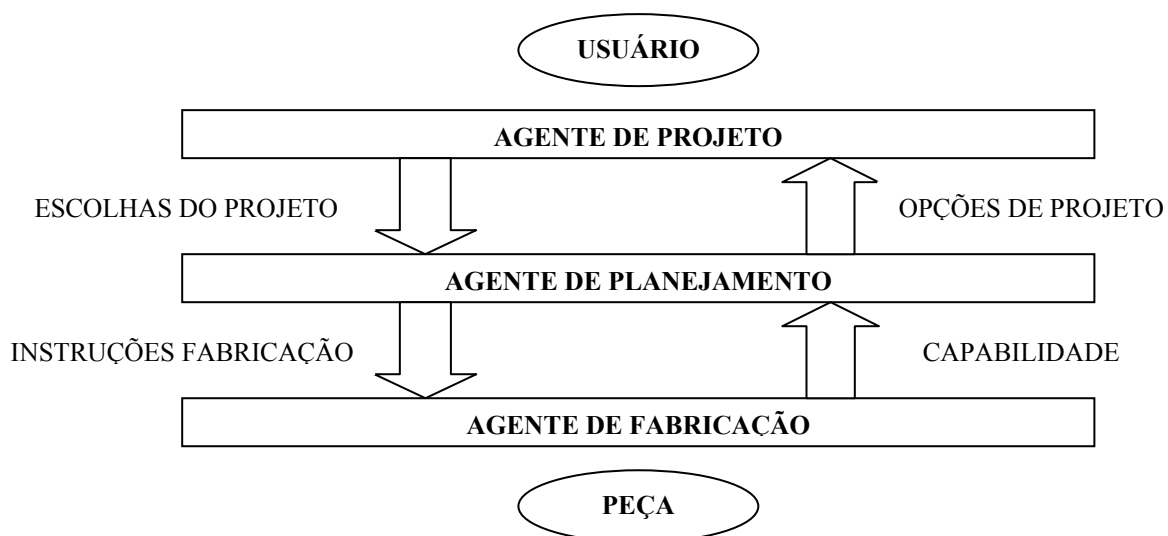


Figura 1. Sistema Multi Agente MAC (Smith e Wright, 2001)

É prevista a utilização da 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 com um Agent Message Router (AMR) usam um nome e um *password*, conectando-se e desconectando-se da Internet, mandando e recebendo mensagens, transferindo arquivos com FTP, e geralmente trocando informações com outros agentes através dos vários computadores onde eles estão sendo executados. Esta ferramenta foi utilizada na implementação do CyberCut. JATLite permite o desenvolvimento de uma infra-estrutura para Typed-Message Agents, definido nos termos de uma comunidade de agentes.

#### 4.8. Teleoperação Máquina-Ferramenta CNC

O sistema de Teleoperação do Torno, denominado WebTurning, é baseado em uma arquitetura Cliente-Servidor utilizando metodologia desenvolvida por Álvares e Romariz (2002) sendo constituído por dois módulos (Álvares, et al., 2002): Servidor, representado pelos programas localizados na estação de trabalho (computador pessoal com Sistema Operacional Linux), conectado à máquina-ferramenta através da interface RS-232C ou Ethernet; cliente, representado por um Applet Java.

O servidor de Teleoperação WebTurning é constituído pelo servidor de vídeo e servidores de teleoperação da máquina que disponibilizam serviços de comando, execução de programas, download e upload de programas, tratamento de erros e demais funções associadas ao protocolo de comunicação disponibilizado pelo fabricante. O servidor de vídeo é responsável pela captura de vídeo e sua distribuição através do protocolo TCP/IP (Internet). Os demais servidores, associados aos serviços de teleoperação, propriamente dito, trabalham de modo bidirecional, recebendo comandos através da Internet e enviando dados de status da máquina.

### 5. DIAGRAMAS IDEF0 DO SISTEMA WEBMACHINING

O sistema integrado CAD/CAPP/CAM - WebMachining foi estruturado através da metodologia IDEF0. A figura 2 apresenta todas as entradas, controles, mecanismos, saídas e funcionalidades que foram propostas.

