

# Um processo de converter Dados Geométricos a partir do formato DXF

Paulo César Monteiro  
Manuel João Próspero

Universidade Nova de Lisboa  
Faculdade de Ciências e Tecnologia  
Departamento de Informática  
P-2825 Monte de Caparica

## Sumário

Descreve-se a estrutura geral dum processo usado para converter dados geométricos contidos em ficheiro no formato DXF. Explica-se a conveniência da introdução dum formato intermédio (TDM), referindo as dificuldades que, no entanto, ainda têm que ser ultrapassadas. É nessa direcção que é descrito um método de edição interactiva, permitindo uma análise estrutural dos dados.

## 1. - Introdução

A Modelação Geométrica é considerada fundamental em áreas como as dos sistemas CAD/CAM e da animação por computador, entre outras. Muito embora existam hoje ferramentas auxiliares com novas concepções, a criação de dados geométricos é ainda um trabalho relativamente pesado. Isto implica que a troca desse tipo de dados entre diferentes aplicações seja bastante solicitada, quando não absolutamente necessária.

Sempre que duas ou mais aplicações pretendam trocar dados geométricos e não possuam formatos de entrada/saída comuns, então a forma de tratamento a usar é a conversão. Esta operação implica alterações aos modelos geométricos. Quanto mais elevado for o grau de abstracção dos modelos e das suas estruturas de dados, mais consistentemente se pode fazer as aplicações comunicar entre si [1].

O tema fulcral da comunicação assenta na escolha do formato DXF como ponto de partida dum processo de conversão de dados. Ele foi escolhido pelo facto de possuir uma grande capacidade para representar dados complexos e ser um dos formatos mais conhecidos e utilizados pelos actuais sistemas de modelação [2].



## 2. - Modelação Geométrica

### 2.1. - Representações

Para representar um sólido não basta reunir um conjunto de planos 3D e superfícies. Estes terão de estar organizados de modo a definir um volume fechado. Em muitas aplicações, contudo, é importante distinguir entre o interior, o exterior e a fronteira de um objecto 3D e de poder calcular propriedades que dependam desta distinção. Em CAD/CAM, por exemplo, se um objecto pode ser modelado de uma forma que capture adequadamente a sua geometria, então uma grande variedade de operações pode ser realizada antes de o objecto ser manufacturado [3]. Entre as várias operações possíveis temos: verificação da interferência entre diferentes objectos, e.g. colisão entre um objecto e outros, deslocando-se o primeiro numa dada trajectória; simulação de mecanismos físicos, tais como engrenagens, caso em que poderá ser importante o cálculo de propriedades como o volume e o centro de massa; interacção com a luz (refracção, reflexão, etc.). Todas estas aplicações são exemplos de modelação geométrica sólida.

A necessidade de modelar objectos como sólidos resultou no desenvolvimento de uma variedade de representações especializadas.

Ligada à representação de um objecto sólido está a noção de completude (*completeness*). Uma representação é completa se corresponde à representação de um e um só sólido. Por outro lado uma representação é única se pode ser usada para representar um dado sólido numa só forma. Se uma representação puder garantir unicidade, então são fáceis operações como a do teste de igualdade entre objectos.

O objectivo da modelação sólida é o de criar representações não ambíguas e completas dos objectos. Para isso têm-se criado várias abordagens diferentes, como sejam a representação pela fronteira B-rep (*Boundary representation*), a geometria sólida construtiva CSG (*Constructive Solid Geometry*), as colecções lógicas de semi-espaços, etc. [4].

No trabalho referido por esta comunicação utiliza-se a representação pela fronteira, com base no modelo topológico desenvolvido por Wu [5].

### 2.2 Estrutura de dados topológica

Acima de tudo, no caso dos modelos de representação pela fronteira existe uma subdivisão entre os modelos geométricos e os modelos topológicos, a qual tem a



vantagem de separar modelos analíticos e algébricos das operações. A parte geométrica é classificada nos modelos poligonais, paramétricos e analíticos. Para os modelos topológicos distinguimos entre os modelos de linhas (*wireframe*), os modelos de superfícies e os modelos de sólidos. Com a separação da informação topológica e geométrica torna-se mais fácil resolver problemas, especialmente na modelação sólida, tais como completude, integridade, complexidade e cobertura geométrica (*geometrical coverage*) [1].

