

Universidade Federal de Santa Catarina
Centro Tecnológico
Curso de Engenharia de Controle e Automação Industrial

**"Desenvolvimento de Interface Gráfica para um Sistema
Visando a Engenharia Concorrente"**

Monografia submetida à Universidade Federal de Santa Catarina
como requisito para a aprovação da disciplina
EEl 5901: Projeto de Fim de Curso

Claudiane Isabel Grando

Florianópolis, Dezembro de 1994

"Desenvolvimento de Interface Gráfica para um Sistema Visando a Engenharia Concorrente"

Claudiane Isabel Grandó

Esta monografia foi julgada no contexto da disciplina
EEL 5901: Projeto de Fim de Curso
e aprovada na sua forma final pelo
Curso de Engenharia de Controle e Automação Industrial

Banca Examinadora:

Norton Paim Moreira
Co-Orientador

João Carlos Espíndola
Orientador

Prof. Augusto Humberto Bruciapaglia
Responsável pela disciplina e Coordenador do Curso

Prof. José Eduardo Ribeiro Cury, Avaliador

Guilherme Ernani Vieira, Debatedor

Marco Antônio Bresola Ferro, Debatedor

Resumo

O presente trabalho apresenta uma abordagem prática de implementação de um sistema CAD baseado em 'features' usando como base um CAD convencional, neste caso em específico, o CAD I/EMS, da Intergraph. O projeto está inserido num contexto de cooperação técnica entre indústria e universidade e visa auxiliar na implantação da Engenharia Concorrente na EMBRAER - EDE (Empresa Brasileira de Aeronáutica - Divisão de Equipamentos), atuando como ferramenta de auxílio às equipes multifuncionais.

Abstract

This work present a practical approach to a feature-based CAD implementation. It has been developed using a conventional CAD - I/EMS, by Intergraph. The project is an industry/university cooperative work and aims at aiding in the Concurrent Engineering implementation in the EMBRAER - EDE. Mainly it is a help tool for the multi-funcional teams.

Sumário

Capítulo 1: Introdução	1
Introdução	1
Modelagem das Informações	2
Estrutura do Sistema Proposto	3
Desenvolvimento das Atividades	4
Estrutura da Monografia	4
Capítulo 2: Engenharia Concorrente	6
Introdução	6
Sistema de Manufatura: Definição	6
O Porquê da Engenharia Concorrente	7
Engenharia Concorrente: Definição	8
Engenharia Concorrente ou Simultânea ?	8
Metodologias de Implantação	9
A Questão da Troca de Informações	11
Capítulo 3: Tecnologia de 'Features'	12
Introdução	12
Tecnologia de 'Features'	12
Definição de 'Features'	13
Identificação, Classificação e Representação de 'Features'	14
Identificação	14
Classificação de 'Features'	15
Representação de 'Features'	16
Reconhecimento de 'Features' X Projeto por 'Features'	16
Reconhecimento de 'Features'	17
Projeto por 'Features'	18
Outros Aspectos Relevantes	19
Capítulo 4: Sistemas CAD	21
Sistemas CAD Tradicionais	21
Histórico	21
Modelagem Geométrica	22
Modelagem Wireframe	22

Modelagem de Superfícies	22
Modelagem Sólida	23
Sistemas CAD Baseados em 'Features'	24
Exemplos de CAD's Baseados em 'Features'	25
Capítulo 5: Descrição do Sistema Proposto	27
Requisitos de CAD em Engenharia Concorrente	27
A Questão da Comunicação	27
Bibliotecas de 'Features'	27
Consulta a Informações	28
Arquitetura Proposta para um CAD Baseado em 'Features'	28
Descrição da Interface Baseada em 'Features'	29
Estruturas de Comunicação Usuário - Sistema	30
Menu de 'Features'	33
Posicionamento de 'Features'	34
Capítulo 6: Descrição da Implementação	36
Ferramentas Utilizadas	36
O CAD I/EMS	36
Geometria Variacional	38
"Profiles"	38
Planos de Referência	38
Restrições	39
Modelando Sólidos no I/EMS	40
I/FORMS	40
PPL	41
Biblioteca de 'Features'	41
Definição das 'Features' na Biblioteca	41
Definição da Forma	42
Definição dos Atributos	43
Instanciando 'Features'	43
Criação de "Forms"	43
Biblioteca de Auxílio ao Projeto	44
Gerenciador de 'Features'	44
Interface de Projeto	44
Exemplo de Utilização da Interface Baseada em 'Features'	45
Capítulo 7: Conclusões	49

Objetivos	49
Resultados Obtidos	49
Trabalhos Futuros	49
Capítulo 8: Referências Bibliográficas	51

Capítulo 1

Introdução

A Engenharia Concorrente chega na indústria como uma proposta nova, capaz de levar à redução dos custos associados à elaboração de um produto. Propondo que os vários fatores que afetam os custos e a performance do produto através do seu ciclo de vida sejam analisados na fase de projeto, a Engenharia Concorrente pretende reduzir tal ciclo e, conseqüentemente, os custos envolvidos na elaboração do mesmo. O presente trabalho é parte fundamental de um projeto que vem sendo desenvolvido em cooperação com a indústria, caracterizando-se, desta forma, não apenas como trabalho acadêmico. Consiste, em linhas gerais, no desenvolvimento de um software, um CAD, visando a Engenharia Concorrente e, em última análise, a integração da manufatura, uma das áreas enfatizadas pelo curso de Engenharia de Controle e Automação Industrial.

Introdução

Um produto é desenvolvido segundo uma seqüência de atividades agrupadas em seu "ciclo de vida", atividades estas que englobam desde a análise mercadológica até a manutenção do produto final. As diversas fases incluídas neste ciclo são tradicionalmente realizadas de forma seqüencial, usualmente por diferentes grupos de pessoas. Desta forma, as várias informações relativas ao produto estão distribuídas entre os departamentos e especialistas da empresa, os quais têm dificuldade de comunicação entre si, ocasionando um inevitável aumento no ciclo de vida do produto, com aumento nos custos e tempos. A figura 1.1 ilustra a abordagem seqüencial, e o grande "lead time" por ela provocado.

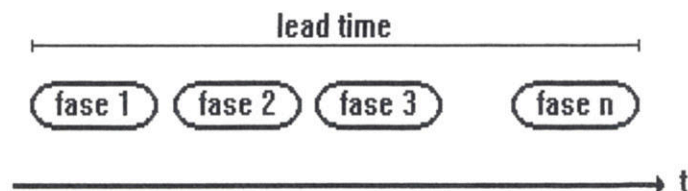


Fig.1.1: Abordagem Seqüencial

A metodologia da Engenharia Concorrente propõe que os vários fatores que afetam os custos e a performance do produto através do seu ciclo de vida sejam analisados na fase de projeto do mesmo; desta forma, pretende-se reduzir tal ciclo, assim como os custos envolvidos na elaboração do produto. Isto requer integração entre as diversas áreas funcionais pertinentes. Observa-se que o tempo de duração de cada etapa aumenta, mas o "lead time" diminui, conforme ilustrado na figura 1.2.

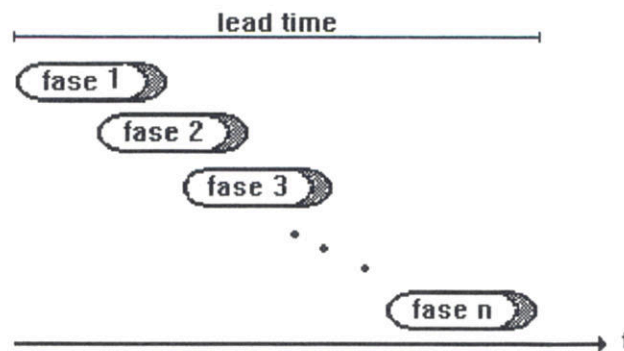


Fig.1.2: Engenharia Concorrente

Foi a partir do Projeto de Cooperação GRUCON - EMBRAER que nasceu a idéia de desenvolver uma ferramenta computacional de auxílio ao uso da Engenharia Concorrente. Trata-se de um ambiente de trabalho para equipes multifuncionais, formadas por especialistas de diversas áreas da EMBRAER - EDE (Empresa Brasileira de Aeronáutica - Divisão de Equipamentos). O sistema proposto está sendo desenvolvido nos moldes de um protótipo, a ser implantado na empresa em questão.

Modelagem das Informações

O principal problema, quando da implementação de um ambiente baseado em Engenharia Concorrente, está relacionado a como representar a variada gama de informações utilizadas pelas diversas atividades do ciclo de vida do produto.

A tecnologia de 'features' permite a definição de um produto, ou das partes do mesmo, em termos de elementos funcionais (como furos, rebaixos, chanfros, etc.) ao invés de simples elementos geométricos (como linhas superfícies ou sólidos). Desta forma, pode-se atribuir informações de caráter não-geométrico, como normas técnicas, ferramental e tecnologia de fabricação (velocidades de corte, avanço, rotação da ferramenta, etc.) a cada um dos componentes ('features') do produto.

Em [MOR94], é apresentada uma definição hierárquica do produto em cinco níveis. Em nosso trabalho, optamos por definir um produto de forma semelhante, ou seja, hierarquicamente, porém com apenas dois níveis, como pode ser visualizado na figura 1.3:

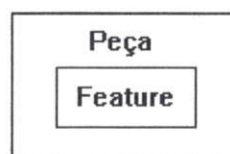


Fig.1.3: Representação hierárquica de um produto

Uma peça é definida como um elemento produzido por um bloco de matéria-prima, sem quaisquer montagens realizadas. Pode ser composta de diversas 'features' já existentes, armazenadas em bibliotecas (de acordo com seu domínio), tantas quanto necessário para representar a funcionalidade da peça. Cada 'feature' representa uma função de projeto utilizada pelo projetista na fase de concepção. A partir deste elemento construtivo, uma peça pode ser gerada através de uma espécie de processo de montagem.

Conforme veremos no decorrer deste trabalho, a tecnologia de 'features' é altamente dependente do tipo de produto envolvido. Assim, delimitamos nossa área de ação a

equipamentos hidráulicos produzidos pela EMBRAER - EDE, como válvulas e atuadores. Estes produtos possuem algumas características comuns como funcionalidade similar, semelhante nível de complexidade geométrica e mesma tecnologia de fabricação. Além disso, seus componentes são, basicamente, peças rotacionais, simplificando assim o modelo para o protótipo.

Estrutura do Sistema Proposto

O sistema proposto é representado esquematicamente pelo Diagrama de Fluxo de Dados da figura 1.4. Neste diagrama, os círculos representam atividades desenvolvidas pelos especialistas nas áreas de Projeto, Planejamento de Processos e Manutenção da Base de Dados. As caixas identificam as entidades externas ao sistema, ou seja, as áreas/pessoas envolvidas. O fluxo de informações é identificado pelas setas nomeadas; os retângulos abertos representam os bancos de dados.

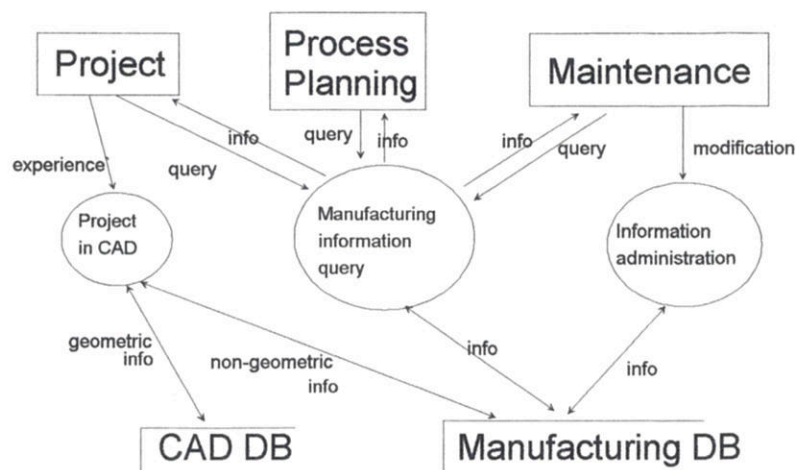


Fig.1.4: Diagrama de Fluxo de Dados do Sistema

Conforme representado na figura, as áreas de Projeto e Planejamento de Processos terão a sua disposição bases de dados, com informações de projeto e manufatura. Estas bases de dados poderão ser acessadas através da interface construída no CAD, de forma a permitir a consulta a antigos projetos (CAD DB), assim como a informações de manufatura.

Como o produto é o elemento em torno do qual as informações de projeto e manufatura estão representadas, então é possível, através do uso da tecnologia de 'features', associar as informações descritivas, que estão armazenadas em um banco de dados, com as formas geométricas características dos produtos da empresa, que serão representadas e armazenadas por um CAD. Esta ligação entre CAD e banco de dados através da tecnologia de 'features' possibilitará as seguintes melhorias:

- melhoria na consulta e na atualização de normas técnicas utilizadas pelo projeto;
- padronização e fácil atualização das informações de manufatura, tais como: normas, ferramentas, dispositivos de fixação, máquinas, etc.;
- padronização dos projetos, evitando a duplicação destes e facilitando a recuperação de informações de antigos projetos Também permite a

padronização das atividades de planejamento de processos, bem como a recuperação de planos de processo;

- redução do número de Solicitações de Serviço à Engenharia (SSE) por parte do planejamento de processos, economizando esforços na revisão e alteração de projetos e acelerando o início da produção do produto.

Dentro do contexto global do sistema acima proposto, o presente trabalho restringe-se à implementação de uma interface com o usuário baseada na abordagem de projeto por 'features', a ser construída sobre o CAD I/EMS, da Intergraph, já em uso pela equipe da EMBRAER. Esta interface permitirá ao projetista definir uma peça a partir da escolha das 'features' em um menu, a medida em que pode consultar informações sobre processos de fabricação e demais informações armazenadas no banco de dados de manufatura. Desta forma, procura-se evitar que uma série de "erros" de projeto sejam detectadas apenas nas fases mais avançadas do ciclo de vida do produto.

Desenvolvimento das Atividades

No sentido de realizar o trabalho a que nos propomos, desenvolveremos as seguintes atividades:

- Estudo da metodologia de Engenharia Concorrente e sua aplicação no âmbito deste projeto;
- Estudo e utilização da Tecnologia de "Features", de maneira adequada às necessidades da EMBRAER - EDE;
- Treinamento de utilização do CAD I/EMS, de forma a possibilitar o entendimento no que se refere aos elementos utilizados pelo mesmo;
- Estudo da linguagem de desenvolvimento PPL e da ferramenta de apoio à construção de interfaces, I/FORMS;
- Definição das funções de interface, que permitirão aos usuários (projetistas) a elaboração de projetos baseados em 'features';
- Implementação das funções definidas, dando origem a uma interface baseada em 'features'.

Estrutura da Monografia

Os demais capítulos desta monografia estão estruturados da seguinte forma:

No capítulo 2 é discutida uma das bases teóricas essenciais, sob a qual todo o trabalho está fundamentado. Define-se o que vem a ser Engenharia Concorrente, seus "porquês", bem como diversas questões associadas ao tema.

O capítulo 3 apresenta a Tecnologia de 'Features', o segundo alicerce teórico de importância fundamental no contexto deste trabalho.

Uma rápida revisão dos sistemas CAD convencionais pode ser encontrada no início do capítulo 4, bem como algumas de suas características e limitações. Alguns aspectos do desenvolvimento de sistemas CAD baseados em “features” são discutidos.

O capítulo 5 descreve a definição do sistema CAD proposto. Inicialmente, são descritos os requisitos do CAD baseado em 'features' inserido na plataforma de auxílio à Engenharia Concorrente. Em seguida, são descritos os módulos básicos que compõe o sistema.

O capítulo 6, por fim, é dedicado à descrição da implementação do modelo desenvolvido. Neste são descritas as ferramentas utilizadas e as soluções encontradas para a implementação de cada um dos módulos do sistema. Um pequeno exemplo de peça desenvolvida com a interface proposta é apresentado no final deste capítulo.

Conclusões pertinentes são feitas no capítulo 7, bem como sugestões de futuros trabalhos a serem realizados nesta mesma linha de pesquisa. Referências bibliográficas utilizadas no decorrer do trabalho são listadas no capítulo 8.

Capítulo 2

Engenharia Concorrente

Introdução

Um produto é desenvolvido segundo uma seqüência de atividades agrupadas em seu "ciclo de vida", atividades estas que englobam desde a análise mercadológica até a manutenção do produto final. As diversas fases inclusas neste ciclo são tradicionalmente realizadas de forma seqüencial, usualmente por diferentes grupos de pessoas. Desta forma, as várias informações relativas ao produto acabam por ficar dispersas entre os departamentos e especialistas da empresa, os quais, devido a dificuldades de comunicação entre si, ocasionam um inevitável aumento no ciclo de vida do produto, com aumento nos custos e tempos.

A metodologia da Engenharia Concorrente propõe que os vários fatores que afetam os custos e a performance do produto através do seu ciclo de vida sejam analisados na fase de projeto do mesmo; desta forma, pretende-se reduzir tal ciclo, assim como os custos envolvidos na elaboração do produto. Isto requer integração entre as diversas áreas funcionais pertinentes. Assim, equipes multifuncionais de projeto, formadas por engenheiros, projetistas, processistas, analistas de mercado, etc., buscam criar, rápida e eficientemente, novos produtos, desde a concepção até a distribuição, através da coordenação de esforços de Engenharia Concorrente, com pensamento criativo e trabalho colaborativo [ROS].

Sistema de Manufatura: Definição

Antes de iniciarmos nossa explanação sobre Engenharia Concorrente, vejamos primeiramente o que vem a ser um Sistema de Manufatura.

Um produto é desenvolvido segundo uma seqüência de atividades agrupadas em seu "ciclo de vida" [REI92]. Estas atividades iniciam na análise de mercado, passam pela aquisição da matéria-prima e estendem-se através de toda gama de atividades de projeto, produção, distribuição e manutenção do produto final [MOR93b]. Alguns autores acrescentam ainda uma etapa de destruição/reciclagem após a etapa de manutenção [ALT91].

Um sistema de manufatura tradicional pode ser representado conforme a figura 2.1, onde cada uma das etapas são realizadas de forma seqüencial.



