

J.R.S. Blaschek

Programa de Engenharia de Sistemas e Computação

COPPE/UFRJ

Caixa Postal 68511

CEP 21945 - Rio de Janeiro - RJ

## RESUMO

A Análise e Projeto Estruturado são métodos bastante conhecidos e amplamente utilizados. A maior desvantagem apontada pelos seus utilizadores é a ausência de ferramentas automatizadas. Este trabalho apresenta um Dicionário de Dados Automatizado para apoio à Análise Estruturada, pertencente a um conjunto mais amplo de ferramentas em desenvolvimento na COPPE/UFRJ.

### 1 - Introdução

O desenvolvimento de um software é um processo extenso e complexo. Reveste-se de problemas ligados à natureza humana, à evolução rápida da ciência da computação e à dificuldade de elaboração de um modelo claro e concreto do que se vai produzir.

Produtos de software produzidos nesse ambiente de complexidade, sem um tratamento adequado, costumam apresentar resultados catastróficos. Comumente excedem orçamentos e cronogramas, não atendem aos anseios dos usuários e possuem um elevado custo de manutenção.

Um ambiente adequado ao desenvolvimento de software é composto de (1):

- um ciclo de vida, que define as fases do processo de desenvolvimento e atividades a serem realizadas em cada fase;
- métodos, usados para organizar o pensamento e o trabalho do usuário ao longo do processo de desenvolvimento; e
- ferramentas, automatizadas ou não, que tornam possível a utilização dos métodos.

Diversos métodos foram criados para permitir um desenvolvimento adequado das fases do ciclo de vida. Os mais bem sucedidos e utilizados, fizeram uso de técnicas estruturadas (Análise e Projeto Estruturado), os quais permitiram a produção de sistemas de boa qualidade (2,3). No entanto, a utilização desses métodos, com as ferramentas manuais propostas, apresentam os seguintes problemas (4):

- as alterações dos documentos produzidos são muito trabalhosas;
- permite que as regras de formação dos documentos não sejam obedecidas;
- a recuperação de informações é trabalhosa;
- a manutenção da consistência dos dados de todos os documentos é trabalhosa, sendo comum a existência de redundâncias; e
- não permite a auto-validação e a capacidade de auto-verificação é pequena.

Para resolver esses problemas, a COPPE/UFRJ vem desenvolvendo um conjunto de ferramentas automatizadas para apoiar a Análise e Projeto Estruturado, além de ferramentas destinadas ao apoio dos processos de avaliação, planejamento e controle das diversas fases do ciclo de vida. O conjunto dessas ferramentas compõe um ambiente de apoio ao desenvolvimento de software apresentado pela figura 1 (5).

Este trabalho apresenta, de uma forma mais detalhada, as características do Editor de Dicionário de Dados.

## 2. Editor de Dicionário de Dados

### 2.1 - Introdução

Qualquer sistema de computação possui como objetivo, pelo menos alguns dos itens abaixo (6):

- ser um sistema confiável com elevado fator de utilidade;
- capacidade de compartilhar e transferir dados entre bancos de dados, sistemas, subsistemas e aplicações;
- independência de dados;
- eliminar redundâncias não planejadas;
- facilidade de verificar que programas usam determinado dado;
- um método eficiente de catalogação;
- um método de documentação automática e de fácil manutenção;
- uma técnica que facilita atualizações, reestruturacões, manutenção e mudança de controle; e
- um eficiente esquema de consulta que forneça informações chaves aos usuários, preferencialmente em tempo real.