Com base nos modelos topológicos, têm sido propostas várias estruturas de dados:

- a estrutura baseada em polígonos contém muito pouca informação. Cada polígono é descrito por si só. Tipicamente, é usada em sistemas gráficos para *rendering* rápido.

- a estrutura baseada em vértices, tem informação sobre os vértices e polígonos simplesmente ligados. Esta estrutura é um bom compromisso entre as estruturas baseadas em polígonos e as baseadas em arestas. Na área da animação e visualização, contudo, não é possível descrever modelos de formas livres.

- a estrutura baseada em arestas pode conter informação acerca de várias faces interligadas para qualquer modelo geométrico. A mais conhecida é a *winged-edge* [6]. Com o desenvolvimento da estrutura de dados *radial-edge* foi feito um primeiro passo para lidar com modelos híbridos (*non-manifolds*).

- finalmente, com a estrutura de dados *double-winged* desenvolvida por Wu [5], foi formalizada a primeira abordagem para o tratamento completo dos modelos híbridos.

### 3. Formato DXF vs. formato TDM

#### 3.1 Formato DXF

Um ficheiro DXF (*Data eXchange Format*) é um ficheiro de texto ASCII especialmente formatado (com a extensão *.dxf* que caracteriza o tipo de ficheiro). Está dividido em quatro secções:

- secção de cabeçalho (*header section*);
- secção de tabelas (*tables section*);
- secção de blocos (*blocks section*);
- secção de entidades (*entities section*).



As duas primeiras secções contêm informação pouco relevante para a conversão, tal como variáveis do ambiente da aplicação, constantes, tabelas de *layers*, registo de nomes de aplicações, etc. As duas secções seguintes, contudo, guardam toda a informação geométrica dos objectos.

A secção de blocos contém a definição dos blocos do desenho. Um bloco é um conjunto de entidades ao qual foi dado um nome e que pode ser manipulado como um todo. Convém salientar aqui a importância que estas estruturas têm para diminuir a complexidade do desenho e facilitar a reutilização de objectos previamente definidos. Um bloco pode referir outros, indirectamente, através do uso de uma entidade especial chamada *INSERT*. Poder-se-á construir, deste modo, uma hierarquia de blocos (ver figura 1).

A secção de entidades é destinada à declaração e registo de todas as entidades que compõem o desenho. Uma aplicação poderá definir qualquer tipo de entidade, desde que obedeça à sintaxe DXF. Em [7] e [8] define-se exactamente um conjunto de entidades que são a base de trabalho para a conversão DXF. Entre estas temos: *insert*, *arc*, *line*, *polyline*, *trace*, *point*, *solid*, *vertex*.

Finalmente, existe em DXF (AutoCAD) um conceito muito importante que é o conceito de *layer*, podendo este ser visto como uma camada. Um desenho é então constituído por um ou mais *layers*, cada *layer* contendo determinadas entidades e/ou blocos (ver figura 1). Um bloco ou entidade pertence sempre a um *layer*, podendo um bloco pertencer a um *layer* e as suas entidades pertencerem a *layers* diferentes.

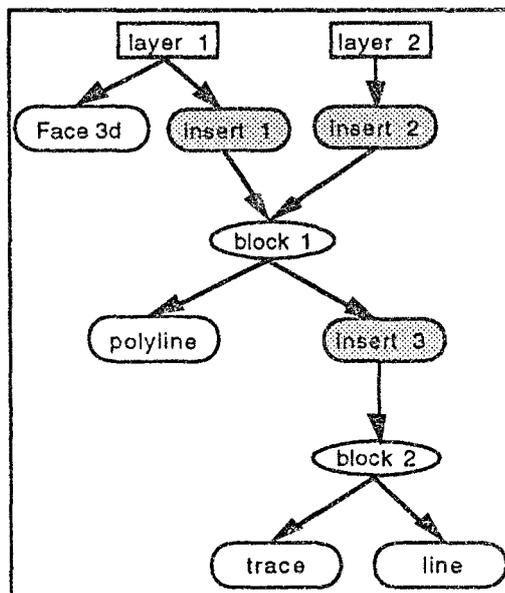


Figura 1  
Grafo DXF típico.

### 3.2. - Formato TDM

A conversão do formato DXF não será feita directamente para o formato de saída pretendido, mas sim através do formato intermédio TDM (*Topological Data Module*).

