

# UM SISTEMA PARA O SUPORTE AO PROJETO REMOTO DE PEÇAS PRISMÁTICAS BASEADO EM FEATURES VIA INTERNET

**Julio César Ticona Benavente**

Universidade Federal de Santa Catarina, Departamento de Engenharia Mecânica,  
GRUCON/GRIMA, caixa postal 476, CEP 88040-900, Florianópolis, SC, metrojulio@yahoo.es

**João Carlos Espíndola Ferreira**

Universidade Federal de Santa Catarina, Departamento de Engenharia Mecânica,  
GRUCON/GRIMA, caixa postal 476, CEP 88040-900, Florianópolis, SC, jcarlos@emc.ufsc.br

**Resumo:** *A globalização tem causado um aumento significativo na competição entre as empresas, mas em contrapartida ela também permite a cooperação entre empresas, mesmo que estejam geograficamente distantes umas das outras. Uma área de pesquisa decorrente dessa aproximação virtual é a fabricação de peças a distância, que conta com a participação dos seguintes atores: (i) um cliente remoto que introduz as encomendas; (ii) uma empresa modeladora responsável pelo desenvolvimento e disponibilização de módulos computacionais para o projeto das peças e para o planejamento das operações a serem executadas sobre as peças encomendadas; e (iii) uma empresa onde a fabricação propriamente dita será efetuada. Deve-se mencionar que estes atores poderão estar localizados em qualquer lugar do mundo. No presente trabalho será descrita uma metodologia para a modelagem de peças prismáticas a distância utilizando a Internet como meio de comunicação. Será apresentado o desenvolvimento de uma interface gráfica disponibilizada para os clientes remotos, sem que eles necessitem instalar nos seus computadores softwares comerciais de CAD. A modelagem das peças será realizada através de uma biblioteca de features, que na presente versão inclui as seguintes features: furo, degrau, ranhura e cavidade retangular. Este módulo computacional será responsável também por verificar a consistência da peça do ponto de vista de manufaturabilidade, isto é, ao introduzir uma feature com os seus respectivos parâmetros, o software verificará se a peça resultante poderá ser fabricada.*

**Palavras chave:** *Features, CAD, Manufaturabilidade, Internet, Cliente Remoto.*

## 1. INTRODUÇÃO

Há vários anos, as empresas de manufatura em todo o mundo têm atuado num ambiente bastante competitivo, em que produtos de qualidade equivalente ou até melhor, com bons preços, têm concorrido pela conquista do mercado consumidor. Ao mesmo tempo, esta competição causada pela globalização tem levado empresas a buscar cooperações com outras empresas para manterem-se competitivas neste mercado<sup>(1)</sup>, Um fator que tem acelerado ainda mais a globalização é a Internet, através da qual o consumidor pode vender ou comprar produtos sem sair de sua casa ou local de trabalho. Uma vez que a Internet permite a aproximação virtual entre pessoas ou empresas localizadas geograficamente distantes entre si, ela pode também ser utilizada como tecnologia para permitir a fabricação de peças a distância. Este tipo de fabricação é motivado pelo fato do cliente não necessariamente possuir os equipamentos e acessórios para a fabricação do produto. Do ponto de vista da empresa que executa a manufatura, ao mesmo tempo em que ela entra diretamente em

contato com o cliente, ao responder prontamente à solicitação quanto à qualidade do produto e tempo de entrega, ela pode assim não somente manter o seu nicho de mercado, mas também aumentá-lo.

Neste artigo descreve-se o desenvolvimento de um sistema para o projeto de peças prismáticas a distância, e o mesmo poderá ser executado via Internet por pessoas em qualquer lugar do mundo.

A modelagem das peças prismáticas será feita através da biblioteca de *features* disponibilizadas pelo sistema. Na atual versão deste software, são disponibilizadas nesta biblioteca as seguintes *features*, que são normalmente encontradas em peças mecânicas: furo, degrau, ranhura e cavidade retangular. Ao serem introduzidas cada uma das *features*, é feita uma verificação da consistência da peça resultante do ponto de vista de manufaturabilidade, de maneira que a peça possa ser fabricada na máquina sendo considerada.

Neste sistema os dados são armazenados numa estrutura de classes implementadas na linguagem Java, e também foi desenvolvida em Java uma interface gráfica para a visualização das peças criadas.

## 2. REVISÃO BIBLIOGRÁFICA SOBRE PROJETO E FABRICAÇÃO DE PEÇAS VIA INTERNET

Segundo Ferreira e Andriolli<sup>(1)</sup>, a Internet pode ser considerada um meio que possibilita uma cooperação direta entre o cliente e empresas de manufatura, e pode também ser usada no suporte às atividades da empresa de manufatura, como, por exemplo, na cooperação entre diferentes departamentos de uma mesma empresa, buscando reduzir os custos de fabricação e a melhoria da qualidade do produto. Uma outra aplicação da Internet é no suporte à fabricação de peças a distância. Um modelo deste procedimento foi proposto por Ferreira e Andriolli<sup>(1)</sup>.