Para que esses objetivos sejam atingidos, há a necessidade de existir uma ferramenta, capaz de armazenar e processar informações sobre todos os dados e funções do sistema. Também há

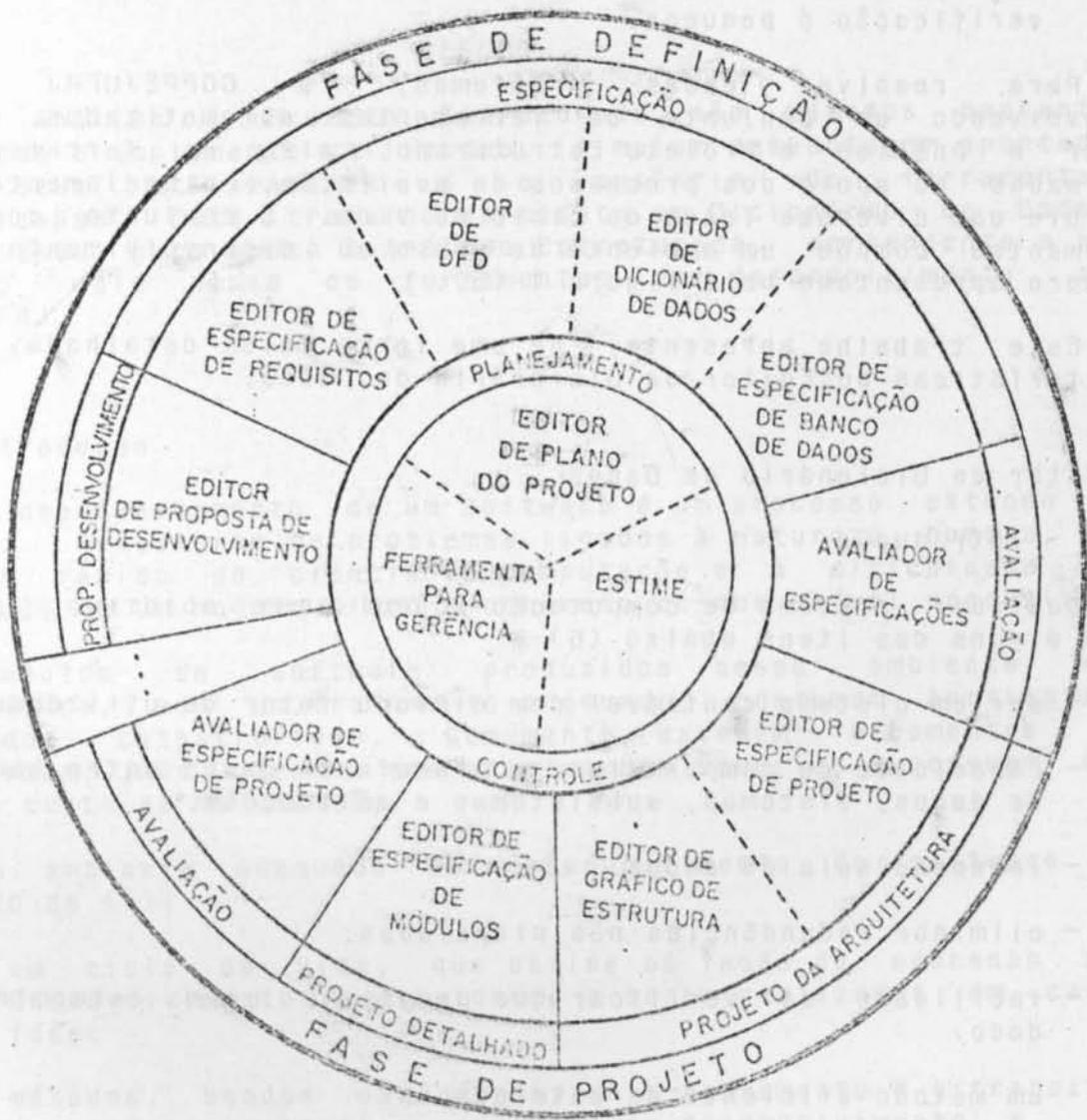


FIGURA 1 - Ambiente de Apoio ao Desenvolvimento de Software

a necessidade de uma ferramenta capaz de auxiliar o desenvolvimento e manutenção de sistemas, assim como no uso e controle de todos os seus dados. Essa ferramenta é um Dicionário de Dados.

Os primeiros dicionários de dados surgiram em meados dos anos 70, tendo como objetivo básico a gerência de informações sobre bancos de dados. Logo a seguir, passaram a fornecer descrições de dados, já compiladas, para inclusão automática em programas de aplicação.