Fig.2.1: Principais Atividades do Processo de Manufatura

Um interessante ponto de vista é apresentado em [LIN92], fazendo uma analogia entre os sistemas de controle e a manufatura. Neste, os processos são as unidades funcionais da fábrica e os documentos usados na comunicação entre estas, os sinais. Aplicando esta analogia às atividades diretamente envolvidas no ciclo de vida do produto, [LIN93] traz-nos algumas considerações qualitativas interessantes.

O Porquê da Engenharia Concorrente

Até há pouco tempo, pintava-se um quadro bastante ambicioso no que diz respeito às fábricas no futuro: indústrias altamente automatizadas, capazes de funcionar por um ou dois turnos sem qualquer auxílio humano. Esta era a visão de CIM - Computer Integrated Manufacturing, a qual, por falta de definição de um padrão, aliado ao fato de que nenhuma instituição a tenha implementado, acabou desacreditada e passou a ter um sentido bem mais abrangente (como "tecnologia de integração") [ALT91, MOR93b].

Hoje em dia, pesquisadores e empresários estão tendo uma visão mais realista do futuro. A idéia difundida atualmente é a de uma fábrica onde o homem seja responsável pela tomada de decisão, devidamente auxiliado por plataformas computacionais; tarefas são realizadas de forma concorrente e, principalmente, o projeto de produtos é realizado de forma tal que todas as atividades do ciclo de vida são consideradas desde o início do estágio conceitual.

Dentro do contexto atual de grande competição entre empresas pela conquista e manutenção de fatias de mercado, a qualidade de um novo produto é tão importante quanto o tempo de seu desenvolvimento. Estudos mostram que cerca de 80% do mercado de um novo produto é direcionado às duas primeiras empresas que o lançarem [BRA90, ROS].

Com relação à redução de custos, estudos mostram que, enquanto as fases de projeto conceitual e preliminar consomem apenas 5% do orçamento alocado, 80% dos custos para desenvolvimento e produção de um dado produto são por estas determinados [ROS]. Com certeza, a fase de concepção de produto também não é a atividade mais demorada do ciclo de vida do mesmo, mas sua influência sobre as demais é bastante intensa. Então, qualquer alteração de concepção nesta fase custa muito pouco, mas pode afetar fortemente o custo final de fabricação [MOR93b]. Creemos ser este o principal motivo que tem levado a aplicação de Engenharia Concorrente comumente apenas à área de projeto, uma vez que os programas de redução de custos centrados nas demais fases tem pouco impacto no custo de ciclo de vida do produto.

No sistema de manufatura apresentado, uma série de problemas pode ser encontrada na fase de projeto [BED91, MOR93b]:

- a definição de detalhes é muito demorada. Seria aconselhável que estes fossem definidos quando o projeto já estiver mais estabilizado;
- características de manufaturabilidade são consideradas muito tarde, somente quando o custo de mudanças é alto;
- planejamento da produção, manutenibilidade, confiabilidade e suporte são considerados somente após a atividade de projeto estar concluída;
- dados de projeto são fragmentados. A documentação inclui informações representadas de muitas formas diferentes, não sendo possível mantê-las consistentemente ao longo do ciclo de vida do produto;
- projetistas não possuem informações sobre custo.

O objetivo da Engenharia Concorrente é exatamente resolver estas deficiências já na fase de projeto utilizando, basicamente, formas de democratizar a informação, permitindo

que qualquer atividade que necessita informações manipuladas por outras atividades tenha acesso a estas.

Engenharia Concorrente: Definição

Segundo Winner [REI92], "Engenharia Concorrente é um método sistematizado para o projeto concorrente e integrado de produtos com seus processos, incluindo produção e suporte. Esta abordagem procura considerar, em princípio, todos os elementos do ciclo de vida de um produto, da concepção até a distribuição" [MOR93b].



Fig.2.2: Conceito de Engenharia Concorrente [MOR93b]

O projeto de um produto é uma atividade que combina conhecimentos diversos, como marketing, usinagem e montagem. No enfoque tradicional, esta tarefa é realizada segundo uma seqüência de atividades estanques, em geral realizadas por pessoas diferentes. Mais do que propor que estas atividades desenvolvam-se paralelamente, ou compatibilizá-las, Engenharia Concorrente traz para a atividade de projeto considerações sobre muitos fatores que afetam o custo total e o desempenho ao longo do desenvolvimento de um produto. Isto envolve a integração de atividades desenvolvidas em diferentes áreas funcionais da empresa. Além das necessidades do usuário, devem ser consideradas informações sobre manufaturabilidade (seleção de material, montagem, planejamento de processos), confiabilidade, testabilidade, distribuição (desmontagem, reciclabilidade), poluição ambiental, etc. [TER92, MOR93b].

Engenharia Concorrente ou Simultânea ?

Conforme nos lembra [SOH92], ambos os termos "Engenharia Concorrente" e "Engenharia Simultânea" vêm sendo utilizados na literatura como sinônimos, referindo-se a "uma forma de trabalho onde várias atividades são integradas e executadas preferivelmente em paralelo do que de forma seqüencial". Se fizermos um pesquisa em dicionários a respeito dos significados de cada uma das duas palavras observaremos que ambas, 'concorrente' e 'simulatâneo', tem o significado de 'acontecer ao mesmo tempo' porém, apenas a primeira possui o sentido de 'estar em acordo, cooperação'.

O processo de manufatura é descrito pelo modelo proposto por [SUH90] como uma série de mapeamentos entre um certo número de "mundos": os Requisitos do Consumidor

são mapeados em Requisitos Funcionais, que por sua vez são mapeados em Parâmetros de Projeto e, finalmente, em Variáveis de Processo. Desta forma, parece óbvio que o processo como um todo não pode ser "simultâneo" no sentido de tudo ser feito ao mesmo tempo; afinal, é um pouco difícil preparar um plano de processo para um produto que ainda não foi projetado. Entretanto, é igualmente óbvio que o processo pode ser concorrente no sentido de que o que vai ser projetado seja fácil de produzir.

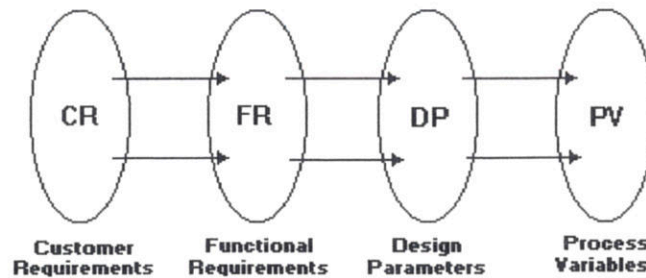


Fig.2.3: Processo de Projeto como uma série de mapeamentos entre diferentes domínios [LIN93]

Assim, o conceito de "Engenharia Concorrente" parece-nos estar mais de acordo com o processo de manufatura do que a denominação "Engenharia Simultânea" [LIN93].

Metodologias de Implantação

A metodologia de implementação da Engenharia Concorrente é uma questão muito discutida. Pode-se classificar as abordagens em dois conjuntos: baseado em grupos multifuncionais ("teamworks") e suporte para a informação auxiliado por computador. Tais abordagens são, na realidade, complementares e devem ser aplicadas em conjunto para que os objetivos sejam alcançados.

Grupos multifuncionais ou multidisciplinares são constituídos por especialistas de diversas áreas que, juntamente com o projetista, definem o projeto do produto. O trabalho em equipe facilita a troca de informações permitindo que os possíveis problemas sejam resolvidos no momento em que são criados. Observando-se a evolução de várias companhias em termos organizacionais, já é possível perceber a proliferação de estruturas baseadas em grupos multifuncionais, isto em todas as atividades e não somente naquelas diretamente ligadas à engenharia. O trabalho em equipe, proposto como forma de integrar contribuições de diferentes especialistas, tem levado a uma redução significativa no tempo do ciclo projeto-produção [CAM94].

A abordagem baseada em computador é muito abrangente, utilizando diversas ferramentas para representar e integrar as informações. O suporte computacional à informação inclui uma gama muito grande de tecnologias.

O trabalho em equipe envolve intensa colaboração e troca de informações. Diversas ferramentas são oferecidas para facilitar este trabalho: entre elas: Projeto para Manufatura ("Design for Manufacturing" - DFM) e Projeto para Montagem ("Design for Assembly" - DFA) são duas das mais importantes e estudadas.

Projeto para Manufatura diz respeito à integração do processo de projeto com o planejamento de processos em uma só atividade, com o objetivo de projetar de forma tal que seja fácil e eficiente produzir. Considerando-se que 70% das decisões tomadas a nível de produção são diretamente determinadas pelo projeto do produto [ASL92], é fácil compreender a difusão do Projeto para Manufatura nas indústrias.

As diretivas de manufatura, cujas considerações sobre as mesmas são parte essencial do DFM, são afirmações práticas da atividade de projetar, definidas empiricamente pela experiência de especialistas em projeto e manufatura, dentre as quais citamos [BED91, MOR93b]:

- projeto com número mínimo de peças;
- projeto modular;
- peças multifuncionais;
- mínima tolerância possível;
- ênfase na padronização
- operações de usinagem e montagem simples;
- tempos de "setups" mínimos.

Quanto ao conceito de Projeto para Montagem, o mesmo visa projetar de forma que o processo de montagem, freqüentemente a operação mais intensiva e que soma a maior porção de custo, seja simples e menos custoso.

Devemos atentar para o fato de que, na realidade, as duas metodologias acima apresentadas são apenas subconjuntos da Engenharia Concorrente, sendo que o todo muitas vezes é esquecido [TER92, MOR93b]. Lembramos que Engenharia Concorrente diz respeito a muito mais do que apenas a projetar para que seja fácil fabricar e/ou montar; minimizar custos, preocupar-se com a reciclagem, são algumas das demais questões relacionadas.

Na realidade, Engenharia Concorrente deve relacionar-se não só a nível de projeto mas em todas as atividades do Sistema de Manufatura. Por exemplo, informações atualizadas sobre planejamento de inspeção devem estar disponíveis ao planejamento de processos, uma vez que implica em procedimentos e sistemas de medição; informações sobre custos de serviços de terceiros são recomendadas tanto a nível de projeto como de planejamento de processos e produção.

Embora as técnicas de CAE/CAD/CAM não sejam implicitamente parte do conceito de Engenharia Concorrente, a recente ênfase nesta metodologia de projeto tem muita relação com a evolução do conceito de CIM nas últimas duas décadas. Neste sentido, os bancos de dados de sistemas CAE/CAD/CAM podem servir como central de informação para as equipes de trabalho e, para tanto, devem constar de informações consistentes e não ambíguas, compreensíveis por todos aqueles que as utilizarem [ROS]. Desta forma, ressaltamos novamente a necessidade de definição de uma linguagem comum, que incorpore todas as informações inerentes ao projeto, relativas às mais diversas fases do ciclo de vida do produto.

Retornando à questão das equipes multifuncionais, gostaríamos de colocar a importância da adequada formação das mesmas, assim como a escolha correta de sua liderança. Vale ressaltar o fato de que o líder não necessita ser um especialista em todas as áreas (o que hoje, com a infinidade de conhecimentos envolvidos na concepção/produção de um produto, é praticamente impossível), embora um sólido conhecimento técnico seja imprescindível, além da habilidade de liderança e habilidades políticas [HUR]. Outro aspecto importante é a motivação da equipe de trabalho;

treinamento da equipe em novas técnicas de projeto e dinâmica de grupo podem auxiliar nesta tarefa [ROS].

Uma observação importante no que se refere à implantação da Engenharia Concorrente, diz respeito à estrutura organizacional da empresa. Não basta dispor de um programa puramente tecnológico (hardware, software, bancos de dados, etc.); acima de tudo, a cultura da empresa deve ser levada em consideração [ASL92].

A Questão da Troca de Informações

Aumento de produtividade e qualidade, assim como redução de custos são os mais importantes critérios do processo de produção. Para satisfazer este critério é imprescindível que informações de projeto, manufatura, etc. estejam disponíveis a todas aquelas que delas necessitarem. Este é o principal requisito da abordagem de Engenharia Concorrente [MOR93a, CAM94].

Olhando para o passado, de volta à era pré-industrial, vemos artesãos fazendo produtos para um mercado, realizando todas as tarefas: ele projeta, produz e vende seu produto. A medida que artesão transformou-se em "manufatureiro", não lhe foi mais possível executar todas as atividades sozinho, surgindo a necessidade de auxílio por parte de novos profissionais. Então, o trabalho que antes era realizado por um indivíduo, agora é executado por um grupo de especialistas cooperando entre si [LIN93].

Esta especialização teve, como todos sabemos, um profundo impacto em nossa sociedade e proporcionou muitos benefícios: aumento da produtividade, variedade, produtos com maior funcionalidade, melhor qualidade, etc. Entretanto, introduziu um sério problema de comunicação, um triplo problema de informação (o problema da comunicação entre os domínios dos diversos especialistas, o problema de controle e o da sincronização das atividades).

Na busca por formas de resolver o problema da comunicação entre os diversos membros, a definição/utilização de modelos comuns é um requisito básico. Modelos que se prestem a este objetivo devem ser capazes de suportar as necessidades dos mais diferentes integrantes do grupo multifuncional. A tecnologia de "features" [CUN88, SHA91, MOR93b, CAM94] é usada para modelar diferentes informações de manufatura, como dimensões da peça, processos de fabricação, ferramentas, tolerâncias, etc.

Muitos trabalhos vem sendo desenvolvidos neste sentido (de minorar os problemas de comunicação), quer a nível conceitual, quer a nível de plataformas computacionais visando a integração.

Capítulo 3

Tecnologia de 'Features'

Introdução

O Sistema de Manufatura tradicional pode ser traduzido por um conjunto de atividades seqüenciais executadas em áreas distintas de uma empresa. Esta prática pode levar a erros, causados pelo desconhecimento de informações em uma certa etapa que, a princípio, seriam de uso "exclusivo" de outras áreas. O fundamento da Engenharia Concorrente está em trazer para uma atividade específica informações que afetam o desenvolvimento do produto como um todo [MOR93b].

A implantação de Engenharia Concorrente baseada em grupos multifuncionais, centra o desenvolvimento de um projeto em torno de diversos especialistas, os quais mantêm o conhecimento cada um sobre seu próprio trabalho. Além disso, cada especialista continua a falar sua própria língua, ocasionando problemas de comunicação com os demais membros da equipe.

Assim sendo, vê-se a necessidade de utilização de uma representação única para modelar o produto. A Tecnologia de 'Features' permite definir o produto como um conjunto de elementos funcionais reconhecidos em todo o Sistema de Manufatura. Cada um destes elementos, ou 'features', armazena informações relativas a seu próprio uso durante todo o ciclo de vida de um produto.

As 'features' são encaradas como entidades que efetivamente permitem a elevação do nível de informação dos dados manipulados pelos sistemas CAD, uma vez que se propõe a representar conjuntos de informações de natureza funcional e semântica dos objetos modelados, além da geometria. Segundo [DIX90], a representação baseada em 'features' no desenvolvimento de projetos de produtos é um pré-requisito essencial para uma nova geração dos chamados sistemas CAD inteligentes, capazes de suportar o projeto conceitual, oferecer sugestões de fabricação, fornecer acesso automático para procedimentos de análise, e capturar e utilizar as intenções do projetista relativas à funcionalidade da peça.

Tecnologia de 'Features'

A atividade de projetar é um processo de criação de peças segundo requisitos e especificações funcionais, sob certas restrições [LIM94]. A grande maioria dos sistemas CAD instalados atualmente na indústria são utilizados apenas para gerar desenhos de forma mais rápida e precisa. Entretanto, a tendência atual de integração das atividades do Sistema de Manufatura, segundo a filosofia da Engenharia Concorrente, tem conduzido à integração do CAD com aplicações subseqüentes, como ferramentas de análise e planejamento de processos.

Segundo [DUA93], 70 a 80% do custo total de um produto é determinado na fase de projeto do mesmo. Desta forma, visando reduzir custos e aumentar a qualidade dos produtos, diversas pesquisas vem sendo desenvolvidas; o alvo: 'features', que permitem

relacionar requisitos de manufatura e das demais fases do ciclo de vida aos dados de projeto.

De acordo com [CHU88], os benefícios trazidos pela utilização de 'features' são inúmeros, e podem ser sintetizados em três aspectos principais:

- os usuários podem expressar mais facilmente sua intenção durante o projeto, diretamente através da manipulação de 'features';
- bancos de dados de 'features' permitem que sistemas dedutivos realizem tarefas como otimização heurística, análise de manufatura, etc;
- 'features' podem armazenar conhecimentos que facilitarão a programação de máquinas NC, o planejamento de processos, etc.

Definição de 'Features'

As 'features' têm sua origem no processo de raciocínio usado em várias atividades de concepção de projeto e fabricação [DIX87a]. Pesquisas referentes à 'features' são relativamente recentes; talvez por isso, o termo 'feature' seja usado com diversos significados, segundo a área onde está sendo aplicado.

Em [BRO93] e [SAL92] encontramos uma série de definições dadas ao termo 'features':

- geometria que corresponde a operações básicas de usinagem ;
- uma peça distinta ou característica de uma mesma peça definindo uma forma geométrica, a qual pode ser tanto específica para processos de usinagem quanto para fixação e/ou medição;
- uma área de interesse na superfície de uma peça;
- uma região de interesse no modelo de uma peça;
- um conjunto de informações relativas à descrição de uma peça;
- elementos usados na geração, análise ou avaliação de projetos;
- uma forma de aspecto funcional para o projeto e fabricação;
- padrões recursivos de informação relacionados à descrição de uma peça;
- um agrupamento semântico usado para descrever uma peça e suas montagens. 'Feature' agrupa de um modo funcionalmente relevante, informações de projeto e fabricação;
- uma forma ou entidade geométrica cuja presença ou dimensões são solicitadas para executar pelo menos uma função CIM e cuja disponibilidade como primitiva permite ao processo de projeto sua ocorrência;
- um transportador de informação do produto que pode auxiliar o projeto ou comunicação entre projeto e fabricação, ou entre tarefas de engenharia;
- qualquer entidade usada em raciocínio sobre projeto, engenharia ou fabricação.