Alguns esforços de pesquisa têm sido empregados no desenvolvimento de sistemas de projeto de peças voltados para a Internet. Shah et al.<sup>(2)</sup> desenvolveram uma arquitetura para padronização da comunicação entre o núcleo de um sistema de modelagem geométrica e as aplicações. Smith e Wright<sup>(3)</sup> 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. Martino et al.<sup>(4)</sup> 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*.

*Features* podem ser definidas como uma forma de representar o significado da geometria de uma peça ou de uma montagem (Shah e Mantyla<sup>(5)</sup>). Exemplos de *features* podem ser furos, chanfros, ranhuras, degraus, etc.

O sistema WebSpiff<sup>(6)</sup> 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<sup>(7)</sup>; Gerenciador de Sessão que fornece funcionalidade para iniciar, associar-se, sair e fechar uma sessão de modelagem, bem como gerencia todas as comunicações entre o sistema SPIFF e os clientes.

Ferreira e Vivian<sup>(8)</sup> desenvolveram um sistema para o projeto de peças rotacionais via Internet, onde as peças são modeladas no computador servidor através de sólidos usando o *kernel* ACIS, aplicando-se JavaServer Pages (JSP)<sup>(9)</sup>. Álvares<sup>(10)</sup> implementou um sistema computacional para o projeto colaborativo de peças rotacionais via Internet usando a abordagem de agentes para a colaboração, e a modelagem das peças sendo feita através de VRML (*Virtual Reality Markup Language*).

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<sup>(5)</sup> 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, etc.) para que estes possam realizar suas correspondentes 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<sup>(11,12)</sup>.

### 3. SISTEMA PARA O PROJETO E FABRICAÇÃO DE PEÇAS VIA INTERNET

Segundo Shah e Mantyla<sup>(5)</sup>, 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 das mesmas.

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.

Na abordagem de projeto por *features*, as peças são criadas diretamente usando *features*, e o modelo geométrico é gerado do modelo de *features*, e isto requer que o sistema de projeto (CAD) tenha definições genéricas de *features* disponibilizadas pela biblioteca de *features*, permitindo a instanciação das mesmas pela especificação de dimensões, parâmetros de localização, a *feature/face/aresta* sobre a qual está localizada e vários outros atributos ou propriedades, restrições e relacionamentos. A abordagem utilizada para desenvolver este aplicativo de CAD é o *Destructive by Machining Feature*.

Este sistema computacional foi desenvolvido utilizando a linguagem Java<sup>(13)</sup>, pois ela é uma linguagem orientada a objetos e que permite o desenvolvimento de aplicativos que podem ser executados através da Internet.

A estruturação deste aplicativo foi desenvolvida utilizando a modelagem baseada na metodologia IDEF (*Integration Definition for Function Modeling*), que é uma abordagem gráfica para a descrição de sistemas. Na figura 1 é apresentado o diagrama IDEF correspondente a este aplicativo.

As três atividades deste sistema CAD são:

- A criação do modelo da peça bruta e acabada;
- A validação do modelo da peça;
- O modelador gráfico no lado do cliente remoto.

A seguir serão descritas cada uma dessas atividades.

#### 3.1. Criação do Modelo da Peça

O modelo da peça bruta é criado pela instanciação de um novo bloco o qual está constituído por seis faces. No momento da criação de um novo bloco, são criadas automaticamente cada uma das seis faces, ou seja, são especificados os valores dimensionais de comprimento e largura para cada uma. Para isto existe uma estrutura de dados organizados em classes. A classe chamada Bloco possui um vetor com seis elementos, cada elemento representa uma face do mesmo. Também existe uma classe chamada Face, a qual contém todos os dados relativos à sua orientação e suas dimensões, e tem funções ou métodos que retornam ao programa dados necessários para saber a quais são as suas quatro faces que são ortogonais e qual é a face paralela. Um atributo importante da classe Face é um

vetor chamado “*features*”, o qual no momento da criação de um novo bloco, se encontra vazio. Este vetor é utilizado para armazenar as *features* criadas pelo usuário. Note-se que cada uma das faces possui este vetor, ficando deste modo o bloco decomposto por faces. A figura 2 apresenta um diagrama da composição de um bloco.

