

UNIVERSIDADE ESTADUAL DE CAMPINAS
FACULDADE DE ENGENHARIA DE CAMPINAS
DEPARTAMENTO DE ENGENHARIA ELETRICA

Este exemplar corresponde a reclusão final
da tese defendida por Wu Shin-Ting e aprovada
pela comissão julgadora em 26/9/1984.

L. P. Magalhães

TEMA: UM MODELO DE DADOS PARA
APLICAÇÃO EM CONTROLE DE PROCESSO

por: Eng.^a WU SHIN - TING

095/84

Orientador: Prof. Dr. Léo Pini Magalhães

Tese de mestrado apresentada à Faculdade de
Engenharia de Campinas da Universidade
Estadual de Campinas

SETEMBRO DE 1984

UNICAMP

Este trabalho foi editado utilizando o sistema
de edição de texto em português
do Laboratório de Sistema/Micro, DEE, FEC, UNICAMP.

Este trabalho contou com o apoio financeiro do
Conselho de Desenvolvimento Científico e Tecnológico
- CNPq -

Dedico à
minha família

AGRADECIMENTOS

A minha mãe, sem a qual dificilmente este trabalho seria realizado.

Ao Professor Dr. Léo Pini Magalhães, pela confiança, incentivo e orientação.

A Professora Gestriz Mascia Dall'ini e ao colega Aguinaldo, pelas sugestões nas atividades computacionais.

Ao Professor Dr. Mário Jino, pelas sugestões e discussões na proposta do modelo MERB.

Ao Professor Dr. Wagner Amaral, pelas sugestões no modelamento do Sistema Metroviário.

Ao Professor José Raimundo de Oliveira, pela cooperação na utilização do sistema de edição de texto.

Aos colegas Ivan, Fátima e Nádia, pela colaboração na implementação do núcleo gerenciador de dados - CORAS-UNICAMP.

Aos amigos Abdo, Célia, Cícero, Helga, Ivonete e Jaime, pela revisão da redação do texto da tese.

Aos funcionários da CPG, pelos serviços administrativos.

Ao CNPq, pelo apoio financeiro.

E a todos aqueles que participaram, direta ou indiretamente, na elaboração deste trabalho.

RESUMO

Neste trabalho é proposto um Modelo de Dados para aplicação em Controle de Processos, que permite um modelamento preciso da informação ligada a Processos; sua motivação deve-se à forte tendência atual de inclusão de Sistemas de Banco de Dados para o tratamento em tempo real do grande volume de informação existente nos Sistemas Automatizados de Controle.

O Modelo Entidade-Relacionamento Binário (MERB) é uma extensão adaptada do Modelo Entidade-Relacionamento (MER); leva em consideração o comportamento dinâmico do processo físico modelado e o tempo de manipulação dos dados, no decorrer da atuação dos controladores.

No MERB são introduzidos os conceitos de MÓDULO, CHAVE SECUNDARIA e EVENTO. Além disso, é redefinida a classificação dos conjuntos de entidades e relacionamentos, adequando-os à formalização dos processos físicos, dentro do enfoque de Controle de Processo.

Como exemplo da aplicabilidade do modelo proposto, modelou-se o Sistema Metroviário Paulistano, sob o ponto de vista do supervisor, utilizando os primitivos do MERB.

Incluem-se, ainda, dois apêndices. No primeiro, é descrita sucintamente a filosofia da implementação de um núcleo gerenciador de dados (CORAS-UNICAMP), no decorrer deste trabalho. No segundo, é proposta uma forma de extensão deste núcleo para torná-lo compatível com o MERB.

INDICE

CAPITULO 1: INTRODUÇÃO	1
CAPITULO 2: SISTEMAS DE BANCOS DE DADOS	
2.1.Introdução	6
2.2.Processamento de Dados - Histórico	9
2.3. Características de Sistemas de Banco de Dados	13
2.3.1.Compartilhamento de Dados	13
2.3.1.1.Redundância	14
2.3.1.2.Inconsistência	14
2.3.1.3.Privacidade e Segurança	15
2.3.1.4.Integridade	15
2.3.2.Independência de Dados	15
2.3.3.Desempenho	16
2.3.3.1.Sintonização	16
2.3.3.2.Migração de Dados	17
2.3.4.Integração	17
2.4.Abstração de Dados	18
2.4.1.Nível Conceitual	19
2.4.2.Nível Descritivo	21
2.4.3.Nível Lógico	23
2.4.4.Nível Físico	24
2.5.Modelos de Dados	26
2.5.1.Modelo Rede (Network)	28
2.5.1.1.Linguagem de Dados do Modelo Rede	32
2.5.2.Modelo Relacional	32
2.5.2.1.Linguagem de Dados do Modelo Relacional	35
2.5.2.2.Normalização	38
2.5.3.Modelo Entidade-Relacionamento	40

2.5.4. Resumo	42
CAPITULO 3: SISTEMAS DE CONTROLE	
3.1. Introdução	44
3.2. Hierarquização de Sistemas de Controle	49
3.2.1. Nível Gerencial	51
3.2.2. Nível Supervisor	52
3.2.3. Nível de Processo	54
3.2.4. Camadas Hierárquicas do Nível de Processo	55
3.3. Evolução dos Sistemas de Controle	57
3.3.1. Sistemas Pneumáticos	59
3.3.2. Sistemas Analógicos	60
3.3.3. Sistemas Digitais	61
3.4. Sistemas de Controle Automático	63
3.4.1. Princípios de Teoria de Controle	64
3.4.1.1. Modelamento de Sistemas Físicos	64
3.4.1.2. Modos de Ação do Controlador	67
3.4.2. Arquitetura de Sistemas de Controle Automático	68
3.4.2.1. Aquisição de Dados	69
3.4.2.1.1. Computador Off- line	69
3.4.2.1.2. Computador In-line	70
3.4.2.1.3. Computador On-line	71
3.4.2.2. Atuação de Controle	73
3.4.2.2.1. Computador em Ma- lha Aberta	73
3.4.2.2.2. Computador em Ma- lha Fechada	73
3.4.3. Tempo de Resposta	73
3.5. Sistemas de Controle Integrado	74
CAPITULO 4: MODELO ENTIDADE-RELACIONAMENTO BINÁRIO	
4.1. Introdução	79
4.2. Sistemas de Banco de Dados Convencional e para Aplicações em Controle de Processo	82
4.3. Primitivos do Modelo Entidade-Relacionamento Binário (MERB)	84

4.3.1. Entidade e Conjunto de Entidades	86
4.3.2. Relacionamento Binário e Conjunto de Relacionamentos	89
4.4. Descrição Estrutural do MERB	91
4.4.1. Declaração de Dados	92
4.4.1.1. Atributos	92
4.4.1.2. Papéis (Roles)	95
4.4.1.3. Versões/Domínio de tempo	96
4.4.1.4. Domínio de Valores	97
4.4.2. Controle de Dados	97
4.4.2.1. Módulo	97
4.4.2.2. Chave Secundária	99
4.4.2.3. Evento	99
4.5. Sintaxe da Descrição Estrutural	100
4.5.1. Descrição da Metalinguagem	102
4.5.2. Definição do Esquema	102
4.5.3. Especificação dos Primitivos	102
4.5.4. Definição do Corpo do Esquema	104
4.5.4.1. Definição do Módulo, Chave Secundária e Evento	104
4.5.4.2. Definição do Domínio de Valores e de Tempo	105
4.6. Integridade e Consistência através do MERB	106

CAPITULO 5: EXEMPLO DE APLICAÇÃO

5.1. Introdução	108
5.2. Descrição do Sistema Metroviário	115
5.2.1. Camada de Otimização	115
5.2.1.1. Recursos Disponíveis	115
5.2.1.2. Especificações Operacionais	118
5.2.1.3. Fluxo de Passageiros	118
5.2.1.4. Despacho Teórico	119
5.2.2. Camada de Adaptação	122
5.2.3. Camada de Regulação	125
5.2.4. Camada de Implementação	127
5.3. Ações de Operadores	127
5.3.1. Despachos Extras	128
5.3.2. Modificação nos Despachos	128
5.3.3. Cancelamento de Despacho	128
5.3.4. Modificação no tempo de parada	129

5.3.5.Modificação no Nível de Desempenho	129
5.3.6.Horário Defasa	129
5.3.7.Trem Direto	130
5.4.Modelamento do Sistema Metroviário através do MERB	130
5.4.1.Informações do Processo	131
5.4.1.1.Recursos Disponíveis	132
5.4.1.2.Especificações Operacionais	134
5.4.2.Informações sobre Fluxo de Passageiros	134
5.4.3.Informações sobre condições operacionais do Processo	135
5.4.4.Informações sobre Operação do Processo	138
5.4.5.Informações sobre localidade funcional de dados	141
5.5.Conclusão	143
 CAPITULO 6: Conclusões	 145
 APENDICE 1: CORAS - Core System for Associative Storage	 148
 APENDICE 2: Mapeamento entre CORAS-UNICAMP e MERB	 154
 BIBLIOGRAFIA	

CAPITULO 1: INTRODUÇÃO

Controle de Processo é um conceito que evoluiu com o progresso humano. Talvez o primeiro processo controlado de que se tenha conhecimento seja o processo de manter o fogo aceso, ainda na Idade da Pedra. Embora desconhecemos qualquer teoria de Controle, os homens da caverna sabiam que, para manter uma fogueira acesa, é necessário que seja colocado constantemente combustível (galhos secos) e caso quisessem alterar a intensidade da chama, bastaria modificar a dosagem do combustível que a alimentava.

O controle de um processo envolve basicamente a interação de dois agentes:

.SISTEMA FISICO: elemento passivo sobre o qual os controladores atuam para que o seu comportamento seja próximo daquele desejado;

.SISTEMA DE CONTROLE: elemento ativo que atua sobre o sistema físico para mantê-lo em torno do ponto de operação desejado.

O fluxo de informação existente entre estes dois agentes tem um papel decisivo nas ações controladoras geradas, pois quanto mais exatas e recentes forem as informações recebidas pelo Sistema de Controle sobre o comportamento do Sistema Físico, melhores poderão ser as decisões sobre a forma como os controladores deverão atuar para que seja mantido o desempenho pré-estabelecido. Conforme a complexidade do processo sob análise, o volume de informações necessárias pelo sistema de controle pode adquirir dimensões consideráveis.

Neste caso, Sistemas de Banco de Dados passam a ter um papel importante como suporte aos Sistemas de Controle, tanto pela sua capacidade de gerir eficazmente informações, através do pré-processamento de todas as informações referentes

ao Sistema Físico, como por possibilitarem um selecionamento, num dado instante, daquelas informações realmente requisitadas pelo Sistema de Controle, evitando que este seja sobrecarregado com informações desnecessárias.

As inovações tecnológicas possibilitaram o tratamento de sistemas cada vez mais complexos, tornando-se portanto conveniente a distribuição das funções controladoras em níveis de hierarquia, associadas à hierarquização das operações e dos equipamentos, onde cada nível hierárquico é autônomo e responde por uma determinada parcela das ações controladoras globais, projetadas para o sistema.

É comum distinguir três níveis de controle nos sistemas físicos industriais:

.GERENCIAL: responsável pela determinação da política e estratégia global do Sistema de Controle;

.SUPERVISOR: responsável pela coordenação e supervisão de cada subunidade do processo;

.PROCESSO: responsável pelo controle da evolução do processo físico em cada subunidade destacada no nível Supervisor.

Cada um destes níveis lida com informações de características diferentes, tendo assim exigências diferentes com relação à forma de manipulação (acesso, atualização) das informações.

As informações referentes ao NÍVEL GERENCIAL caracterizam-se por um grande horizonte temporal de existência e pela generalidade do seu conteúdo. O NÍVEL SUPERVISOR manipula informações de caráter mais repetitivo e rotineiro do que o nível Gerencial. E o NÍVEL DE PROCESSO é caracterizado pela atuação dos controladores em tempo real, em resposta às informações locais e bastante variantes no tempo.

Todas as características e restrições operacionais do Sistema de Controle devem ser descritas com máxima precisão pelos especialistas em Controle, de modo que o Administrador de Banco de Dados (ABD) tenha indicações

suficientes para definir precisamente uma estrutura de dados compatível com as exigências da Aplicação. A formalização desta descrição, através de um MODELO DE DADOS, é uma ferramenta poderosa para evitar a ambiguidade na interpretação dos requisitos a serem atendidos pelo Sistema de Banco de Dados.

Modelos de Dados para aplicações convencionais apresentam recursos suficientes para descrever as semânticas das informações dos níveis GERENCIAL e SUPERVISOR. No nível de PROCESSO, onde predominam atuações em tempo real, a velocidade de acesso às informações é fator preponderante na geração de ações de controle adequadas em intervalos de tempo aceitáveis. Portanto, o ABD deve ter conhecimento da dependência funcional entre as informações e da sua criticidade na ação controladora para garantir que a Base de Dados criada tenha um desempenho que corresponda às exigências do Sistema de Controle.

Este trabalho objetiva apresentar um Modelo de Dados aplicável ao Nível de Processo, através da definição de uma sintaxe simplificada que permita a formalização de todos os requisitos de tempo real impostos pelo Sistema de Controle. O modelo proposto é denominado MODELO ENTIDADE-RELACIONAMENTO BINÁRIO (MERB). A estrutura deste modelo baseia-se essencialmente nos conceitos já definidos no Modelo Entidade-Relacionamento proposto por Chen /CHEN 76/, com devidas extensões e modificações para torná-lo mais apropriado ao tratamento dos aspectos dinâmicos envolvidos num processo a ser controlado. Entre os modelos mais difundidos e para os quais já se encontram implementações ou propostas de implementações (Network, Hierárquico, Relacional, Entity-Set), escolheu-se o Modelo Entidade Relacionamento Binário como modelo-base por apresentar uma semântica mais rica e completa.

Resumidamente, este trabalho aborda os seguintes tópicos:

No Capítulo 2 apresenta-se uma visão geral de Sistemas de Banco de Dados, dando-se ênfase aos seus diferentes níveis de abstração de forma a permitir uma maior independência física e lógica dos dados. São discutidos, sob o ponto de vista

da descrição formal do Mundo Real, três Modelos de Dados convencionais - Network (Rede), Relacional e Entidade Relacionamento. Os primitivos e o grau de abstração de cada modelo são apresentados e explicados.

No Capítulo 3 é feita uma abordagem geral dos princípios básicos de Sistemas de Controle hierárquico (níveis Gerencial, Supervisor e de Processo), dando-se ênfase à natureza e ao volume de informações necessárias para cada nível de hierarquia, com a finalidade de mostrar algumas características particulares do nível de Processo difíceis de serem representadas de forma natural pelos Modelos de Dados existentes.

No Capítulo 4 apresenta-se uma proposta de um Modelo de Dados para descrição formal das informações envolvidas num Sistema de Controle - Modelo Entidade-Relacionamento Binário. Este modelo é descrito sob o ponto de vista conceitual, abordando-se detalhadamente os seus primitivos. É apresentada ainda uma sintaxe para representação do modelo com a descrição formal de todas as exigências a serem atendidas num Sistema de Controle.

No Capítulo 5 é apresentado um exemplo de aplicação, onde o Sistema de Banco de Dados é utilizado para gerir as informações necessárias ao Sistema de Controle de um processo metroviário. Através deste exemplo, é ilustrada a aplicabilidade do MERB para a representação de todas as informações semânticas identificadas.

No Capítulo 6 são apresentadas as conclusões do trabalho, seguidas de sugestões e comentários das diversas possíveis linhas de continuidade para o presente trabalho.

São ainda incluídos dois anexos. No anexo 1, apresenta-se resumidamente o Sistema CORAS, implementado no decorrer deste trabalho. O Sistema CORAS - Core System for Associative Storage - é um pacote de software que contém primitivos sobre os quais pode-se definir um Modelo de Dados. No Anexo 2, apresenta-se um possível mapeamento do MERB aos primitivos (recursos) do CORAS.

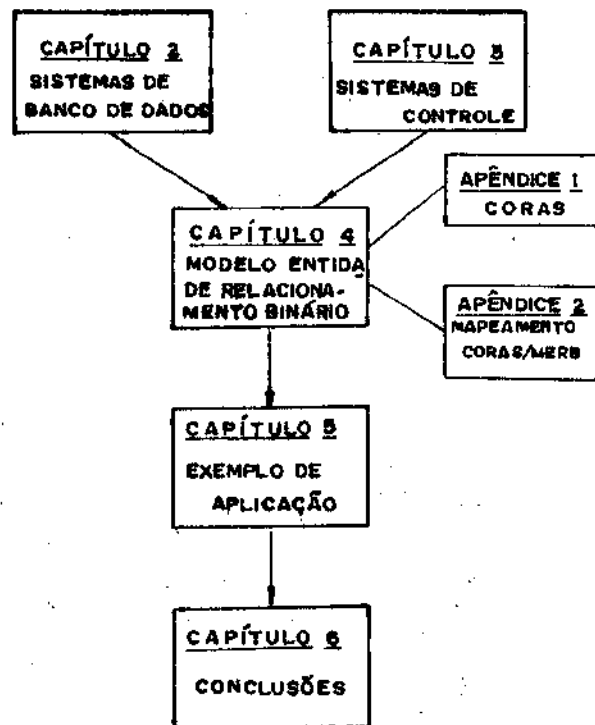


FIG 1.1: Corpo do trabalho

CAPITULO 2: SISTEMAS DE BANCO DE DADOS

2.1. INTRODUÇÃO

Um SISTEMA DE BANCO DE DADOS, ou simplesmente BANCO DE DADOS, consiste em uma coleção de dados interrelacionados que são armazenados, sem redundância prejudicial ou desnecessária, para servir a uma ou mais aplicações. Estes dados são armazenados, de modo que sejam independentes dos programas que os utilizem e tanto as operações de adição de novos dados, como as de modificação e acesso aos dados existentes nos arquivos, sejam uniformes a todos os usuários do Sistema /MART 77/.

O conjunto de dados a partir dos quais serão derivadas as informações necessárias aos programas de aplicação é denominado BASE DE DADOS /ULLM 80/ ou DADOS OPERACIONAIS /DATE 74/. O software que controla o acesso a Base de Dados denomina-se o SISTEMA DE GERENCIAMENTO DE BANCO DE DADOS (SGBD). Existe ainda uma equipe de pessoas conhecida como o ADMINISTRADOR DE BANCO DE DADOS (ABD), que se responsabiliza pelo mapeamento das informações definidas pelos usuários de aplicação aos primitivos do modelo de dados suportado pelo SGBD disponível. Muitas vezes, o ABD é considerado como um membro integrante do Sistema de Banco de Dados /DATE 74/, devido ao fato de que o desempenho de um Banco de Dados depende das decisões sobre as estratégias de acesso e estruturação física/lógica tomadas pelo ABD.

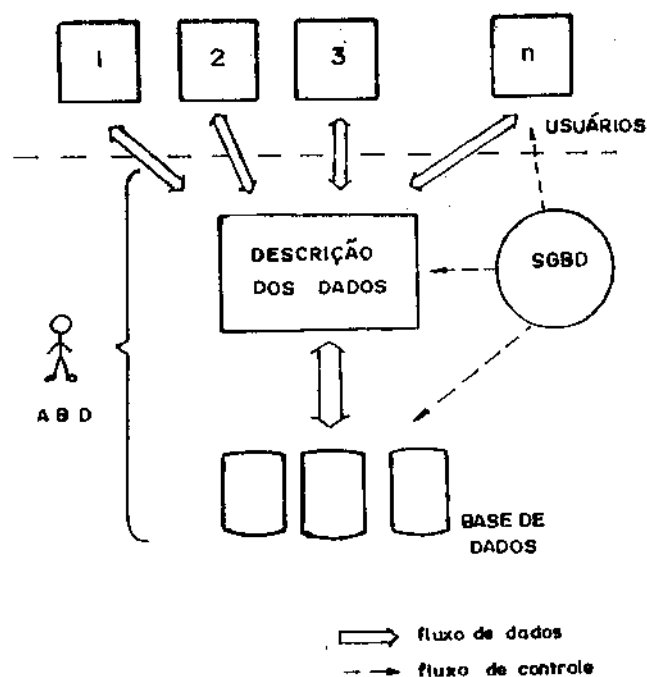


FIG. 2.1: Sistema de Banco de Dados

O SISTEMA DE GERENCIAMENTO DE BANCOS DE DADOS (SGBD) é formado por um conjunto de procedimentos que isola os programas de aplicação dos detalhes referentes à criação, busca, armazenamento, modificação, segurança e estruturação física da Base de Dados /TSIC 77/. Cada usuário deve poder acessar qualquer grupo de dados a que ele tenha direito, sem ter de se preocupar com a real organização e estruturação interna dos mesmos.