Segundo [CUN88], uma 'feature' é "qualquer forma geométrica ou entidade usada no desenvolvimento de um ou mais projetos ou atividades de manufatura." No trabalho realizado por [SHA88b], as várias definições de 'feature' são unificadas por uso e simplificadas em:

- modelagem geométrica: "Agrupamento de entidades geométricas ou topológicas que necessariamente são referenciadas juntas";
- projetos: "Elementos utilizados na geração, análise ou avaliação de projetos";
- planejamento de processos: "Formas ou atributos tecnológicos associados com operações de manufatura e ferramentas".

Como podemos facilmente verificar, a partir das várias definições apresentadas acima, não existe uma definição de 'feature' que seja universalmente aceita. Tal definição é fortemente relacionada ao escopo em que se encontra inserida.

Identificação, Classificação e Representação de 'Features'

Segundo [SHA88b], para uma eficiente utilização das potencialidades da Tecnologia de 'Features', três pontos principais devem ser levados em consideração: a identificação das 'features', sua classificação e representação.

Identificação

No que diz respeito à identificação, uma vez que não existe consenso com relação ao conceito de 'features' e a identificação nele se baseia, os critérios a serem usados neste processo são da mesma forma conflitantes. Uma série de trabalhos foram desenvolvidos no sentido de definir formas de definição de 'features': [MOR93b]

Segundo Shah [SHA91], a identificação de 'features' é realizada sobre um domínio limitado de peças, podendo ser, por exemplo, o conjunto das peças rotacionais ou de produtos produzidos por uma empresa em especial. Para um domínio de problemas específicos é possível identificar um conjunto mínimo de componentes, de forma que as 'features' emergem gradualmente nas atividades de projeto, fortemente relacionadas com a função da peça e podendo ser concebida antes que a peça real o seja.

Os estudos feitos pela empresa John Deere para a CAM-I [BUT86] identificam 'features' fabricadas por processos de usinagem e por conformação de chapas metálicas. Com o objetivo de determinar todos os dados necessários para o planejamento do processo, este trabalho define 'features' em termos de sua forma genérica, parâmetros dimensionais, tolerâncias, etc. As 'features' de forma são classificadas em três categorias: laminares (chapas), não-rotacionais (prismáticas) e rotacionais. Cada uma destas categorias se subdivide em classificações que, nos níveis mais baixos, são bastante específicas. Esta abordagem também comporta classificação para os materiais com base em sua composição (aço, liga de alumínio, etc), forma bruta (lâminas, barras, forjados, etc), tratamento térmico e acabamento de superfícies.

Cunningham et al. [CUN88] identificaram 'features' utilizadas em processos de manufatura, como extrusão de alumínio, forjamento, fundição, estampagem, injeção de moldes e montagem, para a aplicação em análise de custos, avaliação de manufaturabilidade e projeto de ferramental.

Podemos facilmente perceber a forte ligação entre os produtos com que se deseja trabalhar e a identificação de 'features'. A partir de um universo de produtos, a identificação de 'features' deve tomar como base duas perguntas relacionadas: a quem interessam as 'features' e quais as características que devem ser consideradas.

Moreira [MOR93b] chama atenção para o fato de que, a criação de uma 'feature' implica em grande quantidade de informações distribuídas em diversas áreas. Conseqüentemente, uma nova 'feature' deve ser precedida por um estudo de suas características que englobe todo o ciclo de vida dos produtos para os quais representará informações. Todo este processo é lento e complexo, tornando sua prática desaconselhada. Alternativamente, dado um domínio conhecido de produtos, é possível identificar as 'features' características que permitem a modelagem do conjunto. Esta abordagem tem a vantagem de gerar padronização nos elementos construtivos, incrementando a produtividade e a qualidade.

Classificação de 'Features'

Uma vez identificado o conjunto de 'features', estas devem ser classificadas, de forma a permitir a definição dos atributos que as caracterizam. Segundo [SHA91b], o número de 'features' não é finito, mas existe a possibilidade de categorizá-lo em classes, permitindo um tratamento das 'features' a nível das mesmas, ao invés de individualizado a cada 'feature'. Shah [SHA88d] propôs uma divisão em três grupos:

- 'features' de forma - entidades geométricas que definem forma e dimensão de uma peça;
- 'features' de precisão - desvios aceitáveis da geometria nominal da peça (tolerância e acabamento superficial);
- 'features' de material - especificam tipo, classe e propriedades dos materiais em geral.

Wingard [WIN91] propôs uma taxonomia dependente da área de aplicação, sugerindo uma divisão de 'features' de forma em duas sub-classes: atômicas e compostas. As atômicas, que são 'features' simples as quais não podem ser decompostas em 'features' ainda mais simples, que por sua vez são subdivididas em peça ('feature' com existência própria), modificador (existencialmente dependente de outra 'feature') e grupo (reune várias 'features' que possuem coletivamente um significado de engenharia). 'Features' compostas são formadas por 'features' atômicas e sub-divididas em padrões e complexas.

[BRO93] reforça que é muito difícil criar uma classificação independentemente da aplicação; diferentes aplicações simplesmente requerem diferentes 'features', concluindo-se que uma classificação geral de 'features' talvez seja quase impossível. O autor argumenta que as mesmas entidades básicas podem ocorrer em muitos lugares na classificação, porém semanticamente diferentes, exemplificando com um furo cilíndrico que poderia ser uma 'feature' de forma funcional, no Projeto, uma 'feature' de análise com propriedades especiais, em Análise de Elementos Finitos, uma 'feature' de Planejamento de Processo para uma operação de furação, e uma 'feature' de montagem para conexão a outras peças.

O esquema de classificação proposto por Shah [SHA86] foi originalmente concebido para proporcionar uma classificação genérica para aplicações em projeto e manufatura de peças rotacionais, lâminas metálicas, prismáticas, aerofólios, carcaças, fundidos, peças injetadas, estruturas e montagens. Apesar disso, a classificação somente foi detalhada

para peças rotacionais. Segundo esta classificação, as peças rotacionais dividem-se em três tipos: eixos, flanges/rodas e rotacionais gerais. O esquema não é estritamente hierárquico, sendo as 'features' divididas em implícitas e explícitas e ainda em dependentes e independentes. No primeiro caso, as informações geométricas das 'features' são calculadas apenas quando necessário ou são previamente detalhadas. As 'features' são consideradas dependentes quando necessitam de uma outra 'feature' para a sua definição, e independentes em caso contrário.

Pratt e Wilson [PRA85] classificam 'features' através da forma utilizando Tecnologia de Grupo (GT). É criado um código de cinco dígitos; o primeiro dígito fornece a principal classe de forma (depressão, protuberância, etc), o segundo e o terceiro dígitos detalham esta classificação de forma (côncava, convexa, rotacional, etc). Os últimos dois dígitos fornecem a forma de representação (implícita, explícita, representação por fronteira, por geometria sólida construtiva, etc).

Em Chen [CHE89], é definido um código para classificação de 'features' em um domínio de peças de revolução. Nesta abordagem, são utilizadas características específicas da geometria da 'feature', assim como sua funcionalidade. Já em [GIN89], o modelo baseado em GT trata 'features' como volumes envolvidos por entradas, saídas e contornos. Contrariando estes trabalhos, [SHA88b] adverte para o uso de GT, uma vez que nesta, um único código é utilizado para a peça inteira, combinando as 'features' e sua justaposições.

Em Hernandez et al. [HER91a, HER91b, GUT90, ROS90], as 'features' são classificadas em CAD e CAPP. 'Features' CAD são classificadas por sua funcionalidade, como: apoio de mancal, furo de alívio, apoio de rolamentos, etc. As 'features' CAPP, por sua vez, são analisadas do ponto de vista de produção, ou seja: cilindro, chanfro, etc. É proposto um mapeamento automático entre os dois tipos, o que permite sua integração. Em [MOR93] podemos encontrar uma interessante estratégia para definição e classificação de 'features'.

Representação de 'Features'

O terceiro ponto a ser levado em consideração é a forma de modelar e representar uma 'feature' e, conseqüentemente, uma peça. Esta representação proporciona, entre outras coisas, a possibilidade de implementação computacional.

Considerando as representações geométricas oferecidas pelos modeladores sólidos, poderíamos resumir a escolha da representação em usar B-rep, CSG ou um esquema de representação híbrida [LIM94]. [PRA88] apresenta uma discussão sobre a escolha de cada uma das representações. Segundo o autor, a representação híbrida é a mais adequada quando se tratando de projeto por 'features'.

Ainda dentro do contexto de representação de 'features', 'features' podem ser definidas sob o paradigma de orientação a objetos, de forma a simplificar sua manipulação. Dessa forma, 'features' seriam objetos formalizados por uma lista de atributos, métodos e hierarquia herdada.

Reconhecimento de 'Features' X Projeto por 'Features'

Ao longo do tempo o conceito de 'feature' evoluiu de forma a permitir a agregação das informações necessárias à integração do processo de manufatura. Por esse motivo

existem hoje controvérsias a respeito da forma de utilizar 'features'. A primeira corrente reza que se deve projetar a partir de um CAD convencional e depois extrair as 'features'. A segunda afirma que todo o desenvolvimento do projeto deve ser feito a partir de uma biblioteca de 'features'.

Reconhecimento de 'Features'

Reconhecimento de 'features' é a tarefa de extrair 'features' de uma descrição geométrica [MOR93b]. A figura 3.1 representa esta abordagem. A partir do modelo geométrico do produto, e de uma definição dos padrões a serem reconhecidos, procedimentos automáticos de reconhecimento entram em ação, realizando comparações a fim de "reconhecer" que 'features' compõe o modelo. Os procedimentos de reconhecimento lidam com as bases de dados produzidas pelos modeladores geométricos.

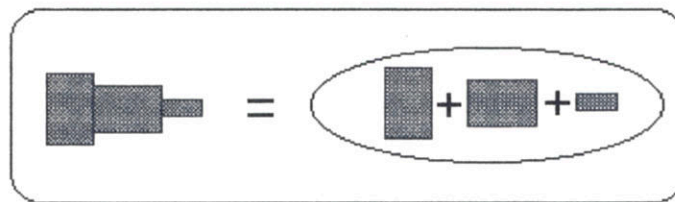


Fig.3.1: Reconhecimento de 'Features'

Uma série de trabalhos já foi realizada nesta área e alguns sistemas "reconhecedores de 'features'" implementados. [JAR84, VAR90, HWA92, WAN93, HEN88, JAR86, JAR89, LEE87, KRA88, TSU89, YEO91].

Sahay et al. [SAH90] desenvolveram um algoritmo de extração de 'features' direcionado para peças cilíndricas. Este domínio simplifica bastante o trabalho de extração. As 'features' são definidas utilizando uma definição baseada em propriedades geométricas. A partir desta premissa, a abordagem primeiro reconhece a existência de todas as 'features' no contorno do objeto, sendo então classificadas como: independente, simples ou complexa. As complexas são subdivididas em primitivas, sendo, mais tarde, classificadas utilizando Sistemas Especialistas.

Stanley et al. [STA83] utilizaram uma abordagem em reconhecimento de padrões orientado por sintaxe para extrair 'features' de um banco de dados de CAD. Essa abordagem é aplicada na classificação de furos num corte seccional (2D) extraído de um banco de dados de modelagem sólida (3D).

Segundo [BRO93], existem diversos problemas nessa abordagem: o processo de reconhecimento é muito complexo e freqüentemente trabalha com uma classe geométrica restrita; os algoritmos de reconhecimento são específicos de cada aplicação. Outro problema é a redundância do reconhecimento de 'features', uma vez que o projetista trabalhou para transformar suas entidades conceituais de projeto de alto nível em formas geométricas de baixo nível, que por sua vez devem voltar a ser vistas como entidades de alto nível pelo reconhecedor de 'features'. [LIM94]

Observando-se que os reconhecedores geralmente exploram diretamente os modelos geométricos e que as 'features' somente passam a existir após a exploração, não se considera que informações não-geométricas, de natureza tecnológica, por exemplo, tenham sido já incorporadas antecipadamente às entidades geométricas. Além do que, esta é uma técnica unilateral, uma vez que possíveis alterações posteriores ao projeto não

podem ser enviadas ao CAD automaticamente, pela impossibilidade de se alterar a base de dados do CAD através de um reconhecedor de 'features'.

Projeto por 'Features'

Segundo Ssemakula et al. [SSE90b], o reconhecimento de 'features' pode ser desnecessário se existir um método que consiga reter as informações explicitamente representadas pelo projetista durante a atividade de projeto. É neste ponto que entra o projeto por 'features', a abordagem mais promissora no tocante a 'features'.

Essa abordagem sustenta-se na idéia do fornecimento de conjuntos de 'features' e procedimentos de utilização desse conjunto ao nível da fase de projeto, de tal maneira que o produto seja já concebido em termos dessas primitivas de engenharia. Ou seja, o produto é criado usando-se entidades de conhecimento de alto nível tanto mais próximas quanto possível da abstração conceitual do projetista [LIM94].

Segundo [CUN88], o projeto por 'features', ao mesmo tempo em que limita o projetista, devido ao número finito de elementos, encoraja a padronização das peças. Sendo esta responsável por aumento de qualidade e manufaturabilidade.

Sistemas desenvolvidos segundo esta abordagem devem cumprir alguns requisitos:

- existência de um mecanismo para definir as descrições de 'features', a fim de se eliminar o problema de conjuntos restritos de 'features', o que se traduz por uma biblioteca aberta de 'features', onde geometria e topologia das 'features' possam ser definidas dinamicamente;
- instanciamento de 'features' deve ser paramétrico e preferivelmente baseado em vários conjuntos de parâmetros;
- restrições devem estar disponíveis para garantir a validade do uso das 'features' e das relações inter-features;
- disponibilizar hierarquias de 'features' para o tratamento de 'features' compostas;
- permitir tanto o projeto "top-down" quanto "bottom-up".

Segundo [SHA91b], duas categorias podem ser definidas dentro dessa abordagem: modelamento destrutivo e modelamento construtivo. No modelamento destrutivo, o modelo da peça é criado por subtrações booleanas de um conjunto de 'features' de fabricação pré-definidas, a partir de um bloco de matéria-prima, permitindo a geração simultânea do plano de fabricação da peça. No modelamento construtivo, o modelo da peça é obtido por adição e/ou subtração de 'features', sem partir de um bloco inicial de matéria-prima. As 'features' são agrupadas em bibliotecas dinâmicas ou estáticas. Alguns sistemas comerciais disponíveis adotam esta abordagem, como o Pro/Engineer.

Utilizando a abordagem de projeto por 'features', os usuários criam modelos através do uso explícito de 'features', sendo estas, desta forma, incorporadas desde o início do ciclo de vida do produto. A utilização deste método implica no uso de uma biblioteca de 'features' e um modelador de sólidos baseado em 'features' [MOR93b]. Diversas propostas são listadas e comentadas em [SHA88b]; abaixo são listadas as mais importantes:

- biblioteca de 'features' desvinculada do modelo geométrico: é utilizada uma linguagem de descrição de 'features', através da qual são introduzidas as informações que a definem;
- modelagem sólida destrutiva (DSG): são utilizadas operações booleanas para construir o modelo a partir da remoção de 'features' de um bloco sólido. Em geral, estas 'features' são em pequeno número;
- modelos baseados na composição de 'features': estes permitem projetar pela manipulação de 'features' pré-definidas. Alguns trabalhos [CHU88, SHA88a] utilizam estas 'features' com sistemas de herança baseado no paradigma de orientação a objetos. A figura 3.2 representa esta abordagem.

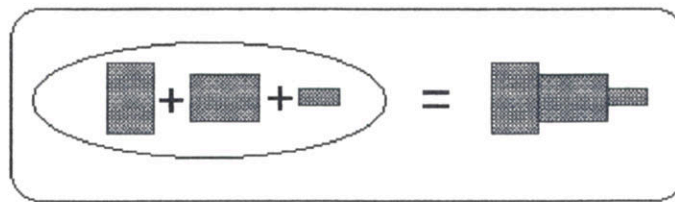


Fig.3.2: Projeto baseado em 'Features'

Em [CUN88] é apresentado um sistema CAD baseado em projeto por 'features'. O projetista constrói a peça manipulando as 'features' armazenadas em uma biblioteca. As operações realizadas sobre a peça são de inserir, modificar e apagar. A partir desta manipulação, uma representação primária da peça é criada. A integridade das operações é garantida por um monitor que controla a manipulação das 'features'. A representação primária é transformada em outras formas, chamadas em conjunto de secundárias, as quais são utilizadas por aplicações específicas. Esta conversão pode envolver alguma forma de reconhecimento de padrões para obter 'features' mais abstratas.

Outros sistemas que utilizam abordagem de projeto por 'features' são apresentados na literatura, em [UNG88, ROL89, TSA89, JUR90, SCH90, LIT91, COR91].

O projeto por 'features' apresenta algumas dificuldades de implementação, pois existe a necessidade de mudar constantemente a forma da peça enquanto se está projetando. Tradicionalmente o projetista concebe uma peça a partir de sua funcionalidade e o processista determina as atividades de produção. Entretanto, esta abordagem coloca o projetista frente à situação de considerar, além da funcionalidade, especificações de produção [KAR91].

Outros Aspectos Relevantes

As diferentes abordagens para o uso de 'features' no sistema de Manufatura visam, basicamente, modelar as informações de produto de forma integrada. A dificuldade está em conseguir fazer convergir as diferenças existentes entre o projeto (composição de funcionalidade) e a produção (decomposição). A nível de projeto as 'features' têm a função de representar funcionalidade através de sua forma e informações encapsuladas. Em termos de produção, o objetivo principal é armazenar informações relativas à execução da peça em termos de definição de operações, seqüenciamento, escolha das ferramentas, dispositivos de fixação, etc.

Respondendo a esta questão, [MOR93b] nos lembra que uma 'feature' existe para todo o ciclo de vida do produto, uma vez que é definida durante o projeto e utilizada nas

atividades subseqüentes. A questão está na maneira como cada fase do Sistema de Manufatura filtra as informações de um mesmo elemento. O elemento é um só, suas características são as mesmas, mas a importância de cada uma delas é relativa, dependendo do ponto de vista adotado.