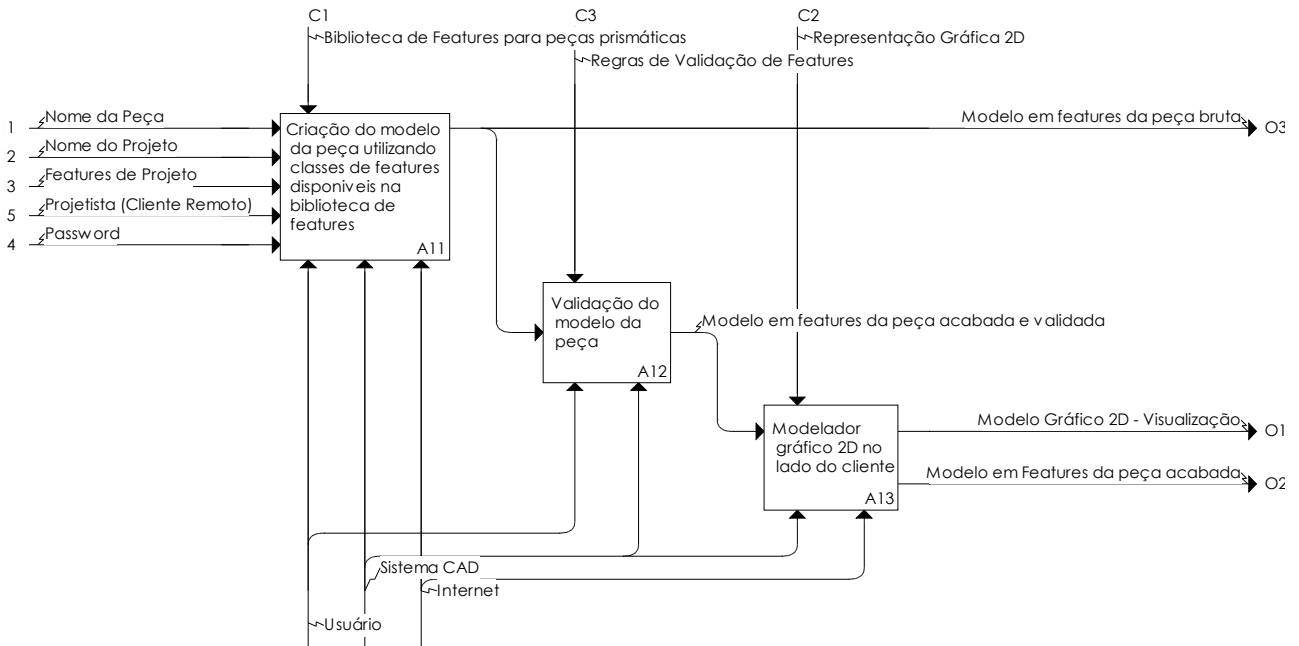


Figura 1. Diagrama IDEF0 do aplicativo CAD

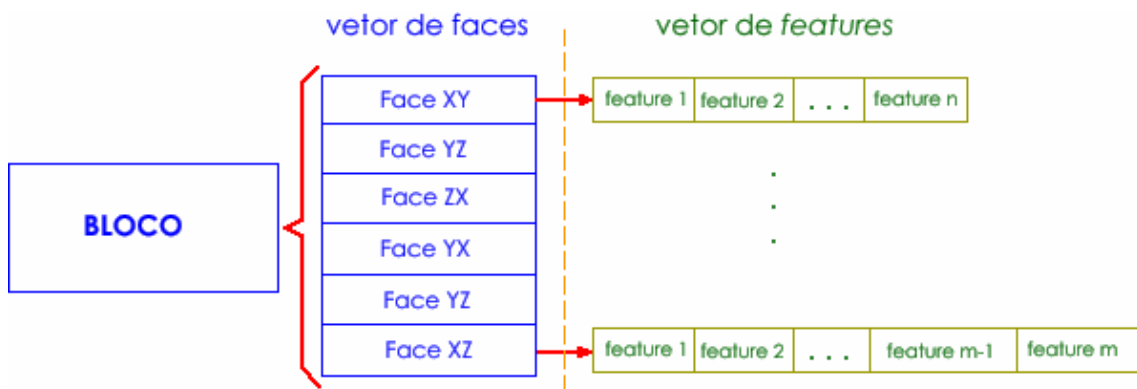


Figura 2. Bloco composto de vetores Face

Uma *feature* está definida neste sistema CAD pela classe “*Feature*”, a qual basicamente contém dados que são comuns a todos os tipos de *feature*, como por exemplo, a face na qual foi criada, ou a posição dentro da face na que está localizada.

O presente sistema CAD disponibiliza em sua versão atual as seguintes *features*: Furo, Degrau, Ranhura e Cavidade Retangular. Estas *features* são instanciadas pelo usuário do sistema, quando ele introduz certos parâmetros necessários, os quais são apresentados na figura 3 em um diagrama UML.

As classes Furo, Degrau, Ranhura e Cavidade são filhas da classe *Feature*, e cada uma delas herda os métodos da classe mãe, e adiciona dados úteis segundo o tipo de *feature* particular.