Assim, um SGBD deve incluir tanto funções que facilitem a definição das estratégias de estruturação dos dados físicos/lógicos por parte do ABD, como também os acessos por parte dos usuários (programas de aplicação) (fig 2.1.). Ao longo deste texto, serão usados indistintamente os termos usuários e programas de aplicação.

Para melhor caracterizar as funções de um SGBD, serão abordadas, em maior detalhe, duas finalidades de um Sistema de Bancos de Dados:

- independência dos dados;
- compartilhamento dos dados.

A INDEPENDENCIA DOS DADOS refere-se ao

isolamento entre o modo de utilização dos dados, por parte dos usuários, e a estrutura de armazenamento e estratégia de acesso dos mesmos. Com isso, objetiva-se a redução da interdependência entre os programas de aplicação e a forma de tratamento dos dados requisitados. O isolamento faz-se com a definição de uma interface, que mapeia a representação lógica na representação física dos dados e vice-versa. Conforme a complexidade e a potencialidade das funções incluídas no SGBD, distinguem-se diferentes graus de independência: INDEPENDENCIA SIMPLEMENTE FISICA, INDEPENDENCIA FISICA E PARCIALMENTE LOGICA e INDEPENDENCIA FISICA E LOGICA /WEDE 77/. Em regra geral, quanto maior for o grau desta independência, menor será o custo de adaptação dos programas já existentes, caso sejam modificadas as representações internas no computador e vice-versa.

Paralelamente aos princípios de independência de dados surgiu o conceito de ABSTRAÇÃO DE DADOS, que enfatiza não só a representatividade como também a semântica dos dados armazenados /HAMM 76/. Um conjunto de dados numa Base de Dados pode ser representado em níveis de abstração semanticamente distintos, cada qual apresentando um diferente grau de independência lógica e/ou física. Quanto maior for o nível de abstração dos dados, maior será a aproximação da semântica destes dados ao Mundo Real visto pelos usuários.

O COMPARTILHAMENTO DOS DADOS é fator importante em aplicações onde o volume de dados é grande e há forte intercâmbio de informações entre os programas de aplicação. O compartilhamento significa que dados de programas distintos ficam armazenados num mesmo "reservatório" cujo acesso depende do grau de privilégio concedido aos usuários (fig 2.2.). A grande vantagem disso é a diminuição da redundância física dos dados, evitando problemas referentes à consistência. O compartilhamento porém, cria, entre outros, problemas: quanto à garantia da privacidade dos usuários em relação aos seus dados particulares; à possibilidade de ocorrência de duas ou mais operações simultâneas e incompatíveis, levando a intertravamento (deadlock), se a coordenação do sistema falhar; etc; de modo que um SGBD deve ser provido de funções para verificação de autorização e de procedimentos de validação.

Os programas de aplicação que compartilham os mesmos dados não precisam apresentar finalidades necessariamente semelhantes, pois o SGBD deve prover recursos que abstraiam diferentes VISÕES do Mundo, a partir de uma mesma Base de Dados. Assim, para cada usuário, é como se os dados fossem estruturados para o seu uso exclusivo.

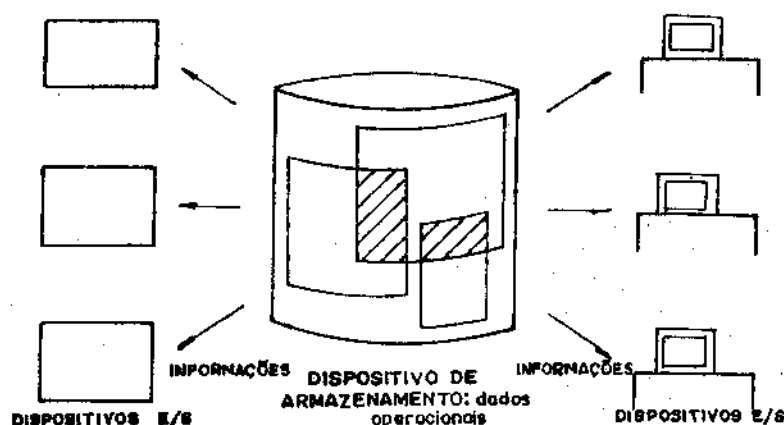


FIG 2.2: Compartilhamento de dados

2.2. PROCESSAMENTO DE DADOS - HISTÓRICO

Tradicionalmente, programas caracterizam-se por ter uma forte dependência com a forma como seus dados estão estruturados. Programas, mesmo quando utilizam dados semelhantes, demandam arquivos distintos. Enquanto o computador é usado como máquina de calcular, isso não se configura como um problema crítico. Mas, à medida que aumenta a complexidade e a quantidade dos dados dos programas de aplicação, isso torna-se indesejável, por fatores como redundância e volume excessivo de dados, etc. A solução será estabelecer uma Base de Dados comum e gerenciá-la por um sistema de software.

No decorrer da evolução das técnicas de gerenciamento de dados, distinguem-se quatro fases marcadas, ou pelos avanços tecnológicos na área de hardware ou pela evolução na área de software:

.primeira fase: programas totalmente dependentes da estrutura física dos dados;

.segunda fase: introduzem-se programas dependentes da organização lógica dos dados;

.terceira fase: introduzem-se programas independentes da organização lógica dos dados;

Os programas totalmente dependentes da estrutura física exigem que o usuário desenvolva não só o software de aplicação como também, o software de estruturação de dados e operações de E/S. A estrutura de dados é totalmente dependente de uma aplicação específica. Quando ocorrem modificações na organização dos dados e/ou nos dispositivos de armazenamento, os programas de aplicação devem ser alterados, recompilados e retestados devidamente. Muitas vezes, diferentes programas de aplicação que necessitam dos mesmos dados, porém sob formas diferentes, demandam arquivos distintos (fig 2.3.).

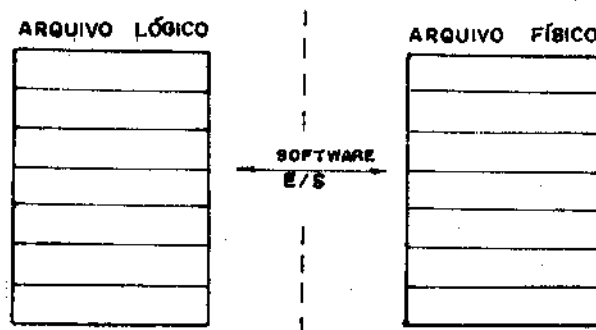


FIG 2.3: Arquivos distintos antes da década de 60:
-dispositivo de armazenamento lento:fita,etc;
-arquivos físicos equivalentes aos arquivos lógicos;
-elevada interdependência entre dados e dispositivos de armazenamento;
-elevada redundância de dados.

O surgimento de discos possibilitou acesso direto aos arquivos. Assim, os usuários passaram a localizar "diretamente" os dados requisitados utilizando seus endereços físicos, ao invés de ter de varrer o arquivo para verificar dados um por um. Esta nova tecnologia reduziu substancialmente o tempo de acesso a um grande volume de dados ALEATÓRIOS. Entretanto, a dependência entre os programas de aplicação e seus respectivos dados ainda persistia.

Programas somente dependentes da organização lógica dos dados caracterizam-se pelo isolamento entre programas de aplicação e estrutura física dos arquivos, através de um pacote de software, incluído no próprio Sistema Operacional, que mascara, através de uma interface Base de Dados/Programas de aplicação, todos os detalhes da organização física dos dados (fig 2.4.), tornando independentes programas dos dispositivos de armazenamento.

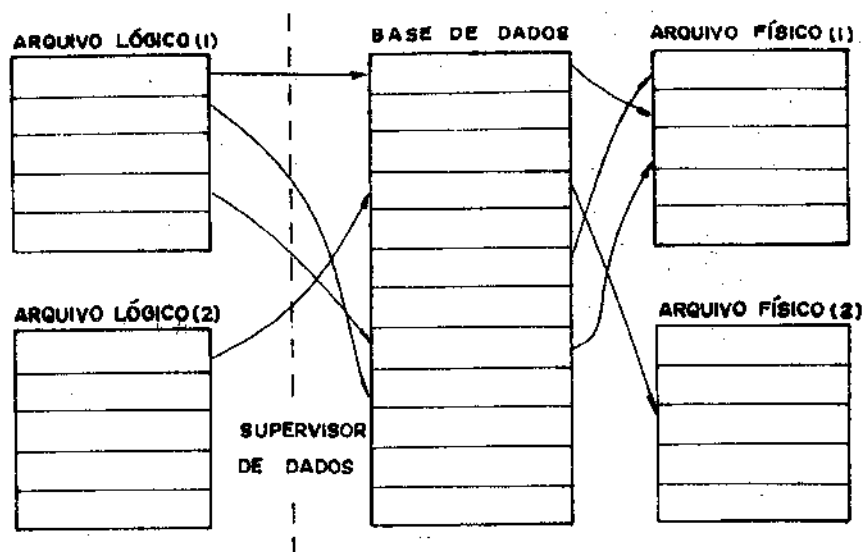


FIG 2.4: Características dos primeiros sistemas de gerenciamento de dados:
 . redução de dados redundantes;
 . independência da organização física de dados;
 . compartilhamento de dados;
 . dependência dos programas de aplicação à organização lógica dos dados.

Na década de 70, a grande difusão do uso de computadores em todas as áreas causou um aumento no interesse de isolamento entre programas e sua estrutura lógica de dados (por ex.: tabelas de acesso, listas de células lógicas ligadas para guardar informações de tamanho variável e imprevisível, etc). Desta forma, os programas manter-se-iam inalterados, mesmo que ocorressem modificações nos endereços e nas estratégias de acesso lógico aos dados, pois a complexidade da estrutura lógica dos dados requeridos pelos diferentes usuários fica transparente a eles. Sob o ponto de vista destes, os dados estariam estruturados numa forma simples segundo um Modelo de Dados. Para isso, surgiu o conceito de SISTEMA DE GERENCIAMENTO DE DADOS, que provê recursos para mapear a estrutura do Modelo de Dados na estrutura

de dados lógicos e físicos implementada (fig 2.5.).

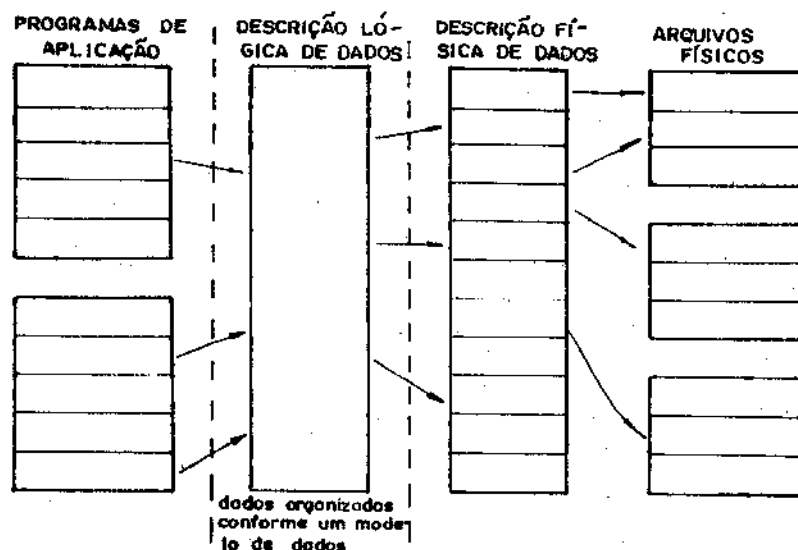


FIG 2.5: Banco de dados da década de 70:

- programas de aplicação com vida útil longa;
- independência dos endereçamentos lógicos dos dados;
- software bastante complexo para o controle de privacidade, segurança e integridade de dados;
- provido de linguagem de descrição e linguagem de manipulação de dados para unificar a visão lógica dos dados;
- maior flexibilidade quanto à reestruturação dos dados.

Modelos de Dados constituem um meio eficiente de comunicação entre usuários e Sistema de Banco de Dados. Nos últimos anos, dezenas de Modelos de Dados foram propostas, objetivando encontrar um que apresente primitivos e regras bem definidas de sintaxe e semântica, para descrever sem ambiguidade as percepções dos usuários sobre o Mundo Real que os envolve. Classificam-se estes modelos em duas grandes classes (fig 2.6.):

- os direcionados à concepção computacional;
- os direcionados à concepção do Mundo Real.

Sob o ponto de vista de implementação, os SGBD's comercialmente disponíveis hoje em dia suportam Modelos de Dados próximos à concepção computacional. Pesquisas e estudos tem sido feitos no sentido de estender estes SGBD's através da inclusão de funções que estabeleçam a correspondência biunívoca

entre os primitivos definidos nestas duas diferentes percepções, automatizando assim a transformação da estrutura de informações familiar aos usuários em estrutura de dados suportada pelo computador.

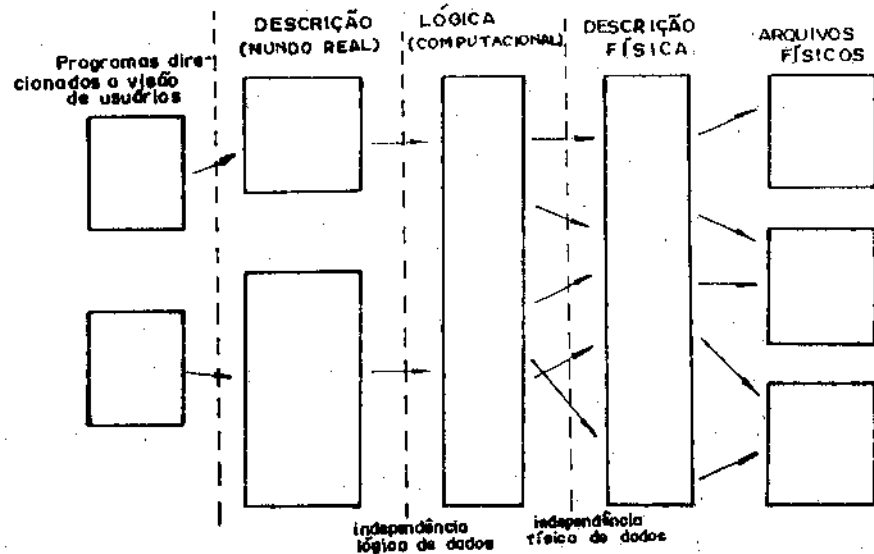


FIG 2.6: Geração atual dos Sistemas de Banco de Dados: automatização do mapeamento entre o o modelo de dados próximo à concepção computacional e o próximo à concepção do Mundo Real.

2.3. CARACTERÍSTICAS DE SISTEMAS DE BANCO DE DADOS

Conforme definido anteriormente (item 2.1), os principais objetivos, a serem atendidos no uso de um Sistema de Banco de Dados, podem ser resumidos basicamente em: proporcionar INDEPENDENCIA e COMPARTILHAMENTO DE DADOS, almejando uma MELHOR UTILIZAÇÃO dos mesmos e INTEGRANDO-os como um conjunto global. Em seguida, abordar-se-á cada um destes aspectos.

2.3.1. COMPARTILHAMENTO DE DADOS

Com um sistema que permita o compartilhamento dos dados, o volume de dados redundantes pode ser reduzido. Isto, embora signifique uma redução do custo por bit de armazenamento, torna necessário um conjunto de algoritmos complexo e extenso

para controlar o acesso aos mesmos dados operacionais pelos diferentes programas de aplicação.

O compartilhamento de dados permite não só a redução do volume do espaço ocupado (redução de dados redundantes) como também possibilita a utilização dos dados já existentes por novos programas de aplicação. Ligado ao compartilhamento de dados, devem-se ressaltar os aspectos relativos ao controle de redundância, inconsistência, segurança, privacidade e integridade de dados.

2.3.1.1. REDUNDANCIA

Dados redundantes podem ocupar desnecessariamente espaços físicos e encarecer as operações de manipulação. Por exemplo, se for atualizado um dado, todos os outros redundantes devem ser alterados adequadamente, para evitar inconsistência. Porém, nem sempre a redundância é indesejável. Na realidade, em muitos casos, recomenda-se a redundância para melhorar o tempo de acesso e simplificar os métodos de endereçamento. Em algumas situações, dados são duplicados fisicamente para assegurar a recuperação destes, caso ocorram falhas acidentais. Portanto, um Sistema de Banco de Dados deve ser capaz de eliminar os dados redundantes desnecessários, para obter um ponto ótimo entre a eficiência do Sistema e o volume de dados redundantes.

2.3.1.2. INCONSISTENCIA

Um Sistema é dito CONSISTENTE num instante t , quando quaisquer grupos idênticos de dados geram resultados de valores iguais para inferências idênticas, satisfazendo ao mesmo tempo as restrições impostas. Note-se que, em certos casos, a inconsistência é uma propriedade intrínseca do sistema que contém os dados redundantes, pois, raramente, eles são atualizados simultaneamente. A sequência de procedimentos que transfere a Base de Dados de um estado consistente a um novo estado consistente é denominada TRANSAÇÃO. Durante o processo de

transformação de estado, a Base de Dados fica "temporariamente" em estado inconsistente /NEUM 82/. Um Sistema de Banco de Dados deve prover controles adequados para distinguir uma inconsistência temporária de uma permanente e tomar as ações corretivas necessárias.

2.3.1.3. PRIVACIDADE E SEGURANÇA

São características mais ligadas aos fatores sociais que aos computacionais propriamente ditos.

A PRIVACIDADE refere-se ao direito que o usuário tem para determinar quando, como e quanto das suas informações privadas podem ser acessadas pelos outros que utilizem a mesma Base de Dados. SEGURANÇA significa a proteção contra alterações, destruições acidentais e/ou não autorizadas.

2.3.1.4. INTEGRIDADE

Entende-se como INTEGRIDADE, a manutenção de dados verdadeiros nos arquivos, satisfazendo sempre as restrições impostas pelo Sistema e/ou usuários. Em quaisquer tipos de acidentes (falhas de hardware ou pessoal), o Sistema deve prover meios para recuperar os dados atingidos, garantindo a manutenção da integridade. Alguns recursos utilizados para este fim são: log-file, imagem da sessão, etc.

2.3.2. INDEPENDENCIA DE DADOS

Um Sistema de Banco de Dados deve prover basicamente duas visões independentes de dados - representação física de dados e representação lógica de dados, de forma a garantir que modificações na estrutura física dos dados não afetem os programas de aplicação existentes e vice-versa. A importância desta característica já foi ressaltada no item 2.1.

2.3.3. DESEMPENHO

O desempenho de um Sistema de Banco de Dados está intimamente ligado à organização dos dados (física e lógica), pois, envolve como parâmetros de avaliação, o volume do espaço físico ocupado, o acesso aos dados e os tipos de interação usuário/Sistema disponíveis. Por exemplo, para os usuários on-line, o tempo de diálogo homem-máquina (intervalo de tempo entre a entrada do comando e a saída de mensagem) é um fator importante para avaliar o desempenho do sistema.

Até pouco tempo atrás, só existiam Sistemas de Banco de Dados não inferenciais, o que permitia que o Sistema fosse organizado com certo desempenho a priori. Entretanto, como a tendência atual é a implementação de sistemas inferenciais, estes devem prover recursos adequados para selecionar melhores caminhos de busca em cada inferência não prevista durante o projeto, assegurando, sempre que possível, o melhor desempenho.