Existem treze tipos de entidades em TDM, com a possibilidade de se disporem hierarquicamente, e todas as funções se agrupam em quatro classes. Na figura 2 podemos observar algumas dessas entidades: *assembly*, *object*, *component*, *face*, *chain*, *loop*, *cycle*, *edge* e *vertex*. A forma como estas diversas entidades se relacionam foi definida formalmente em [5].

Interessa reter, sobretudo, que o modelo TDM oferece, não só as estruturas de dados, como as funções para operar sobre as mesmas. Das quatro classes de funções existem duas muito importantes: as funções de inquirição e as funções de construção. Na figura 2 pode-se também observar como se relacionam as estruturas TDM e DXF.

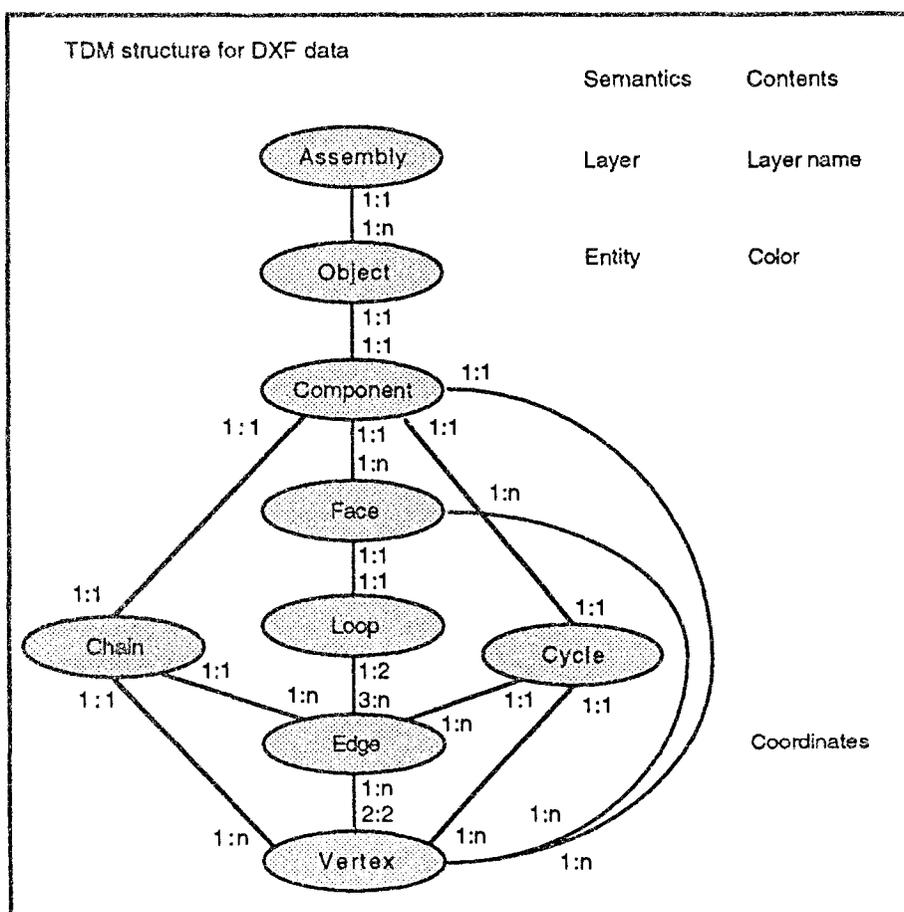


Figura 2  
Relações entre as estruturas TDM e DXF.

#### 4. - Processo de Conversão

O processo de conversão é feito em vários passos. Um destes passos é a conversão para um formato intermédio, possibilitando assim a conversão para vários formatos finais [9]. Pretende-se, deste modo, obter generalidade no processo de conversão. Para cumprir este objectivo é necessário que o formato intermédio seja suficientemente abstracto, ainda que possibilitando guardar informação própria de cada formato. No caso dos objectos sólidos, o uso de um modelo topológico de dados parece apropriado, uma vez que:

- toma em consideração apenas a fronteira, e, portanto, todos os modelos podem ser convertidos;
- tem a vantagem de separar os modelos analíticos e algébricos das operações;
- o uso do TDM, em particular, permite o tratamento completo dos modelos híbridos [1].