Atualmente, os dicionários de dados são utilizados nos mais diversos tipos de aplicações, sendo parte obrigatória do desenvolvimento e manutenção de qualquer sistema de computação. Os novos sistemas de dicionário de dados a serem desenvolvidos, certamente envidarão esforços para suportar todo o ciclo de vida do software.

É importante ressaltar que um dicionário de dados contém apenas metadados. Para um dado elemento de dados, por exemplo, o dicionário de dados conterá os atributos deste elemento e informações sobre seu uso por sistemas de processamento e usuários. Assim sendo, o dicionário de dados não armazena nenhuma instância do elemento de dados, as quais estarão armazenadas em um banco de dados ou em arquivos convencionais.

O acesso a um dicionário de dados pode ser feito tanto em batch como on-line. Em geral, os relatórios periódicos e longos são gerados em batch, enquanto que as transações menores são executadas on-line.

Como um dicionário de dados é um banco de dados, (na verdade, é um banco de metadados, ou seja, de dados que descrevem dados), o seu desenvolvimento segue as etapas normais do desenvolvimento de um banco de dados qualquer, podendo, portanto, utilizar qualquer método destinado a esse fim.

## 2.2 - Características do Editor de Dicionário de Dados

O sistema já desenvolvido e implementado é uma ferramenta destinada a auxiliar analistas de sistemas, na tarefa de construção do dicionário de dados da análise, em ambientes de microcomputadores de 16 bits.

Embora podendo ser utilizado por diversos métodos, sua utilização foi basicamente voltada para o método "Análise Estruturada de Sistemas", proposta por Gane e Sarson. Seu desenvolvimento foi voltado para os seguintes objetivos:

- possibilitar que as alterações na documentação produzida sejam realizadas mais facilmente;
- facilitar o armazenamento de todas as informações;
- possibilitar o fácil acesso a informações mais complexas;
- ter certeza de que regras pré-definidas de formação dos documentos estão sendo obedecidas;
- possibilitar a auto-verificação da consistência e

### Integridade dos dados:

- permitir que as informações possam ser obtidas mais rapidamente;
- permitir que as informações obtidas sejam as mais atualizadas;
- permitir que um grande número de usuários tenha acesso ao sistema, e,
- ser um sistema portátil e de fácil manutenção.

Uma visão genérica do sistema, a nível lógico, é apresentada pelas figuras 2 e 3.

O processo "Descrever Entradas" permite que o usuário descreva e armazene no dicionário de dados elementos de dados, estruturas de dados, depósitos de dados, processos, fluxos de dados, entidades externas e itens de glossário. Para cada entidade citada, existe uma tela específica, onde o usuário fornece, além dos atributos básicos de cada entidade, os relacionamentos existentes. O sistema utiliza esses relacionamentos para criar, caso não exista, a ocorrência da entidade relacionada. A figura 4 apresenta um modelo de tela, utilizada para descrever estruturas de dados.

O processo "Atualizar Entradas" atualiza ou deleta entidades criadas pelo processo anterior. Caso uma ocorrência de uma entidade seja deletada, todos os relacionamentos desta ocorrência são automaticamente deletados.

O processo "Gerar Relatório de Entradas" gera seis relatórios diferentes:

- relatórios completos contendo todas as informações armazenadas sobre: uma determinada ocorrência de uma entidade; todas as ocorrências de uma determinada entidade; ou todas as ocorrências de todas as entidades;
- relatórios resumidos contendo apenas o código (quando aplicável), nome e descrição de todas as ocorrências de uma determinada entidade ou de todas as ocorrências de todas as entidades;
- relatório contendo apenas o código (quando aplicável), nome e descrição de todas as ocorrências de entidades indicadas, pelo usuário, como parcialmente descritas nos processos de descrição ou atualização. Indica as ocorrências que ainda não foram completamente descritas.