Para melhorar o desempenho do Sistema, recursos como SINTONIZAÇÃO e MIGRAÇÃO DOS DADOS podem ser incorporados entre as facilidades oferecidas pelo SGBD.

2.3.3.1. SINTONIZAÇÃO

A medida que os dados forem sofrendo alterações, a sua estrutura pode também sofrer modificações (por exemplo, caminhos de acesso, número de registros, índices, apontadores, etc). Sabe-se que o desempenho dos programas de aplicação depende da estrutura dos dados - sempre existe uma estrutura mais adequada para uma aplicação específica. Portanto, é desejável que o Sistema de Gerenciamento de Banco de Dados possa organizar os dados dos programas de aplicação em uma das estruturas disponíveis mais adequadas, assegurando sempre um bom desempenho. Esta propriedade é importante para os sistemas inferenciais. Nos sistemas existentes, estes ajustamentos (escolha de uma estrutura lógica mais adequada) são feitos baseados em dados estatísticos.

2.3.3.2. MIGRAÇÃO DE DADOS

Em certos sistemas, é interessante agrupar os dados localizados conforme a sua frequência de uso. Uma das estratégias utilizadas é armazenar os dados obsoletos em memórias lentas (fitas, por exemplo) e os de consulta corriqueira em memórias rápidas (por exemplo, discos). Periodicamente, são feitas verificações de todos os dados existentes na Base de Dados e reorganizadas suas localidades. Este remanejamento de dados é conhecido como MIGRAÇÃO DE DADOS. Deve-se ressaltar que, em muitos casos, a redução do tempo de acesso aos dados não é devida à migração física dos dados e sim, devida à redefinição de novas estratégias de acessos (tabelas hashing, novas árvores de busca, etc).

2.3.4. INTEGRAÇÃO

Entende-se como INTEGRAÇÃO das Bases de Dados relacionados a uma aplicação, a administração de todas as informações contidas num Sistema de Banco de Dados como um conjunto global. Cada vez mais, esta é uma necessidade imperiosa das aplicações, trazendo vantagens tais como: menor redundância, maior controle de integridade, maior compartilhamento de dados, melhor utilização de recursos disponíveis, etc. Existem, porém, alguns problemas que devem ser levados em consideração:

- maior vulnerabilidade à destruição total de dados, se ocorrerem falhas;

- maior susceptibilidade à quebra de sigilo dos dados dos usuários que compartilhem a mesma Base de Dados;

- complexidade do software para gerenciar um grande volume de dados destinados às aplicações que podem ser de natureza antagônica, como por exemplo, aplicações em tempo real e aplicações altamente confidenciais.

Face a estes problemas, o grau de integração dos dados, a ser adotado por uma instituição que pretende implementar um Sistema de Banco de Dados, dependerá da natureza das informações, das aplicações em análise, da disponibilidade do

software (por exemplo, o "overhead" necessário ao controle dos problemas citados) e hardware (por exemplo: mecanismos de acesso físico aos dados).

2.4. ABSTRAÇÃO DE DADOS

Para que todas as informações relevantes a uma aplicação possam ser armazenadas num Sistema de Banco de Dados, elas devem ser identificadas e modeladas através de primitivos aceitáveis pelo SGBD para serem, então, convertidas em dados internos compatíveis com a estrutura suportada pelos dispositivos físicos de armazenamento.

As diferentes percepções de uma mesma informação são denominadas NÍVEIS DE ABSTRAÇÃO DOS DADOS. Não há ainda uma padronização na classificação destes níveis. Aqui serão considerados quatro níveis /CHEN 76//WGIO 77/(fig 2.7.):

- .Nível conceitual;
- .Nível descritivo;
- .Nível lógico:
 - .independente dos caminhos de acesso;
 - .dependente dos caminhos de acesso;
- .Nível físico.

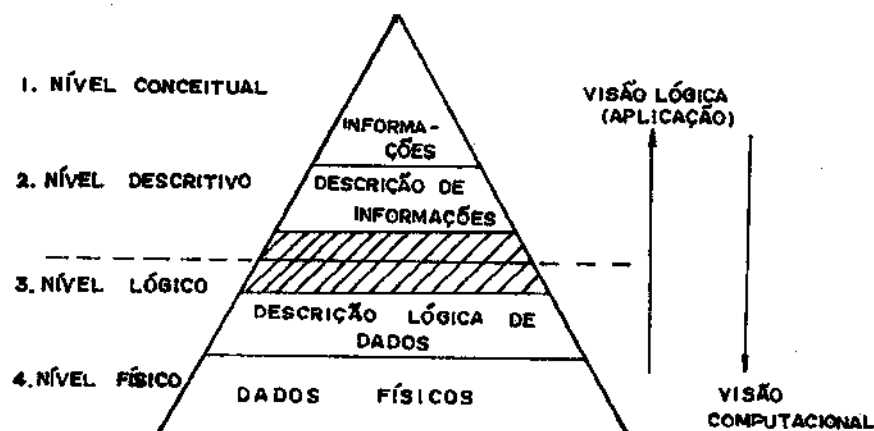


FIG 2.7: Quatro níveis de abstração de dados

Os níveis Conceitual e Descritivo referem-se a visão do Mundo Real, enquanto os níveis lógico e físico estão mais próximos da concepção do Mundo Computacional. Através de uma sequência de procedimentos conhecida como MAPEAMENTO, as informações modeladas em mente dos usuários (Nível Conceitual) serão transformadas em dados físicos armazenados nas memórias físicas dos computadores (Nível Físico).

2.4.1. NÍVEL CONCEITUAL

Este nível trata as informações (percepções) do Mundo Real que os usuários têm em mente. Para processar estas percepções e transmiti-las às outras pessoas, existem diversas formas para representar o Mundo Real como, por exemplo, através de sentenças obedecendo as regras gramaticais e semânticas de cada língua. Com a crescente importância do computador como ferramenta de apoio às atividades humanas, tornaram-se necessários meios apropriados de comunicação entre os homens e os computadores. Idealmente, esta comunicação deveria ser em linguagem natural dos usuários (por ex., português, inglês, etc). Porém, no estágio atual do desenvolvimento do software, os computadores não adquiriram ainda a capacidade de lidar com vocabulários volumosos e estruturas gramaticais sofisticadas, havendo assim a necessidade de limitar a semântica da sua "linguagem" e estabelecer regras gramaticais mais simplificadas.

O Mundo Real pode ser modelado satisfatoriamente através dos seguintes primitivos: ENTIDADES (substantivos), RELACIONAMENTOS (verbos ou ações) e ATRIBUTOS (adjetivos e advérbios).

As ENTIDADES são os objetos e/ou eventos do Mundo Real que interessam aos usuários. É comum agrupar as entidades similares em conjuntos ou classes de entidades, denominadas TIPOS DE ENTIDADES. Por exemplo, <FORNO> é um tipo de entidade. Cada tipo de entidade é qualificado ou caracterizado por um conjunto de ATRIBUTOS. Por exemplo, fabricante, tipo, capacidade, n.^o de série, etc, podem ser atributos do tipo de entidade <FORNO>. As informações sobre cada ocorrência (cada

entidade particular) de um tipo de entidade são expressas em VALORES que descrevem qualitativa e/ou quantitativamente os ATRIBUTOS. Por exemplo, fabricante=VILLARES, tipo=FORNO DE REAQUECIMENTO, capacidade=1 t, etc.

Em quase todas as situações reais, existem muitos tipos de entidades relevantes (de interesse aos usuários) que, quando associados, compõem outras informações. A associação que se estabelece entre as ocorrências de dois tipos de entidades é denominada RELACIONAMENTO. Essencialmente, classificam-se três tipos de relacionamentos - 1:1, 1:N e M:N. Considerando as associações entre dois tipos de entidades X e Y como correspondências

$$f_x: X \rightarrow Y$$

$$\text{e } f_y: Y \rightarrow X,$$

os três tipos de entidades podem ser definidos formalmente como /TSIC 77/ (fig 2.8.):

.1:1, quando ambas as correspondências f_x e f_y são funcionais;

.1:N, quando somente uma das correspondências f_x ou f_y é funcional;

.M:N, quando nenhuma das duas correspondências é funcional, isto é, cada 'x' pode ter muitos 'y' associados e vice-versa.

As correspondências funcionais podem ser classificadas em dois subgrupos: CORRESPONDENCIAS TOTALMENTE FUNCIONAIS e CORRESPONDENCIAS PARCIALMENTE FUNCIONAIS, que correspondem respectivamente às funções injetoras e sobrejetoras definidas na teoria de funções. Este conceito de funcionalidade das associações torna explícito o tipo de relacionamento (1:1, 1:N, M:N) que existe entre os componentes identificados num contexto de aplicações.

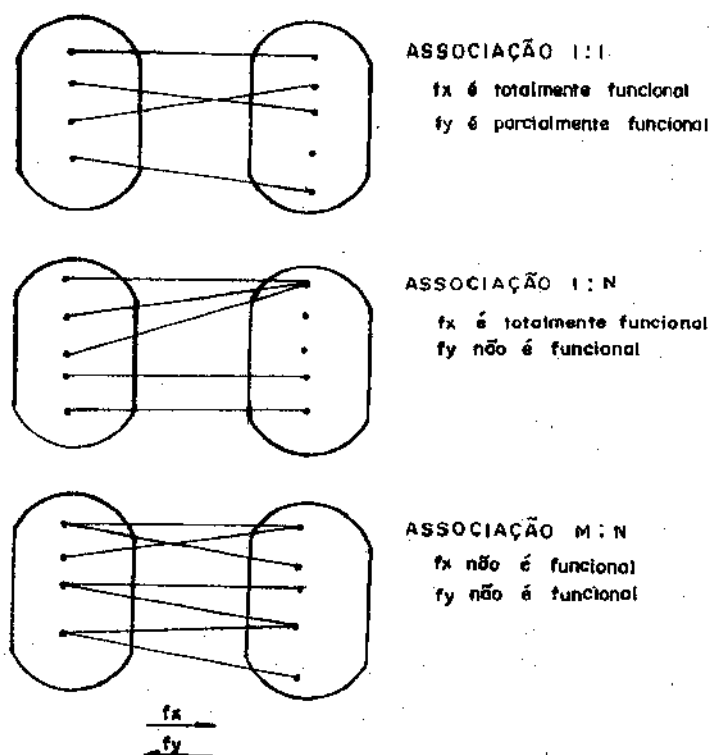


FIG 2.8: Tipos de relacionamentos

2.4.2. NÍVEL DESCRITIVO

Entre as funções sob responsabilidade do ABD inclui-se o projeto da estrutura dos dados para a aplicação em estudo, utilizando o modelo oferecido pelo Sistema de Gerenciamento de Banco de Dados disponível. Para isso, o ABD deve conhecer os tipos e a finalidade das informações que o Sistema de Banco de Dados armazenará. Como estes tipos de informações são específicos, dependentes das aplicações, eles são usualmente definidos pelos usuários especializados em Aplicações. Com isso, surgiu um problema quanto à exatidão da comunicação das informações, que os usuários têm em mente, aos ABD, que podem interpretá-las sob ângulos diferentes e decidir por uma organização ineficiente para as exigências dos usuários.

Como solução a esta possível ambiguidade de interpretação das informações sobre o Mundo Real, definidas no Nível Conceitual, são estabelecidos os MODELOS DE DADOS, através dos quais os usuários especificam formalmente os seus requisitos

e definem detalhadamente os primitivos identificados no Nível Conceitual como, por exemplo, os tipos de valores associados às entidades, os graus de proteção a serem atribuídos aos seus dados e o grau de velocidade de manipulação destas informações em relação à execução das tarefas de uma Aplicação. Enfim, Modelos de Dados direcionam o usuário a descrever suas exigências numa forma concisa e precisa. Assim, os ABD's terão uma noção global e correta sobre a natureza das informações que devem constar na Base de Dados, para suportar os prováveis programas que a utilizarão, e definir uma estrutura adequada aos recursos oferecidos pelo SGBD escolhido.

Dependendo da capacidade de abstração do SGBD, o mapeamento do modelo de dados definido pelos usuários ao Modelo de Dados suportado pelo SGBD pode, ou não, ser imediato. Quando for imediato (corresponde à parte hachurada da fig 2.7.), os ABD's definem o lay-out da estrutura dos dados na Base de Dados utilizando diretamente os esquemas do Modelo de Dados fornecidos pelos usuários. Estes esquemas serão compilados e usados pelo SGBD, tanto para codificar e decodificar corretamente os dados, como também para controle de integridade e consistência destes dados. Para o segundo caso, as informações devem ser transformadas e estruturadas em recursos aceitáveis pelo SGBD e a documentação da elaboração dos esquemas constitui uma importante fonte de referência para os usuários de como se deve manipular os dados armazenados (fig 2.9). Estes esquemas correspondem ao Nível Lógico na hierarquia de Abstração de Dados.

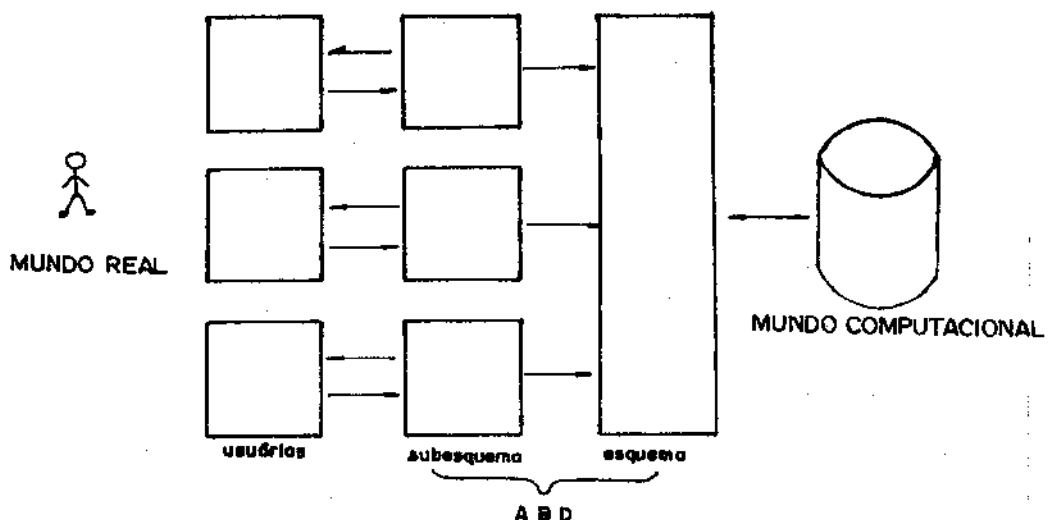


FIG 2.9: Interface entre os usuários (Mundo Real) e ABD (Mundo Computacional)

2.4.3. NIVEL LÓGICO

Este nível reflete a capacidade do SGBD de produzir automaticamente um mapeamento correto entre o Nível Físico (os dados organizados na Base de Dados) e as representações lógicas dos dados denominadas também Modelos de Dados. Estes Modelos de Dados são utilizados pelo ABD para organizar os dados em uma estrutura compatível com a oferecida pelo SGBD a partir das informações especificadas pelos usuários através dos Modelos de Dados correspondentes ao Nível Descritivo. Classificam-se os modelos do Nível Lógico em duas categorias quanto ao seu grau de abstração:

- .dependente dos caminhos lógicos de acesso;
- .independente dos caminhos lógicos de acesso.

Os Modelos de Dados independentes dos caminhos lógicos de acesso (por ex.: Modelo Relacional) não oferecem recursos explícitos para os usuários interferirem na organização física dos dados. Normalmente, um SGBD que os suporte é mais complexo, devendo conter algoritmos para mapear as descrições formais em estruturas físicas mais convenientes aos programas de aplicação.

Qualquer que seja o grau de abstração, uma Base de Dados pode ser acessada por diferentes usuários com diferentes visões da informação armazenada. Portanto, para facilitar a interação usuário/Sistema e proteger os dados de acessos inadequados, é interessante que se considere a parte as informações úteis a um determinado usuário das informações globais. Assim, sob este enfoque, destacam-se ainda dois subníveis de abstração (fig 2.10.):

.SUBESQUEMAS (ou DEFINIÇÃO DE SUBMODELOS DE DADOS): referem-se às descrições formais dos dados utilizados pelos programas de aplicação. É a visão que cada usuário tem sobre a Base de Dados;

.ESQUEMAS (ou DEFINIÇÃO DE MODELOS DE DADOS): referem-se às descrições formais do contexto global da Base de Dados. A partir dela, diferentes subesquemas podem ser derivados.

E a visão que um ABD deve ter sobre a Base de Dados, para poder analisar e supervisionar o desempenho do Sistema como um todo.

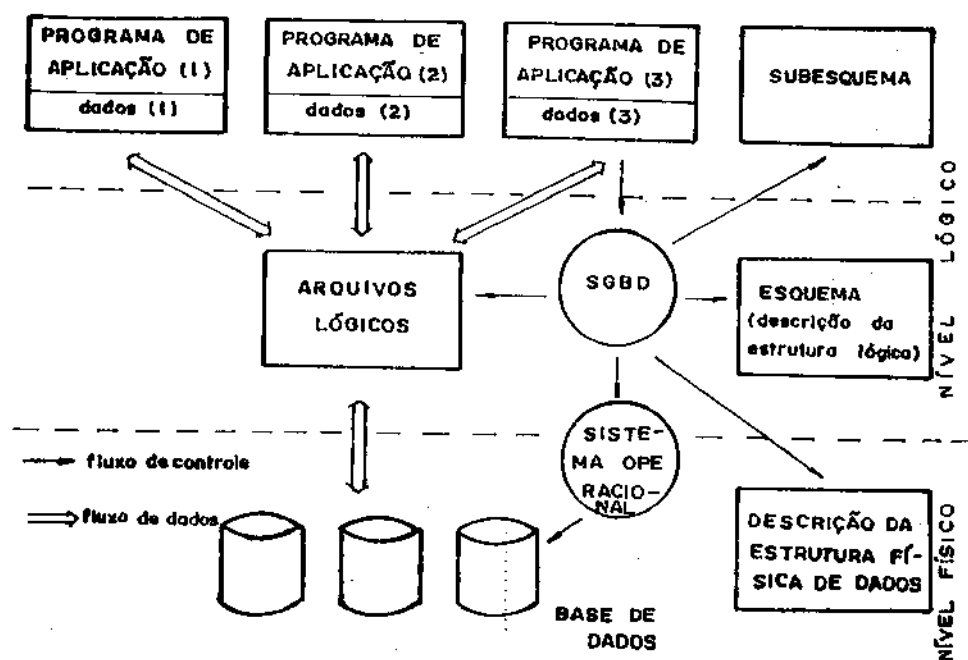


FIG 2.10: Distinção de dois subníveis de abstrações no Nível Lógico de Dados (esquemas e subesquemas)

Tanto os esquemas quanto os subesquemas constituem o denominado conjunto de MASCARAS utilizadas pelo SGBD para mapear dos dados físicos as abstrações correspondentes aos diferentes níveis/subníveis e vice-versa (fig 2.11).

2.4.4. NÍVEL FÍSICO

Os critérios usados para definir a organização física dos dados são diferentes daqueles usados para a organização lógica dos dados. Como, na maioria dos sistemas, a implementação da estrutura física dos dados fica a cargo do Sistema Operacional sobre o qual o Sistema de Gerenciamento de Banco de Dados está apoiado, só serão citadas as características físicas das Bases de Dados que devem ser analisadas pelo ABD para verificar o grau de compatibilidade entre o SGBD e as exigências das Aplicações.

É importante ressaltar aqui que, muitos sistemas operacionais existentes não são realmente apropriados para serem utilizados especificamente como suporte a Sistema de

Banco de Dados. Portanto, já existem estudos no sentido de desenvolver um que propiciasse suporte adequado a isso /CARV 82/.