Partindo deste princípio, a "chave de acesso" para as informações do ciclo de vida de um produto deve ser definida pelas informações de funcionalidade, sendo passível de análise considerando diferentes pontos de vista. Na prática, uma 'feature' possui todas as informações necessárias. O acesso a esta informação é realizado por consultas a partir das características referenciais para cada atividade do Sistema de Manufatura. A figura 3.3 representa esta abordagem. Nesta, cada usuário (x,y e z), "olhando" para uma mesma 'feature', através de um filtro próprio, encontra os atributos que lhe interessa [MOR93b].

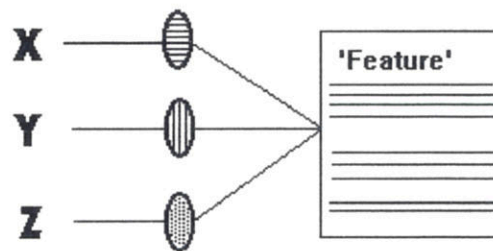


Fig.3.3: Uma mesma 'feature' vista sob diferentes aspectos

Capítulo 4

Sistemas CAD

Este capítulo introduz os sistemas CAD do ponto de vista da representação das informações. Por fim, tece algumas considerações sobre sistemas CAD baseados em 'features'.

Sistemas CAD Tradicionais

Histórico

Computer-aided design (CAD) - Projeto assistido por computador - pode ser simplesmente descrito como a "atividade de projetar com o auxílio do computador". Uma grande variedade de aplicações podem ser vistas, sob este conceito, como aplicações de CAD: projeto de peças mecânicas, circuitos integrados, layout de chips, e por aí fora.

O CAD apareceu na década de 50, quando pela primeira vez o computador foi usado para criar objetos gráficos [CHA85]. Inicialmente baseavam-se em modeladores geométricos (formas de representação de objetos tridimensionais) pobres, e que não eram especificamente desenvolvidos para aplicações em engenharia, mas para uma grande faixa de utilização, como animações para televisões, comerciais, computação gráfica, etc., Assim sendo, originariamente simplesmente serviam como ferramentas de substituição de pranchetas, oferecendo aos projetistas todas as ferramentas de desenho através de um programa computacional.

Entretanto, dadas as necessidades dos usuários a cada novo instante, os sistemas CAD evoluíram, e atualmente estão voltados para a modelagem de objetos na engenharia, o que inclui não apenas funções de desenho mas certas funções de engenharia, como análise de elementos finitos, cálculo de massa, checagem de interferência e comando numérico, entre outras. [SHA88d]

A figura 4.1 [SHA88d] mostra a evolução dos modeladores nos últimos 20 anos, desde os mais simples até a última geração, que são de modeladores de 'features' sobre representações sólidas. A diferença fundamental entre cada geração de modeladores é o nível de informação suportado.

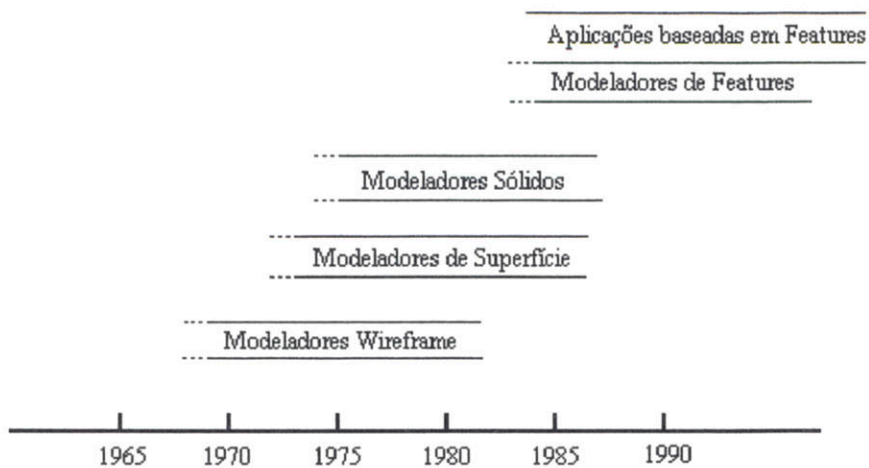


Fig.4.1: Modeladores Geométricos

Modelagem Geométrica

Existem três esquemas básicos de representação tridimensional de um objeto. Sejam eles: a modelagem wireframe, de superfícies e a modelagem sólida.

Modelagem Wireframe:

A representação de um modelo geométrico composto por uma coleção de linhas (bordas) e vértices é conhecida como "wireframe". Uma vez que é composta apenas por tais estruturas, não sendo dotada de superfícies, esta forma de representação geralmente leva à ambigüidades.

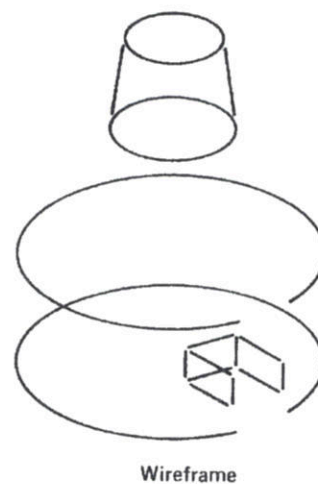


Fig.4.2: Wireframe

Modelagem de Superfícies:

Na modelagem de superfícies, a representação de um objeto 3D é feita usando superfícies para modelar seus contornos. Este tipo de modelagem é comumente usada em aplicações de engenharia em que apenas informações a respeito dos contornos do objeto são suficientes [FER90].

Modelagem Sólida:

Um dos grandes avanços dos modeladores geométricos foi a capacidade de representação e tratamento de formas 3-D de objetos sólidos, o que é conhecido como modelamento de sólidos. Modeladores de sólido são de grande importância, sobretudo na engenharia, uma vez que são capazes de representar objetos sem ambigüidade.

De acordo com [LEE80], existem cinco diferentes esquemas de representação de sólidos, entre os quais são mais difundidos o B-rep - Boundary representation - e o CSG - constructive solid geometry. Sistemas comerciais apresentam, normalmente, uma forma híbrida, combinando CSG para o modelamento e derivando as informações da representação B-rep associada.

A representação CSG é baseada em um conjunto de primitivas sólidas (cubos, esferas, cilindros, etc) que podem ser combinadas, a fim de construir entidades mais complexas, através de um conjunto de operações booleanas (diferença, união e intersecção) [CHA85]. Nessa representação, ao contrário da B-rep, as informações sobre as primitivas são implicitamente definidas por um conjunto de parâmetros que instanciam os sólidos correspondentes.

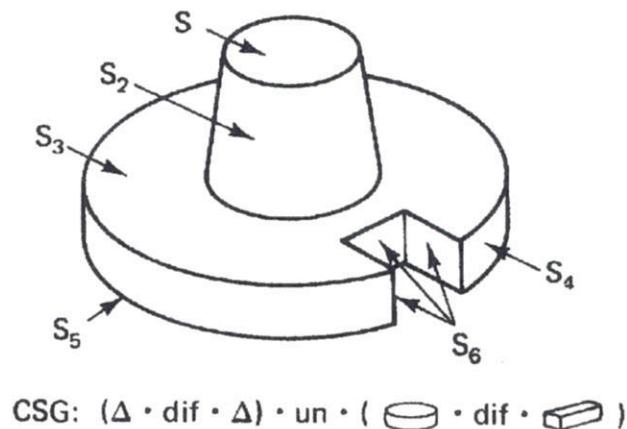


Fig.4.3: Modelagem CSG

A representação B-rep considera que um sólido pode ser visto como sendo limitado por um conjunto de faces limitadas por um conjunto de arestas, estas últimas limitadas por dois vértices. A informação é armazenada sobre esses elementos de contorno e suas relações de adjacência, as quais representam as informações topológicas do sólido e possibilitam a representação de um sólido B-rep como um espaço fechado [BRO93]. As faces e arestas são representadas por equações matemáticas apropriadas, enquanto que os vértices são representados por suas coordenadas (x,y,z) . Muitos modeladores baseados nessa representação fornecem somente faces planares devido a facilidade de representação, simplicidade e velocidade de operação [LIM94].

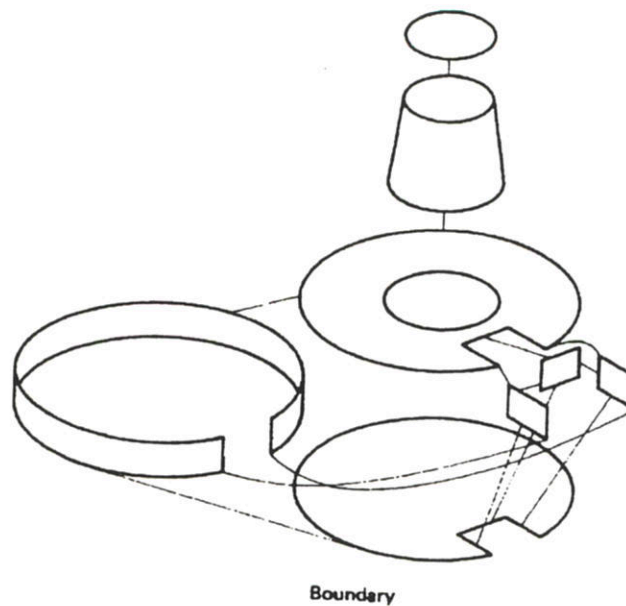


Fig.4.4: Modelagem B-rep

Sistemas CAD Baseados em 'Features'

Partindo-se do desejo de integrar todas as atividades envolvidas com o ciclo de vida do produto, os requisitos impostos aos sistemas CAD por seus usuários têm-se tornado cada vez mais exigentes em termos de qualidade da representação do conhecimento envolvido. Muitas aplicações de engenharia e fabricação auxiliadas por computador requerem uma definição geométrica mais completa de um produto, a qual os modeladores geométricos atuais não conseguem fornecer.

Assim, pode-se resumir os requisitos atualmente cobrados dos sistemas de auxílio ao projeto em uma única frase: "elevação do nível de representação das informações pertencentes à especificação do projeto de um produto" [LIM94].

Os modeladores sólidos, que significaram uma abstração superior aos modeladores 2D e 2½D, abriram caminho para a representação de outras informações, de natureza não-geométrica, dentro das bases de dados dos sistemas CAD. Algumas áreas de aplicação, como planejamento de processos e geração de programas NC, por exemplo, efetivamente aproveitam-se desses modeladores para inferir e/ou associar informações adicionais às geométricas, elevando ainda mais o grau dos conhecimentos específicos diretamente associados aos modelos suportados pelos sistemas CAD.

É dentro deste contexto que surge a nova geração de modeladores geométricos, os modeladores baseados em 'features', que efetivamente permitem a elevação do nível de informação dos dados manipulados pelos sistemas CAD, uma vez que se propõe a representar conjuntos de informações de natureza funcional e semântica dos objetos modelados, além da geometria. Segundo [DIX90], a representação baseada em 'features' no desenvolvimento de projetos de produtos é um pré-requisito essencial para uma nova geração dos chamados sistemas CAD inteligentes, que possam suportar o projeto conceitual, oferecer sugestões de fabricação, fornecer acesso automático para procedimentos de análise, e capturar e usar as intenções do projetista relativas a função e manufaturabilidade [LIM94].

Exemplos de CAD's Baseados em 'Features'

A maioria dos modeladores de sólidos desenvolvidos até então não permitem ao usuário definir 'features'. O que alguns sistemas CAD comerciais estão oferecendo é a conveniência de agrupar primitivas geométricas básicas em 'features'. No entanto, um agrupamento desta forma não pode ser considerado como um 'feature' no seu sentido mais amplo, uma vez que a única informação de manufatura associada é aquela de natureza geométrica.

Segundo [DUA93], "corrigir" as deficiências dos modeladores de sólidos, acrescentando-lhes funcionalidades em termos de 'features', não é o melhor caminho a ser seguido. Procedendo-se desta maneira, a forma de modelagem não é modificada, e gasta-se tempo e esforço em inserir melhorias em uma ferramenta que fora na verdade projetada para fins mais simplórios, o do desenho auxiliado por computador. Na opinião do autor, já que se deseja acrescentar informações de todas as fases do ciclo de vida de um produto na etapa de projeto, é fundamental que se faça um estudo do sistema como um todo, usando teorias modernas de software, visando o "reprojeto" dos sistemas CAD.

Seguindo esta linha de raciocínio, Duan et al. [DUA93] desenvolveram o FSMT - Feature Solid Modelling Tool. Esta ferramenta é composta por um sistema de gerenciamento e definição de 'features' (FDMS), um processador de operações booleanas (BOP), um "checador" de consistência de geometria e atributos (CVS), uma base de conhecimento (KB), uma base de dados central (DB) e um sistema de gerenciamento de interface (UIMS). Bibliotecas de 'features' (FL) são utilizadas.

O FDMS permite ao usuário definir 'features' de acordo com a aplicação, de forma que a peça seja projetada com aquelas 'features' que são mais adequadas à sua função, geometria e plano de processo. O BOP é utilizado para realizar operações de união, diferença e intersecção, além de cálculos de concavidade. Verificações de consistência são realizadas pelo CVS. O módulo UIMS é responsável pelo gerenciamento da interface com o usuário. A figura 4.5 apresenta a arquitetura básica deste sistema. Para maiores detalhes a respeito do sistema vide [DUA93].

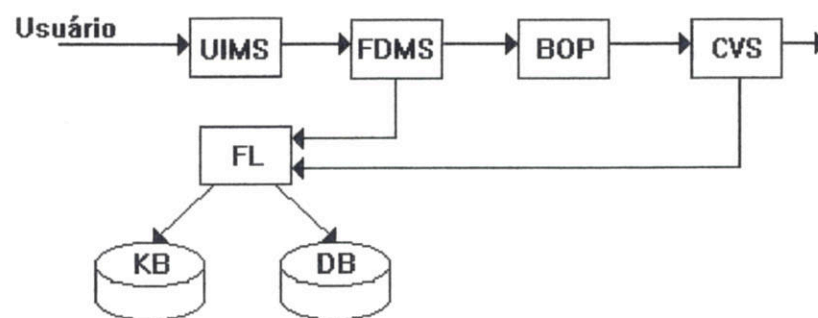


Fig.4.5: Arquitetura do FSMT

O trabalho realizado por [LIM94], desenvolvido a nível de dissertação de mestrado, introduz um modelo baseado em 'features' concebido para representação de informações de projeto e fabricação referentes a um produto (peça mecânica) relacionadas às análises de manufaturabilidade, tecnologia de grupo e seleção de processos de fabricação. Foram implementados dois módulos específicos, afim de validar o modelo conceitual, que trabalham de forma integrada: um para contemplar a fase de projeto propriamente dita (Feature Modeller), o qual recebeu maior enfoque, e outro para efetuar os raciocínios necessários e pertinentes à fabricação de um produto (Subsistema Abstrato).

O trabalho seguiu a abordagem de projeto por 'features'. Ambos os módulos trabalham sob o paradigma de orientação a objetos, sendo que o módulo abstrato é também um sistema especialista baseado em regras e fatos.

'Feature' foi definida dentro do escopo deste trabalho como "um conjunto de informações ou entidades usadas em atividades de projeto ou fabricação, dependentes de modelo e contexto".

O módulo gráfico, Feature Modeller, foi desenvolvido no AutoCAD, CAD comercial da AutoDesk. As 'features' foram implementadas valendo-se das primitivas sólidas fornecidas pelo AutoCAD/AME - modelador de sólidos do AutoCAD - para representar suas formas. Atributos de material, processos de fabricação, de classificação de simetria, de tipo de 'feature' e de estado das faces foram adicionados às primitivas geométricas para atribuir significado de engenharia. O sistema oferece ao projetista a possibilidade de criar novas 'features', as macro-features, segundo suas necessidades próprias. Tais 'features' são armazenadas em bibliotecas abertas.

Capítulo 5

Descrição do Sistema Proposto

Requisitos de CAD em Engenharia Concorrente

Conforme já apresentado em capítulos anteriores, um produto é desenvolvido segundo uma seqüência de atividades agrupadas em seu ciclo de vida. No Sistema de Manufatura tradicional as várias informações relativas ao produto estão distribuídas entre os diversos departamentos. Esta falta de integração acaba por acarretar um aumento no ciclo de vida do produto e, por consequência, dos custos envolvidos. A Engenharia Concorrente propõe-se a reduzir tais custos, através da integração das diversas áreas funcionais envolvidas, direta ou indiretamente, com a elaboração do produto. Discutiremos, a seguir, alguns aspectos relativos à construção de um ambiente de auxílio à Engenharia Concorrente.

A Questão da Comunicação

O principal problema quando da implementação de um ambiente baseado em Engenharia Concorrente está relacionado a encontrar uma forma de representar informações de diversas naturezas, utilizadas pelas várias atividades do ciclo de vida do produto, de modo a suportar as necessidades dos mais diferentes setores. Ou seja, deve existir uma forma normalizada para a comunicação entre as atividades.

A tecnologia de 'features' surge, então, como uma possível solução para o problema da troca de informações. Seu uso permite a definição de um produto, ou das partes do mesmo, não apenas a partir de simples elementos geométricos, mas também em termos de informações não-geométricas, como normas técnicas, ferramental e tecnologia de fabricação. Desta forma, o uso de 'features' permite englobar, sob uma mesma entidade, informações de diversas naturezas, relativas às várias formas com que uma mesma peça é vista por cada uma das atividades do ciclo de vida.

Bibliotecas de 'Features'

Já vimos que a identificação das 'features' é fortemente dependente dos produtos que são escolhidos como modelos. Assim, delimitado o universo de trabalho, é possível, através de um detalhado trabalho de identificação, definir quais são as 'features' básicas, a partir das quais projetos pertencentes ao domínio em questão podem ser construídos.

É possível, portanto, definir bibliotecas de 'features', que servem como fonte de elementos construtivos para a criação de novas peças. A utilização destas bibliotecas possibilitam, entre outras coisas, a melhoria na consulta e na atualização de normas técnicas utilizadas pelo projeto, a padronização e fácil atualização das informações de manufatura, tais como: normas, ferramentas, dispositivos de fixação, máquinas, e a padronização dos projetos.