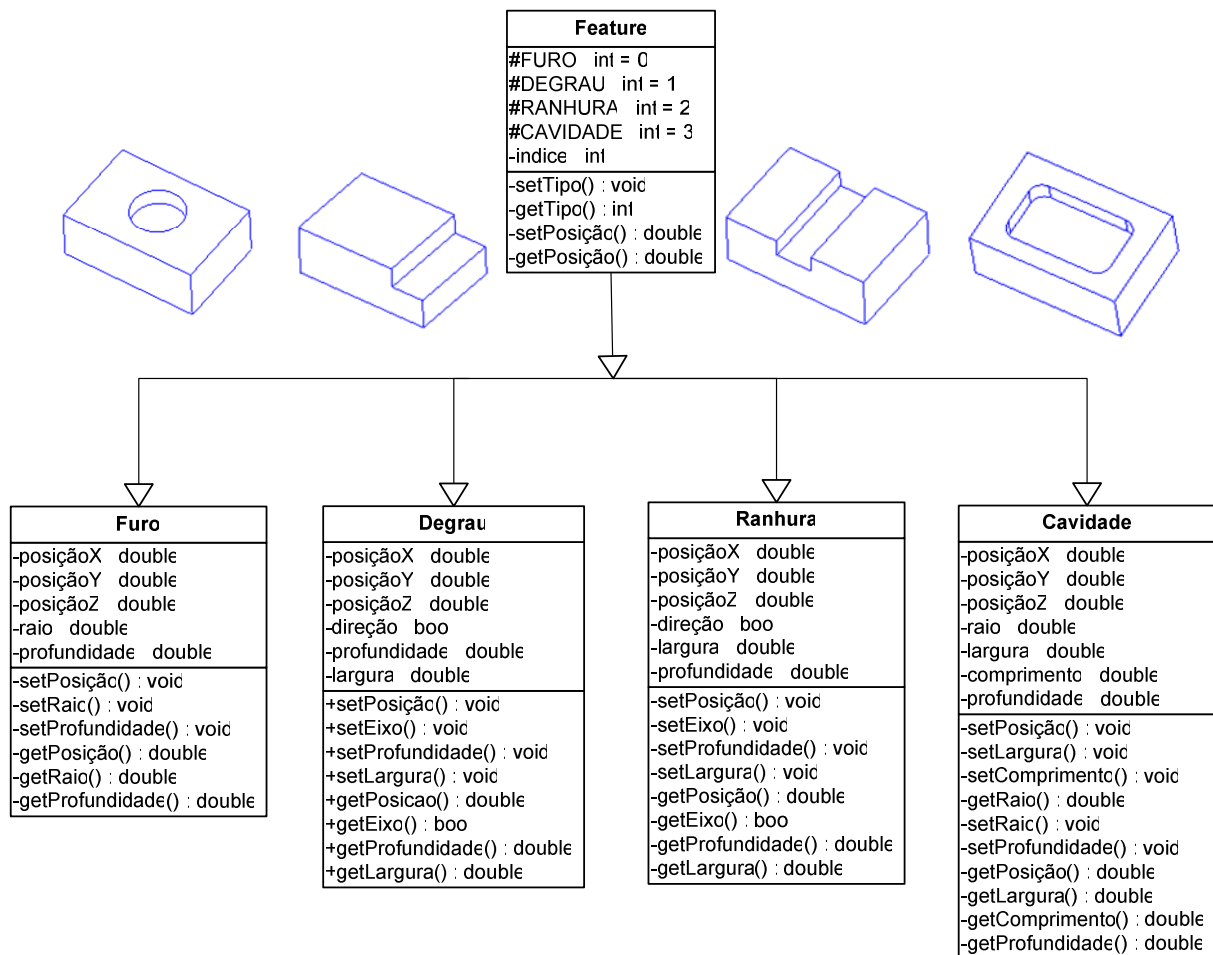


Figura 3. Diagrama UML da biblioteca de *features*

As ferramentas que o sistema disponibiliza estão associadas a menus, botões para a visualização da peça em diferentes vistas laterais, etc. Uma destas ferramentas é a que permite ao usuário do sistema alterar os pontos de referência da *feature* a ser criada. Por exemplo, na figura 4a o projetista colocou as referências do furo a partir do vértice inferior direito. Para criar este furo, ter-se-ia que calcular as cotas “a” e “b” (figura 4b), porém se a peça for girada 90 graus em sentido anti-horário de modo que a peça fique como na figura 4c, então as referências originais podem ser utilizadas ( $a_1 = b_0$  e  $b_1 = a_0$ ).

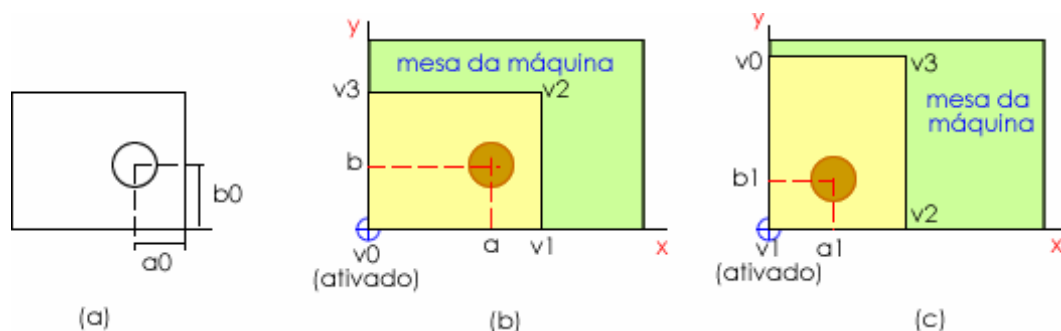


Figura 4. Modificações das referências de projeto

### 3.2. Validação do modelo da peça

Antes de passar os dados para o modelador gráfico, é necessária fazer uma avaliação da consistência das *features* inseridas na peça prismática. Para isto, algumas regras de validação foram implementadas neste aplicativo CAD para não permitir a criação de *features* inconsistentes, e estas regras são:

(a) uma *feature* é válida se ela está contida dentro dos limites da face de trabalho; se o usuário tentar criar uma *feature* que ultrapasse os limites máximos da face, o programa emitirá uma mensagem de erro. A figura 5a apresenta um furo válido, enquanto a figura 5b apresenta um furo inválido, e neste último caso o programa não que esta *feature* seja criada.

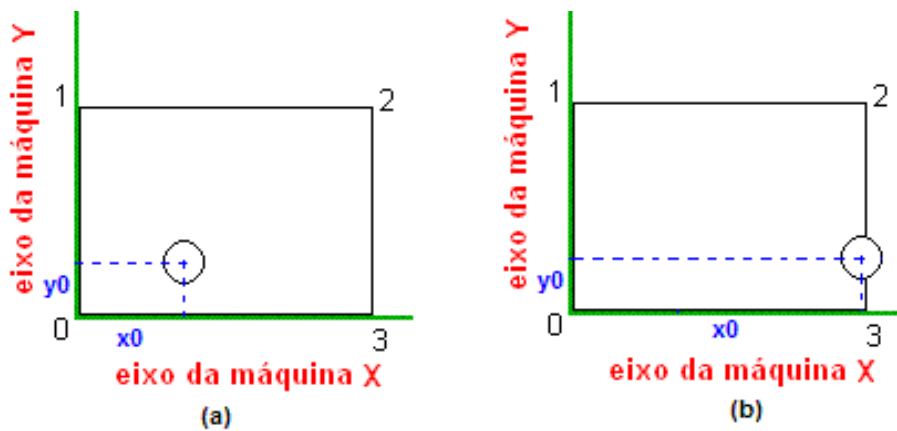


Figura 5. a) Furo válido, b) Furo inválido

(b) a posição Z da nova *feature* deve ser consistente. Ou seja, a posição Z da nova *feature* deve coincidir com a superfície da face da peça bruta. Na figura 6a é apresentado um furo válido, enquanto os outros dois não são válidos.

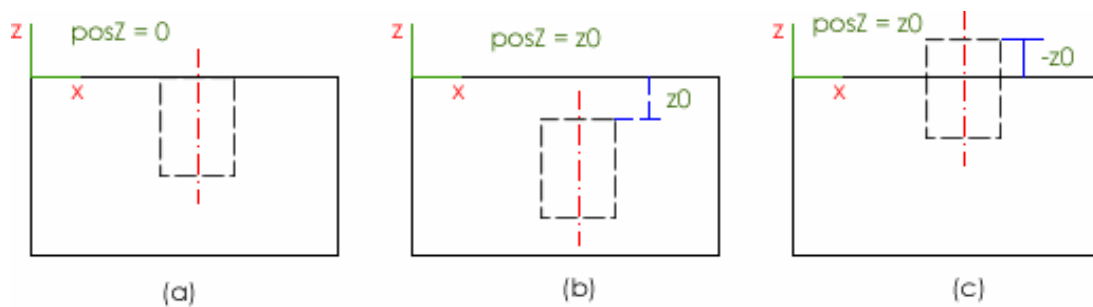


Figura 6. (a) furo válido, (b) e (c) furos inválidos

O sistema permite a criação de *features* aninhadas, desde que a posição Z da nova *feature* a ser aninhada dentro da outra coincida com a superfície base da *feature* anterior (figura 7).

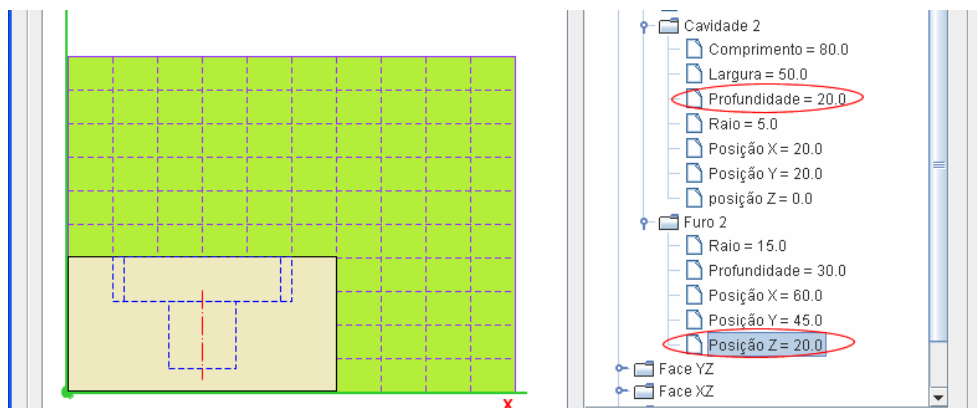


Figura 7. Furo aninhado dentro de uma cavidade retangular

(c) a *feature* não pode ter profundidade maior do que a profundidade do bloco (figura 8).