A organização física dos dados está intimamente ligada à eficiência operacional: tempo de operação (caminhos de acesso, localidade física dos dados, etc.) e custo (volume ocupado do espaço físico). Abaixo são relacionadas algumas características que devem ser verificadas na escolha de uma estrutura física dos dados /MART 77/:

- .espaço de armazenamento;
- .redundância física de dados;
- .tempos de operação desejáveis (inserção, remoção, leitura, etc);
- .tempos de resposta desejáveis;
- .volatilidade dos dados armazenados: tempo de permanência dos dados no Sistema;
- .tipos de dados armazenados: tamanho dos registros, bloco, etc;
- .estrutura dos dados lógicos que ela suporta;
- .grau de confiabilidade e eficiência necessário;
- .baixo custo.

Note-se que muitas destas características são incompatíveis entre si. Por exemplo, sistemas de alta densidade de armazenamento (pequeno espaço físico ocupado) requerem maiores tempos de busca (tempo de operação maior) (fig 2.11.).

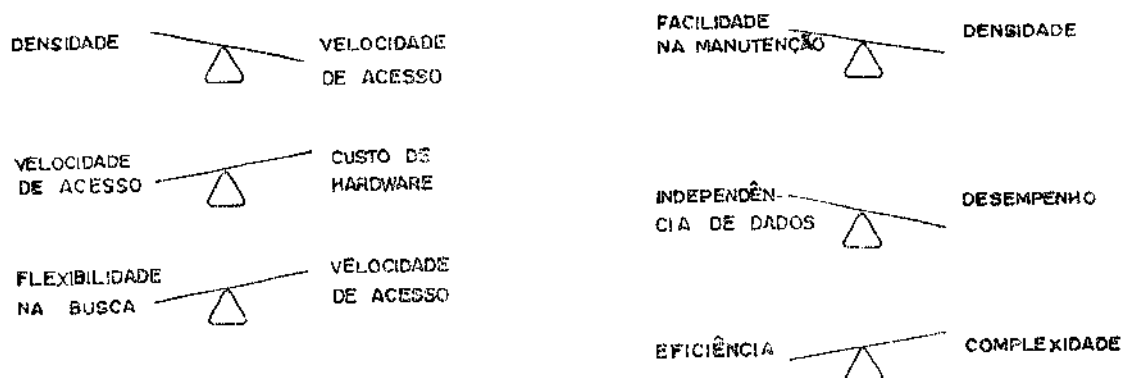


FIG 2.11: Características antagônicas de uma estrutura física

2.5. MODELOS DE DADOS

Um MODELO DE DADOS é:

- .um conjunto de diretivas que padroniza a forma de representação da estrutura lógica dos dados contidos na Base de Dados /TSIC 77/;

- .a visão que os usuários têm sobre a Base de Dados /DATE 74/;

- .uma ferramenta para os usuários descreverem precisamente a sua percepção do Mundo Real /SMIT 80/.

Note-se que o termo Modelo de Dados é usado indistintamente entre os autores para referir o Modelo de Dados correspondente aos Níveis Descritivo (item 2.4.2) e Lógico (item 2.4.3.).

A utilização de Modelos próximos à concepção do Mundo Real (programas de aplicação) é mais natural aos usuários não especializados em computação. A transformação desta concepção do Mundo Real em dados físicos armazenados na Base de Dados pode ser realizada automaticamente pelo SGBD ou através do Administrador de Banco de Dados (ABD), que mapeia manualmente as especificações formais dos usuários (Modelos do Nível Descritivo) em Modelos de Dados suportados pelo SGBD /KAHN 80/.

O conjunto de formalizações que permite a estruturação e manipulação dos dados conforme a forma estabelecida pelo Modelo de Dados é denominado LINGUAGEM DE DADOS. Esta linguagem se classifica em três grupos /TSIC 77/:

- .Linguagem de definição de dados (LDD);
- .Linguagem de manipulação de dados (LMD);
- .Linguagem de definição de armazenamento (LDA).

Uma LINGUAGEM DE DEFINIÇÃO DE DADOS descreve e define a estrutura dos dados relevantes aos programas de aplicação. Esta linguagem deve comportar algumas das seguintes capacidades de:

- .especificar o modelo de dados, segundo o qual

a Base de Dados é organizada pelo SGBD (se vários modelos estiverem disponíveis no Sistema);

.atribuir identidade à Base de Dados e especificar as unidades lógicas que a comporão como, por exemplo, entidades, atributos, etc;

.identificar e descrever as associações que existem entre as unidades lógicas dos dados;

.especificar os atributos dos tipos de entidades que são utilizados como chaves de identificação;

.especificar as restrições de integridade e/ou consistência;

.especificar as regras de acesso; etc.

O grupo de operações para manusear os dados organizados na Base de Dados é exprimido através da LINGUAGEM DE MANIPULAÇÃO DE DADOS. A operação básica de manipulação é a SELEÇÃO DE DADOS /TSIC 77/, que consiste em um conjunto de procedimentos para identificar, localizar e isolar os dados necessários para efetuar neles operações, tais como, inserção, remoção, atualização, etc. Distinguem-se dois tipos de critérios utilizados pelo SGBD para selecionar logicamente os dados:

.endereçoamento pelo conteúdo: quando a seleção é feita através dos valores dos atributos (ou qualificações) das entidades. Por exemplo, seleção de todas as entidades do tipo de entidade <FORNO>, que tem como valor de atributo <ESTADO DE OPERAÇÃO=NORMAL>;

.endereçoamento pela associação: quando a seleção baseia-se nas associações (relacionamentos) estabelecidas entre os tipos de entidades. Por exemplo, obter os alarmes correspondentes às falhas do forno X, onde <ALARMES> e <FORNO> são classificados como tipos de entidades distintos.

Sob o ponto de vista de acesso físico dos dados, estes dois critérios de seleção se reduzem a um problema comum: SEARCHING (busca) /KNUT 73/, descrito formalmente como o mapeamento

$$M:v \rightarrow (i_1, i_2, i_3, \dots, i_n),$$

onde cada elemento i_j , $j \in N$, representa a entidade que se

associa ao valor 'v', através da correspondência M. Existem diferentes técnicas para implementar estes mapeamentos: listas invertidas, Quad-trees, B-trees, busca exaustiva, etc. As técnicas que são incluídas no SGBD são transparentes aos usuários comuns. Normalmente, cabe aos ABD's a responsabilidade de selecionar os recursos de busca mais apropriados e estabelecer a correspondência entre os comandos da LMD e os recursos escolhidos. Portanto, os aspectos da linguagem de manipulação de dados referente a um Modelo de Dados não indicam necessariamente os mecanismos de acesso físico/lógico.

É possível distinguir ainda em alguns modelos de dados uma terceira classe de comandos que descreve as estruturas de armazenamento físico, tais como: área de armazenamento, localidade dos registros lógicos, etc. Estes comandos são enquadrados na categoria de LINGUAGEM DE DEFINIÇÃO DE ARMAZENAMENTO (LDA), que é capaz de:

- .selecionar o dispositivo de armazenamento (s. existir mais de um disponível);
- .descrever o mapeamento entre a estrutura lógica e a estrutura física dos dados;
- .especificar os identificadores utilizados para acessar fisicamente as unidades lógicas dos dados;
- .especificar a ordenação física dos dados;
- .especificar o tipo de conversão dos formatos de dados. Por exemplo, binário -> inteiro.

A seguir, serão descritos sucintamente três modelos de dados que apresentam nível de abstração diferente: Modelo Rede, Modelo Relacional e Modelo Entidade-Relacionamento (fig 2.15.).

2.5.1. MODELO REDE (NETWORK)

MODELO REDE é o modelo de dados que, dentre os que serão apresentados, aproxima-se mais do Mundo Computacional, ou seja, da estrutura organizacional física dos dados. A proposta clássica do Modelo Rede originou-se do relatório do Grupo CODASYL do DBTG, em 1971, cujos conceitos e definições servirão de base de descrição do Modelo Rede que se segue.

No Modelo Rede, distinguem-se dois primitivos: TIPOS DE REGISTROS e TIPOS DE CONJUNTOS. No mapeamento do Nível Descritivo ao Nível Lógico, normalmente os TIPOS DE REGISTROS são utilizados para representar os tipos de entidades, definidos pelos usuários (item 2.4.1.1), e os TIPOS DE CONJUNTOS para representar as associações entre os tipos de entidades.

Os TIPOS DE REGISTROS são caracterizados pelos ITENS DE DADOS. Os valores (OCORRENCIA DOS ITENS DE DADOS) assumidos pelos itens definem unidades distintas, denominadas OCORRENCIAS DE REGISTRO, de um mesmo tipo de registro. Um grupo de valores de item que define univocamente ocorrências de um tipo de registro é designado CHAVE do tipo de registro.

Por exemplo, <LAMINADOR> é um tipo de registro, que pode ser caracterizado pelos seguintes itens: capacidade, precisão de ajuste, fabricante, n.^o da série, etc. Um grupo de valores atribuídos a estes itens define uma ocorrência (um laminador particular) do tipo de registro <LAMINADOR>. Como a associação entre os valores do 'n.^o da série' e as ocorrências do <LAMINADOR> é unívoca, o 'n.^o da série' é um atributo candidato a chave do tipo de registro <LAMINADOR>.

Neste Modelo, a correspondência fx entre dois tipos de registros distintos X e Y , tal que $fx: X \rightarrow Y$ é denominada TIPO DE CONJUNTO, onde X é considerado TIPO DE REGISTRO PROPRIETARIO (owner) e Y , TIPO DE REGISTRO MEMBRO (member). A associação definida no Modelo Rede representa a abstração da conexão física existente entre os dados na Base de Dados. Alguns autores como, por exemplo /TSIC 77/, consideram tipos de conjuntos como ELOS explicitamente definidos, ligando diferentes tipos de registros. Para obter os registros de seu interesse, os usuários são obrigados a seguir os caminhos estabelecidos entre os registros pelos elos na Base de Dados. Este tipo de acesso é conhecido informalmente como NAVEGAÇÃO - processo em que se seguem explicitamente os caminhos de acesso para obter as informações desejadas.

Cada associação fx entre a ocorrência X do

tipo de registro proprietário X e um conjunto de ocorrências do tipo de registro-membro Y, y_i (este conjunto deve ter quantidade de elementos ≥ 0) é designada uma OCORRENCIA DO CONJUNTO, devendo satisfazer as seguintes regras de correspondência /TAYL 76/:

1. dado um registro-proprietário, é possível acessar os registros-membros;

2. dado um registro-membro, é possível obter o seu registro-proprietário;

3. dado um registro-membro, é possível obter os outros registros-membros pertencentes à mesma ocorrência do tipo de conjunto;

4. cada registro-membro só pode estar associado a um registro-proprietário em um tipo de conjunto.

O Modelo Rede suporta os três tipos de relacionamentos: 1:1, 1:N, M:N. Observe-se que o relacionamento M:N entre registros de diferentes tipos de registros violaria a regra (4), pois um registro-membro poderia estar associado a diferentes registros-proprietários. Este problema é contornado no Modelo CODASYL, definindo-se um terceiro tipo de registro - o PSEUDO-REGISTRO /FUSA 79/ ou TIPO DE REGISTRO RELACIONAMENTO. Com este tipo de registro, é possível decompor um relacionamento original M:N em dois relacionamentos 1:M e 1:N /TAYL 76/ (fig 2.12.).

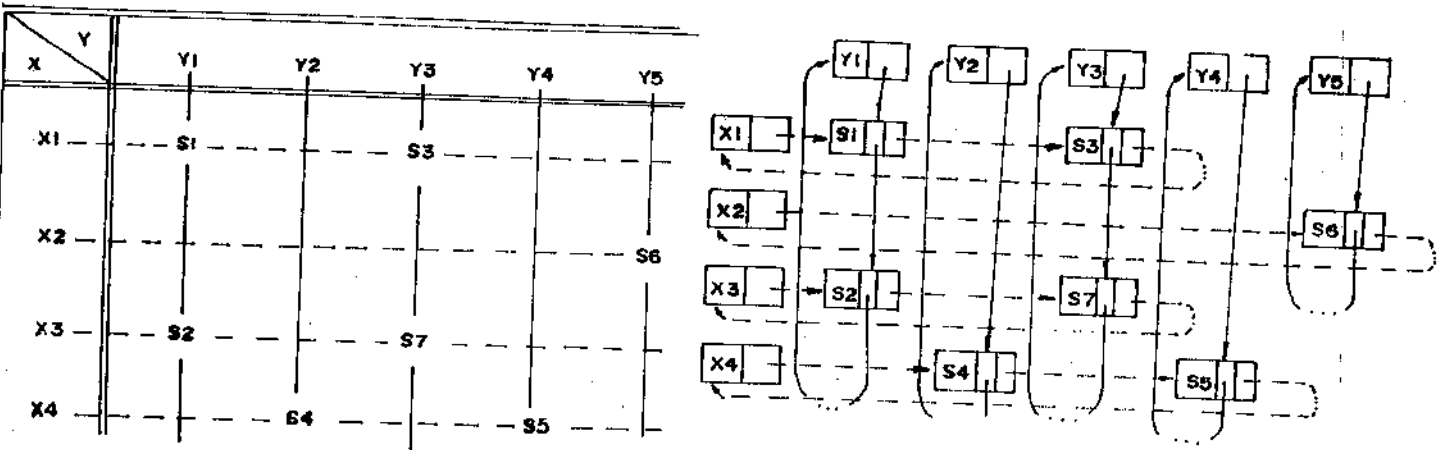


FIG 2.12: Exemplo de relação entre os tipos de registros X e Y, através da definição dos pseudo registros S

Relação:
 tipo de registro $X = \{x1, x2, x3, x4\}$ e
 tipo de registro $Y = \{y1, y2, y3, y4, y5\}$

O tipo de registro relacionamento é usado também para solucionar problemas referentes às estruturas recursivas, onde os registros-proprietários e membros pertencem ao mesmo tipo de registro, como por exemplo, a composição de peças formadas por peças componentes que, por sua vez, são formadas por outras peças componentes, etc. Neste caso, pode-se decompor o relacionamento entre os registros do mesmo tipo de registro em dois relacionamentos entre dois tipos de registros diferentes - o tipo de registro original e o tipo de registro relacionamento (pseudo-registro) criado /TAYL 76/ (fig 2.13.).

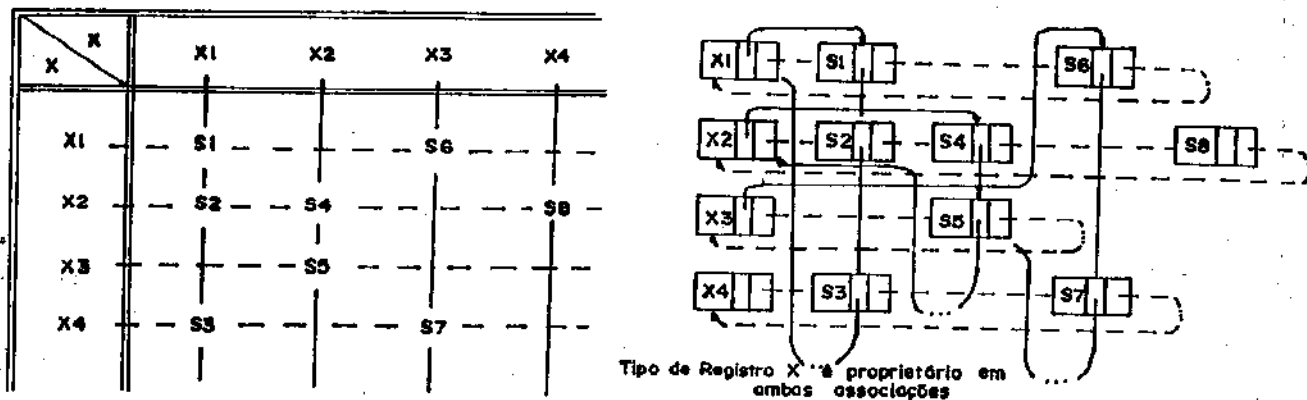


FIG 2.13: Exemplo de relação recursiva entre os elementos de um mesmo tipo de registro $X = \{x1, x2, x3, x4\}$ utilizando pseudo-registros S

2.5.1.1. LINGUAGEM DE DADOS DO MODELO REDE

Como o Modelo Rede apresenta um grau de abstração que se aproxima do nível de organização lógica dos dados, são incluídos explicitamente nas suas linguagens de dados comandos referentes aos pormenores dos acessos que foram estabelecidos entre os registros para obter as informações desejadas, tais como: FIND NEXT, GET NEXT, GET, GET...VIA..., MOVE...TO..., etc. Deste modo, a eficiência das operações de manipulação programadas, usando os recursos disponíveis na LMD, dependerá em grande parte do conhecimento dos usuários sobre os detalhes da estruturação lógica do Banco de Dados.

No Modelo Rede, conforme o grau de privilégio concedido, o usuário poderá intervir na organização física e endereçamento lógico dos dados, através de comandos específicos da LDA, tais como: LOCATION MODE, AREA, ORDER, etc. Para assegurar a privacidade dos dados de cada usuário, a linguagem prevê-se do comando PRIVACY, que deve ser declarado junto com os outros comandos da LDD, de modo que os dados protegidos, só poderão ser acessados e/ou manipulados, quando for fornecido corretamente o código de PRIVACY /ULLM 80/.

Com a LDD, o ABD define e declara todos os tipos de dados (tipos de registros com seus respectivos atributos, tipos de conjunto) que comporão a Base de Dados. A visão global do ABD de todas as descrições referentes aos dados contidas na Base de Dados é conhecida como ESQUEMA do Modelo e a visão parcial, correspondente aos dados de interesse aos usuários, é chamado SUBESQUEMA.

2.5.2. MODELO RELACIONAL

O Modelo Relacional é um modelo baseado na teoria matemática de relações para representar as associações tanto entre os atributos das entidades de um tipo de entidades como entre os diferentes tipos de entidades. Por exemplo,

(FABRICANTE, N.^o DA SERIE, CAPACIDADE, CARGA MAXIMA) é uma associação dos atributos do tipo de entidade <MOTOR> e (ACIONADOR, TIPO DE CARGA, MOTOR) seria um exemplo da associação entre três tipos de entidades: MOTOR, CARGA e PESSOA.

Matematicamente, as relações podem ser definidas como segue /DATE 74/:

Dado um conjunto de domínios $C_1, C_2, \dots, C_n$ não necessariamente distintos, define-se como uma RELAÇÃO R sobre os n conjuntos, o conjunto de n tuplas, ou simplesmente TUPLAS,

$\langle d_1, d_2, \dots, d_n \rangle,$

onde $d_1 \in C_1, d_2 \in C_2, \dots, d_n \in C_n$.

Em outras palavras, uma relação é um subconjunto do produto cartesiano de 'n' conjuntos $C_1 \times C_2 \times \dots \times C_n$, onde $n \geq 1$.

Fazendo uma analogia grosseira com o Modelo Rede, uma tupla corresponderia a uma ocorrência de um tipo de registro e uma relação corresponderia tanto a um tipo de registro como a um tipo de conjunto. Enquanto no Modelo Rede faz-se a distinção entre as entidades (registros) e os elos que as interligam logicamente na Base de Dados, no Modelo Relacional todas as associações, tanto entre os diferentes tipos das entidades como entre os atributos das entidades, são representadas indistintamente como relações. Para evitar a ambiguidade na conceituação, são usadas diferentes terminologias nos dois Modelos /ULLM 80/.

Cada RELAÇÃO é um conjunto de tuplas definidas em um conjunto de 'n' domínios $C_1, C_2, \dots, C_n$, não necessariamente distintos. A quantidade 'n' define o GRAU da relação R e a quantidade de tuplas define a sua CARDINALIDADE /CHAM 76/. Os valores que definem cada elemento de uma tupla são chamados COMPONENTES da tupla. Um conjunto de componentes referentes a uma tupla define uma OCORRÊNCIA. Graficamente, uma relação R n-ária pode ser representada como uma tabela. Esta representação apresenta as seguintes propriedades /CODD 70/:

.cada linha representa uma tupla da relação R (ocorrência);

- .a ordem das linhas é irrelevante;
- .todas as linhas são distintas, ou seja, pelo menos um componente de uma linha é diferente dos componentes das outras linhas;
- .a quantidade de colunas representa o grau da relação;
- .a quantidade de linhas representa a cardinalidade da relação;
- .cada coluna corresponde a um domínio da relação.