O processo "Recuperar Referências Cruzadas" recupera as informações armazenadas sobre as ocorrências de entidades relacionadas com uma dada ocorrência de uma entidade, fornecida pelo usuário.

O processo "Verificar Consistência" verifica a consistência e integridade de todo o dicionário de dados, fornecendo relatórios para análise pelo usuário.

O processo "Recuperar Entrada" recupera uma ocorrência de

Figura 2 - DFD Escrever Dados (nível 2)

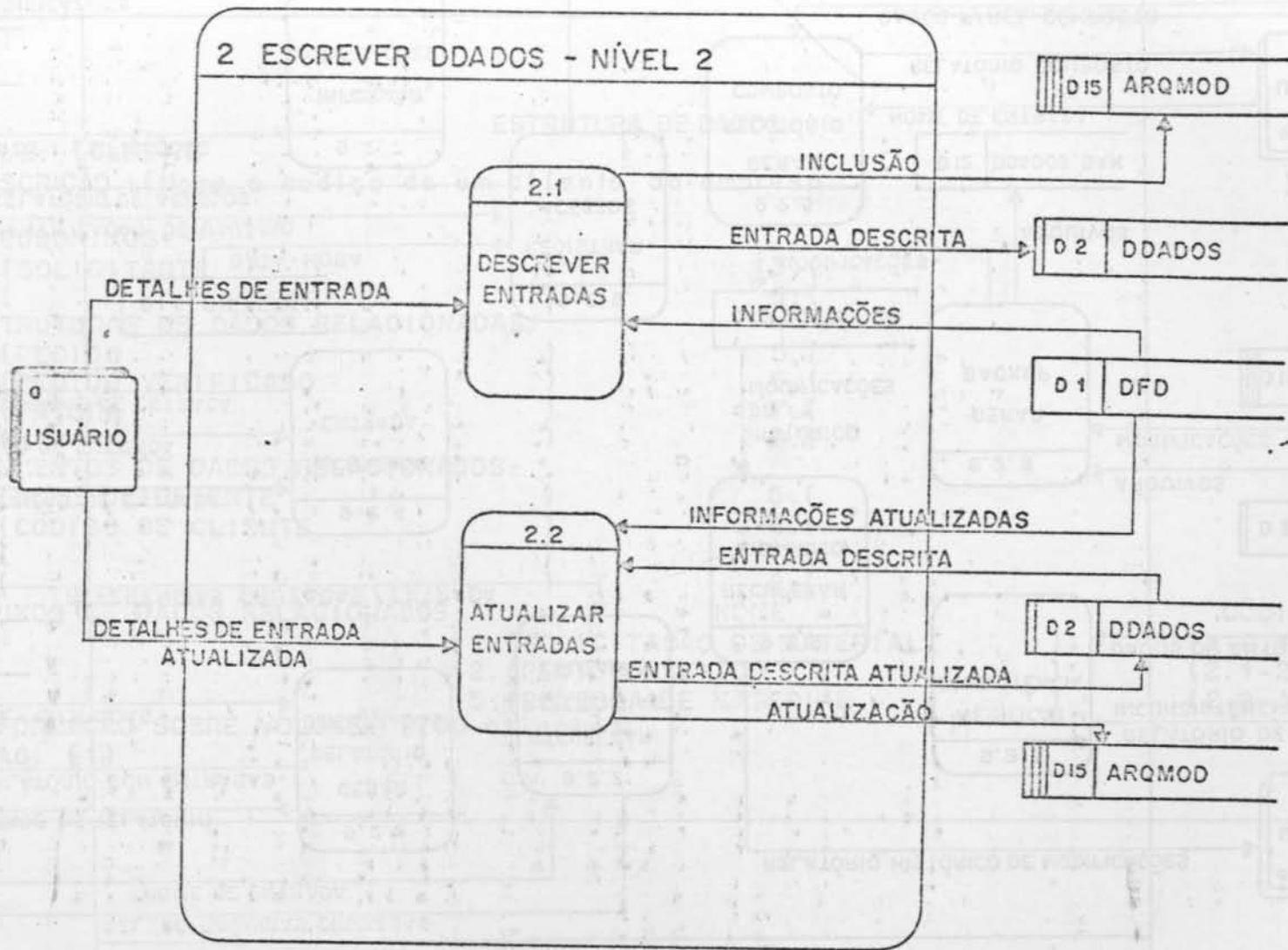


Figura 3 - OFD Emitir Dados (nível 2)