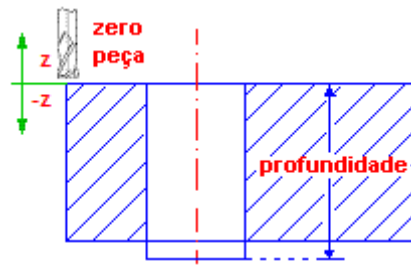


Figura 8. Inconsistência devida à profundidade

(d) uma regra que avalia a criação da *feature* do ponto de vista da manufacturabilidade é que uma *feature* não pode ser criada se, com a ferramenta disponível, ela não puder ser usinada (figura 9).

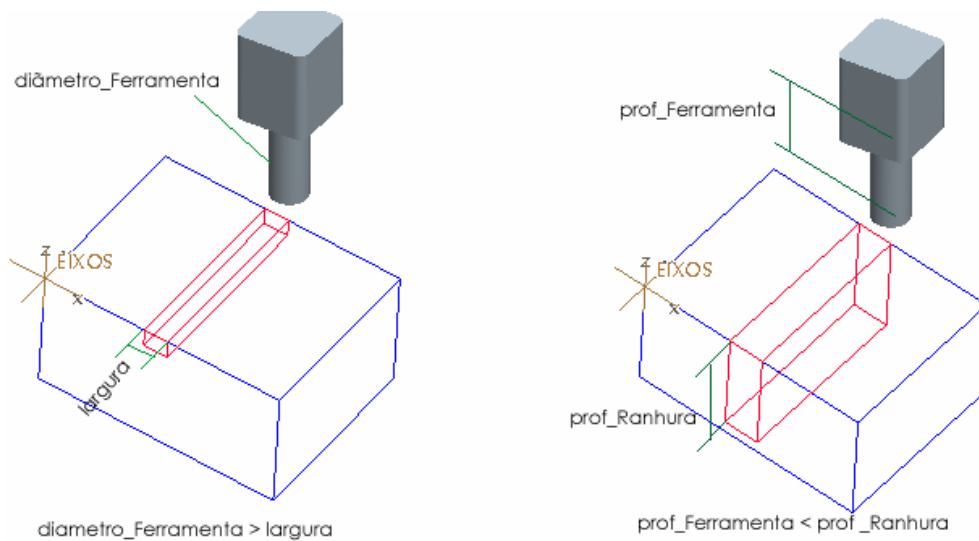


Figura 9. Inconsistência na *feature* ao ser comparada com a geometria da ferramenta

(e) não é possível criar uma *feature* em uma face que não possa ser fixada dentro da máquina. Por exemplo, na figura 10, pode se notar que é possível criar *features* na face XY, mas não na YZ.

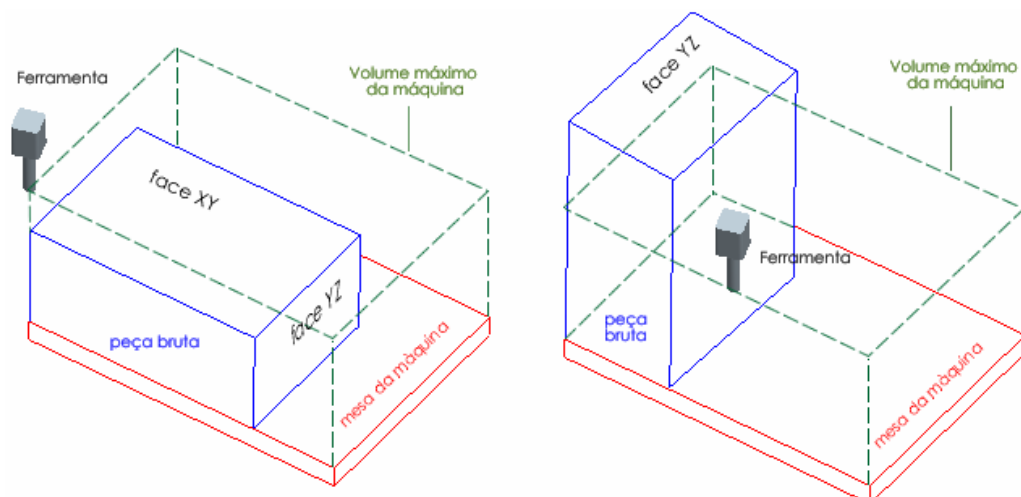


Figura 10. Inconsistência na criação da *feature* devido à máquina

### 3.3. Modelador Gráfico no Lado do Cliente

No lado do cliente remoto, o “*applet*” (aplicativo na linguagem Java) é executado, e o modelador gráfico percorre o vetor de faces e os vetores de *features* de cada uma das faces do bloco

para verificar se em alguma das faces existem *features*. No caso de *features* estarem presentes, o modelador cria a representação gráfica em 2D do bloco e das *features* por faces. Caso houver a presença de *features* em várias faces, o aplicativo CAD efetua a representação das *features* que não estão na face visualizada através de linhas pontilhadas, como ilustrado na figura 11.