##### 4.1. - Problemas inerentes à conversão

A conversão de um formato para outro pode, por vezes, ser bastante complexa e exigir, da parte do conversor, grande manipulação geométrica. Quando a diferença entre os modelos é quase toda sintáctica, então a conversão é relativamente pacífica, resumindo-se esta a copiar dados de um formato para outro. Os problemas surgem quando os formatos têm por detrás modelos bastante diferentes.

A conversão exige, com frequência, aproximações e/ou transformações geométricas sobre os objectos. Por exemplo, a conversão de um círculo definido analiticamente para um modelo poligonal exige a aproximação do círculo a um polígono. Outros problemas surgem ligados à precisão da representação. Se não for suficientemente precisa, pode levar a distorções na geometria do objecto, devido às manipulações geométricas. Existem ainda casos em que não é possível converter determinadas características de um modelo. Serve como exemplo a informação numa árvore CSG, que se pode perder na conversão para um modelo que não suporte CSG.

Outro problema que um conversor tem de enfrentar são os erros sintácticos ou semânticos dos formatos originais. Este problema é relativamente frequente e pode ser abordado de diferentes maneiras. O modo mais simples é abortar a conversão assim que se detecta um erro. Esta abordagem oferece pouco conforto ao utilizador, especialmente se o erro é pequeno e não compromete a restante conversão. Por outro lado, pode-se alertar para o erro e prosseguir a conversão, se tal for viável. Seria um pouco melhor, mas ainda assim não muito satisfatório. Melhor seria, de facto, que

além de alertar para os erros, o conversor propusesse uma correcção por omissão ou que permitisse a correcção interactiva com o utilizador.

#### 4.2. - Conversão interactiva

Entende-se por conversão interactiva a conversão que fornece e recebe dados do utilizador durante o processo, de modo a conduzi-lo de uma maneira ou de outra. O objectivo principal deste método é obter maior flexibilidade na conversão.

Ligada à conversão interactiva está a possibilidade de analisar estruturalmente os dados a converter, permitindo a visualização e manipulação dos mesmos [10]. Esta análise é feita recorrendo a ferramentas apropriadas (por exemplo: um editor de grafos, conforme se refere na secção 4.4.). O resultado da análise pode ser registado e servir de entrada para uma fase posterior da conversão.

#### 4.3. - Conversão do formato DXF

O processo de conversão, na sua globalidade, pode ser dividido em quatro fases principais:

- fase 1) Análise prévia dos dados DXF;
- fase 2) Conversão para um formato intermédio;
- fase 3) Análise posterior;
- fase 4) Conversão para o formato final.

As fases indicadas apresentam-se esquematizadas na figura 3. É de salientar que não existem restrições quanto aos formatos de saída, desde que modelem sólidos e sejam *boundary oriented*.

Após a análise sintáctica obtém-se um grafo DXF estrutural. Na figura 1 mostrou-se um grafo típico, relacionando os elementos ELB. Por ELB pretende-se designar os diferentes tipos de elementos DXF: entidades, camadas (*layers*) e blocos. Note-se, no entanto, que existem outras relações entre os diferentes elementos e que não são aqui mostradas, nomeadamente as relações entidade-entidade, bloco-*layer* e algumas entidade-*layer*.

Nesta altura é possível a utilização de um editor de grafos interactivo, permitindo uma análise completa da estrutura apresentada, assim como a especificação de critérios para a conversão:

- cc1) conversão por camadas;
- cc2) conversão por blocos;



cc3) conversão mista, por camadas e por blocos;

cc4) conversão por um dos critérios anteriores, mas até certo nível.

Com a figura 4 mostra-se um exemplo de grafo TDM que resulta da conversão do grafo DXF da figura 1 pela escolha do critério cc1), enquanto que a figura 5 se obtém a partir do mesmo grafo inicial mas pela utilização dos critérios cc3) e cc4). De notar que, neste último caso, a informação da hierarquia de blocos se perdeu pelo facto da conversão por blocos ter sido interrompida a um determinado nível.