Bibliotecas de 'features' são manipuladas em dois diferentes níveis. O primeiro, o nível construtivo, diz respeito à construção da biblioteca em si. Sempre que se percebe a

necessidade de se definirem novas 'features', estas precisam ser adicionadas à biblioteca; possivelmente 'features' já definidas necessitam ser modificadas ou mesmo retiradas da mesma. Este nível deve estar sob a responsabilidade de um administrador, uma pessoa ou equipe, que possua conhecimento suficiente para definir a 'feature' em termos de sua utilização em todo o Sistema de Manufatura.

O segundo nível, o de utilização das bibliotecas, é aquele com que trabalha o projetista. Por utilização de uma biblioteca de 'features' entende-se a capacidade de instanciar 'features' nela definidas, atribuindo os valores desejados para seus atributos.

Consulta a Informações

Uma 'feature' é uma entidade que existe para todo o ciclo de vida do produto, sendo composta por informações de diversas naturezas, que dizem respeito às diversas atividades que o compõe.

Uma vez que uma 'feature' é definida de forma a englobar todas as informações relativas a ela ao longo das diversas atividades, um sistema CAD baseado em 'feature' deve oferecer ao usuário, neste caso o projetista, a possibilidade de consultá-las. Este procedimento tem por objetivo procurar evitar que possíveis "erros" de projeto, ocasionados pelo não conhecimento de aspectos que dizem respeito a outras etapas do ciclo de vida, sejam detectados apenas nas fases subseqüentes a esta. Isto implicaria na necessidade de retorno do projeto para esta área e conseqüente aumento dos custos envolvidos.

Arquitetura Proposta para um CAD Baseado em 'Features'

O sistema proposto pode ser representado como na figura 5.1. Conforme ilustrado, o usuário (o projetista) interage com o sistema através de uma interface; esta, por sua vez, interage com um bloco de gerenciamento, responsável pela interligação com os demais elementos, quais sejam a biblioteca de 'features', a biblioteca de auxílio ao projeto e os arquivos de descrição de peça.

A *biblioteca de 'features'* possui a definição de todas as 'features' disponíveis no sistema. É onde são descritos os atributos de cada 'feature', bem como a forma geométrica associada. Uma série de valores comumente utilizados é inicialmente atribuída, podendo ser modificada pelo usuário no instante da instanciação.

A *biblioteca de auxílio ao projeto* contém as informações das diversas atividades do ciclo de vida que dizem respeito a uma 'feature'. São informações tais como normas técnicas, tecnologia de fabricação, controle de qualidade, vendas, etc. É formada por um conjunto de arquivos de descrição, cada um deles correspondendo a uma definição de 'feature' da biblioteca de 'features'. A ligação entre estes dois elementos é feita através de um apontador (um simples atributo cujo valor, atribuído pelo administrador, é o nome do arquivo de descrição), definido na descrição da 'feature'.

O bloco de *descrição de peça* é responsável por armazenar arquivos, contendo a *estrutura de descrição de peça*, juntamente com os atributos de cada uma das 'features' instanciadas pelo usuário para aquela peça. A *estrutura de descrição de peça* se presta a armazenar indicadores para cada uma das instâncias de 'feature' que irão compor a peça, na ordem devida em que foram instanciadas.

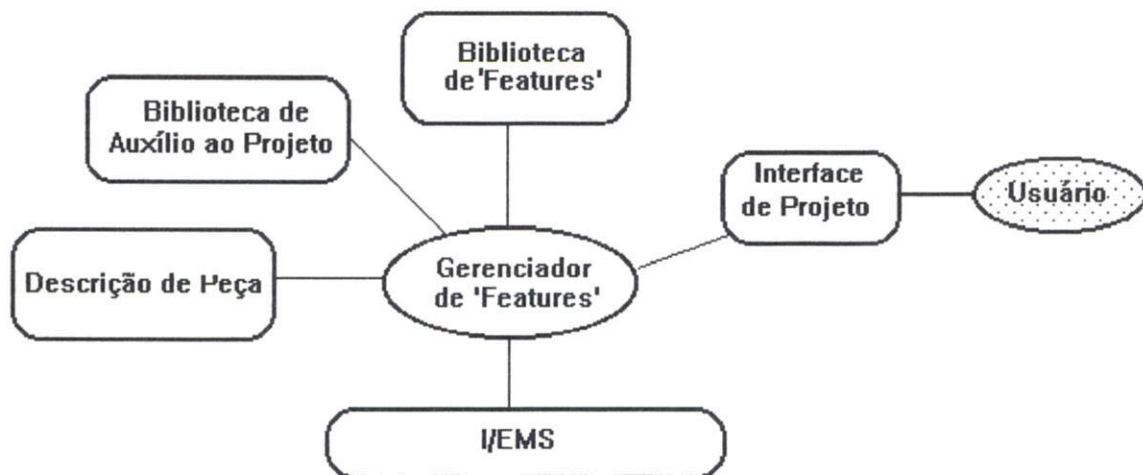


Fig.5.1: Arquitetura do Sistema Proposto

Servindo de base para toda esta estrutura, temos o I/EMS, sob o qual a interface (CAD) baseado em 'features' está definida.

A *interface de projeto* é uma interface gráfica composta por menus baseados em ícones. É através desta que o usuário interage com o restante do CAD, instanciando 'features' através da atribuição de valores a seus atributos, modificando-as posteriormente, deletando-as se necessário e consultando informações a ela associadas. Esta comunicação é feita através de um *gerenciador de features* que controla o sistema como um todo.

Descrição da Interface Baseada em 'Features'

Com base no modelo proposto, a atividade de projeto pode ser descrita pelo seguinte procedimento básico:

Por meio da *interface de projeto*, o projetista inicialmente cria uma nova peça. Só a partir deste momento, 'features' podem ser instanciadas. Este procedimento faz com que o *gerenciador* abra uma nova *estrutura de descrição de peça* para a peça em questão. Esta estrutura ficará armazenada em um arquivo de *descrição de peça*, juntamente com as 'features' instanciadas pelo usuário.

Em seguida, o usuário seleciona qual a 'feature' a ser instanciada. O *gerenciador de features* recebe esta informação e busca na biblioteca de 'features' a descrição correspondente; conhecida a descrição, ou seja, o conjunto de atributos que definem a 'feature', o usuário é questionado a respeito dos valores de atributos, e da posição em que a 'feature' deve ser colocada. Feito isso, o gerenciador armazena a descrição da 'feature' em um arquivo de *descrição de peça* e aciona uma série de funções do I/EMS a fim de posicionar graficamente a 'feature' instanciada.

Quando o usuário requisita um pedido de consulta através da *interface de projeto*, o *gerenciador* o questiona a respeito de qual a 'feature' a consultar, solicitando que o usuário a localize na tela. Assim, identificada a instância, seus atributos são recuperados do arquivo de *descrição de peça* correspondente, e as informações associadas recuperadas na *biblioteca de auxílio ao projeto*. Desta forma, o usuário pode verificar os valores dos atributos da 'feature', como também as informações de normas, processos de tratamento, materiais, etc. a ela associadas.

Operações de modificação e deleção são realizadas de fora semelhante ao procedimento de consulta. No caso de modificação, localizada a 'feature' na tela, o *gerenciador* recupera qual é o tipo de 'feature' instanciada, o que permite saber quais seus atributos. O usuário então atualiza os valores desejados, que são utilizados para modificar o arquivo de *descrição de peça*, bem como a representação geométrica na tela. Em se tratando de deleção, localizada a entidade, o *gerenciador* "apaga" sua definição no arquivo de descrição de peça, e posteriormente, sua representação na tela.

Uma série de verificações precisa ser realizada pelo *gerenciador* tanto no caso de pedidos de modificação quanto no caso de pedidos de deleção. Se, por exemplo, requisita-se uma modificação em um cilindro ao qual está associado um chanfro, faz-se necessário alterar, não apenas o cilindro mas também o chanfro. Da mesma forma, se requisitamos a deleção de uma entidade a qual diversas outras são associadas, o *gerenciador* deve questionar o usuário a respeito de qual o procedimento a ser seguido: ignorar a operação ou deletar todos os elementos associados, e então efetuar todas as alterações necessárias. A *estrutura de descrição de peça* auxilia nesta tarefa.

Ao usuário é permitido "reabrir" uma peça. Basta selecionar o ícone correspondente na *interface de projeto* e entrar com o nome da peça quando requisitado. O gerenciador, então, recupera o arquivo de descrição da peça correspondente, permitindo que se continue o trabalho sobre a mesma.

Gostaríamos de lembrar que a interface baseada em 'features' é definida sobre a interface padrão do I/EMS, à qual são acrescentadas novas funcionalidades no sentido de permitir o projeto por 'features'. Todas as ferramentas de construção próprias do I/EMS continuam a disposição do usuário.

Estruturas de Comunicação Usuário - Sistema

Conforme descrito acima, o usuário interage com o sistema como um todo através de uma *interface de projeto*. Esta interface constitui-se, basicamente, de um "menu de ícones", um conjunto de "forms" e um conjunto de janelas de "help".

O "menu de ícones" é responsável por permitir ao usuário a seleção da operação a ser realizada, de forma simples e clara. Cada ícone (botão) apresenta uma imagem/palavra-chave característica da função que realiza. A figura 5.2 ilustra um menu contendo todos os botões necessários para a realização das funções acima especificadas. Alguns botões têm por função apenas acionar pop-up menus, também compostos por ícones, enquanto que outros, os dos níveis inferiores, são responsáveis pela execução da operação em si.

A entrada/saída de dados é realizada por meio de "forms", uma espécie de janela gráfica contendo uma série de campos de diferentes tipos que recebem/exibem dados, promovendo a interação entre o usuário e o sistema. A figura 5.3 ilustra um exemplo de "form" no I/EMS. Quando uma entrada/saída de dados simples, como selecionar determinado elemento ou entrar/apresentar um único valor, é necessária, então o sistema apenas envia uma mensagem a um campo do I/EMS próprio para este fim, sem necessidade de um usar um "form".

As informações armazenadas na *biblioteca de auxílio de projeto* podem ser visualizadas a partir de arquivos de "help". Veja a figura 5.4 As informações contidas em cada um destes arquivos dizem respeito a uma 'feature' em particular, e podem ser visualizadas através de um pedido de consulta, realizado pela seleção do botão

correspondente no menu de ícones. Efetuando-se tal seleção, o sistema pede ao usuário que identifique a instância

"Forms", menus e arquivos de help podem ser construídos, no I/EMS, através da ferramenta I/FORMS. Maiores detalhes podem ser encontrados no próximo capítulo.



Fig.5.2: Menu de Ícones

feature forma: **Ranura Interna** X ✓
 funcao: **Anel de Vedacao**

diametro: + 0.00 1.00
 -

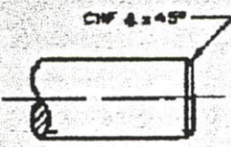
largura: + 0.00 0.50
 -

tolerancia de forma: tolerancia de posicao:
 circularidade concentricidade
 cilindridade

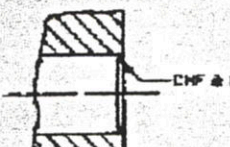
processo de fabricacao:

Fig.5.3: Form

Chanfro em Rosca Externa



a) Rosca Externa



b) Rosca Interna

FILETES POR 25,4 mm (1 pol)	Ø mm.
De 40 para cima	0,4
De 20 a 36	0,8
De 10 a 18	1,6
De 8 para baixo	3,2

Novo
Up
Press
Next
Back
Mark
Print

Fig.5.4: Help

Menu de 'Features'

O aspecto da funcionalidade associada a uma 'feature' é ponto fundamental para a atividade de projeto. Por este motivo, optou-se por dispor as 'features' dentro da estrutura de menu de acordo com sua funcionalidade. Para esclarecer este aspecto, vejamos o exemplo mostrado na figura 5.5.

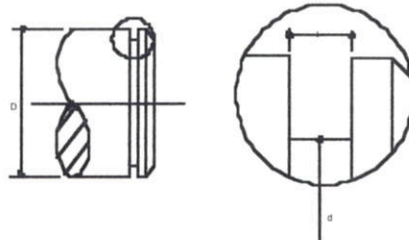


Fig.5.5: 'Feature' ranhura externa

Veja que, a partir deste mesmo elemento de forma, ranhura externa, é possível definir duas 'features' funcionais: ranhura para anel de vedação (atributos como inclinação nas paredes da ranhura, tolerância dimensional nos diâmetros, etc) e ranhura para colocação de anel de retenção (atributos como tolerância dimensional no comprimento, por exemplo). Estas, por sua vez, englobam diversos conhecimentos das várias áreas relacionadas ao produto, além da forma.

Assim, sendo, optamos por colocar as 'features' na estrutura de menus segundo uma hierarquia de dois níveis. As diversas 'features' que possuem funções diferentes, o que as leva a possuir atributos diferentes, mas possuem a mesma forma geométrica são reunidas em termos de um único botão, relacionado à forma. Selecionado este botão, abre-se o segundo nível, com uma série de 'features' funcionais.

As 'features' de forma apresentadas na figura 5.6 foram identificadas nos projetos da EMBRAER-EDE, e são suficientes para modelar a grande maioria dos produtos hidráulicos fabricados [MOR94].

'Features' de forma, entretanto, podem se tornar muito mais poderosas quando transformadas em 'features' funcionais pelo acréscimo de informações como acabamento superficial, tolerâncias, processos de tratamento térmico, etc. Uma vez que esta segunda etapa da identificação das 'features' ainda não foi concluída na EMBRAER-EDE, pela equipe responsável, tratamos apenas algumas das 'features' do conjunto, as quais já haviam sido definidas.

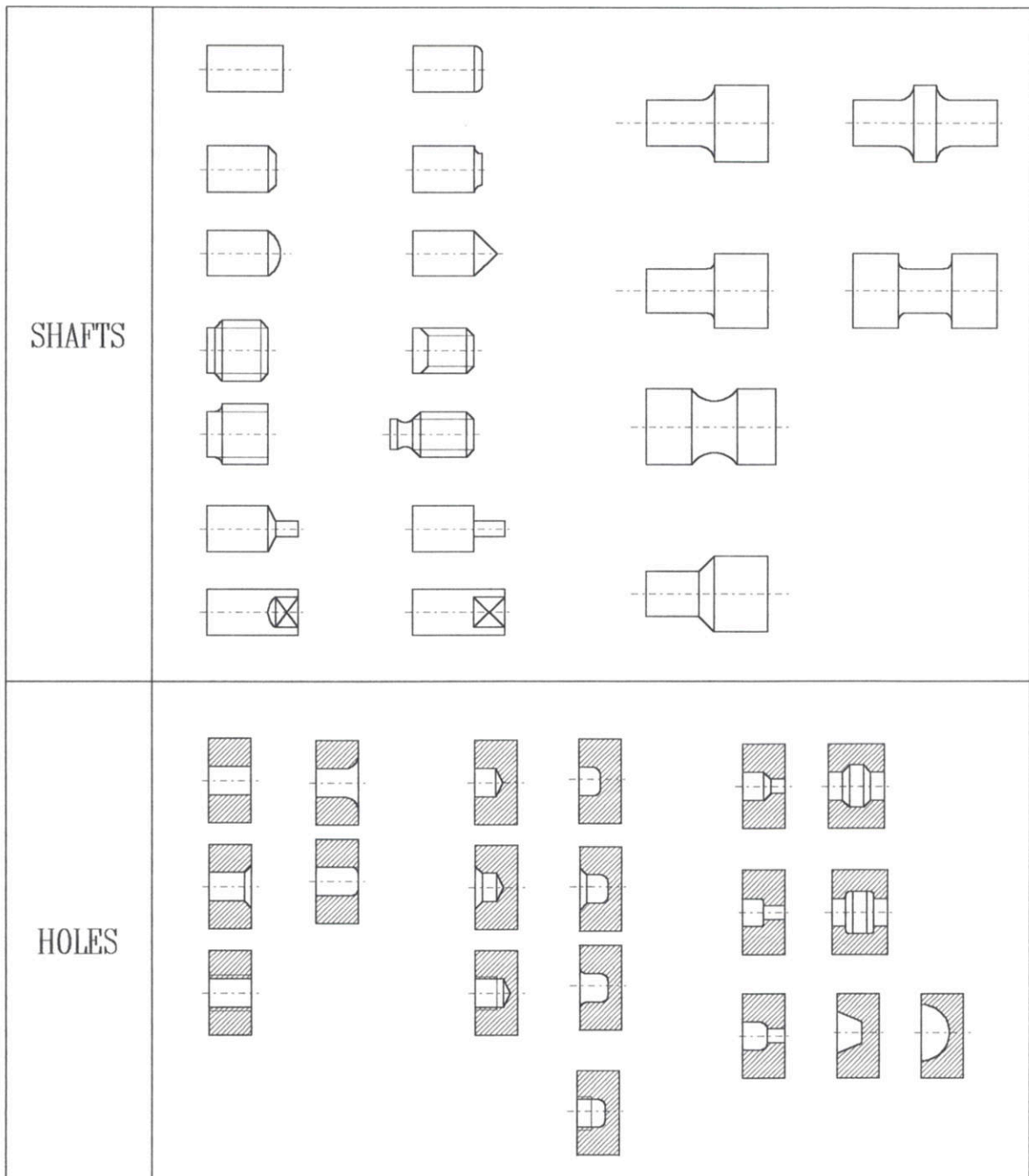


Fig.5.6: 'Features' de forma na EMBRAER-EDE

Posicionamento de 'Features'

Segundo a abordagem de projeto por 'features', uma peça é criada a partir de elementos construtivos básicos, as 'features'. Quando da construção de um modelo, faz-se necessário posicionar adequadamente as instâncias das 'features' na tela.

A definição da localização geométrica de um sólido de revolução qualquer é univocamente determinada a partir de um ponto de ancoragem e de um eixo de rotação, bem como o plano de referência que os contém. Dado que todas as 'features' a que este sistema se presta são do tipo sólido de revolução, faremos uso desta peculiaridade para efetuar o posicionamento das instâncias criadas.