A figura 3 detalha um pouco mais o sistema WebMachining, "explodindo-o" em três módulos associados às suas principais funcionalidades: CAD by *Features*, CAPP e CAM. A figura 4 apresenta as atividades associadas ao módulo CAD by *Features*. A figura 5 descreve as atividades do módulo CAPP e a figura 6 descreve as atividades do CAM.

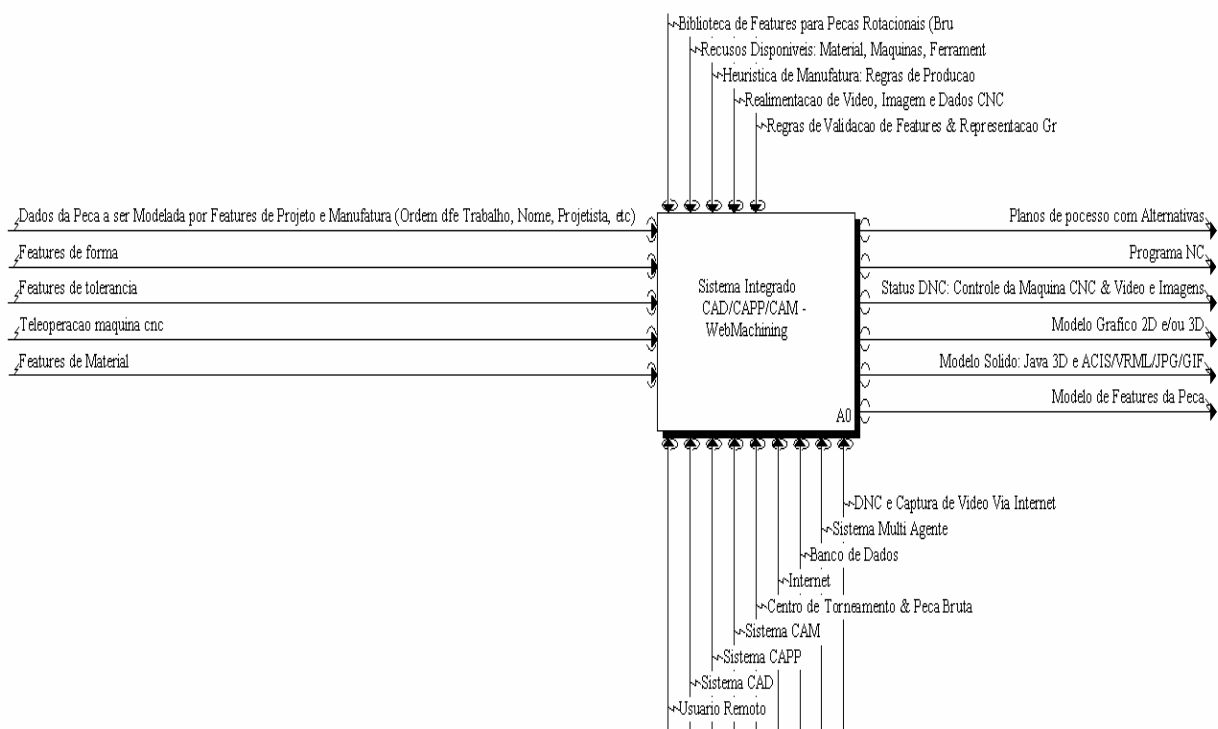


Figura 2. Modelo IDEF0 do sistema integrado CAD/CAPP/CAM - WebMachining