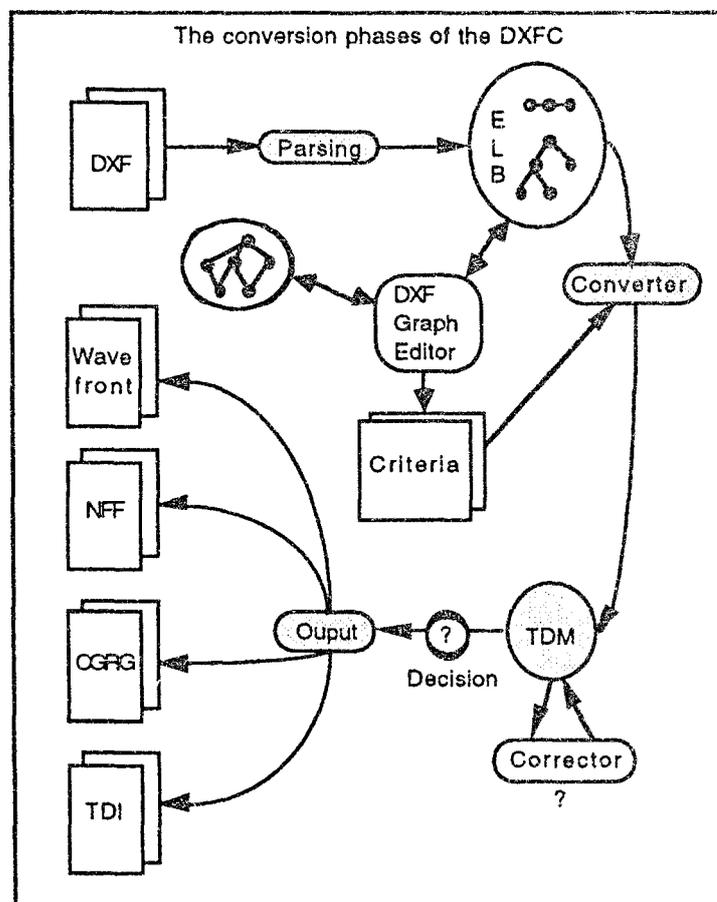


Figure 3  
As diferentes fases de conversão do DXF.

#### 4.4. - Edição de grafos

O editor de grafos desempenha, aqui, um papel fundamental na conversão interactiva. É através do editor que o utilizador tem a oportunidade de observar, modificar ou seleccionar elementos do grafo DXF. É também aqui que se podem estabelecer as regras de conversão, ou seja, o que é que vai ser convertido e como o

será. É neste sentido que o editor de grafos pretende adicionar flexibilidade à conversão.

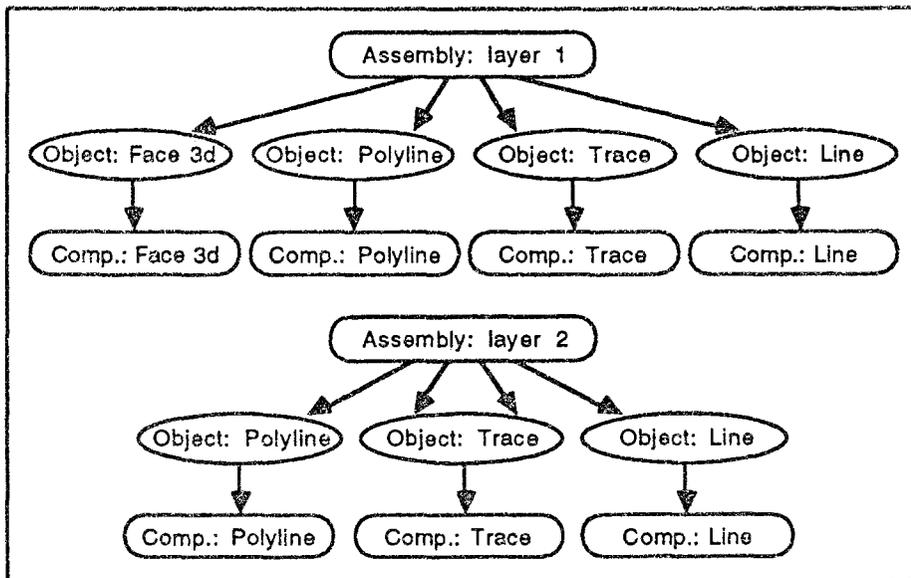


Figura 4

Grafo TDM resultante da conversão do grafo da figura 1 utilizando o critério cc1).