Distinguem-se dois tipos de relações: RELAÇÃO DE DOMÍNIOS ORDENADOS e RELAÇÃO DE DOMÍNIOS NÃO ORDENADOS. No primeiro grupo, a ordem das colunas é relevante, pois corresponde à ordem dos conjuntos $C_1, C_2, \dots, C_n$, sobre os quais a relação R é definida. Assim, para uma relação R definida em $C_1, C_2, \dots, C_n$, a tupla $\langle d_1, d_2, \dots, d_n \rangle$ pertencente a R significa que $d_1 \in C_1, d_2 \in C_2, \dots, d_n \in C_n$. Entretanto, se cada coluna for referida pelo nome do domínio, e não pela sua posição relativa dentro da relação, a ordem das colunas passa a ser irrelevante.

Quando dois ou mais atributos são definidos em um domínio comum numa mesma relação R , ou seja, R definido em $C_1, C_2, \dots, C_n$ e existe $C_i = C_j, i \neq j$ e $1 \leq i, j \leq n$, devem-se distinguir as diferentes funções ou papéis (roles) desempenhadas pelo mesmo domínio para que a correspondência entre a coluna (domínio) e a identificação do domínio sobre o qual ela é mapeada seja unívoca.

Por exemplo, para uma relação REL definida em três domínios: Quantidade, Quantidade, Componente $\langle C \rangle$, onde o domínio Quantidade desempenha duas funções: quantidade de refugos Q-R e quantidade de produtos reaproveitáveis Q-PR, qualquer tupla pertencente a ela poderá ser definida de modo inequívoco através de $(v_1 \in C, v_2 \in Q-R, v_3 \in Q-PR)$.

A coluna ou conjunto de colunas (domínios), cujos valores identificam univocamente cada elemento (tupla) em uma relação é denominada CHAVE-CANDIDATA. É possível que uma relação tenha duas ou mais chaves-candidatas. Neste caso, seleciona-se arbitrariamente uma destas chaves como a CHAVE

PRIMARIA, ou simplesmente CHAVE, da relação /CODD 71/.

No Modelo Relacional, não há declaração explícita das associações entre diferentes relações (lembrando-se que nele só existe basicamente um primitivo: relação). A referência das tuplas de uma relação às tuplas de uma outra (ou a mesma) relação é feita através da REFERENCIA CRUZADA, usando uma CHAVE ESTRANGEIRA (foreign key).

Denomina-se uma chave, CHAVE ESTRANGEIRA da relação R na referência à relação S, se ela não for a chave primária da relação R, mas seus elementos são valores da chave primária da relação S, de tal sorte que, a partir dela pode-se acessar os valores dos atributos definidos na relação S, associando o elemento da relação R ao elemento da relação S. Por exemplo, na relação R (COMPONENTE, FABRICANTE), S (FABRICANTE, ESTADO), a chave estrangeira da relação R na referência à relação S é FABRICANTE.

2.5.2.1. LINGUAGEM DE DADOS DO MODELO RELACIONAL

O Modelo Relacional é um modelo que apresenta um grau de abstração de dados maior que o Modelo Rede, pois nele é suposto que os ABD's não precisam ter conhecimento dos detalhes de endereçamento lógico dos dados para acessá-los. Por exemplo, os procedimentos de navegação citados no item 2.5.1.1 ficam transparentes aos seus usuários. Assim, a sua linguagem de dados é menos procedural que a do Modelo Rede e muitos autores preferem classificá-la como linguagem de inquerição (query language) /ULLM 80//FRY 76/. Destacam-se três tipos clássicos de representação desta linguagem:

.ALGEBRA RELACIONAL: consiste em uma coleção de operadores que manipula as relações para criar novas relações, isto é, as tuplas buscadas podem ser consideradas como uma nova relação que resulta de uma sequência ordenada de operações em um conjunto de relações já existentes. Os principais operadores da álgebra relacional são: PROJEÇÃO, PERMUTAÇÃO, UNIAO, COMPOSIÇÃO e RESTRIÇÃO /CODD 70/. Como exemplo da linguagem de álgebra relacional, tem-se ISBL /ULLM 80/;

.CALCULO RELACIONAL SOBRE TUPLAS: a operação

entre as n-tuplas armazenadas na Base de Dados faz-se através da especificação das qualificações (valores dos atributos) que estas tuplas devem satisfazer, utilizando operadores condicionais, tais como: $\langle, \wedge, \vee, \neg, =, \neq, \theta, \exists$, etc. Deste modo, as linguagens baseadas em cálculo relacional apresentam um nível de abstração maior que as de álgebra relacional, porque, na linguagem de álgebra relacional, a ordem das operações a serem efetuadas para obter uma informação é definida pelos usuários e na linguagem de cálculo relacional, isso fica a cargo de um compilador ou interpretador. Pode-se provar, porém, que a linguagem de cálculo relacional e a de álgebra relacional são equivalentes /ULLM 80/. Das linguagens de cálculo relacional, citam-se ALPHA, QUEL /ULLM 80/;

.CALCULO RELACIONAL SOBRE DOMINIOS: ao invés de aplicar os operadores condicionais sobre as tuplas, aplicam-se estes operadores sobre os domínios, em que são definidas as relações. Há implementações desta linguagem em forma gráfica, isto é, as relações são representadas no display como tabelas, onde os usuários preenchem os valores conhecidos e marcam os locais (domínios), onde esperam que os valores desejados sejam retornados. Por exemplo, Query-by-example /ULLM 80/.

Além destes três tipos de linguagens, existem linguagens intermediárias entre linguagens procedurais (álgebra relacional) e linguagens não procedurais (cálculo relacional). As linguagens intermediárias provêm-se de recursos de cálculo relacional (como, selecionar e comparar as tuplas existentes na Base de Dados através das qualificações, ao invés de, através da especificação de uma sequência de operações aritméticas, como ocorre na de álgebra relacional) e de álgebra relacional (como, operações entre relações distintas: união, intersecção, etc). Por exemplo, na linguagem SEQUEL /ULLM 79/, no comando `SELECT...FROM...WHERE...`, a palavra $\langle WHERE \rangle$ antecede os requisitos impostos pelos usuários; estes requisitos são definidos, utilizando operadores condicionais ($\langle, \rangle, =$, etc.) ou operadores de conjunto (união, intersecção). Enquadra-se, também, nesta categoria de linguagem, a linguagem DBASE /DBAS 80/;

A descrição do universo de dados armazenados na Base de Dados, utilizando a LDD, é mais simples no Modelo Relacional que no Modelo Rede, pois é suposto que o SGBD se

encarregaria de todas as operações referentes à organização física e lógica dos dados, de modo que as funções dos ABD's se restringiriam à declaração dos tipos de dados que serão armazenados e a definição de alguns detalhes específicos de aplicação. Isso apresenta aspectos positivos e negativos: a vantagem é que os ABD's não precisam ter grandes conhecimentos da Base de Dados e a desvantagem é que eles não tem recursos suficientes para interferir na estruturação dos dados e melhorar o seu desempenho.

No Modelo Relacional, diferentes relações (conjunto de tuplas definidas em diferentes domínios) podem ser declaradas. Algumas destas relações tem existência própria, pois, constituem os dados operacionais essenciais. Tais relações são denominadas **RELAÇÕES PRIMÁRIAS**. As relações que são derivadas das operações relacionais (por exemplo, através do cálculo relacional ou de álgebra relacional), a partir das relações primárias são consideradas **RELAÇÕES DERIVADAS** /TSIC 77/. Estas relações correspondem às visões particulares que determinados usuários tem sobre a Base de Dados, num determinado instante. Uma vez definidas, estas relações passam a ter existência e sua vida útil dependerá dos programas de aplicação.

As alterações que um usuário particular introduz nas suas relações (derivadas) não afetam diretamente as relações projetadas para os outros programas de aplicação. Por exemplo:

.na operação remoção: os domínios das relações, as tuplas das relações ou as próprias relações não são realmente removidas da Base de Dados; mas sim, são redefinidas pelo SGBD as novas relações derivadas, usando os operadores relacionais;

.na operação inserção: os novos dados que são acrescentados à Base de Dados só serão percebidos pelos programas de aplicação que usam as relações definidas sobre estes dados;

.na operação atualização: os dados modificados são automaticamente comunicados a todos os usuários que os utilizem, pois, a alteração dos dados corresponde a modificação dos dados operacionais essenciais, sobre os quais são geradas as relações derivadas. Portanto, pode-se considerar as relações

derivadas como IMAGENS das relações primárias são obtidas a partir das operações relacionais sobre os dados essenciais em qualquer instante de tempo.

Estas imagens que definem as visões dos usuários particulares são denominadas DEFINIÇÃO DE SUBMODELO DE DADOS, enquanto a visão global das relações primárias é conhecida como DEFINIÇÃO DE MODELO DE DADOS. Note-se que existe uma diferença entre a definição de esquema/subesquema do Modelo Rede e a de modelo/submodelo de dados do Modelo Relacional /ULLM 80/. No Modelo Rede, o subesquema é um SUBCONJUNTO do esquema, apresentando os mesmos comandos de manipulação e definição, inclusive os comandos referentes a estruturação de armazenamento, tais como LOCATION MODE, elos de acesso, etc. Assim, modificações introduzidas por um usuário no seu subesquema podem afetar diretamente o esquema definido e refletir-se nos outros subesquemas, demandando possivelmente reestruturação dos programas de aplicação já existentes. No Modelo Relacional, como já visto, existe uma interface nítida entre a definição do Submodelo de Dados e a definição do Modelo de Dados, de modo que as manipulações ocorridas em cada programa de aplicação não interfiram necessariamente nas relações primárias, e estas, por sua vez, tem suas alterações somente percebidas pelos programas de aplicação que as acessem.

2.5.2.2. NORMALIZAÇÃO

O projeto da estruturação dos dados, segundo o modelo relacional, baseia-se na TEORIA DE NORMALIZAÇÃO, que consiste em um conjunto de diretivas para escolher dentre as possíveis alternativas dos esquemas (ou definição do modelo de dados) do Banco de Dados, aquela que apresenta propriedades mais favoráveis: menor redundância, menor probabilidade de ocorrência de anomalias de atualização, inserção, remoção, etc /ULLM 80/. A teoria se fundamenta na redução do universo de relações ao conjunto de relações normalizadas. As formas mais conhecidas de normalização são: primeira, segunda, terceira e quarta formas normais /CODD 71//FAGI 77//ULLM 80/ (fig 2.14.).

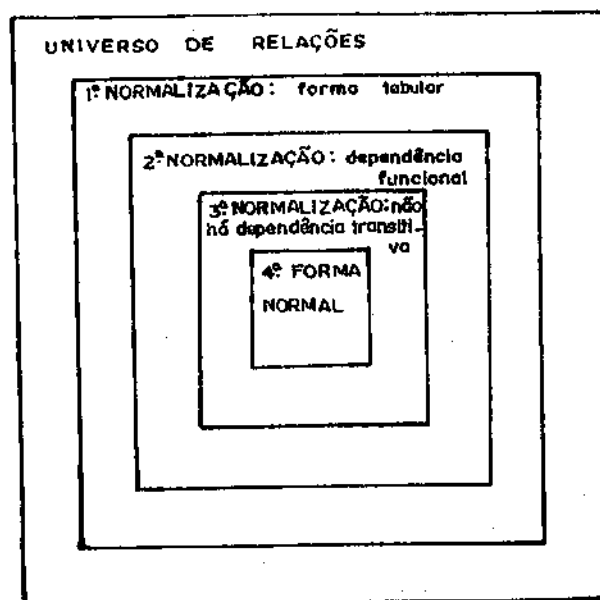


FIG 2.14: Quatro formas normais

Uma relação está na 1.^a FORMA DE NORMALIZAÇÃO, quando todos os componentes das tuplas das relações são atômicas e indivisíveis, isto é, os componentes não são listas de valores nem outras relações /CHAM 76/ e as relações podem ser representadas como uma tabela bidimensional.

Uma relação está na 2.^a FORMA DE NORMALIZAÇÃO, se e somente se, estiver na 1.^a forma de normalização e a correspondência entre os domínios dos atributos que não forem candidatos à chave primária e os candidatos é sobrejetiva, isto é, na correspondência $f_X: X \rightarrow Y$, os domínios Y não incluídos na chave são funcionalmente dependentes dos domínios X pertencentes à chave.

Uma relação está na 3.^a FORMA DE NORMALIZAÇÃO, se e somente se, estiver na 2.^a forma de normalização e não existir nenhuma correspondência bijetiva entre os domínios dos atributos que não forem candidatos à chave primária, isto é, os domínios não incluídos na chave NÃO SÃO TRANSITIVAMENTE DEPENDENTES DA CHAVE (não são mutuamente dependentes).

Uma relação está na 4.^a FORMA DE NORMALIZAÇÃO ou FORMA NORMAL DE BOYCE-CODD GENERALIZADA, se e somente se, estiver na 3.^a forma normal e não houver nenhuma dependência

multivalor entre os domínios não incluídos na chave, ou seja, não há nenhuma correspondência sobrejetiva entre dois subconjuntos dos domínios não incluídos na chave.

2.5.3. MODELO ENTIDADE-RELACIONAMENTO

O Modelo Entidade-Relacionamento enquadra-se no conjunto de Modelos de Dados que apresentam grau de abstração correspondente ao nível Descritivo. Ele dá maior ênfase à definição da semântica de informações, direcionando os usuários a definir com precisão e numa forma natural, emrela à sua concepção usual, os seus requisitos.

No MODELO ENTIDADE-RELACIONAMENTO distinguem-se dois grupos de primitivos: ENTIDADES e RELACIONAMENTOS. A ENTIDADE é um objeto que tem existência própria. Um grupo de entidades similares (correspondentes às ocorrências dos tipos de entidades do Modelo Rede), ou seja, que apresentem propriedades comuns, é denominado CONJUNTO DE ENTIDADES. O RELACIONAMENTO define a associação entre as entidades por meio das FUNÇÕES ou PAPEIS (role) desempenhadas por elas nesta associação. O CONJUNTO DE RELACIONAMENTO define todas as associações estabelecidas entre 'n' (n≥1) entidades, cada qual pertencente a um conjunto de entidades, não necessariamente distinto.

Conceitualmente, o relacionamento definido no Modelo Rede significa a conexão lógica que se estabelece entre os registros. Por sua vez, no Modelo Entidade-Relacionamento, um relacionamento é mais que uma conexão lógica, ele deve ser considerado como um elemento concreto com existência própria, da mesma forma que as entidades, isto é, sem necessariamente estar estabelecendo associação entre entidades.

Tanto as entidades quanto os relacionamentos podem ser caracterizados pelos valores de qualificações, através do mapeamento via FUNÇÃO-ATRIBUTO. O grupo de funções-atributo que estabelece a correspondência unívoca entre as entidades/relacionamentos de um conjunto de entidades/relacionamentos e os valores, que as/os qualifiquem, é denominado a CHAVE do

conjunto.

Por exemplo, MOTOR e CARGA podem ser consideradas entidades e ACIONAMENTO, o relacionamento que as associa. Cada um pode ser caracterizado por um conjunto de funções-atributo, tais como:

.N.^o de série, fabricante, tipo, etc para MOTOR;

.N.^o de série, tipo, peso, etc, para CARGA;

.chaves das entidades associadas, tempo, desempenho, etc, para ACIONAMENTO.

Como o atributo (n.^o de série) estabelece a correspondência unívoca entre o valor do domínio (número), e <MOTOR> ou <CARGA>, esta função-atributo é considerada a chave das entidades <MOTOR> e <CARGA>. Analogamente, pode-se considerar que <chave das entidades associadas> como a chave do relacionamento <ACIONAMENTO> desde que seja garantida a univocidade da correspondência entre os valores definidos no domínio <chave das entidades associadas> e ocorrências do <ACIONAMENTO>.

Um conceito importante introduzido neste Modelo é a definição do relacionamento sobre um conjunto de funções, ao invés da sua definição sobre os domínios de valores, como ocorre no Modelo Relacional. Dentre as vantagens desta conceituação, destacam-se:

.as funções desempenhadas pelas entidades num relacionamento são únicas, isto é, não há possibilidade da ocorrência de ambiguidade da interpretação de uma informação contida numa tupla;

.um conjunto de relacionamento pode estabelecer associações idênticas entre conjuntos de entidades distintos, desde que eles desempenhem as mesmas funções, o que se aproxima mais da concepção real, pois certas funções podem ser desempenhadas pelos diferentes conjuntos de entidades e vice-versa.

Uma outra noção que favorece o Modelo Entidade-Relacionamento é a classificação das entidades e relacionamentos em FRACO e REGULAR. Os dados regulares são

aqueles que tem existência própria, independentemente dos outros dados e cujo acesso pode ser feito diretamente pelas chaves; enquanto os dados fracos são aqueles cuja existência depende da existência de algum outro dado e, normalmente, a chave utilizada para acessá-los é composta da chave do seu proprietário e sua própria chave. A vantagem desta diferenciação é a facilidade do controle de consistência e integridade dos dados /CHEN 76/.

NÍVEL DE ABSTRAÇÃO \ MODELO		REDE	RELACIONAL	ENTIDADE RELACIONAMENTO
NÍVEL 1 CONCEITUAL				
NÍVEL 2 DESCRITIVO			4 ^o forma normal	tuplos de entidade- des tuplos de relacionamentos
NÍVEL 3 LÓGICO	independente do caminho de acesso		relações (tabelas)	
	dependente do caminho de acesso	registros/tipos de registros tipos de conjunto		
NÍVEL 4 FÍSICO		bloco, registros, área, etc	bloco, registros, área, etc	

FIG 2.15: Níveis de abstrações nos três diferentes modelos

2.5.4. RESUMO

Na fig 2.16. são representadas todas as terminologias apresentadas nos itens anteriores, mostrando ao mesmo tempo a equivalência entre elas /FRY 76/.

CONCEITOS	RELACIONAL	R E D E	ENTIDADE RELACIONAMEN- TO
A) ENTIDADES QUALIFICADOR VALOR DO {SIMPLES QUALIFICADOR {MÚLTIPLO TIPO DE ENTIDADES OCORRÊNCIA	função / domínio componente — relação tuplo	tipos de item de dados ocorrência grupo tipo de registros ocorrência do tipo de regis- tro	atributo / domínio valor — conjunto de entidades entidade (tupla)
B) ASSOCIAÇÕES FUNÇÕES ENTIDADES ASSOCIADAS ESTABELECIMENTO DE ASSOCIAÇÃO UNIDADE	— — chave estrangeira —	— — tipo de conjunto ocorrência do tipo de conjunto	função (papel) entidade conjunto de relacionamento relacionamento (tupla)
C) MODELAMENTO DE DADOS DEFINIÇÃO SOB VISÃO DO ADO DEFINIÇÃO SOB VISÃO DE USUÁRIOS	definição de modelo de dados definição do submodelo de dados	esquema subsquema	— —
D) IDENTIFICAÇÃO DE CADA ELEMENTO DA BA- SE DE DADOS	chave primária	chave	chave primária

FIG 2.16: Resumo das terminologias usadas nos três modelos

CAPITULO 3: SISTEMAS DE CONTROLE

3.1. INTRODUÇÃO:

Ao longo deste texto, SISTEMAS DE CONTROLE terão uma conotação abrangente, sendo definidos como:

..uma combinação de elementos agindo em um Sistema Físico, visando atingir o desempenho especificado.

Estes elementos são (fig 3.1.):

..os SISTEMAS DE DECISÃO, que procuram selecionar a melhor solução dentre as factíveis, atendendo, ao mesmo tempo, o objetivo pré-estabelecido;

..os SISTEMAS SENSORIAIS, que observam o comportamento do sistema isoladamente e/ou em relação ao seu meio-ambiente;

..os SISTEMAS ATUADORES, que realizam as ações de controle decididas.