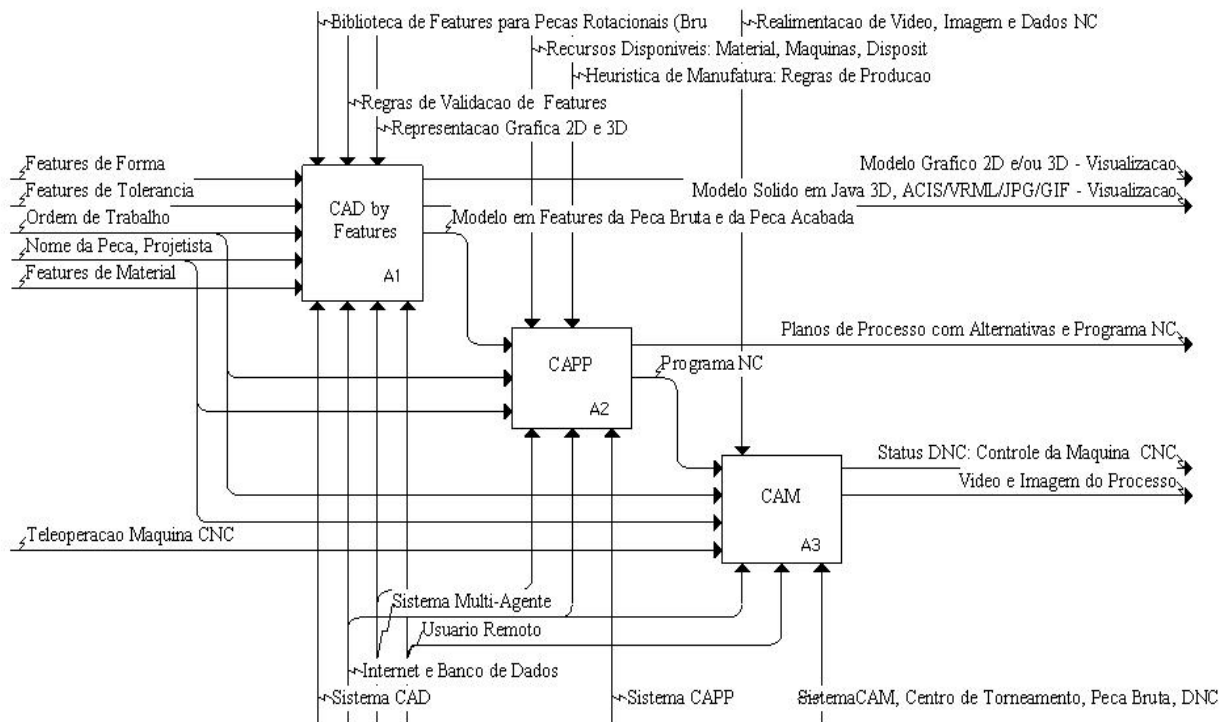


Figura 3. IDEF0: Os três principais módulos do Sistema WebMachining

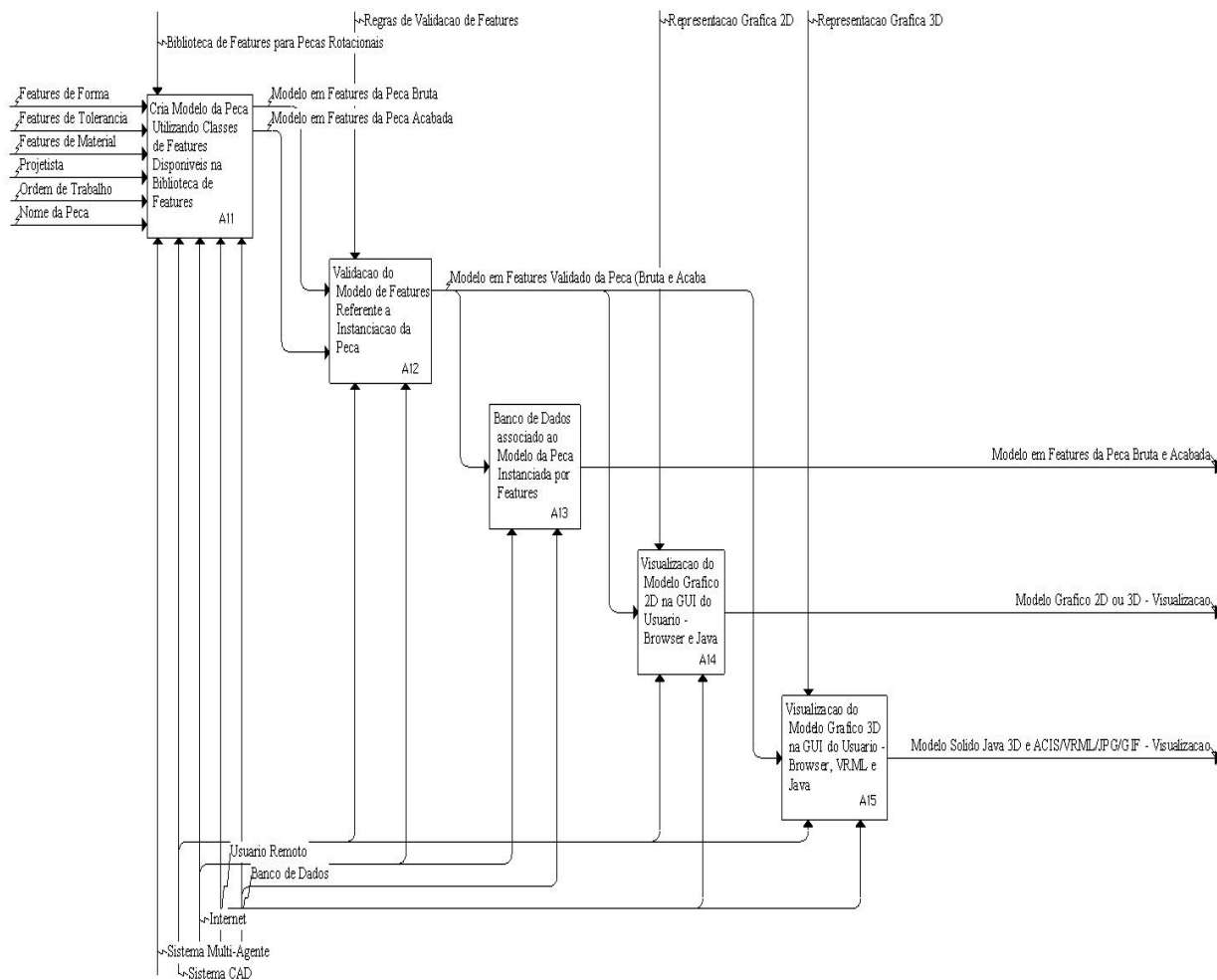


Figura 4. IDEF0: Atividades associadas ao módulo CAD by Features