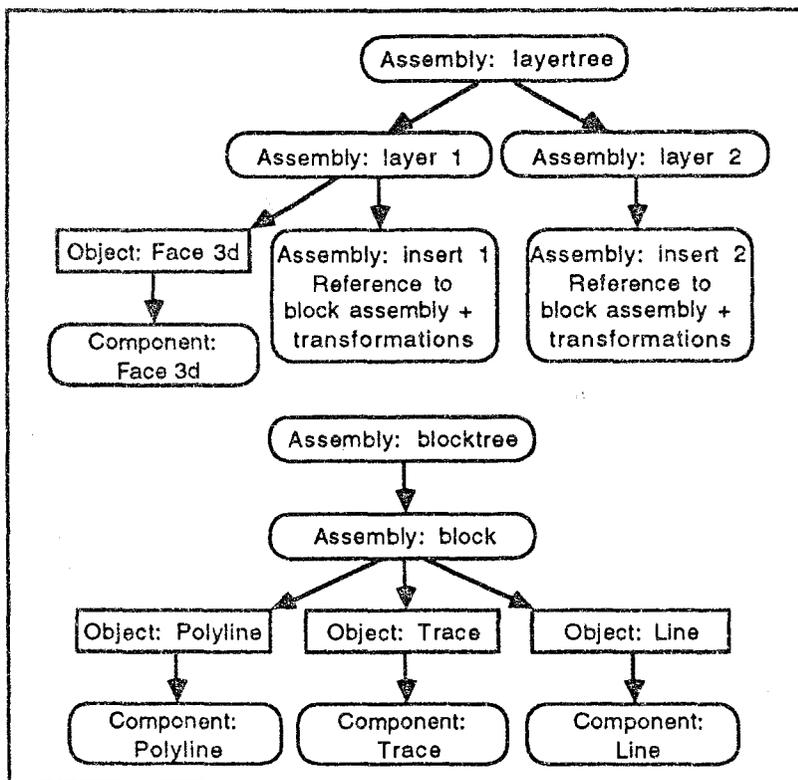


Figura 5

Grafo TDM resultante da conversão do grafo da figura 1 usando os critérios cc3) e cc4).

#### 4.4.1. - Modelo do grafo DXF

Mostrar muitas centenas de objectos (frequentemente milhares), as relações entre eles e a maior quantidade possível de informação sobre cada um, por modo fácil de apreender e manipular pelo ser humano, não é tarefa fácil. Existem vários factores a considerar:

- o meio de representação;
- a escolha da informação relevante para apresentar em cada momento, de forma a não sobrecarregar o meio de representação;
- métodos de compactação da informação;
- modos de interacção com o utilizador;
- mecanismos para protecção e manutenção da coerência da informação.

Estes são alguns dos desafios que um sistema de representação de informação enfrenta. Passa-se exactamente o mesmo com o editor de grafos, excepto que aqui o meio de representação já foi seleccionado: o grafo. De facto, o grafo parece ser o meio ideal para modelar a estrutura DXF, com os seus vários elementos e os vários tipos de relações.

O grafo DXF é um grafo conexo e dirigido, com uma única raiz. Todos os nós do grafo partilham as mesmas propriedades gerais, existindo no entanto cinco tipos de nós: DXF, camada, bloco, entidade e colectivo. Os nós representam assim os principais elementos do formato DXF. São excepção o nó DXF, que representa o próprio DXF, e o nó colectivo, que representa um conjunto de outros quaisquer nós. Existem três tipos de relações:

- relação de contenção. Um objecto (nó) contém outro objecto (nó). Por exemplo, um nó camada contém muitos nós entidade.
- relação de parentesco. Um objecto é pai de outro objecto. Esta relação usa-se sobretudo para representar as árvores CSG.
- relação de colectividade. É uma relação que representa um conjunto de quaisquer outras relações. Usa-se entre um qualquer nó e um nó colectivo.

Note-se que todos estes tipos de relações obedecem a restrições quanto aos tipos de nós que lhes estão ligados.

#### **4.4.2. - Apresentação e compactação da informação**

Como já foi referido, a informação estrutural do DXF é apresentada sob a forma de um grafo. Há, no entanto, dois pontos importantes a salientar: em primeiro lugar, só a informação mínima é apresentada (como na figura 1), devendo o utilizador seleccionar os objectos para os quais deseje informação detalhada; em segundo lugar, e sempre que possível, o grafo é compactado, i.e. todos os nós que podem ser agrupados por uma propriedade comum vão constituir um nó colectivo, passando-se o mesmo para os respectivos arcos. Existe, contudo, a possibilidade de compactar ou descompactar apenas partes específicas do grafo através de selecção.