Neste contexto, SISTEMA FÍSICO é qualquer objeto a ser controlado, independentemente de sua natureza, podendo ser, por exemplo, um processo de fabricação, a situação econômica de uma empresa, a alocação de pontos de abastecimento ou estocagem, o escalonamento dos horários de trabalho dos funcionários, a folha de pagamento, etc.

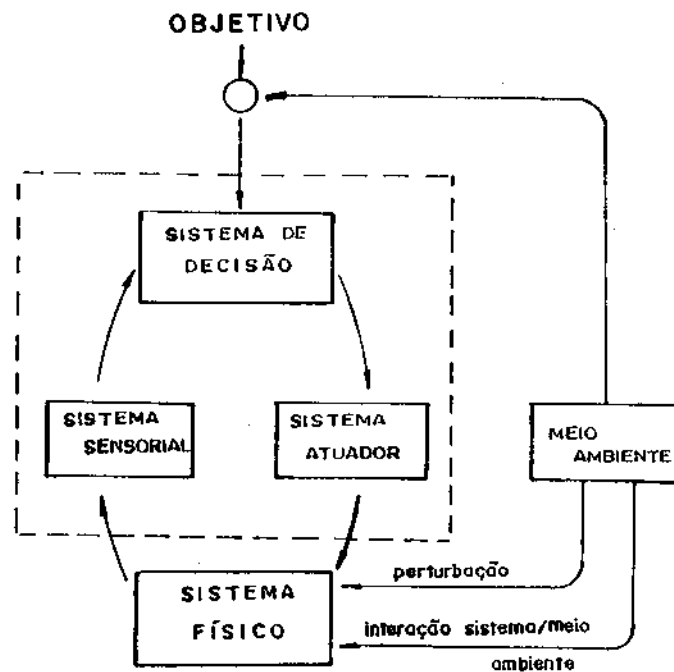


FIG 3.1: Elementos básicos de um Sistema de Controle

O SISTEMA DE DECISÃO, por sua vez, compreende dois componentes básicos /STRE 78/:

„FONTE DE INFORMAÇÃO: é constituído de uma coleção de dados sobre o Sistema Físico e de um conjunto de procedimentos e/ou programas responsável em mantê-lo num nível operacional satisfatório. Enquanto os dados referentes ao estado do sistema são coletados pelos sensores, os procedimentos de ações de controle, segundo os quais os atuadores agem, são normalmente estabelecidos pelas unidades de decisão;

„UNIDADE DE DECISÃO(UD): é a entidade que decide sobre os tipos de ações a serem executadas para que o desempenho do Sistema Físico se mantenha em um nível pré-estabelecido. Para isso, a UD precisa ter acesso às informações sobre o Sistema Físico. Quanto mais precisas e atualizadas forem estas informações, mais confiáveis serão as tomadas de decisão (fig 3.2.).

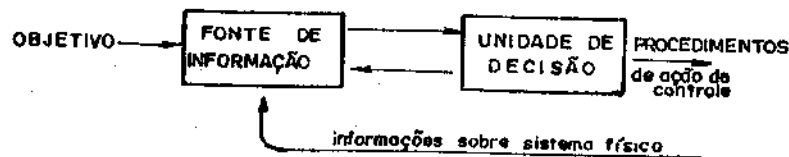


FIG 3.2: Sistema de decisão

Nas últimas décadas, as inovações tecnológicas possibilitaram o tratamento de sistemas cada vez mais complexos. As informações sobre o Sistema Físico tornaram-se volumosas, o que dificulta uma tomada de decisão precisa e rápida numa configuração de sistema de controle centralizado, pois todas as UD's tem um limite tolerável quanto à capacidade de assimilação de informações e à geração de respostas adequadas. A tendência atual é implementar sistemas de controle com a arquitetura DISTRIBUIDA e INTEGRADA /PAA 76/.

A filosofia de CONTROLE DISTRIBUIDO baseia-se na distribuição de responsabilidades de controle entre várias UD's. Assim, o campo de ação de cada UD se delimita a setores menores, o que envolve uma quantidade menor de informação e facilita a tomada de decisão. A divisão deve ser feita de tal modo que a interferência entre as sub-unidades seja mínima para poder demarcar bem as tarefas das UD's /IMTO 83/.

Entre as vantagens de uma arquitetura distribuída para o Sistema de Controle, citam-se:

- maior flexibilidade à automação das UD's, pois, partindo-se de um projeto de sistema de controle com funções decisórias corretamente distribuídas entre as UD's, pode-se automatizá-las gradativamente conforme a disponibilidade de recursos, com mínimas modificações no projeto inicial do sistema;

- maior disponibilidade, pois, mesmo que alguma UD falhe, o Sistema Físico como um todo poderá continuar operando, embora com uma certa degradação no seu desempenho global;

.melhor desempenho no que diz respeito ao tempo de resposta, uma vez que as unidades individuais de cada sistema podem ser controladas simultaneamente pelas diferentes UD's;

.maior facilidade de manutenção, pois um sistema distribuído é, normalmente, composto de unidades padronizadas menores e mais simples;

.baixo custo de operação e instalação, pois as UD's estarão mais próximas dos Sistemas Físicos, o que reduzirá o custo de transmissão de informações entre estes sistemas e a UD.

A idéia do CONTROLE INTEGRADO partiu do fato de que a ausência de intercâmbio de informações entre as UD's sobre as tomadas de decisão efetuadas causam sérios problemas na sintetização de um controle ótimo. É difícil que uma UD possa estimar os estados de outros subsistemas, baseando-se somente nas informações referentes ao seu campo de ação. Ela precisa de informações adicionais de outros subsistemas ou de um agente coordenador central para garantir que suas tomadas de decisão sejam compatíveis com o resto do sistema. Assim, numa configuração de sistema de controle com tarefas distribuídas, é imprescindível que exista INTEGRAÇÃO, de modo que, sob o ponto de vista operacional, o conjunto de UD's atue visando a um objetivo comum. Fisicamente, a integração pode ser implementada através de seguintes configurações (fig 3.3.):

.SISTEMAS DE BANCO DE DADOS, centralizando o gerenciamento e coordenação de fontes de informações das UD's /STRE 78/;

.interconexão física entre as UD's /WILL 74/;

.configuração mista ou combinação das duas configurações anteriores.

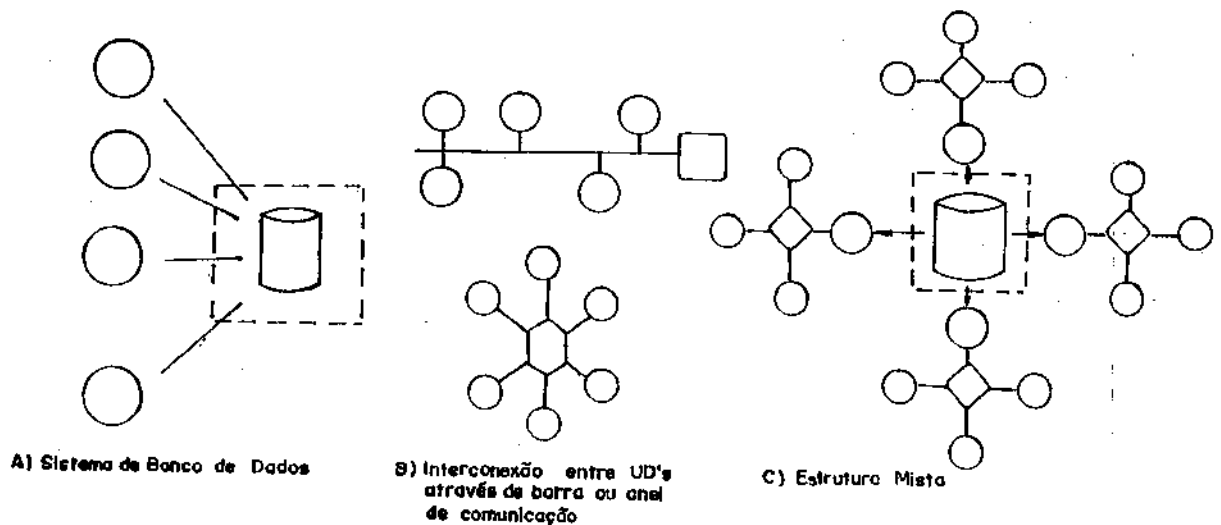


FIG 3.3: Diferentes configurações de integração das UD's

Independentemente da arquitetura selecionada, um sistema de controle moderno deve ser projetado visando às seguintes características:

.INTEGRAÇÃO: deve-se procurar integrar os diversos componentes de um sistema de controle, pois o desempenho e a eficiência globais são dependentes da comunicação e do grau de interação entre estes componentes;

.TEMPO REAL: a função controladora deve ter suas ações executadas dentro de um intervalo de tempo aceitável pelo Sistema Físico;

.FLEXIBILIDADE: quando ocorrerem perturbações imprevisíveis, deve ser sempre possível conduzir o sistema ao estado operacional desejado dentro de um intervalo de tempo compatível com o processo;

.PRODUTIVIDADE: deve aumentar o rendimento da produção;

.SIMPLICIDADE: para facilitar a manutenção e coordenação global do sistema;

.CUSTO: o investimento inicial deve ser justificado pelo desempenho do sistema implantado, tendo como meta a estratégia global estabelecida pela empresa;

.CONFIABILIDADE: assegurar que o Sistema Físico esteja sempre em regime operacional aceitável. Caso perturbações

sejam introduzidas no sistema, as UD's devem ser capazes de detectá-las e providenciar ações corretivas;

.TRANSPORTABILIDADE: procura-se usar módulos padronizados não orientados a aplicações específicas, o que facilita a manutenção;

.SERVIÇO CONTINUO: se a natureza de operação dos Sistemas Físicos for contínua, o seu sistema de controle deve ser ininterrupto, isto é, caso ocorra alguma falha na sua UD, devem haver recursos de retaguarda para garantir a continuidade operacional, mesmo com certa degradação no desempenho global;

.AUTOMAÇÃO: nas tomadas de decisão repetitivas e monótonas, procura-se evitar a intervenção humana que é muito susceptível a erros;

.INTERFACE NITIDA: devido ao crescente nível de automação, o computador está desempenhando, cada vez mais, um papel importante no sistema de controle. A interação homem-sistema tende a se ampliar para homem-computador-sistema, demandando, assim, uma interface amigável entre homem e computador.

3.2. HIERARQUIZAÇÃO DE SISTEMAS DE CONTROLE

Num sistema de controle distribuído e integrado, atribui-se a cada UD uma parcela da responsabilidade de controle, de tal modo que, se forem satisfeitas isoladamente as exigências para obter o desempenho local pré-estabelecido, a estratégia global almejada será alcançada. Dependendo do campo de ação das UD's, a natureza das informações necessárias às tomadas de decisão é diferente, pois uma mesma tarefa pode ser vista sob diferentes ângulos pelas distintas UD's.

Por exemplo, o processo <LAMINAÇÃO CONTINUA UNIDIRECIONAL A QUENTE> pode ser visto como um conjunto de laminadores subdivididos em seguintes unidades: forno de reaquecimento, trem contínuo, leito de resfriamento, corte a frio, bobinadeiras e trem de arame. Até a etapa 'trem contínuo', a linha de processo é comum aos produtos perfis, bobinas e arame. A partir daí, o material, dependendo do produto final, é

encaminhado a uma das seguintes linhas: leito de resfriamento, bobinadeira ou trem de arame /CSGM 93/. Para se obter um certo padrão de produto final (barras ou chapas), as variáveis, como velocidade do processo, temperatura e pressão impostas ao material, devem ser mantidas dentro de limites estabelecidos. Um outro enfoque é considerar-se (LAMINAÇÃO) como um subprocesso na produção de chapas de aço - conformação mecânica de lingôtes ou placas (lingotamento contínuo) - demandando, num certo intervalo de tempo, o trabalho-conjunto de uma certa quantidade de operadores e equipamentos para obter uma dada quantidade de chapas com a qualidade pré-estabelecida. Neste caso, as UD's devem ser capazes de manter um ritmo de produção alocando adequadamente os recursos disponíveis. E, finalmente, o processo (LAMINAÇÃO) pode ser visto como uma atividade do departamento de processo de uma indústria siderúrgica. Estes diferentes ângulos de visão podem ser organizados numa estrutura hierárquica como mostra a fig 3.4.

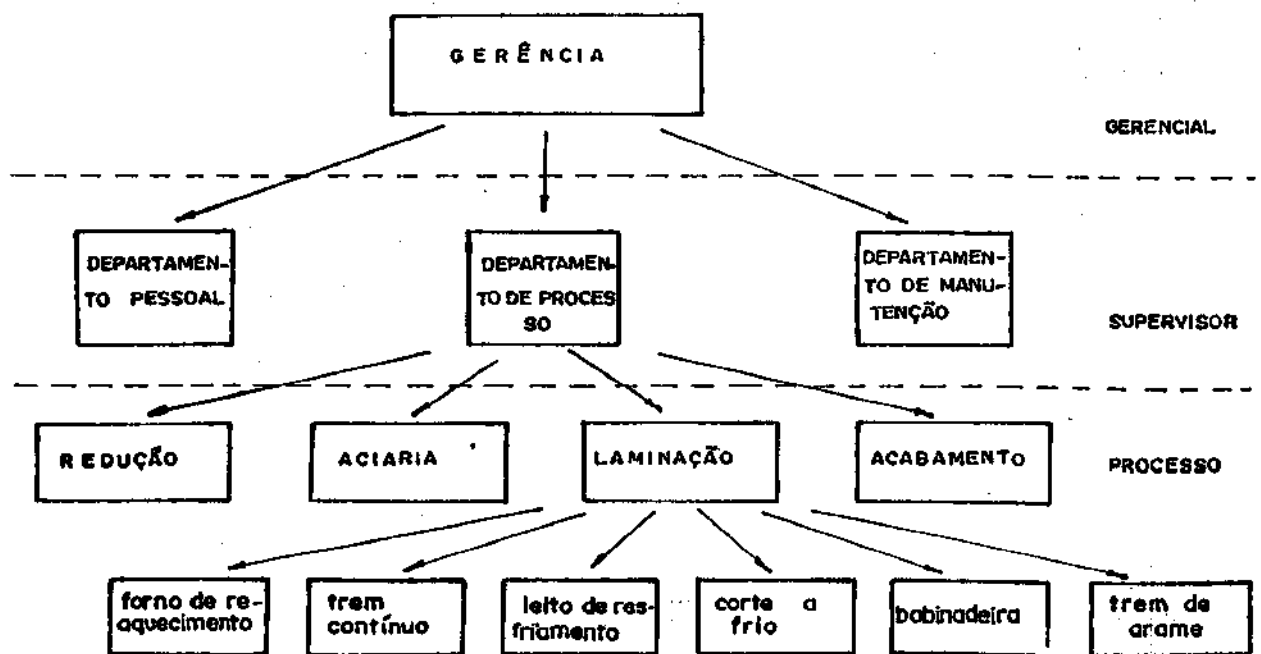


FIG 3.4: Diferentes visões do processo LAMINAÇÃO

Não há, ainda, um consenso geral sobre uma estrutura hierárquica padrão. Isto pode ser justificado pelo fato de que a Teoria de Controle e as inovações tecnológicas dos

equipamentos de controle não foram desenvolvidas em função de uma arquitetura de controle hierárquica básica, mas sim, tiveram seus desenvolvimentos independentes, e a automação foi implementada gradativamente, à medida que a complexidade do problema de controle foi sendo solucionada. No final da década de 60, surgiram as primeiras formalizações dos conceitos de um sistema hierárquico, culminando com o trabalho de Mesarovic e outros /MMTA 70/, no início da década de 70.

Dependendo do enfoque dado ao problema de controle, a classificação dos níveis numa estrutura hierárquica pode ser feita sob o ponto de vista funcional /JUZE 83//LEFK 77/, organizacional (automação através de uso de dispositivos digitais) /WILL 78/ e da natureza de informações necessárias para tomadas de decisões. Todas estas classificações são corretas, de acordo com a teoria de sistema hierárquico multinível.

Sob o ponto de vista de informações, destacam-se três níveis hierárquicos (fig 3.4.):

- . Nível Gerencial;
- . Nível Supervisor;
- . Nível de Processo.

3.2.1.NIVEL GERENCIAL

Este nível é responsável pela determinação da política global de uma empresa, levando em consideração a capacidade das instalações, a disponibilidade de matérias-primas e energia, a disponibilidade de recursos humanos, a situação do mercado, etc. Dentre as tarefas de elaboração da política global, citam-se /WILL 78/:

.estabelecimento e/ou reestruturação do cronograma básico do plano de trabalho da empresa, conforme a demanda do mercado e/ou a ocorrência de algum imprevisto;

.determinação do nível de estoque das matérias-primas, conforme a capacidade de instalação e o nível de produção da empresa;

.determinação do nível de estoque dos produtos finais de uma empresa, quando, por exemplo, há necessidade de um

fornecimento contínuo deste produto ao mercado;

.estabelecimento do nível máximo admissível de custo de produção, para que seja viável a concorrência dos produtos no mercado;

.estabelecimento do nível de qualidade dos produtos, etc..

Além da determinação da política global, que será usada para fixar os objetivos almejados para os níveis hierárquicos inferiores, o Nível Gerencial se encarrega também da produção de relatórios sobre a situação financeira, produtiva e pessoal da empresa. Estas informações são utilizadas pelos gerentes ou grupo de pessoas especializadas no assunto (denominados UD's), para avaliar o desempenho global e determinar as ações necessárias de controle fig 3.5. As decisões neste nível não têm características críticas no tempo, ou seja, têm uma constante de tempo elevada (dias, semanas, meses)..

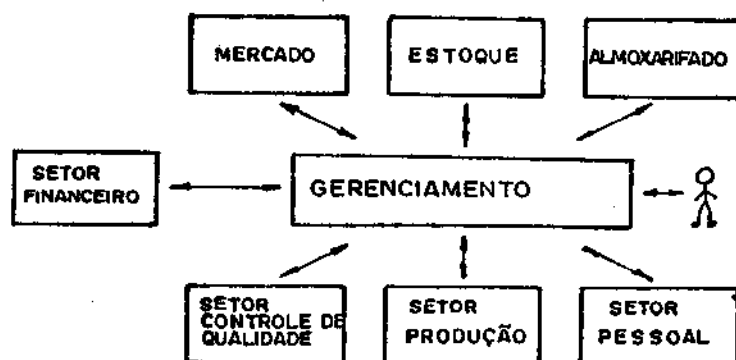


FIG 3.5:Nível Gerencial

3.2.2. NÍVEL SUPERVISOR

Normalmente, este nível é responsável pela coordenação e supervisão do desempenho de cada setor de uma organização industrial, baseado na política estabelecida pelo nível Gerencial, ou seja, o campo de ação das UD's se restringe a

um dos setores mostrados na fig 3.5. Sem perda de generalidade, será estudado, mais detalhadamente, o setor de produção. Para a execução das tarefas deste nível, as UD's devem ter acesso tanto às informações do Sistema Físico sob o seu controle como também às metas estabelecidas pelos níveis superiores.

Entre as funções atribuídas a este nível, destacam-se:

- .detalhar o cronograma estabelecido pelo nível gerencial e adaptá-lo conforme a situação real de cada setor;

- .estabelecer e otimizar o sequenciamento e alocação dos recursos disponíveis (tanto materiais como humanos), dentro das restrições impostas pelo planejamento básico do nível gerencial, para reduzir o custo operacional;

- .estabelecer as folgas de produção para que, quando ocorrerem imprevistos, não ocorra um atraso excessivo no ritmo de produção de toda a empresa;

- .supervisionar a operação correta dos sistemas de controle dos níveis inferiores, etc.

Neste nível, são também produzidas as informações sobre o histórico da produção de cada setor, tais como: quantidade e tipo de material utilizado no processo, o desempenho e a eficiência do setor, a quantidade e qualidade de produtos finais produzidos por turno de trabalho em cada setor e/ou subsetor, etc. Estas informações são utilizadas pelas UD's de cada setor ou subsetor para estabelecer decisões de controle dentro do seu campo de ação (fig 3.6.). As decisões deste nível, se comparadas com as do nível gerencial, são mais críticas no tempo pois influenciam a produção a curto prazo (constante de tempo da ordem de dezenas de minutos ou horas).