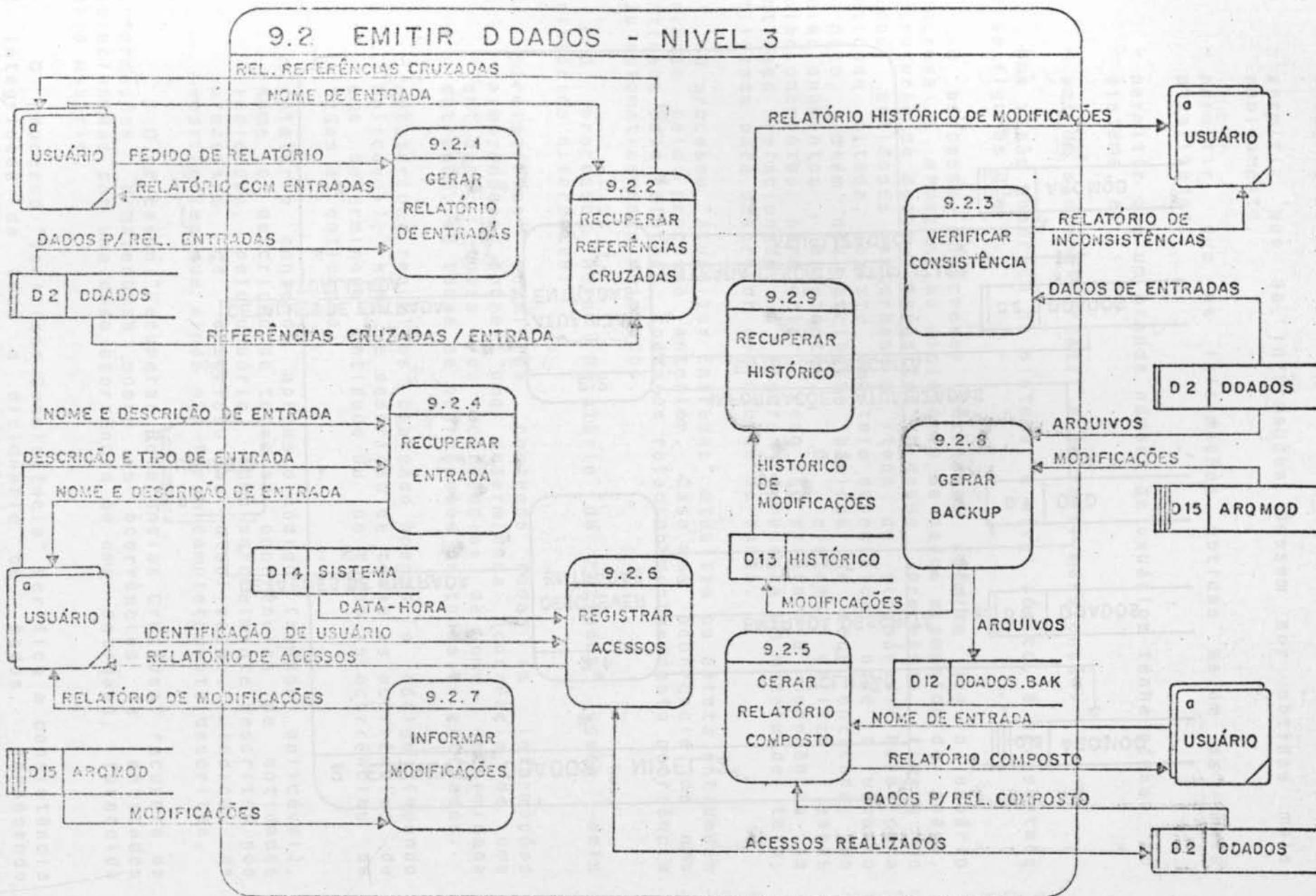


Figura 4 - Exemplo de Tela do Editor de Dicionário de Dados

ESTRUTURA DE DADOS			
NOME:	(CLIENTE	)	
DESCRIÇÃO:	(Nome e código de um cliente da empresa.	)	
PSEUDÔNIMOS:			
1.	(SOLICITANTE	)	3.(
2.	(	)	4.(
ESTRUTURAS DE DADOS RELACIONADAS:			
1.	(PEDIDO	)	5.(
2.	(PEDIDO VERIFICADO	)	6.(
3.	(FATURA	)	7.(
4.	(	)	8.(
ELEMENTOS DE DADOS RELACIONADOS:			
1.	(NOME DE CLIENTE	)	5.(
2.	(CÓDIGO DE CLIENTE	)	6.(
3.	(	)	7.(
4.	(	)	8.(
FLUXOS DE DADOS RELACIONADOS:		NOME	CÓDIGO
	1.	(SOLICITAÇÃO DE MATERIAL	) (A-2.1
	2.	(PEDIDO ACEITO	) (2.1-2.2
	3.	(REMESSA DE MATERIAL	) (2.2-A
INFORMAÇÃO SOBRE VOLUME: (700 clientes )			
FLAG: (1)			

uma determinada entidade, a partir de descrições fornecidas pelo usuário. É utilizado quando o usuário não tiver a certeza de já ter descrito a ocorrência.

O processo "Gerar Relatório Composto" fornece informações sobre ocorrências relacionadas com ocorrências, relacionadas com uma determinada ocorrência fornecida pelo usuário.

Os demais processos executam atividades que podem ser entendidas pelo seu próprio nome, sendo desnecessário maiores explicações.

Para armazenamento dos dados, foi projetado um banco de dados relacional. A estrutura do banco permite uma completa navegação por todos os seus arquivos, possibilitando a recuperação de diversos níveis de relacionamentos e verificações de consistência e integridade com relativa simplicidade. Os arquivos foram indexados por diversos campos de modo a agilizar as pesquisas realizadas.