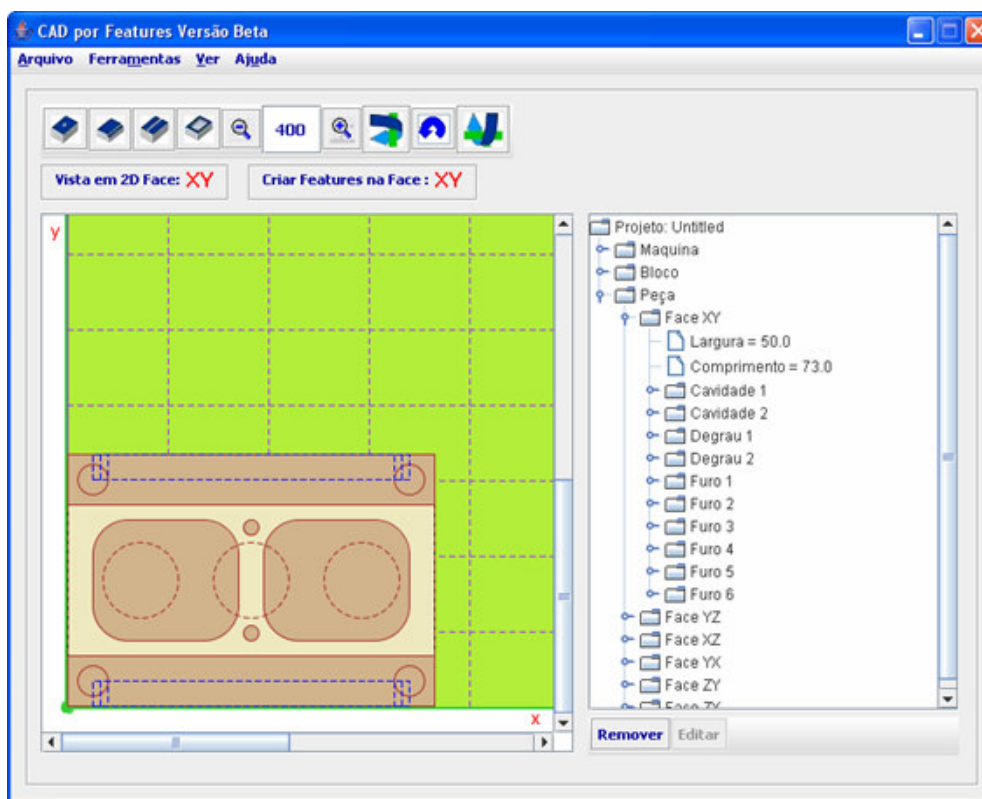


Figura 11. Exemplo de peça modelada no sistema

#### 4. RESULTADOS

Na figura 11 é mostrado um exemplo ilustrando uma peça prismática simples modelada por este aplicativo de CAD baseado em *features*. As *features* presentes nesta peça estão apresentadas na tabela 1.

Tabela 1. *Features* presentes na peça modelada

Face	Furo	Degrau	Ranhura	Cavidade	Total features
XY	6	2	0	2	10
YZ	-	-	-	-	-
XZ	0	0	0	4	4
YX	3	0	1	0	4
ZY	-	-	-	-	-
ZX	0	0	0	4	4

O sistema disponibiliza janelas para a visualização da peça em diferentes faces (nos eixos X, Y, e Z), informando ao usuário a face que está sendo visualizada no momento atual e a face na qual ele está introduzindo as *features*. Deve-se notar que os eixos que aparecem na figura 11 (no caso, X e Y) são os eixos absolutos da máquina de usinagem e por isso ficam fixos. A área de cor verde com linhas pontilhadas representa a área da mesa de trabalho da máquina, e o usuário remoto tem a possibilidade de desabilitar esta área se ele desejar. O lado direito do aplicativo contém a



representação da peça prismática baseada em *features*, onde pode-se notar que cada uma das *features* criadas aparecem com os seus parâmetros. Também são apresentados os dados referentes à máquina onde a peça será fabricada (suas dimensões máximas de largura, comprimento e profundidade), além dos dados relativos à peça bruta.

## 5. CONCLUSÕES

O sistema CAD proposto neste trabalho utiliza a Internet e sistemas de comunicação para contribuir para o desenvolvimento de um ambiente integrado CAD/CAPP/CAM. O sistema proposto, que pode ser executado no link <http://www.grima.ufsc.br/software/CAD>, contribui para o desenvolvimento de sistemas de projeto de peças prismáticas baseados na *web*, podendo depois ser utilizado para integrar as atividades de projeto, planejamento de processos e de fabricação através da modelagem por *features*.

Como trabalho futuro pretende-se disponibilizar mais *features* na biblioteca, como por exemplo ranhura cega, cavidade genérica e entalhe.