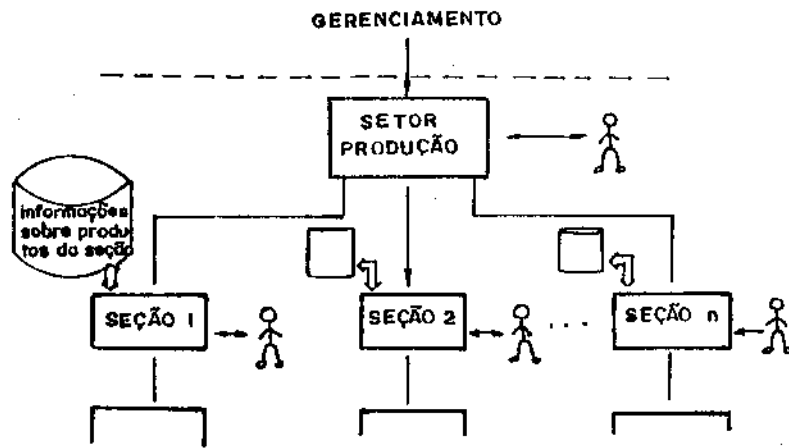


FIG 3.6: Nível Supervisor

3.2.3. NÍVEL DE PROCESSO

Este nível é responsável pelo controle direto do processo. Inclui as funções de aquisição de dados diretamente do processo físico e geração de ações de controle, objetivando-se manter os estados de operação dos Sistemas Físicos (por exemplo, equipamentos de produção) dentro de limites aceitáveis, estabelecidos conforme as estratégias de produção definidas nos níveis superiores. Dependendo do grau de automação da empresa, a participação humana neste nível pode ser mínima, ou seja, os Sistemas Físicos podem ficar ligados diretamente aos controladores digitais, de modo que os operadores humanos se restringirão a desempenhar o papel de monitor das operações e atuador em situações de emergência (fig 3.7.). Por isso mesmo, as decisões neste nível são críticas no tempo, já que influenciam imediatamente o processo em andamento (constante de tempo da ordem de minutos ou segundos).

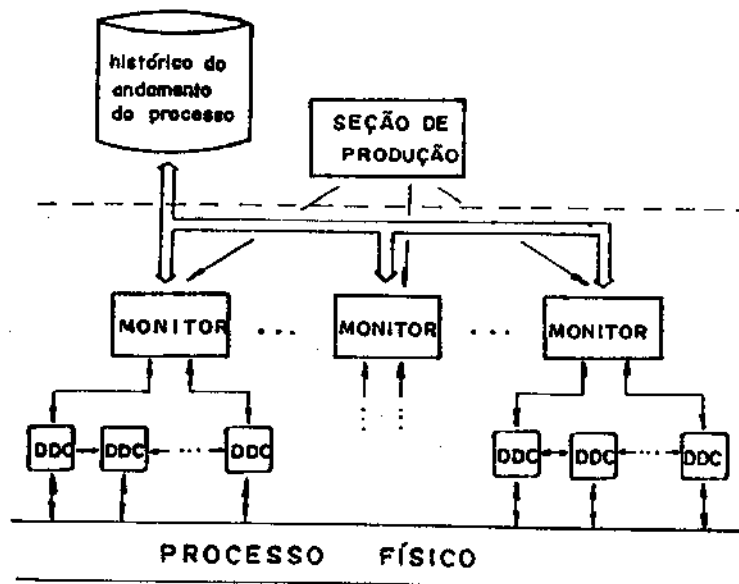


FIG 3.7: Nível de Processo

3.2.4. CAMADAS HIERARQUICAS DO NIVEL DE PROCESSO

Uma possível estrutura hierárquica é obtida distinguindo-se no nível de processo de um sistema de controle quatro camadas funcionais (fig 3.8.) /ATHA 78/:

- . implementação;
- . regulação;
- . adaptação;
- . otimização.

Esta classificação permite que um problema de controle de processo seja subdividido em diversos subproblemas mais simples e mais fáceis de serem resolvidos. Embora, teoricamente, também pudesse ser aplicada aos outros dois níveis - Gerencial e Supervisor - a utilização da Teoria de Controle moderna para análise, projeto e implementação de sistema de controle nestes dois níveis ainda se encontra em estágio embrionário e não será tratada aqui.

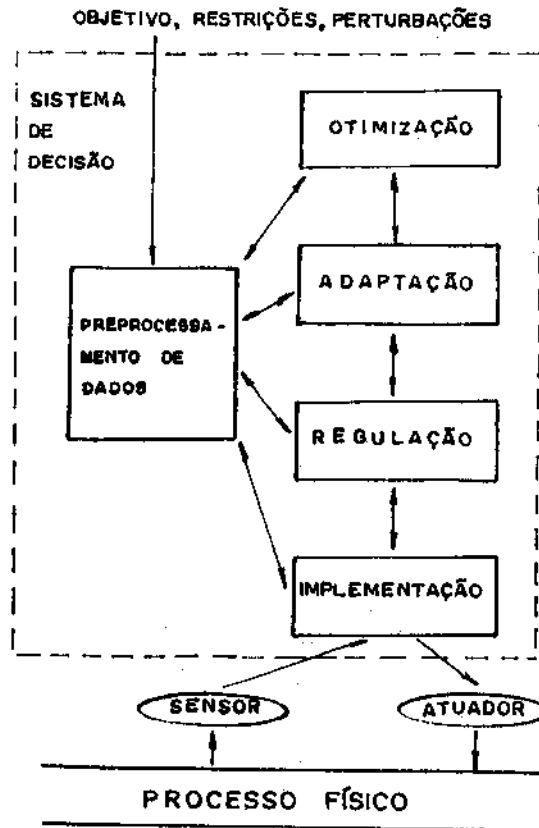


FIG 3.8: Quatro camadas hierárquicas no Nível de Processo

A camada de OTIMIZAÇÃO tem como finalidade traduzir e otimizar a função-objetivo estabelecida no nível hierárquico superior, levando em consideração as restrições técnicas e operacionais.

A camada de ADAPTAÇÃO é importante no controle do processo, principalmente quando a decisão otimizada foi tomada baseada num modelo simplificado do Sistema Físico, válido somente em determinadas condições e estados operacionais. Portanto, as variações que afetam o comportamento do sistema devem ser identificadas para introduzir as devidas correções no modelo simplificado /SAVA 65/.

A camada de REGULAÇÃO refere-se à seleção da melhor ação de controle, dentre as factíveis, que minimize os desvios do comportamento do processo em torno da função-objetivo estabelecida pela camada de Otimização e corrigida pela camada de Adaptação. A camada de IMPLEMENTAÇÃO corresponde à atuação das

ações de controle e à coleta de informações sobre o Sistema Físico.

Sob o ponto de vista de análise do comportamento do sistema, esta abstração de um mesmo sistema físico em diferentes camadas oferece o seguinte aspecto positivo: muitos aspectos dinâmicos e locais podem ser tratados nas camadas inferiores (por exemplo, de implementação), permitindo assim o uso de modelos estáticos e simplificados, para representar com fidelidade o sistema, nas camadas superiores (otimização, adaptação).

3.3. EVOLUÇÃO DOS SISTEMAS DE CONTROLE:

Antes da década de 60, a organização de um sistema industrial era dividida nitidamente em duas partes:

- . administração (gerencial e supervisão);
- . produção (processo).

Os primeiros conceitos de controle surgiram no campo de Engenharia e se aplicavam exclusivamente ao nível de Processo (controle dos equipamentos físicos). Os estudos de controle eram baseados na solução de equações diferenciais por intermédio de técnicas clássicas. A não ser para casos de sistemas simples, a análise segundo esta abordagem é trabalhosa e não oferece recursos eficientes para o estudo do comportamento dinâmico dos sistemas. Portanto, os primeiros controladores tiveram suas aplicações limitadas a sistemas simples, tais como: controle de velocidade, controle de posição, etc.

A introdução da descrição e análise dos sistemas físicos através das variáveis de estado marcou o início da fase de Teoria de Controle moderna. Embora a teoria convencional continue aplicável a muitos problemas práticos, a teoria moderna vem sendo utilizada cada vez mais nos projetos dos Sistemas de Controle /SILJ 77/. A potencialidade das técnicas desenvolvidas foram logo reconhecidas em outras áreas de Ciência, tais como: Medicina, Biologia, Administração, Economia, etc

Portanto, a tendência atual é aplicar as técnicas de Teoria de Controle moderna, não só no Nível de Processo como também nos níveis Gerencial /STRE 78//AOKI 78/ e Supervisor /ROLI 78/.

O desenvolvimento dos dispositivos de controle (sensores, atuadores e UD's) influenciou também, de forma marcante, na evolução da configuração dos sistemas de controle. Os primeiros sistemas de controle eram distribuídos e os dispositivos de controle existentes eram basicamente sensores e atuadores. Estes formavam um único bloco de controle local, montado diretamente sobre o ponto de atuação. As UD's eram constituídas de operadores humanos que percorriam toda a instalação para coletar as informações, interpretá-las e tomar decisões sobre ações de controle. Para facilitar a verificação e análise do andamento do processo, era comum acrescentar-se registradores gráficos mecânicos junto aos sensores. Mesmo assim, a monitoração integrada era uma tarefa difícil, pois, o campo de ação dos homens é limitado pela sua capacidade de assimilação, processamento e resposta aos estímulos.

O advento dos computadores possibilitou a automatização das tomadas de decisão pré-programáveis, de modo que, as funções das UD's passaram a ser divididas entre a máquina e o homem. Os computadores não só tomam decisões como também, pré-processam e sintetizam informações necessárias para os operadores humanos atuarem como supervisor. Estas informações sintetizadas são mais agregadas e portanto, um operador humano passa a ter condição de monitorar simultaneamente uma quantidade maior de unidades físicas.

Sob o ponto de vista das inovações tecnológicas dos dispositivos de controle, distinguem-se nitidamente três fases na evolução dos Sistemas de Controle /KOMP 81/:

- .sistemas de controle predominantemente pneumáticos;
- .sistemas de controle predominantemente eletrônicos/elétricos (analógicos);
- .sistemas de controle digitais ou computadorizados.

3.3.1. SISTEMAS PNEUMATICOS

Os primeiros controladores eram de natureza puramente pneumática, constituídos de foles, diafragmas ou tubos de Bourdon. Os sinais de resposta e atuação eram pneumáticos e traduzidos como movimentos físicos (informações) destes dispositivos. Estes controladores atuavam somente no nível do processo (mais especificamente, nas unidades físicas do processo).

Usando as leis hidráulicas, desenvolveram-se as primeiras linhas de pressão, que possibilitavam a transmissão de dados dispersos nos controladores locais a uma sala de comando central. Porém, tecnologicamente, não foi possível obter um nível de confiabilidade razoável que possibilitasse a implementação desta arquitetura centralizada.

Com o desenvolvimento de amplificadores pneumáticos, foi possível padronizar os dispositivos para operações em baixas pressões - em 3 a 15 psig. Esta padronização aumentou a transportabilidade dos equipamentos pneumáticos e ocasionou a sua grande proliferação. Além disso, a operação em baixas pressões tornou tecnicamente viável a transmissão de sinais pneumáticos. Com isso, surgiram os primeiros sistemas centralizados, em que, com exceção dos sensores e atuadores que ficavam próximos do local de controle, todos os dispositivos, tais como indicadores e registradores, ficavam concentrados numa sala de comando central, facilitando a visualização global e o controle integrado do processo (fig 3.9.).

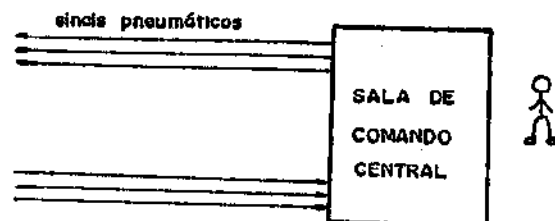


FIG 3.9: Controle pneumático centralizado e

O grande problema destes transmissores pneumáticos era o seu volume físico e a velocidade de transmissão de sinais, que limitavam a distância entre o ponto de atuação e a sala de comando, e a quantidade de sinais manipulados num consolo. Assim, nas instalações de processos mais complexos, permaneciam configurações de controle distribuídas e isoladas funcionalmente em diversas unidades de comando, o que impossibilita um controle integrado em tempo real.

3.3.2. SISTEMAS ANALÓGICOS

A década de 50 é marcada pelo advento da eletrônica no campo do controle industrial, quando se desenvolveram os equipamentos de controle e instrumentos que se comunicavam através de sinais elétricos. Os primeiros controladores eram semelhantes aos controladores pneumáticos, diferindo, basicamente, só na velocidade de transmissão dos sinais - velocidade da luz (sinais elétricos) versus velocidade do som (sinais pneumáticos).

Apesar da relutância inicial, os progressos na área de eletrônica foram tão rápidos que estes equipamentos conseguiram atingir um nível de confiabilidade comparável aos equipamentos pneumáticos, apresentando, ainda, a vantagem de ter uma interface mais simples e ocupar um espaço menor, de modo que, um volume maior de dados sobre o processo podia ser coletado e agrupado num mesmo painel de comando. Com isso, para um sistema de certa complexidade, o grau de integração do controle passou a ser limitado, principalmente, pela capacidade humana de processar corretamente quantidade muito grande de dados num intervalo de tempo menor que a constante de tempo do processo (fig 3.10.).

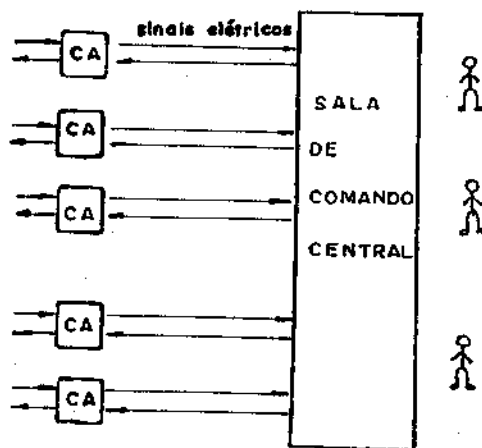


FIG 3.10: Sistemas analógicos

3.3.3. SISTEMAS DIGITAIS

Sem dúvida a evolução da microeletrônica foi marcante no apoio ao controle de processo e possibilitou a implantação dos sistemas automáticos, nos quais os computadores auxiliam os operadores nas tomadas de decisão tanto no nível de processo /ANDR 82//LERO 78//LUWI 81/, como também nos níveis gerencial e supervisor /WILL 78//AOKI 78//ROLI 83/.

Nos níveis gerencial e supervisor, os computadores não só ajudam nos cálculos complexos, como pré-processam e armazenam as informações que poderão ser úteis para as tomadas de decisão. Devido à complexidade e à imprevisibilidade dos fatos envolvidos nestes dois níveis, as tomadas de decisão são dificilmente pré-programáveis e são executadas, em grande parte, pelos homens. Na última década, iniciaram-se estudos e pesquisas na direção de automatizar estes dois níveis. O grande desafio a isso é a formulação de modelos matemáticos que representem apropriadamente o comportamento de sistemas econômicos e administrativos /SILJ 77/.

No nível de processo, os elementos analógicos foram substituídos pelos elementos digitais, apresentando as

seguintes características favoráveis:

- .capacidade de manipular maior quantidade de dados concorrentemente (os primeiros computadores dedicados chegaram a manipular até 10 malhas de controle concorrentemente) (fig 3.11.);

- .transportabilidade (programáveis);

- .maior precisão e menor susceptibilidade a erros.

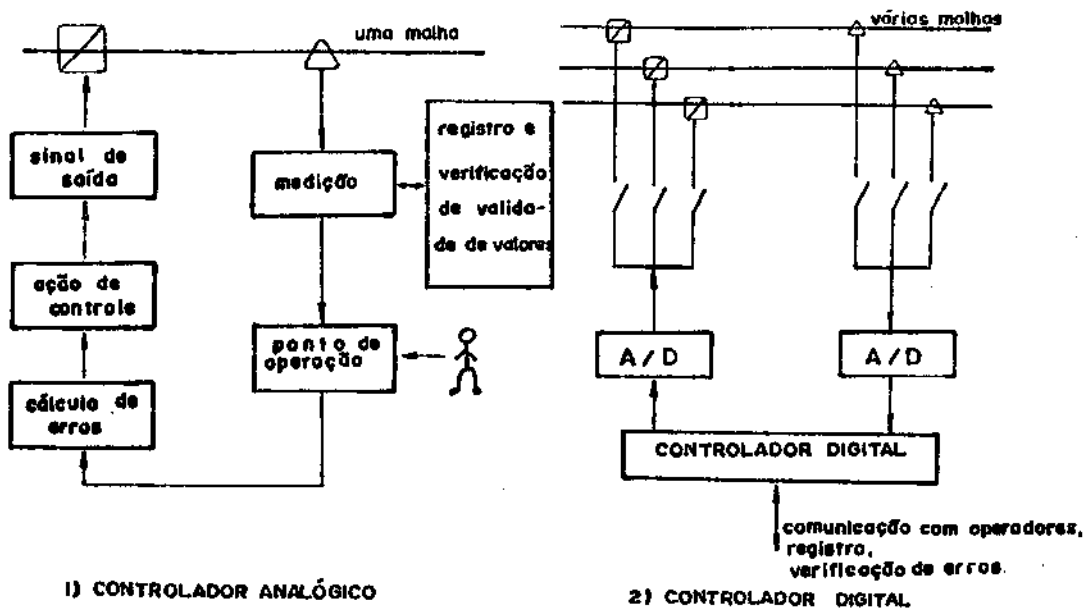


FIG 3.11: Controlador analógico versus controlador digital

Embora, teoricamente, todas as UD's do nível de processo possam ser centralizadas em um único computador de grande porte para facilitar o controle de integridade, esta arquitetura pode apresentar alguns sérios problemas /JUZE 83/:

- .disponibilidade: quando o computador falhar, provocará a parada de todo o processo ocasionando grandes prejuízos;

- .simplicidade: um sistema de controle centralizado requer maior complexidade no desenvolvimento de unidade central de processamento, sequenciamento das tarefas em tempo real, controle de operações de entrada/saída, comunicação de tarefas, etc;

- .custo de transmissão: um grande volume de

dados será coletado e multiplexado para ser transmitido a um controlador computarizado central.

Face a estes problemas, foram propostas as estruturas de controle hierarquizadas, dentro do nível de processo (item 3.2.4.), com as seguintes vantagens:

- .novas camadas hierárquicas podem ser adicionadas conforme os recursos disponíveis e automação desejada, sem invalidar as outras camadas já existentes;

- .os dispositivos de hardware e recursos de software podem ser menos genéricos, dedicados a cada nível, de modo que eles atendam melhor as características desejadas;

- .maior disponibilidade do sistema, pois caso ocorram falhas em alguma camada, outras camadas imediatamente superiores e/ou inferiores poderão servir como retaguarda, mesmo que isto implique em uma certa degradação no desempenho total do sistema.

Um sistema de controle computadorizado, com a arquitetura distribuída e hierarquizada, tornou-se viável na década de 70 com a introdução de microprocessadores e minicomputadores no mercado /LERO 79//LUWI 81/. Atualmente, os microprocessadores tem capacidade suficiente para monitorar uma grande variedade de sensores simultaneamente, tomar decisões e gerar sinais de controle em tempo real. Estes microcomputadores são normalmente acoplados a um minicomputador, que desempenha a função de otimizador e/ou retaguarda na hierarquia de controle.

3.4. SISTEMAS DE CONTROLE AUTOMATICO

Entre as vantagens oferecidas por um sistema de controle automático, distinguem-se /HUNT 78/:

- .aumento da produção com a queda sensível da ocorrência de erros comuns aos operadores humanos;

- .possibilidade de impor restrições mais rigorosas na qualidade dos produtos;

- .substituição dos operadores em tarefas monótonas e rotineiras, ou de alto grau de precisão, ou em

ambientes hostis e perigosos;

maior conveniência e conforto, etc.