Todas as opções de processamento são selecionadas pelo usuário através de diversos menus de seleção.

### 3 - Conclusões

Termos como CASA (Computer-Aided Systems Analysis) e CAP (Computer-Aided Programming) passaram recentemente a despertar o interesse de analistas e programadores, preocupados em melhorar o desempenho de suas funções.

A partir dessa tendência, pretendeu-se desenvolver e implementar o protótipo de um sistema de dicionário de dados automatizado (7). O sistema tem apresentado bons resultados nas avaliações a que vem sendo submetido, constituindo-se em uma boa fonte de pesquisas. Alguns aperfeiçoamentos foram sugeridos, devendo, em breve, serem implementados.

### BIBLIOGRAFIA

- (1) Rocha, A.R.C. da; Aguiar, T.G.; Blaschek, J.R.S.; Ambientes para Desenvolvimento de Software: Definição de Termos; Relatório Técnico; COPPE/UFRJ; 1987.
- (2) Gane, G.; Sarson, T.; Structure Systems Analysis: Tools and Techniques; Prentice-Hall; 1979
- (3) Yourdon, E.; Constantine, L.; Structured Design; Prentice Hall; 1979
- (4) Aguiar, T.G.; Ferramentas de Apoio à Análise Estruturada; Tese de Mestrado; IME; Fevereiro de 1986.
- (5) Rocha, A.R.C. da; Aguiar, T.G.; Blaschek, J.R.S.; Nogueira, D.; Ferramentas Automatizadas para Apoiar as Fases de Análise e Projeto; XX Congresso Nacional de Informática; SUCESU; São Paulo; 1987

(6) Duyn, J.V.: Developing a Data Dictionary System: Prentice-Hall; 1982

(7) Blaschek, J.R.S.: Dicionário de Dados Automatizado para Análise Estruturada; Tese de Mestrado; IME; fevereiro de 1987.