O modo mais intuitivo de interagir com o utilizador é através de uma interface gráfica, onde ele possa seleccionar, por simples apontar, os comandos e os objectos desejados. Mas, de acordo com os princípios já enunciados, tem de haver um compromisso entre a quantidade de informação apresentada e a quantidade de informação capaz de ser abarcada pelo utilizador.

#### **4.5. - Protecção da coerência da informação**

Após a interacção com o utilizador, ficam registadas as suas opções de conversão sob a forma de regras. Estas regras vão, por conseguinte, ditar o comportamento do conversor na sua fase seguinte (fase 2).

As operações críticas são as de apagar objectos, criar, copiar, mover ou ainda outras, específicas à estrutura DXF. Certo tipo de operações, como a de eliminar uma relação provocando a desconexão do grafo, não são permitidas.

Depois da fase de conversão para o formato intermédio (TDM), que é a mais susceptível de criar erros, podem surgir, por exemplo: ciclos abertos, orientação incorrecta das normais das faces, problemas de intersecção, erros de construção, imprecisão nas transformações, etc. É neste contexto que a fase 3, por estar integrada junto ao processo de conversão, se torna importante.

Erros daquele tipo podem ser complicados de corrigir se de forma tradicional. Suponhamos, por exemplo, que se queria transformar um desenho DXF traduzindo-o directamente em OFF (antigo CGRG). Havendo um erro na conversão, e não sendo possível corrigi-lo em OFF, ter-se-ia que converter o desenho de OFF para TDI, efectuar as correcções e converter novamente para OFF [10]. Ora as facilidades associadas à referida fase 3, colocadas à disposição do utilizador, evitam a complicação dessas soluções cruzadas e criam um ambiente de trabalho de conversão em que a eficiência é muito maior.



## 5. - Conclusão

O problema da representação, análise e conversão de dados geométricos é ainda um tema activo de investigação. Especialmente no que diz respeito à animação e à realidade virtual, as exigências na eficiência, capacidade e qualidade dos modelos são muito grandes.

No que diz respeito à conversão, nesta comunicação foram focados dois pontos importantes: em primeiro lugar, a tentativa de generalização para os modelos de representação pela fronteira, utilizando o formato intermédio TDM, e, em segundo lugar, a tentativa de aumentar a flexibilidade na pré-análise e nos próprios processos de conversão, recorrendo a métodos interactivos. A própria fase de conversão para o formato final, além de parametrizável, também pode ser interactiva.

Todo o processo de conversão referido ao longo das secções anteriores tem vindo a ser desenvolvido no Instituto Fraunhofer, Universidade Técnica de Darmstadt. Para o estado actual do trabalho, que ainda não está completamente realizado, contribuíram, nos últimos dois anos, P. Pires [9] e P.C. Monteiro [10]. Como alunos finalistas da licenciatura em Engenharia Informática da UNL, colaboraram no âmbito dos programas ERASMUS e COMETT, sob a orientação directa de Fritz Loseries.

## 6. - Bibliografia

- [1] José L. Encarnação: "Advanced R & D Topics on Animation and Scientific Visualization", internal report — THD, 1993.
- [2] Paulo Pires, Shin-Ting Wu, "A Converter of DXF-Format to Any Boundary-Oriented Data Format", *Proc. of the Compugraphics'92*, pp. 261-269, Lisbon, December 1992.
- [3] J. Foley, A. van Dam, S. Feiner and J. Hughes, *Computer Graphics — Principles and Practice*, Addison Wesley, 1990.
- [4] Michael E. Mortenson, *Geometric Modeling*, John Wiley & Sons, 1985.
- [5] S.T.Wu, "Topologie von Hybriden Objekten", *Phd Dissertation*, THD, Darmstadt, 1991.
- [6] Mäntylä, M., *Introduction to Solid Modeling*, Computer Science Press, Rockville, Maryland, 1988.
- [7] "AutoCAD 10.0 - Reference Manual", AutoDesk Inc., 1988.
- [8] "Autodesk Inc.: AutoCAD Release 12: Referenzhandbuch", 1992.
- [9] Paulo A.L.R.Pires, "A Converter of DXF Format to Any Boundary-Oriented Data Format", Computer Graphics Project — Final Report, FhG-IGD, August 1992.
- [10] P.César Monteiro, "Concepts on generic and adaptative DXF-Conversion for Visualization and Animation", Computer Graphics Project — Final Report, FhG-IGD, August 1993.