Como os parâmetros das *features* neste sistema estão disponíveis, o módulo de CAPP, que estará sendo desenvolvido no futuro, poderá selecionar as ferramentas e estratégias de usinagem mais adequadas para a fabricação de cada uma das *features*.

## 6. REFERÊNCIAS

1. FERREIRA, J.C.E. e ANDRIOLLI, G.F., "Uma Metodologia para a Fabricação de Peças à Distância", Revista Máquinas e Metais, Aranda Editora, Ano 41, No 476, p. 84-99, setembro 2005.
2. SHAH, J. J., DEDHIA H., PHERWANI V. e SOLKHAN S., "Dynamic Interfacing of Applications to Geometric Modeling Services Via Modeler Neutral Protocol", Computer-Aided Design, 29, p. 811-824, 1997.
3. SMITH, C.S. e WRIGHT, P.K., "Cybercut: An Internet-based CAD/CAM System", ASME Journal of Computing and Information Science in Engineering, Vol. 1, No. 1, p. 1-33, 2001.
4. MARTINO, T.D., FALCIDIENO, B. e HASINGER, S. "Design and Engineering Process Integration Through a Multiple View Intermediate Modeller in a Distributed Object-oriented System Environment", Computer-Aided Design, 30(6), p. 437-452, 1998.
5. SHAH, J. J. e MÄNTYLÄ, M., "Parametric and Feature-Based CAD/CAM: Concepts, Techniques, and Applications", John Wiley & Sons, New York, 1995.
6. BIDARRA, R., VAN DEN BERG, E. e BRONSVOORT, W.F., "Collaborative Modeling with Features", Proceedings of DET'01, 2001 ASME Design Engineering Technical Conferences, Pittsburgh, EUA, 2001.
7. CORNEY J. e LIM T., "3D Modeling with ACIS", Saxe-Coburg Publications, 2a Edição, 2001.
8. FERREIRA, J.C.E. e VIVIAN, D., "Feature Recognition in Axisymmetrical Parts Modeled by Solids in an Internet-Oriented CAD/CAM System", Journal of Materials Processing Technology, Vol. 179, No. 1, 2006.
9. SUN MICROSYSTEMS, "JavaServer Pages™", <http://java.sun.com/products/jsp>, último acesso em 7/8/2006
10. ÁLVARES, A.J., "Uma Metodologia para Integração CAD/CAPP/CAM Voltada para Manufatura Remota de Peças Rotacionais Baseada na Internet", Tese de Doutorado, Universidade Federal de Santa Catarina, Florianópolis, SC, outubro de 2005.
11. TÖNSHOFF, H.K., AURICH, J.C. e BAUM, T., "Configurable Feature-Based CAD/CAPP System". Proceedings of the IFIP International Conference on Feature Modeling and Recognition in Advanced CAD/CAM Systems. Valenciennes, France, p. 757-769, 1994.
12. SALOMONS O.W., HOUTEN F.J.A.M. VAN e KALS H.J.J., "Review of research in feature-based design, Journal of Manufacturing Systems", Vol.12, no.2, p. 113-132, 1993.
13. SUN MICROSYSTEMS, "Java Technology", <http://java.sun.com>, último acesso em 7/8/2006.

# **A SYSTEM FOR THE SUPPORT TO THE REMOTE FEATURE-BASED DESIGN OF PRISMATIC PARTS THROUGH THE INTERNET**

**Julio César Ticona Benavente**

Universidade Federal de Santa Catarina, Departamento de Engenharia Mecânica,  
GRIMA/GRUCON, Caixa Postal 476, CEP 88040-900, Florianópolis, SC, metrojulio@yahoo.es

**João Carlos Espíndola Ferreira**

Universidade Federal de Santa Catarina, Departamento de Engenharia Mecânica,  
GRIMA/GRUCON, Caixa Postal 476, CEP 88040-900, Florianópolis, SC, jcarlos@emc.ufsc.br

***Abstract:** Globalization has been causing a significant increase in the competition among companies, but it also allows the cooperation among them, even if they are geographically distant from one another. A research area resulting from that virtual approximation is the remote manufacture of parts, which has the participation of the following actors: (i) a remote customer who inputs the orders; (ii) a modeling company responsible for the development and provision of software modules for the design of the parts, and for the operations planning to be executed on the ordered parts; and (iii) a company where the manufacture will actually take place. It should be mentioned that these actors can be located anywhere in the world. In the present work a methodology for the remote modeling of prismatic parts using the Internet is described. The development of a graphic interface is depicted, which is made available to the remote customers, without the need to install any commercial CAD software in their computers. The modeling of the parts is accomplished through a library of features, which includes the following features in the present version: hole, step, slot and rectangular pocket. This software module is also responsible for verifying the consistence of the part with regard to its manufacturability, that is, when a feature is input with its respective parameters, the software verifies whether the resulting parts can be manufactured.*

**Keywords:** Features, CAD, Manufacturability, Internet, Remote Customer.