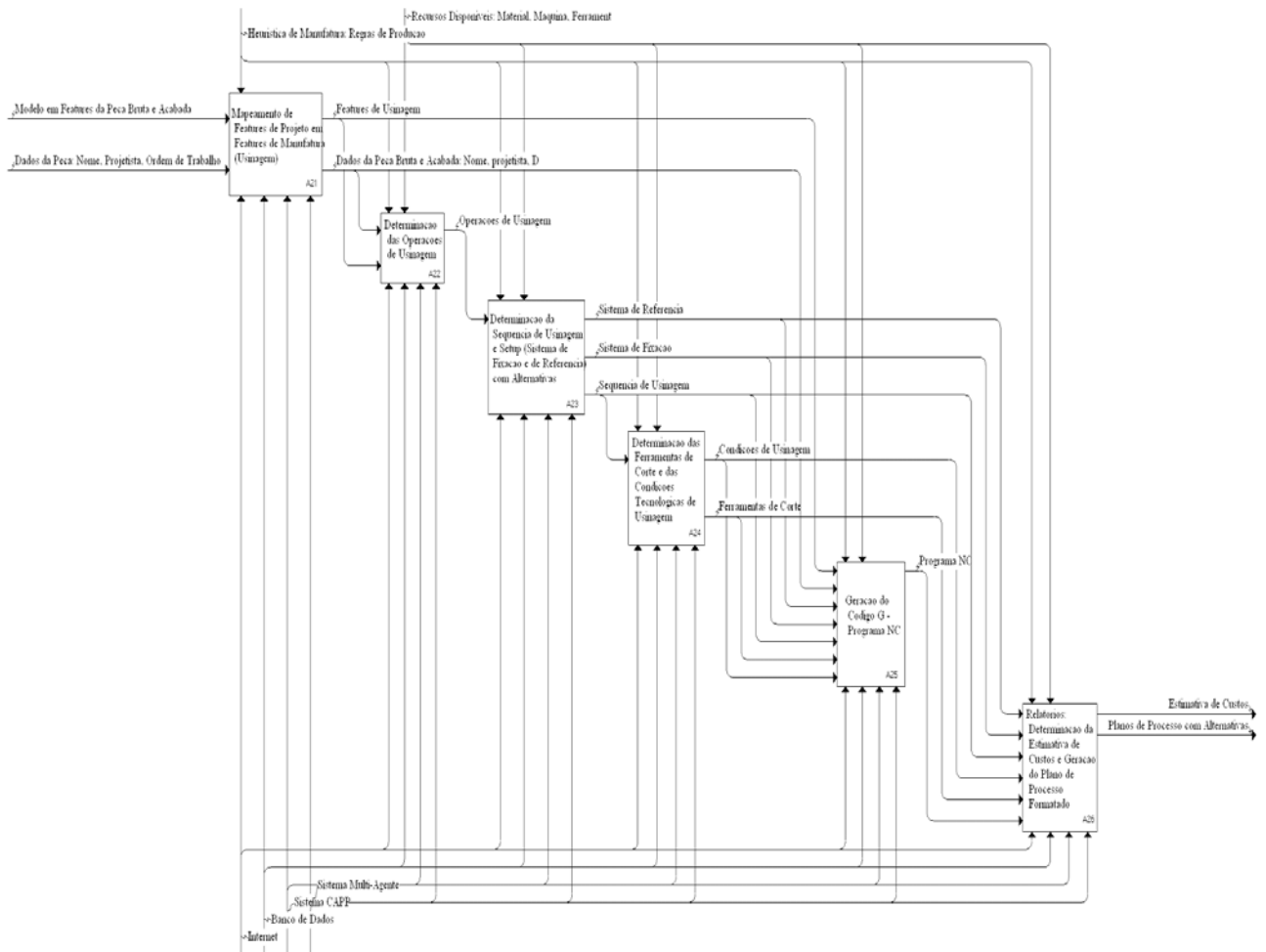


Figura 5. IDEF0: Atividades associadas ao módulo CAPP

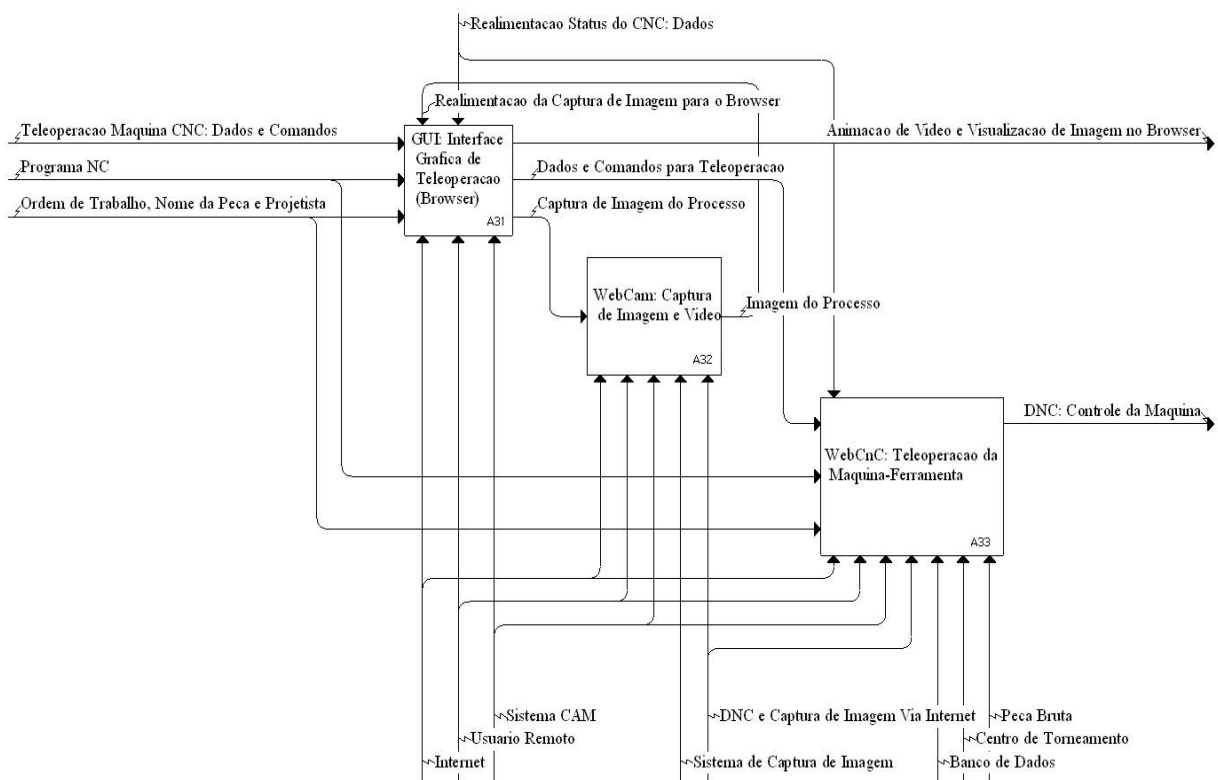


Figura 6. IDEF0: Atividades associadas ao módulo CAM (Teleoperação - WebTurning)