Desta forma, sempre que da colocação de uma nova 'feature' na tela, o sistema interroga o usuário quanto ao ponto, eixo e plano em que esta deve ser colocada. Devido a limitações próprias do I/EMS, a linha que serve como eixo de rotação deve ser sempre paralela ao eixo x do sistema de referência do plano em que será posicionada (figura 5.7). Se desejarmos colocar uma instância formando um ângulo α com o plano em que estamos trabalhando, basta que definamos um outro plano sobre este, cujo sistema de referência forme este mesmo ângulo α com aquele do primeiro plano, e sobre este coloquemos o ponto de ancoragem e o eixo de rotação.

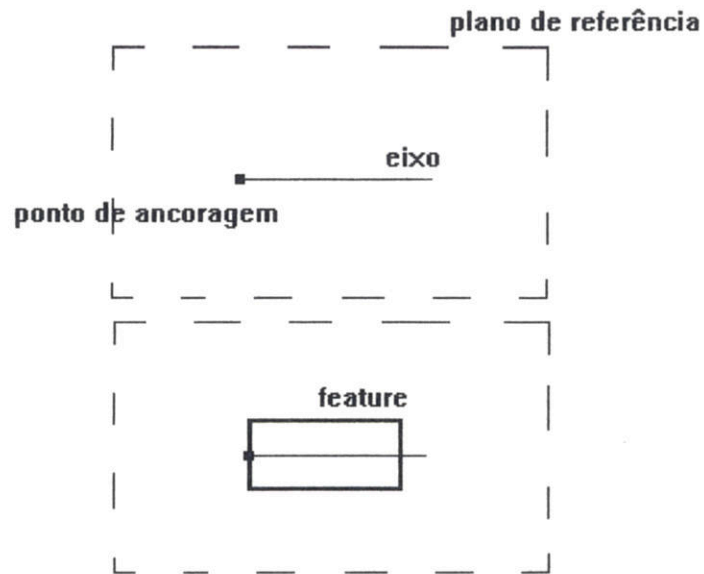


Fig.5.7: Posicionando 'features' rotacionais

Capítulo 6

Descrição da Implementação

O presente capítulo destina-se a descrever a forma de implementação do CAD baseado em 'features', sobre o CAD convencional da Intergraph, o I/EMS. Iniciamos o capítulo apresentando uma rápida descrição das ferramentas utilizadas. Na seqüência, são descritas as formas de implementação de cada um dos blocos da estrutura proposta, apresentada no capítulo anterior. Por fim, apresentamos um exemplo.

A fim de compreender exatamente o que foi feito em termos de implementação da arquitetura proposta, seria necessário que tivéssemos dedicado várias páginas deste trabalho a esclarecer o funcionamento do CAD I/EMS. Por crer que este não seja o objetivo do projeto, apenas apresentaremos a solução por nós encontrada, esclarecendo os aspectos de caráter geral.

Ferramentas Utilizadas

As ferramentas utilizadas são a seguir relacionadas, juntamente com algumas de suas características gerais.

O CAD I/EMS

O I/EMS - Intergraph Engineering Modelling System - é, basicamente, uma ferramenta de desenho para uso em estações de trabalho, combinando o poder de processamento destes equipamentos com as estruturas de dados e técnicas avançadas de programação para oferecer uma ferramenta versátil para a geração de modelos e desenhos de peças mecânicas. Consiste em um sistema que integra diversos tipos de modelagem, a modelagem wireframe 3D, a modelagem por superfícies e modelagem sólida. O I/EMS produz gráficos utilizando a tecnologia NURBS (B-splines racionais não uniformes).

Através do uso da geometria variacional associativa, o I/EMS permite a criação de modelos "dirigidos por dimensões". Construindo um modelo e colocando as dimensões necessárias, é possível controlar seu tamanho, forma e localização através de restrições.

O I/EMS oferece uma série de facilidades, no sentido de facilitar as tarefas de projeto. Funções como "shading" estão a disposição para auxiliar na visualização de entidades geométricas criadas. Janelas auxiliares podem ser criadas, e técnicas de manipulação de visualização permitem olhar o modelo a partir de várias perspectivas, simultaneamente.

O usuário interage com o I/EMS através de uma tela gráfica, na qual estão disponíveis diversos comandos, sob a forma de ícones, organizados em uma estrutura de menus. Um poderoso "help on-line" está a disposição do usuário, para tirar dúvidas com relação aos comandos do I/EMS que este possa ter durante a atividade de projeto.

A figura 6.1 mostra a tela do I/EMS, juntamente com uma peça nele projetada.

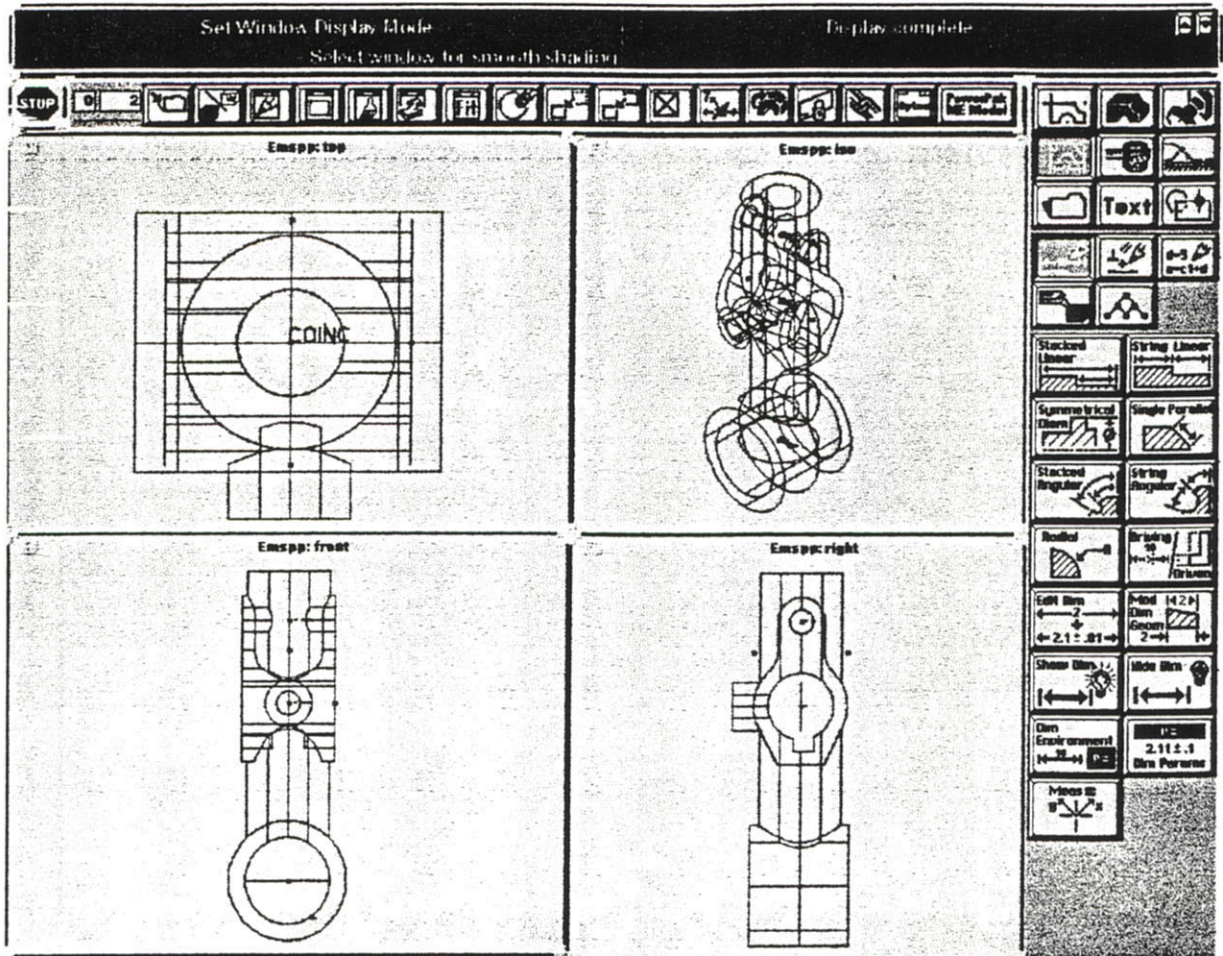


Fig.6.1: O CAD I/EMS

Algumas características do I/EMS:

- facilidade de utilização, uma vez que o CAD é formado por menus de tela baseados em ícones;
- grande número de comandos disponíveis ao usuário, permitindo efetuar as mais diversas operações;
- "help on-line", que possui descrição dos comandos do I/EMS;
- geometria variacional, que permite modificar modelos alterando apenas as dimensões a ele associadas;
- associatividade, que fornece uma maneira simples de associar objetos, e fazer com que as relações entre eles sejam mantidas quando um ou mais objetos são manipulados;
- possibilidade de visualização dos modelos sob diversas perspectivas (ISO, vista frontal, lateral, de topo, etc.) através de um sistema de janelas.

Apresentaremos, em seguida, sucintamente, alguns dos conceitos básicos do I/EMS.

Geometria Variacional

Geometria Variacional consiste de elementos wireframe (elementos tipo pontos, linhas, polígonos, círculos, etc), cujo tamanho, forma e localização podem ser controlados por meio de restrições. As restrições permitem, de uma forma simples, mudar a geometria sem ter que alterar cada elemento individualmente. Por exemplo, você pode modificar a geometria variacional mostrada na figura 6.2 apenas alterando o valor da restrição dimensional relativa à altura.

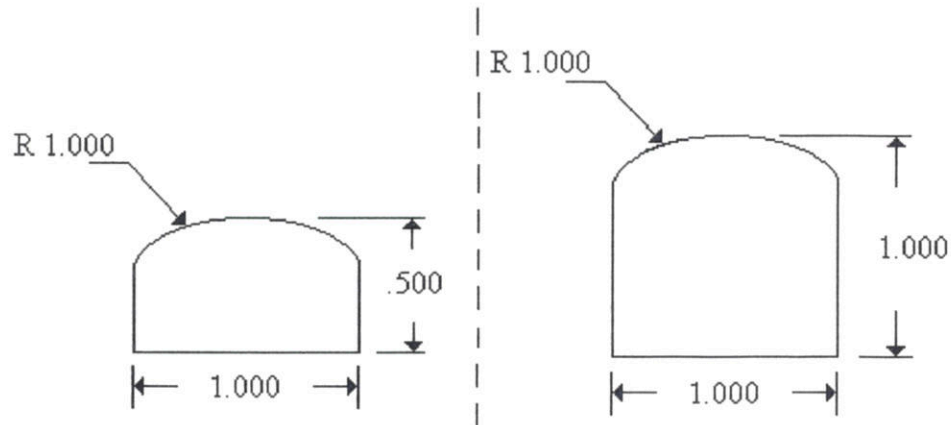


Fig.6.2: Alterando um modelo via restrições

Um elemento associativo é um elemento que depende de um ou mais elementos. Quando você modifica algum dos elementos aos quais este elemento associativo está ligado, ele automaticamente se altera, de forma a manter sua relação com aqueles. Os conceitos de variacionalidade e associatividade funcionam conjuntamente, permitindo construir e modificar modelos de uma forma rápida e simples.

"Profiles"

Elementos variacionais (apenas pontos, linhas, arcos e círculos), juntamente com restrições sobre estes elementos, podem ser utilizados para construir perfis ("profiles"). Estes elementos, por sua vez, podem ser utilizados para gerar sólidos e superfícies com alto grau de complexidade.

Existem duas formas básicas de criar perfis no I/EMS: criando diretamente uma entidade de natureza variacional, ou criando uma entidade de natureza não-variacional, e convertendo-a para variacional. Este segundo procedimento é muitas vezes interessante pois o conjunto de comandos para elementos tipo não-variacional disponíveis no I/EMS é muito grande, desta forma possibilitando construir alguns modelos mais facilmente do que diretamente com geometria variacional.

Planos de Referência

Existem três planos de referência padrão no I/EMS, cada um deles paralelo a uma das vistas frontal, lateral direita e de topo. Planos de referência são associativos aos elementos neles criados. Assim, quando o plano é movimentado na tela, os elementos a ele associados também o são. Além dos planos padrão do sistema, é possível criar tantos planos quanto quisermos, e em qualquer posição do espaço.

Apenas um plano de referência pode estar ativo a cada instante. Se estivermos manipulando elementos associativos, estes garantidamente estarão sendo criados sobre este plano, e serão associativos a ele.

Planos de referência são de fundamental importância, já que perfis precisam de um plano de referência para serem criados.

Restrições

Um ponto no espaço possui três graus de liberdade, correspondendo a movimentos nos três eixos coordenados. Em se tratando de pontos pertencentes a um perfil, apenas dois graus de liberdade são permitidos, já que um perfil está sempre sobre um plano de referência. De forma equivalente, uma linha qualquer é definida por dois pontos, logo, seis graus de liberdade; se a linha em questão for um perfil, serão quatro os graus de liberdade.

Depois de criada a forma geométrica de um perfil, é necessário que se coloque as restrições sobre este, de forma tal a restringi-lo por completo ("fully constrained"), não deixando que este possua nenhum grau de liberdade. Isto é feito por meio de restrições.

Restrições não só garantem que o modelo geométrico conservará sua forma quando forem feitas quaisquer manipulações sobre ele, como também facilitam, em muito, a atividade de construção do mesmo. Restrições só podem ser aplicadas a elementos variacionais, e podem ser de três tipos: dimensionais, geométricas e algébricas. A figura 6.3 ilustra um profile "fully constrained".

Restrições dimensionais são aquelas que atribuem valores numéricos a elementos, ou a relações entre eles. Editando restrições dimensionais é possível alterar um elemento, conforme ilustrado na figura 6.2. Restrições geométricas, por sua vez, controlam relações geométricas entre elementos, sem fazer qualquer uso explícito de valores numéricos. Existem em torno de dez restrições deste tipo definidas no I/EMS. Por exemplo, aplicando-se uma restrição *horizontal* a uma reta, garante-se que esta será sempre horizontal em relação ao seu plano de referência.

Restrições algébricas usam expressões para definir restrições dimensionais em termos de outras restrições dimensionais ou variáveis. Um restrição algébrica pode fazer com que a dimensão de um elemento seja sempre o dobro da dimensão de um outro elemento, por exemplo.

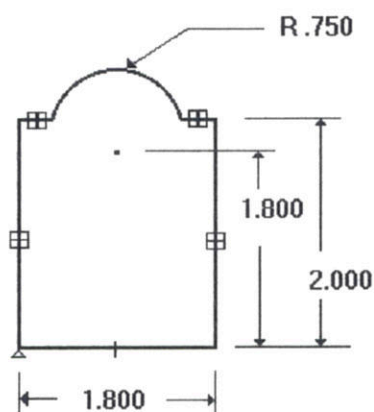


Fig.6.3: Uso de restrições - profile "fully constrained"

Modelando Sólidos no I/EMS

Um sólido é um objeto tridimensional que cerca e preenche um volume no espaço. Superfícies cercam um volume, mas não o preenchem. Wireframe, por sua vez, apenas contornam um volume, mas não o preenchem nem o fecham.

Sólidos podem ser construídos no I/EMS através de primitivas sólidas ou via "profiles". Primitivas sólidas podem ser criadas isoladamente, sem necessitar de outros elementos para serem geradas. São exemplos primitivas como cilindro, esferas, cones e paralelepípedos.

Sólidos derivados são criados a partir de outros elementos. Construindo "profiles", é possível, através de operações como rotação, projeção, "skinning" e "Drafted" dar origem a modelos sólidos.

Muitos modelos geométricos sólidos não podem ser criados apenas pelo uso destes elementos. Deste forma, o I/EMS oferece a possibilidade de efetuar modificações sobre os sólidos acima definidos. Tais modificações podem ser feitas através da aplicação de operações booleanas (diferença, intersecção e união), que combinam dois ou mais sólidos a fim de obter um outro, ou através de modificações de características, que se aplicam a um sólido individualmente. São exemplos a colocação de furos e bossas, assim como o arredondamento de cantos.

I/FORMS

O I/FORMS - Intergraph User Interface Development Toolkit - é um conjunto de utilitários e bibliotecas de funções que permitam ao usuário:

- modificar a interface com o usuário de aplicações baseadas no I/FORMS já existentes, quer sejam elas associadas ao próprio I/EMS, quer definidas para a aplicação do usuário ;
- criar um interface com o usuário para suas próprias aplicações.

O I/Forms é composto por quatro módulos básicos:

- Forms - permite ao usuário criar "forms", que podem receber os mais diversos tipos de dados. A ferramenta gráfica Form Builder oferece uma forma simples de criar tais entidades gráficas. Uma vasta biblioteca de funções é utilizada para manipular o "form" criado.
- Menus - criação de grupos de botões que podem ser utilizados para executar uma seqüência de operação definida pelo usuário. Uma ferramenta gráfica, o Menu Builder, está disponível para auxiliar na tarefa de construção de menus.
- Help - permite a criação de "help on-line".
- Messages - oferece a possibilidade de definir mensagens de erro e de diálogo com o usuário.

Obs: Não há ferramentas gráficas disponíveis para auxiliar na definição de "helps" e "messages".

PPL

Embora o I/EMS ofereça ao usuário uma grande quantidade de comandos de construção/manipulação gráfica, é possível que surja a necessidade de se definirem novos comandos para aplicações específicas do usuário.

O I/EMS permite que novas definições de comandos sejam feitas através da utilização de uma linguagem de desenvolvimento própria, o PPL - Intergraph Parametric Programming Language. Inspirada na linguagem de programação C, a sintaxe do PPL é muito similar a daquela linguagem.

Utilizando PPL, um novo comando pode ser criado através da edição de um arquivo de texto ASCII, via um editor qualquer. Desta forma, por exemplo, uma seqüência de comandos utilizada com freqüência pode ser "transformada" em um único comando, e entidades gráficas podem ser criadas e manipuladas automaticamente através de dados armazenados em um arquivo, sem necessidade de interação direta com o usuário.

Através de um arquivo de comando escrito em linguagem PPL é possível:

- criar, deletar, localizar e manipular elementos gráficos;
- fazer chamadas a comandos já existentes;
- receber dados do teclado, mouse ou arquivo;
- fornecer entradas para comandos;
- executar outros arquivos de comando, bem como funções do sistema operacional, entre outros.