## 6. CONCLUSÃO

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. Estes ambientes serão globais, centrados em rede e espacialmente distribuídos, tendo como front-end com o sistema CAD/CAPP/CAM navegadores baseados na Web e a linguagem Java, permitindo a independência da plataforma computacional do usuário.

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 manufatura através da modelagem por *features* e utilizando os protocolos TCP/IP. O sistema é inédito no que concerne à 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 associada 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 2004. Maiores informações sobre o projeto WebMachining podem ser consultadas em <http://WebMachining.AlvaresTech.com> ou <http://WebMachining.graco.unb.br>.

## 7. REFERÊNCIAS

- 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., 2002, "Monografia da Disciplina de Estudo Dirigido: TeleManufatura Aplicada a Operações de Usinagem", Programa de Pós-Graduação em Engenharia Mecânica, UFSC, <http://WebMachining.AlvaresTech.com/telemanufatura>.
- 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.
- Álvares, A. J., 2001, "Monografia da Disciplina de Estudo Dirigido: Métodos para o Projeto, Planejamento do Processo e Fabricação de Peças Assistidos por Computador", Programa de Pós-Graduação em Engenharia Mecânica, UFSC, <http://WebMachining.AlvaresTech.com/capp>.
- Bidarra, R., Van den Berg, E. e Bronsvort, W. F., 2001, "Collaborative Modeling with Features", Proceedings of DET'01, 2001 ASME Design Engineering Technical Conferences, Pittsburgh, USA.
- Bliznakov, P., Rogers, M., Ali, A., Shirur, A., Shah, J., 1995, "User interface design for feature modelers and feature based applications", ASME Computers in Engineering Conference, Boston, USA.
- Cay, F. e Chassapis, C., 1997, "An IT view on perspectives of computer aided process planning research", Computers in Industry, 34, págs 307-337.
- 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.
- Ferreira, J.C.E., 1996, "Planejamento do Processo Assistido por Computador - CAPP", Apostila, Universidade Federal de Santa Catarina, Departamento de Engenharia Mecânica, Florianópolis.
- 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.
- Halevi, G. e Weill, R.D., 1995, "Principles of Process Planning: A Logical Approach", Chapman & Hall.
- Ham, L. e Lu, S.C.Y., 1988, "Computer-aided process planning: the present and the future", Anais CIRP 37 (2), págs 591-601.
- 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.
- Harrington, J.Jr., 1984, "Understanding the Manufacturing Process - Key to Successful CAD/CAM Implementation", Marcel Dekker, New York.
- Hartley, J. R., 1992, "Concurrent Engineering - Shortning Lead Times, Raising Quality and Lowering Costs", Productivity Press (1992), Capítulo 2.
- Huang, H., 1988, "A Generative Process Planning System for Turned Components". Tese de Doutorado, University of Manchester Institute of Science and Technology, Inglaterra.

- 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 1999.
- 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.
- 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.
- Sundaram, R. M., 1986, "Process Planning and Machining Sequence", Computers & Industrial Engineering, Vol. 11, págs 27-31.
- 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.

## **A CAD/CAPP/CAM INTEGRATION METHODOLOGY FOR THE REMOTE MANUFACTURE OF CYLINDRICAL PARTS THROUGH THE WEB**

**Alberto José Álvares**

Universidade de Brasília, Departamento de Engenharia Mecânica e Mecatrônica, Grupo de Automação e Controle (GRACO), 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 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. This methodology specifies the modules of the system, and how these modules interact with each other. In this article it is presented the architecture for implementing the integrated system CAD/CAPP/CAM applied to the remote manufacture of cylindrical parts, consisting of a group of specifications, techniques and algorithms used for the functional definition of the modules of the system. 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. 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.