A linguagem PPL pode ser utilizada em conjunto com a ferramenta I/FORMS, a fim de controlar "forms" definidos pelo usuário da forma mais adequada, assim como manipular "forms" que não estejam sob seu controle (os "forms" do próprio I/EMS, por exemplo).

Biblioteca de 'Features'

Bibliotecas de 'features' servem como fonte de elementos construtivos para a criação de peças, segundo a abordagem de projeto baseado em 'features'. Para que o projetista possa fazer uso de um CAD baseado em 'features', é necessário que os elementos construtivos de que ele necessita, as 'features', estejam previamente definidas em uma biblioteca de 'features'. Ainda não existe consenso quanto a forma de realizar este procedimento. Devem os projetistas possuir esta liberdade? Ou deve a equipe multifuncional fazê-lo?

Da forma como definimos a construção de uma biblioteca de 'features', qualquer pessoa pode inserir, deletar ou modificar 'features', desde que conheça o I/EMS, e o mecanismo desta definição.

Definição das 'Features' na Biblioteca

A definição das 'features' na biblioteca correspondente é feita de forma manual. Sua geometria é definida através da construção de uma *macro*, enquanto que seus atributos são definidos em uma estrutura.

Definição da Forma

Por trabalhar com geometria variacional associativa, o I/EMS apresenta certas facilidades quando se deseja manipular elementos geométricos. É possível, construindo um modelo e colocando as dimensões necessárias, controlar seu tamanho, forma e localização através de restrições.

Entretanto, o I/EMS possui certas limitações com relação à movimentação e reprodução de elementos gráficos de forma direta, via comandos tipo "move" e "copy" disponíveis do menu do sistema. Por exemplo, não é possível movimentar elementos variacionais na tela, bem como não é possível copiá-los, e fazer com que continuem a ser variacionais; e se deixam de ser variacionais, então não mais temos o poder de alterar sua geometria facilmente. Desta forma, não é possível instanciar uma 'feature' de forma, apenas copiando um modelo geométrico e posicionando-o na tela.

A maneira adequada de solucionar esta questão de definição das 'features' de forma, é fazer uso de uma ferramenta oferecida pelo I/EMS, a possibilidade de se definirem "macros".

Uma "macro" é um modelo geométrico definido pelo usuário, devidamente armazenado em uma biblioteca de macros, e que pode ser reproduzido em qualquer ponto do espaço, não necessariamente com os mesmos valores dimensionais da definição. Assim, como podemos facilmente perceber, é possível definir e instanciar geometricamente uma 'feature' utilizando a estreita relação entre estas e as macros oferecidas pelo I/EMS. Até mesmo a idéia de armazenamento em uma biblioteca de 'features' é característica presente nas macros. Atentemos para o fato de que macros são capazes de manipular apenas geometria. Desta forma, só se prestam a definir as 'features' de forma, não as 'features' funcionais.

Então, para cada uma das 'features' que definimos, criamos um modelo geométrico 2D, usando "profiles", colocamos as devidas restrições, e geramos o sólido 3D via operação de revolução. Em seguida, criamos uma definição de macro a partir deste modelo e a armazenamos em uma biblioteca de macros. Esta ainda não pode ser verdadeiramente chamada de biblioteca de 'features', uma vez que contém apenas dados de ordem geométrica.

Maiores detalhes de procedimento de geometrias e macros podem ser encontrados nos manuais do I/EMS ou no "Help on-line". Apenas alguns aspectos relevantes, não contidos na documentação citada, e que foram "descobertos" usando a técnica da tentativa-erro, merecem ser aqui citados:

- Os planos de referência padrão do sistema jamais devem ser utilizados como planos de referência na construção da geometria que definirá a macro. Macros definidas utilizando tais planos não podem ser instanciadas em qualquer lugar do espaço;
- Defina a geometria da macro tendo por plano de referência, um plano paralelo a um dos planos padrão. A distância entre eles deve ser nula para que seja garantido o correto posicionamento da instância criada
- Utilize geometria variacional na criação dos modelos. Isto não só permite a facilidade de construção, como também a de instanciação. Lembre-se do conceito de geometria variacional.

- Quando definindo uma macro do tipo sólido de revolução, construa um perfil (elemento variacional 2D) e coloque um ponto de ancoragem neste. Não utilize a restrição "ground", mas sim coloque um ponto auxiliar na tela, não variacional, e restrinja um ponto qualquer do perfil com relação àquele, usando uma restrição de coincidência.

Definição dos Atributos

Macros não suportam atributos do tipo não-dimensionais. Portanto, os atributos das 'features' de origem não-geométrica, como material e tecnologia de fabricação, devem ser definidos em algum outro local dentro do sistema, e associados à macro (que define a forma). A fim de armazenar atributos não-geométricos, optamos por definir uma estrutura para cada 'feature' criada e armazenada na biblioteca de 'features'.

Desta forma, para cada 'feature' definida, geometricamente falando, construímos um pequeno programa, em linguagem PPL, onde armazenamos os atributos da 'feature', na forma de uma estrutura. Este programa será chamado quando o usuário selecionar o botão relativo à criação de uma 'feature'.

Instanciando 'Features'

Definidas as 'features' na biblioteca, estas já podem ser instanciadas. Para tal, o usuário seleciona o botão correspondente à 'feature' desejada no menu de ícones, efetuando a chamada do programa correspondente.

O programa chamado apresenta o "form" de entrada daquela 'feature' na tela, e espera que o usuário dê valores aos atributos da mesma. Quando o usuário seleciona o botão de execução, este pequeno programa faz uma chamada a um outro programa, responsável por criar e posicionar a instância de macro na tela, com os valores dados pelo usuário. Por fim, retornando ao primeiro programa chamado, um "part" é criado para aquela instância.

O fato de termos uma PPL apenas para a construção gráfica da 'feature' na tela deve-se ao fato de que, todas as 'features' funcionais associadas a uma mesma 'feature' de forma utilizam-se da mesma definição de macro, e conseqüentemente no mesmo procedimento para sua apresentação gráfica.

A questão da criação de um "part", acima citada, diz respeito à necessidade de se armazenar os valores dos atributos da 'feature', já que o usuário pode desejar visualizá-los a posteriori. Através do uso de "parts" é possível acrescentar dados à estrutura de diretórios interna do I/EMS, onde são guardadas todas as informações a respeito das entidades gráficas criadas. Posteriormente, via funções PPL, é possível recuperar essas informações (é assim que a função de consulta recupera os valores dos atributos das 'features'). Para maiores detalhes a respeito do uso de "parts" consulte os manuais do I/EMS.

Criação de "Forms"

Como cada uma das 'features', de um modo geral, possui atributos diferentes das demais, faz-se necessário ter um "form" de entrada/saída para cada uma delas, no qual o usuário atribui/visualiza valores aos atributos da 'feature' que deseja criar ou consultar (via procedimento de consulta).

Assim sendo, para cada uma das 'features' definidas construímos os "forms" necessários, utilizando a ferramenta Form Builder, do I/FORMS. A manipulação de cada "form" é realizada através de um arquivo, escrito segundo a linguagem de programação PPL, por meio de uma série de funções do I/FORMS.

Biblioteca de Auxílio ao Projeto

Definida a 'feature', a próxima etapa é construir o *arquivo de auxílio ao projeto* associado. Este deve permitir ao usuário visualizar informações relativas a 'feature' a que está relacionado.

Os arquivos de auxílio ao projeto são implementados através de "helps" do I/EMS. "Helps" possuem uma série de facilidades relativas à visualização das informações neles contidas. Possuem um formato padrão, e todo um sistema de controle próprio. Basta apenas editar um arquivo de texto (com possível chamadas de figuras), em um editor qualquer, de acordo com um conjunto de regras. Então, é só compilá-lo e "ligá-lo" a um botão da interface, para que o usuário, ao clicar sobre o botão, possa ver as informações nele contidas.

A ligação entre as informações e um ícone na tela é também feita através de programa PPL, que efetua uma chamada ao sistema e executa a ferramenta "Browser". Esta, por sua vez, apresenta a janela de "help" na tela e efetua o controle necessário.

Quando o usuário efetua um pedido de consulta (clizando sobre o ícone correspondente), um programa PPL é executado. Este solicita que o usuário selecione na tela a 'feature' a respeito da qual deseja receber informações. Os dados armazenados no "part" correspondente, no instante da instanciação, são então recuperados e apresentados ao usuário num "form", juntamente com o arquivo de auxílio ao projeto.

Gerenciador de 'Features'

Na verdade não existe um bloco gerenciador como entidade única, conforme arquitetura proposta. O gerenciamento é executado por vários blocos, cada um deles associado a uma função disponível na interface de projeto.

Interface de Projeto

A interface de projeto é composta basicamente por um menu de ícones. Cada ícone possui uma função associada (um programa PPL), que é executado quando o usuário faz uma seleção no menu. Estas funções, entre outras coisas, abrem "forms" para entrada/saída de dados, e "helps" associados às informações de auxílio ao projeto.

A construção de menus pode ser realizada via "Menu Builder", uma ferramenta da I/FORMS, ou de forma semelhante aos "helps", via edição de um arquivo que segue um determinado padrão.

A fim de inserir o menu desejado na interface do CAD I/EMS, é necessário compilá-lo com o menu padrão, além de alterar uma série de arquivos do sistema, tais como o *product_data* e o arquivo de chamada.

Exemplo de Utilização da Interface Baseada em 'Features'

O seguinte modelo foi construído usando a a estrutura proposta:

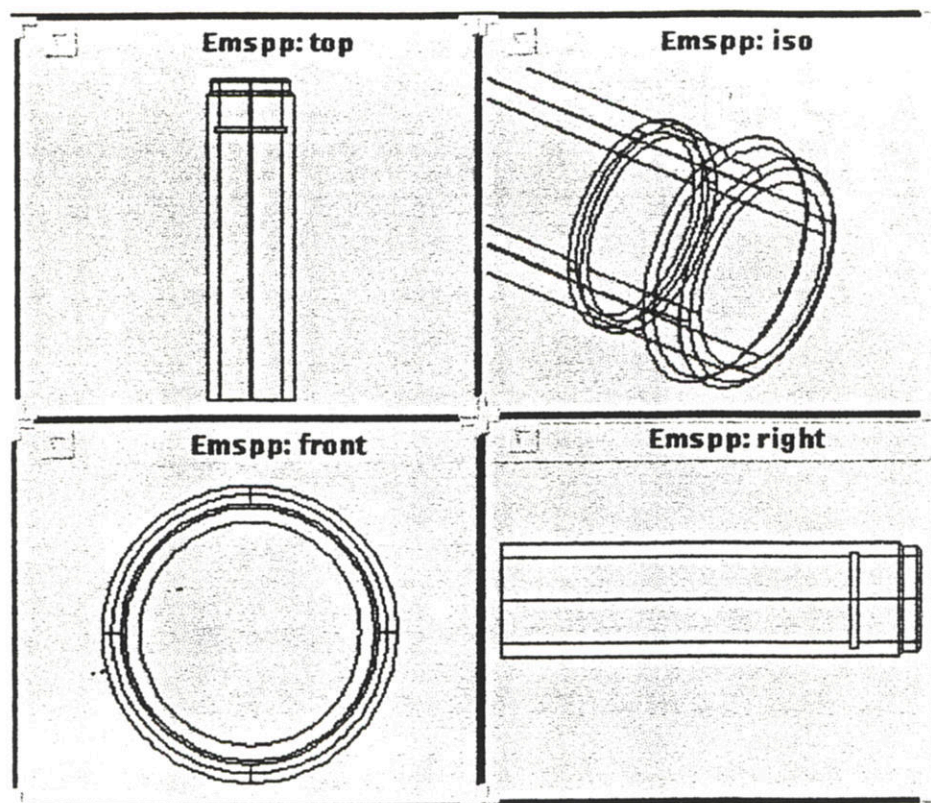


Fig.6.4: Peça construída usando a interface baseada em 'features'

Utilizando a interface baseada em 'features' poderemos proceder como segue:

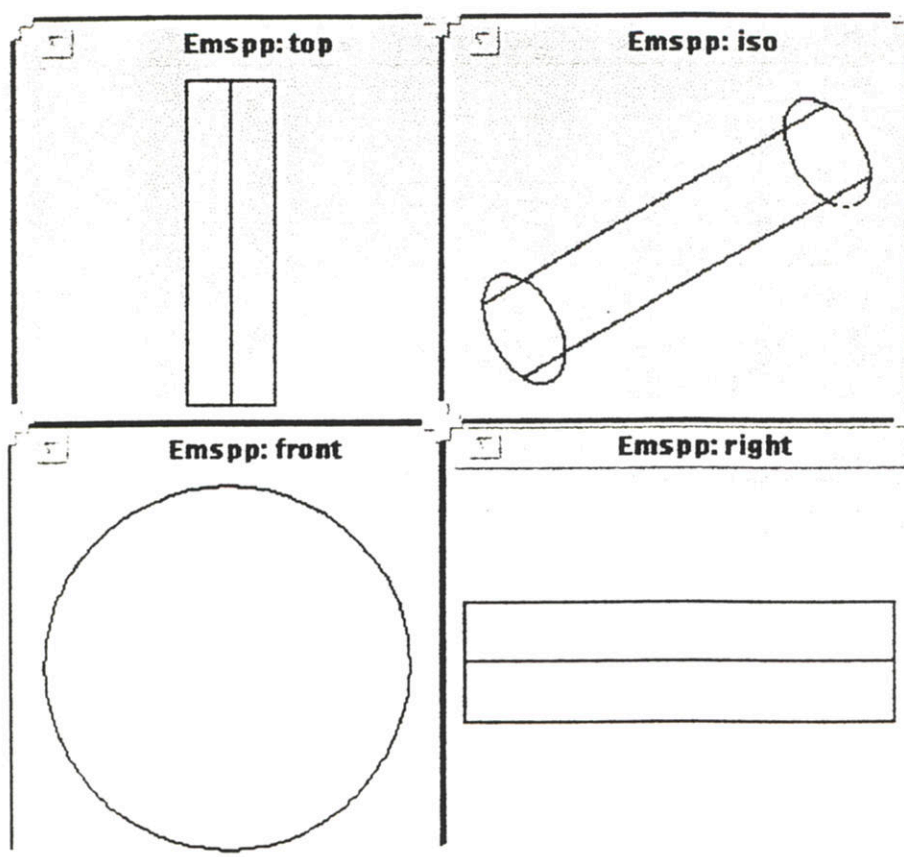


Fig.6.5: Posicionamento de cilindro 1

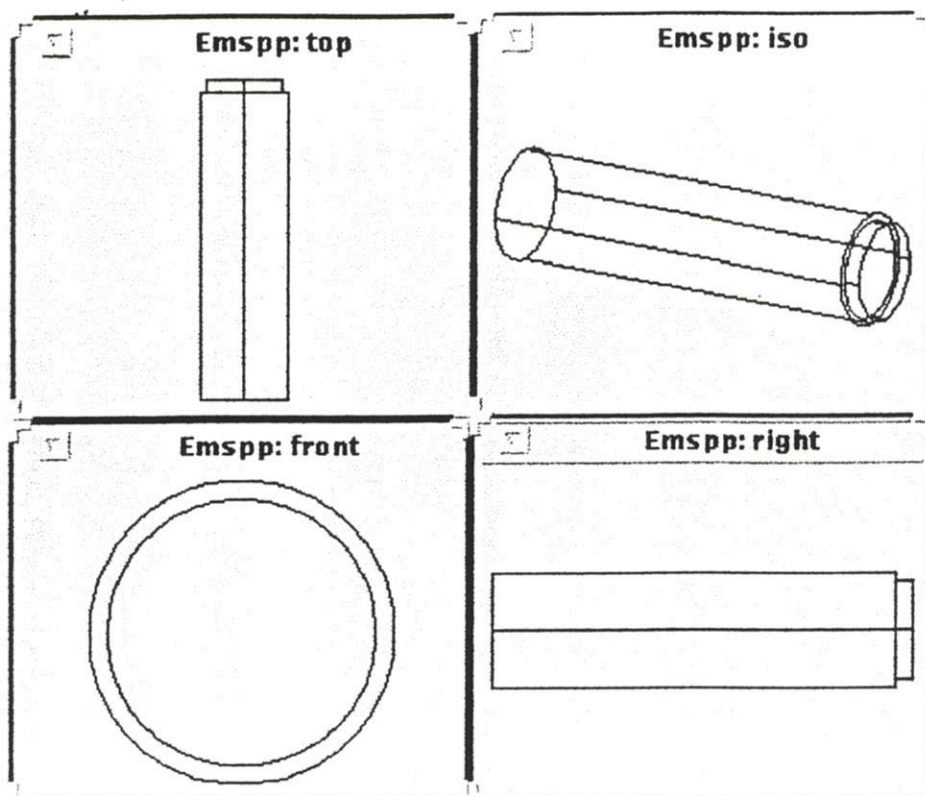


Fig.6.6: Inserção de cilindro2

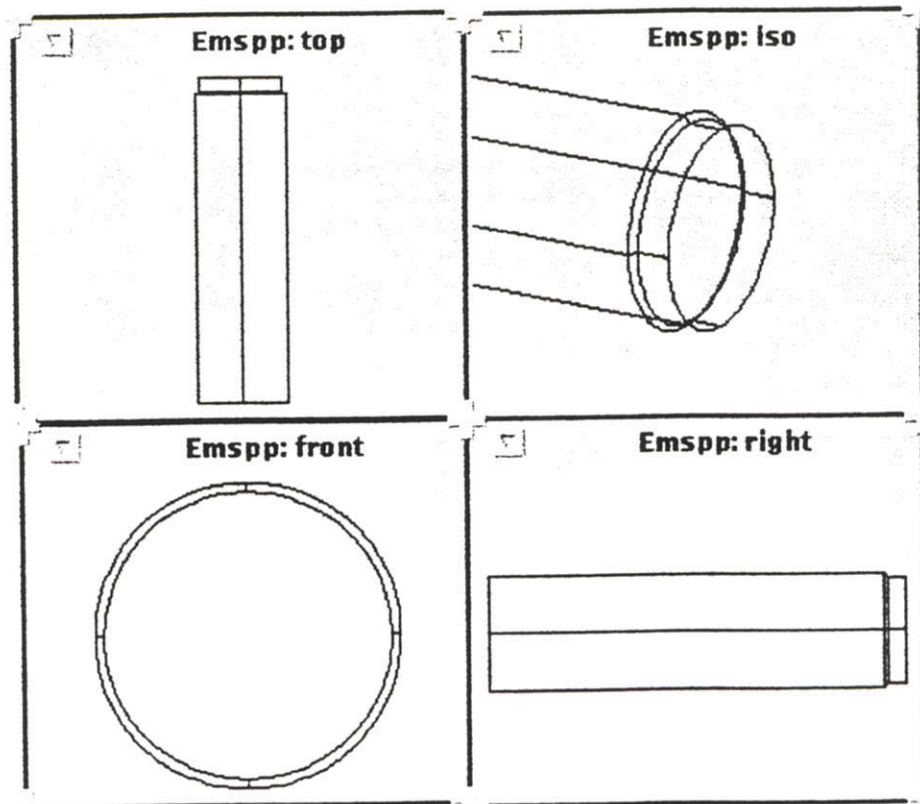


Fig.6.7: Inserção de chanfro 1

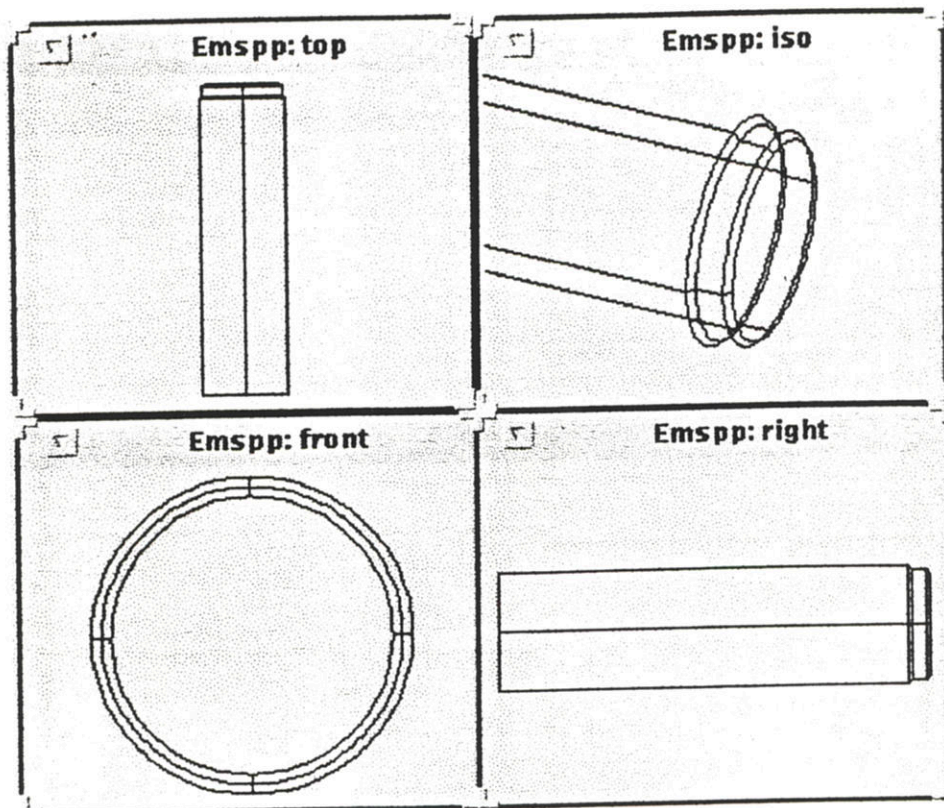


Fig.6.8: Inserção de chanfro 2

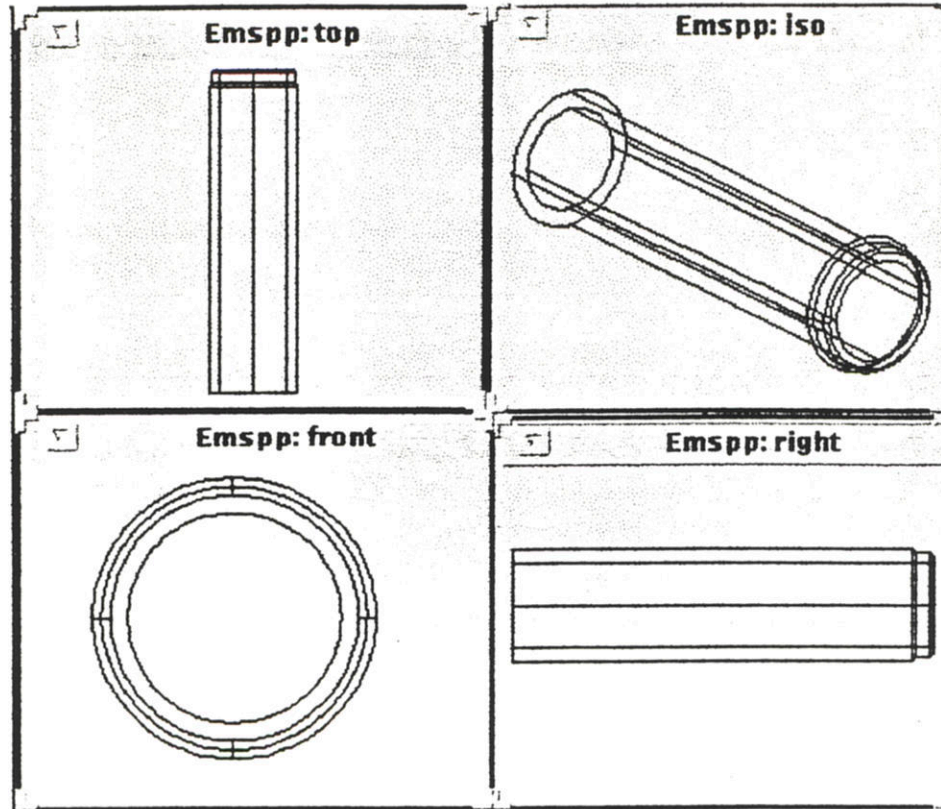


Fig.6.9: Inserção de furo

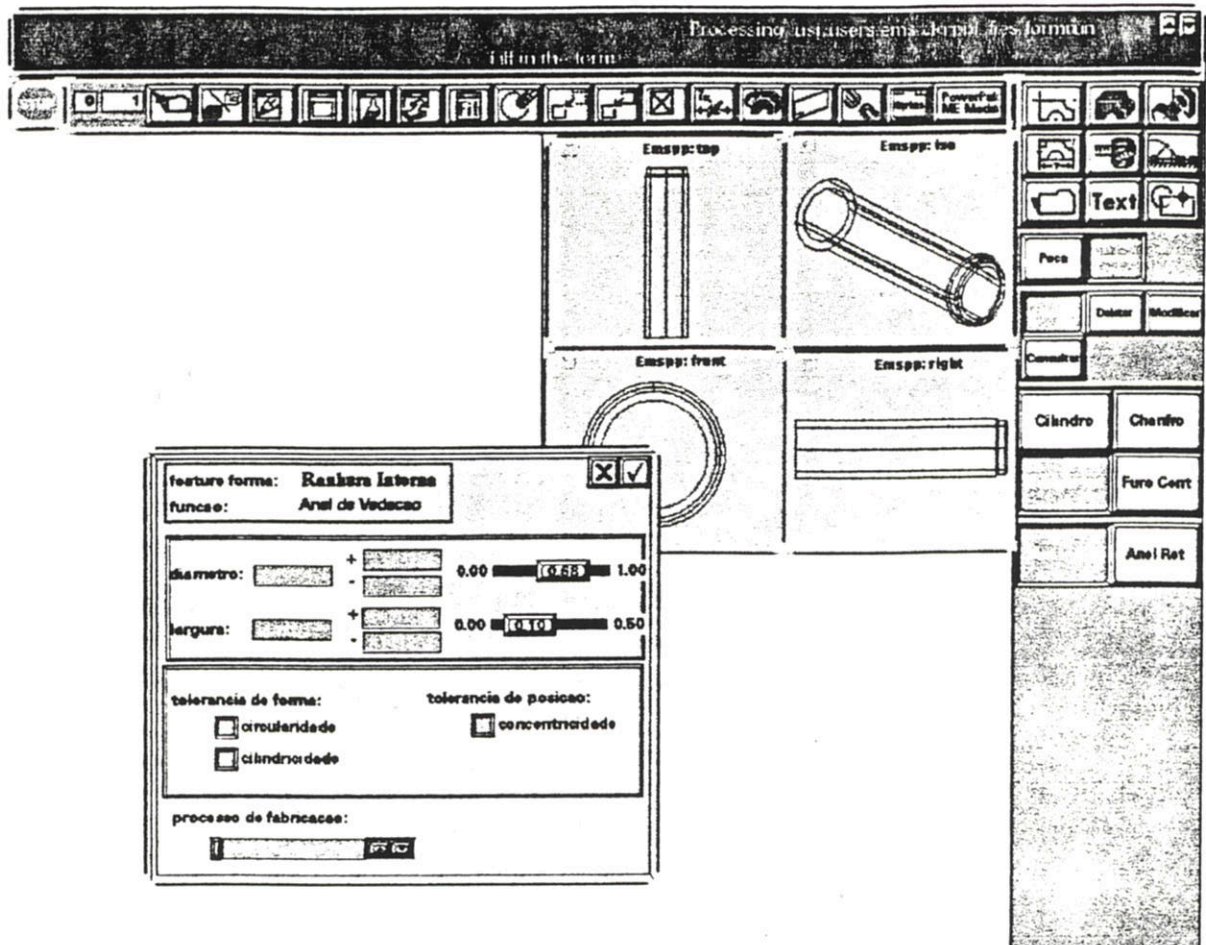


Fig.6.10: Inserção de ranhura interna

Capítulo 7

Conclusões

Objetivos

O presente trabalho visa a definição e implementação de um CAD baseado em 'features', utilizando, como base, um CAD convencional: o I/EMS, da Intergraph. Esta ferramenta tem por objetivo colaborar na implantação da Engenharia Concorrente na EMBRAER - EDE, servindo de ferramenta de auxílio às equipes multifuncionais, formadas por especialistas das mais diversas atividades do ciclo de vida do produto.

Resultados Obtidos

Grande parcela do tempo que dedicamos a este trabalho foi despendida na busca de soluções, no I/EMS, para cada uma das questões envolvidas com a inclusão de 'features' a um sistema CAD convencional, transformando-o em um CAD baseado em 'features'. Diversas possibilidades foram analisadas, uma série de testes efetuada sobre o CAD I/EMS.

Após os estudos realizados, consideramos que a implementação completa do sistema tomará ainda mais tempo de desenvolvimento até que se possa, efetivamente, solucionar as dificuldades encontradas. O principal problema encontrado se refere à identificação dos objetos depois de instanciados e alterados por operações booleanas (nesta situação, eles perdem seu identificador interno).

Acreditamos que este trabalho seja uma contribuição significativa no sentido de oferecer uma proposta de estrutura para um CAD baseado em 'features', bem como de indicar qual o caminho para realizá-la no I/EMS.

Trabalhos Futuros

Uma vez definida a estrutura da interface baseada em 'features', e descobertos os mecanismos do I/EMS a partir dos quais é possível realizá-la, sua implementação está pronta para ser efetivamente colocada em prática. Definido o conjunto de 'features' adequado às necessidades particulares de cada empresa, e seguindo os procedimentos aqui propostos para criação, instanciação e consulta de 'features', pode-se construir um CAD baseado em 'features' capaz de acrescentar inúmeras facilidades à atividade de projeto e consulta a informações a respeito do significado de uma 'feature' para as várias atividades da empresa.

Quanto ao modelo proposto, este não foi definido de forma a cobrir todas as questões envolvidas na construção/utilização de um CAD baseado em 'features'. O procedimento de deleção de 'features', por exemplo, constitui um problema relativamente complexo, com o qual não nos preocupamos em nosso modelo.

Outro aspecto importante é o fato de que as 'features' são definidas manualmente pelo usuário que, para tal, deve possuir uma boa compreensão do I/EMS, como também da

idéia envolvida por trás desta definição. Assim, outra grande colaboração que pode ser dada ao sistema é a de se construir um "compilador" capaz de fazer a conversão do desejo de usuário, expresso de uma forma simples, em algumas linhas de comando do I/EMS.

Capítulo 8

Referências Bibliográficas

A bibliografia apresentada a seguir diz respeito, em sua grande maioria, à literatura encontrada a respeito de Tecnologia de 'Features' e engenharia Concorrente. Quanto à utilização do CAD I/EMS, e das ferramentas PPL e I/FORMS, fizemos uso dos diversos manuais disponíveis, bem como do "Help on-line".

[BOO] Boothroyd, G. Dewhurst, P. "Design For Assembly - Selecting the Right Method"

[BRA90] Brazier, D. Leonard, M. "Concurrent Engineering: Participating in Better Design." Mechanical Engineering. January, 1990.

[BRO93] Bronsvort, W.F. Jansen, F.W. "Feature Modelling and Conversion: Key Concepts to Concurrent Engineering", 1993, Computers in Industry 21, Elsevier (1993), pp. 61-86.

[CAM94] Camarinha-Matos, L.M. Osório, A.L. "An Integrated Platform for Concurrent Engineering" Annals of III CIMIS.net Workshop. June, 1994.

[CHA85] Chang, T.C. Wysk, R.A. "An Introduction to Automated Process Planning Systems", 1985.

[CUN88] Cunningham, J.J. Dixon, J.R. "Design with Features: The Origin of the Features" ASME Computers in Engineering Conference. San Francisco August, 1988.

[DIX87a] Dixon, J.R. Cunningham, J. Simmons, M.K. "Research in Designing with Features", in Intelligent CAD, I, edited by H. Yoshikawa and D. Gossard, Proceeding IFIP TC 5/WG. Workshop on Intelligent CAD, Elsevier, 1987, pp.137-148

[DIX90] Dixon, J.R. Libardi, E.C. Nielsen, E.H. "Unresolved Research in Development of Design-with -Features Systems" Geometric Modelling for Product engineering, , Elsevier Science Publishers B.V, 1990.

[DUA93] Duan, W. "FSMT: a feature solid-modelling tool for feature-based design and manufacture" , January, 1993

[FER90] Ferreira, J.C.E. "A Contribution to Process Planning for Prismatic Components", Ph.D. Thesis, The University of Manchester, November 1990.

[HUR] Hurtwaite, B. "Design for Competitiveness" The Institute for Competitive Design.

[HWA92] Hwang, J.L. Henderson, M.R. "Applying the Perceptron to three-dimensional feature recognition" Journal of Design and Manufacturing, 1992

[JAR84] Jared, G.E.M. "Shared Features in Geometric Modelling", 1984

[LEE80] Lee, Y.T. Requicha, A.A.G. "Algorithms for Computing the Volume and Other Integrals Properties of Solids: I - Know Methods and Open Issues" Commun. ACM, Vol.25, No.9, september 1982.

[LIM94] Lima, Celson Pantoja "Modelamento Baseado em Features em um Conceito de Projeto para Fabricação e Montagem" Dissertação de Mestrado Universidade Federal de Santa Catarina setembro, 1994.

[LIN92] Lindberg, L. "The Product Structure - the Backbone of CIM" Annals of CIRP Vol.41/1/1992.

[LIN93] Lindberg, L. "Notes of Concurrent Engineering" Annals of CIRP Vol.42/1/1993

[MOR93a] Afsarmanesh, H. Wiedijk, M. Moreira, N.P. "Design of a Distributed Database for a Concurrent Engineering Environment" Annals of III CIMIS.net Workshop. june, 1994.

[MOR93b] Moreira, N.P. "Integração da Manufatura: Uma Proposta de Base" Dissertação de Mestrado Universidade Federal de Santa Catarina. setembro, 1993.

[MOR94] Moreira, N.P. Cabral, P.H.A.P. Filho, S.O. "An Approach to Concurrent Engineering Implementation in EMBRAER-EDE"

[PRA88] Pratt, M.J. "Synthesis of an Optimal Approach do Form Feature Modelling" ASME Computers in Engineering Conference, San Francisco, USA, July-August 1988

[REI92] Reimann, M.D. Huq, F. "A comparative Analisis Approach for Evaluating the Effect that Concurrent Engineering has on Product Life Cycle Cost" Flexible Automation and Information Management - FAIM. july, 1992.

[ROS] Rosa, E. et al. "CAE/CAD/CAM e Engenharia Simultânea" GRANTE/GRUCON. Universidade Federal de Santa Catarina.

[SAL92] Salomons, W. Houten, F.J.A.M. van Kals, H.J.J. "Review of Research in Feature-based Design" Journal of Manufacturing systems, Vol.12, No.2, 1992

[SHA88b] Shah, J. Sreevalsan P. Rogers, M. Billo, R. Mathew, A. "Current Status Of Features Thechnology" CAM-I Report R 88-GM-04.1. november, 1988.

[SHA88d] Shah, J. Rogers, M.T. "Expert Form Feature Modelling Shell" Computer Aided Design, Vol.20 No.9, 1988.

[SHA91] Shah, J. "Assesment of Feature Technology" Computer Aided Design Vol 23. Nro 5. june, 1991.

[SOH92] Sohlenius, G. "Concurrent Engineering" Annals of the CIRP Vol.41/2/1992.

[SUH90] Suh, N.P. "Principles of Design" Oxford University Press. 1990.

[TER92] Terpeny J.P. Deisenroth, M.P. "A Concurrent Engineering framework: three basic components" Flexible Automation and Information Management - FAIM. july, 1992.

[VAR90] Várady, T. Gaál, B. Jared, G.E.M. "Identifying Features in Solid Modelling", 1990

[WAN93] Wang, M.T. Chambrelain, M.A. Jonea, A. Chang, T.C. "Manufacturing Feature extraction and Machined volume Decomposition in a Computer-Integrated Feature-Based Design and Manufacturing Planning Environment" Computers in Industry, 1993

[WIN91] Wingard, L. "Introducing Form Features in Product Models, a Step Towards CAD/CAM with engineering Terminology", Licenciata Thesis, Department of Manufacturing Systems, Royal Institute of Technology, Stockholm, 1991