Estas vantagens só podem ser usufruídas se o sistema de controle for adequadamente projetado para o problema específico. O projeto de um sistema de controle não é uma tarefa simples. É necessário que os projetistas tenham um bom conhecimento do comportamento estático e dinâmico do Sistema Físico para poder selecionar os equipamentos adequados e implementá-los nos pontos cruciais, de modo a obter a função controladora desejada.

3.4.1. PRINCIPIOS DE TEORIA DE CONTROLE

O projeto de um sistema de controle consiste em utilizar as técnicas da Teoria de Controle para analisar e prever os possíveis comportamentos do Sistema Físico e especificar os recursos (humanos e materiais) necessários para manter o processo em um nível operacional pré-estabelecido.

As técnicas de controle podem ser aplicadas a qualquer tipo de Sistema Físico, desde que estes sejam apropriadamente representados pelos modelos. Quanto mais próximo da realidade for o modelo, melhor será o sistema de controle projetado. Portanto, o problema fundamental no projeto de um sistema de controle é definir um modelo para o Sistema Físico.

3.4.1.1. MODELAMENTO DE SISTEMAS FISICOS

O modelo de um Sistema Físico pode ser constituído por um conjunto de equações matemáticas que descreve o seu comportamento estático e/ou dinâmico - MODELOS ANALITICOS. A representação matemática é, normalmente, formulada a partir das leis físicas, ou de dados experimentais e/ou estatísticos, em que devem ser consideradas todas as possíveis situações, para garantir a representatividade do modelo. Quando não se consegue estabelecer uma relação matemática coerente entre os dados experimentais, pode-se definir como o modelo do Sistema Físico

uma 'tabela' de correspondência entre as possíveis saídas e entradas dos Sistemas Físicos e suas respectivas ações de controle, ou sequência de operações e instruções recomendadas pelas experiências anteriores - MODELOS EMPIRICOS /SAVA 65/.

Os MODELOS ANALITICOS expressam os Sistemas Físicos através de equações matemáticas. Eles podem ser derivados teoricamente e/ou a partir dos dados experimentais. Enquanto os modelos teóricos são definidos a partir dos conhecimentos das leis físicas e matemáticas que regem o comportamento do Sistema Físico, os modelos desenvolvidos experimentalmente são resultados do uso de métodos de projeto experimental, análise de correlações, regressões, etc, para obter a relação entre as variáveis do sistema. A prática mais comum é desenvolver o modelo, usando estas duas aproximações, isto é, através da análise teórica, obtêm-se as equações e os fatores dominantes e os valores dos parâmetros das equações são determinadas com mais precisão, utilizando técnicas de regressão ou correlação /LEFK 77/.

Na formulação dos modelos, as variáveis que devem entrar nas equações devem ser identificadas, conforme a sua função na equação. Esta distinção é importante para a análise dos sistema através das técnicas de Teoria de Controle. Basicamente, classificam-se três grupos (fig 3.12.):

.VARIAVEIS INDEPENDENTES (variáveis de entrada);

.VARIAVEIS DEPENDENTES (variáveis de saída);

.VARIAVEIS DE ESTADO.

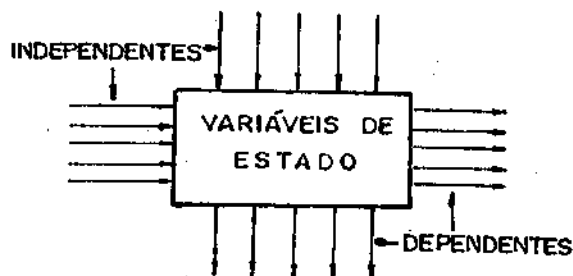


FIG 3.12: Sistema Físico modelado como sistema multivariável

As VARIÁVEIS INDEPENDENTES são as variáveis que determinam o comportamento do Sistema Físico. Elas se classificam em dois grupos: VARIÁVEIS CONTROLÁVEIS e VARIÁVEIS NÃO CONTROLÁVEIS (ruídos ou perturbações). Os controladores podem manipular as variáveis controláveis para contrabalançar os efeitos negativos das variáveis não controláveis, de modo a obter uma saída desejada /SAVA 65/.

Quanto às VARIÁVEIS CONTROLÁVEIS, classificam-se ainda em:

.VARIÁVEIS BÁSICAS: são as variáveis que, quando modificadas, afetam diretamente o comportamento do Sistema Físico;

.VARIÁVEIS TRANSFORMADAS: são as variáveis que derivam da combinação de duas ou mais variáveis básicas e tem participação direta nos modelos, tanto nas suas formulações como na análise do comportamento do Sistema Físico.

As VARIÁVEIS DEPENDENTES são também conhecidas como variáveis de saída. Através delas, pode-se obter e analisar os efeitos da transformação das variáveis independentes nos Sistemas Físicos. Em muitos processos, tais como processos térmicos, termoquímicos e termomecânicos, é impossível de se obter os valores de certas variáveis dependentes através das medições diretas. Por exemplo, no processo de laminação é impossível medir temperaturas internas em um esboço que está sendo laminado. Dessa forma, os valores desses dados inacessíveis devem ser inferidos através de medidas indiretas.

As VARIÁVEIS DE ESTADO são as variáveis que contêm toda informação sobre a dinâmica do sistema. No caso dos sistemas determinísticos e dinâmicos, o vetor das variáveis de estado compacta todo o passado do sistema, de modo que, conhecendo-o e os valores das variáveis de entrada de um intervalo de tempo, pode-se determinar univocamente os valores das variáveis de saída /CTSG 70/.

3.4.1.2. MODOS DE AÇÃO DO CONTROLADOR

Um problema de controle resume-se na solução das seguintes etapas /AZHÓ 75/:

- .estabelecer um conjunto de especificações de desempenho;
- .definir o problema de controle em decorrência das especificações;
- .formular um conjunto de equações que descreva o Sistema Físico;
- .especificar os equipamentos adicionais e as técnicas de controle para melhorar o desempenho do sistema original;
- .otimizar o desempenho do sistema.

A técnica mais difundida para melhorar a resposta de um Sistema Físico é o controle pela MALHA FECHADA ou COMPENSADA. Este modo de ação só é satisfatoriamente realizável, se o comportamento do sistema puder ser modelado analiticamente e o tempo de atuação dos equipamentos for menor ou próximo da constante de tempo do sistema /COKÓ 65/. Utilizando os modelos do Sistema Físico disponíveis, pode-se deduzir, a partir das variáveis de saída, as prováveis perturbações ocorridas na entrada e estabelecer medidas de controle para compensar os efeitos negativos destas perturbações - FEEDBACK ou REALIMENTAÇÃO (fig 3.13.- A). Esta compensação pode ser:

- .analógica ou contínua: as UD's acompanham o comportamento do sistema continuamente;
- .digital ou descontínua: o andamento do sistema é avaliado através de amostragens periódicas.

Outra técnica muito difundida é o CONTROLE PRE-PROGRAMADO ou MALHA ABERTA, em que as ações de controle necessárias para manter o sistema num determinado nível operacional são previstas e pré-estabelecidas, a partir dos históricos do seu comportamento e/ou informações sobre sistemas semelhantes a ele. É também possível, baseando nas perturbações medidas, prever os prováveis desvios que serão causados nas variáveis de saída e definir medidas de controle. Este tipo de controle é conhecido como CONTROLE FEEDFORWARD ou CONTROLE

ANTECIPATIVO (fig 3.13.).

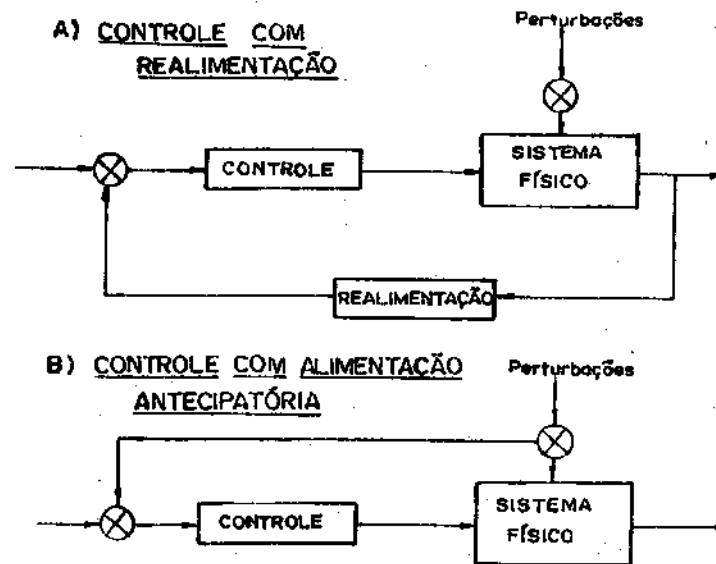


FIG 3.13: Configurações de sistemas de controle em malha fechada

3.4.2. ARQUITETURA DE SISTEMAS DE CONTROLE AUTOMÁTICO

Na última década, o elevado grau de automação nas instalações industriais, não só no nível de processo como nos níveis gerencial e supervisor, deve-se à grande evolução tanto na área de microeletrônica, que possibilitou a construção de computadores e controladores digitais cada vez mais robustos, confiáveis e potentes, como na área de controle, com o desenvolvimento de técnicas e algoritmos de controle mais rigorosos e sofisticados.

Se as decisões a serem tomadas forem estruturáveis ou programáveis, elas poderão ser automatizadas, ficando os operadores humanos na posição supervisora e atuando somente em casos de emergência. Quando as decisões requerem intuição ou criatividade, impossíveis de serem reduzidas a um procedimento padrão, as tomadas de decisão do homem poderão ser facilitadas com o auxílio do computador, que poderá preprocesar os dados para uma forma mais compreensível. Em resumo, as atividades do homem e do computador se complementam, tendo como o

principal objetivo aumentar a eficiência do sistema de controle /ZEMU 75/.

Na arquitetura do sistema de controle moderno, as formas de interação homem-computador-sistema podem ser classificadas quanto a /SAVA 65//SANT 80/:

- .forma de aquisição de dados;
- .atuação de controle.

3.4.2.1. AQUISIÇÃO DE DADOS

A seguir, serão examinadas, sob o ponto de vista de aquisição de dados, as funções dos computadores no Sistema de Controle de Processos.

3.4.2.1.1. COMPUTADOR OFF-LINE

O fluxo de dados entre o computador e o processo faz-se por intermédio dos operadores humanos. Todos os dados são fornecidos manualmente aos computadores. A coleta de dados pode ser direta, quando já estão em formato compatível com o formato de dados aceitável pelo computador (por exemplo, fitas magnéticas), ou manual, quando é necessária a intervenção humana para codificar os dados (por exemplo, em forma de documentos ou relatórios). O computador processa e gera resultados que são utilizados pelos operadores humanos. Normalmente, nestes computadores, os dados são processados em batch, portanto a sua utilização só é recomendada em situações onde o tempo não é um fator crítico, como, por exemplo, nos níveis gerencial (por exemplo, controle de balanço da empresa) e supervisor (por exemplo, controle de escalamento dos turnos de serviço) (fig 3.14.).

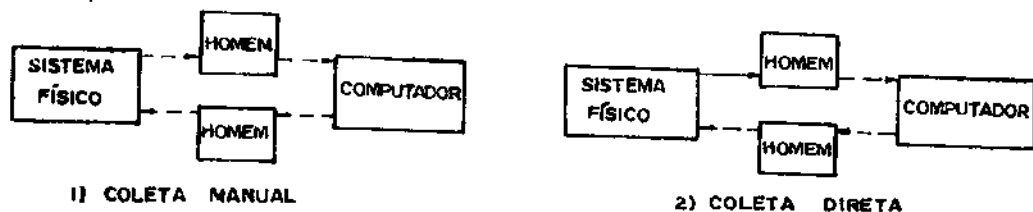


FIG 3.14: Computador off-line

3.4.2.1.2. COMPUTADOR IN-LINE

Os operadores podem interromper as atividades correntes dos computadores e solicitar o processamento imediato dos dados introduzidos, segundo a ordem de prioridade. Esta configuração é utilizada em sistemas nos quais o monitoramento humano é necessário e o tempo que se leva para corrigir as perturbações influi no desempenho global do sistema (fig 3.15.), como por exemplo, a função supervisora no nível de processo.

No nível de processo, os computadores já são capazes de desempenhar duas atividades importantes: controle e supervisão, baseando-se nas informações coletadas e processadas. Devido à sua capacidade de processar grande volume de dados com uma velocidade maior que a dos homens, a tendência atual é substituir o quanto possível os operadores humanos pelos computadores em atividades programáveis. Entretanto, para assegurar um bom desempenho, a presença humana é ainda necessária e importante - não como elemento da malha de controle, mas como supervisor e/ou retaguarda - principalmente nas situações imprevisíveis.

Para tomar decisões corretas, o homem precisa interpretar adequadamente a mensagem recebida. Hoje em dia, o computador oferece recursos para facilitar esta comunicação, preprocessando os dados em informações facilmente assimiláveis pela capacidade humana (por exemplo, sob forma gráfica).

Ressalte-se aqui que, o projeto da interface

homem-máquina envolve não só o fluxo de informação, como também as condições ambientais em que as informações são trocadas. Elas devem estar dentro do campo de percepção (fisiologia) e do campo de ação (ergonomia) dos operadores, para que estes possam agir imediata e livremente em uma emergência /SCHW 80/.

No nível de processo, as atividades mais comuns dos operadores são:

- .supervisionar os desvios das variáveis do processo na faixa de valores admissíveis;
- .supervisionar o estado dos elementos do sistema de controle (sensores e atuadores) e equipamentos físicos que compõem o processo;
- .detectar e interpretar os alarmes e/ou avisos;
- .tomar as ações corretivas corretas e rápidas, segundo a ordem de constante de tempo das unidades no processo;
- .acompanhar a evolução do comportamento do processo, logo após as medidas corretivas ou de manutenção;
- .verificar periodicamente a calibração e o desempenho global do processo.

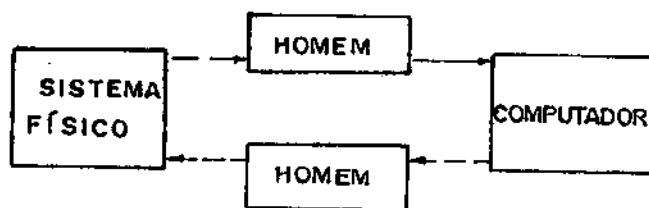


FIG 3.15: Computador In-line

3.4.2.1.3. COMPUTADOR ON-LINE

O computador é provido de periféricos que lhe permitem receber diretamente os dados do Sistema Físico. Estes dados podem ser armazenados e são utilizados pelas UD's para gerar ações de controle. Normalmente, computadores que apresentam estas características são designados também como COMPUTADOR DE CONTROLE ou CONTROLADOR DIGITAL DIRETO (DDC) (fig 3.16.).

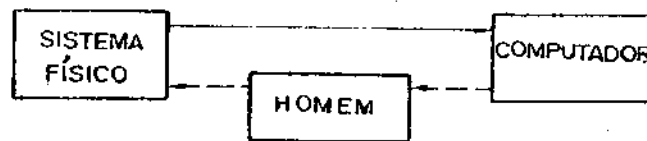


FIG 3.16: Computador On-line

Atualmente, existem no mercado sistemas de aquisição de dados bastante sofisticados, que incluem num único módulo conversores analógicos/digitais, seguradores (sample-holders) e multiplexadores, compatíveis com os microprocessadores e que podem ser ligados diretamente aos Sistemas Físicos.

Os dados referentes ao estado do sistema no nível de processo (por exemplo, temperatura, pressão, etc) devem ser obtidos periodicamente e a frequência de sua coleta é função da necessidade da manutenção do perfil operacional num nível estabelecido e dinâmica do sistema. Eles são utilizados pelos controladores locais para ajustes em tempo real.

Sob o ponto de vista supervisor, estes dados coletados são importantes para o acompanhamento da evolução do processo. Por exemplo, pode-se precisar da média e do desvio dos valores das variáveis medidos num intervalo de tempo pré-estabelecido, para se ter a idéia do andamento do processo. Assim, os DDC funcionam também como preprocessadores de informações e comunicam-se com o computador supervisor, encarregado das decisões mais abrangentes. Esta configuração distribuída contribuiu de forma definitiva para que os sistemas de alta complexidade pudessem ser controlados sem a sobrecarga de um único computador (item 3.1).

3.4.2.2. ATUAÇÃO DE CONTROLE

A atuação de ações decisórias ocorre basicamente sob duas formas: Controle em Malha Aberta e Controle em Malha Fechada.

3.4.2.2.1. COMPUTADOR EM MALHA ABERTA

Diz-se que o controle pelo computador é em malha aberta, quando as informações (ações de controle) elaboradas pelo computador, baseadas nos dados do sistema observado, são transmitidas ao sistema, via operador humano. Se estas ações de controle forem explicitamente emitidas, em forma de instruções, a ação humana se restringirá apenas em executá-las numa sequência estabelecida - denominada também como CONTROLE EM MALHA FECHADA MANUAL; caso contrário, os operadores devem ser capazes de processar as informações e decidir pela sequência de operações mais adequadas (fig 3.16.).

3.4.2.2.2. COMPUTADOR EM MALHA FECHADA

Não há intervenção manual na atuação dos sinais de controle. As ações de controle são decididas e aplicadas diretamente no sistema pelo controlador digital DDC (fig 3.17.).

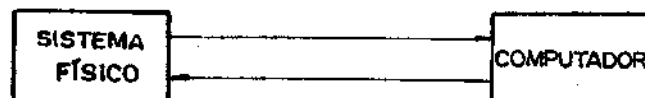


FIG 3.17: Computador em malha fechada

3.4.3. TEMPO DE RESPOSTA

Nos sistemas multivariáveis, a interação homem-

computador-Sistema Físico depende dos modos de entrada e saída dos dados em relação ao Sistema Físico. Estes modos de entrada e saída se combinam resultando em diferentes tipos de configurações de Sistemas de Controle, como por exemplo: modo de aquisição de dados on-line e modo de atuação em malha fechada, on-line e malha aberta, in-line e malha fechada, etc. A opção por uma dada configuração dependerá das exigências e características do Sistema Físico, existindo sempre um compromisso entre a flexibilidade e a rapidez de resposta do sistema decisório, conforme mostra a fig 3.18 /SANT 80/.

ASPECTOS EM RELAÇÃO A	AGENTE DECISÓRIO	HOMEM	COMPUTADOR
Capacidade na aquisição de dados		limitada	muito rápida
Confiabilidade		dependente do estado emocional dos operadores	dependente da confiabilidade dos dispositivos microeletrônicos
Capacidade de tomada de decisão		indefinida e ilimitada	só age conforme o que foi programado
Tempo de resposta		dependente da complexidade do problema	é rápida
Integração e Manipulação de informações complexas		lento	se for adequadamente programado, a velocidade é grande
Atuação nos casos imprevisíveis		eficiente	impossível
Atuação em falhos previstos		velocidade relativamente lenta; depende das reações humanas	atuação rápida e eficiente

FIG 3.18-Homem x computador como agentes decisórios

3.5. SISTEMAS DE CONTROLE INTEGRADO

Como já mencionado (item 3.1), a idéia da automação de um sistema de controle está intimamente ligada à integração das informações sobre o sistema, de maneira a tornar as tomadas de decisão mais precisas e condizentes com a realidade. Portanto, a automação industrial só vai ter seu potencial totalmente explorado quando for suportada por um sistema compatível de tratamento de informação e de suporte à decisão /JLFL 88/. Uma das propostas mais aceitas e difundidas na área de controle é integrar as informações através de Sistemas de Banco de Dados.

Um Sistema de Banco de Dados que se proponha a auxiliar a UD, ou o conjunto de UD's em determinado processo decisório para o controle de um Sistema Físico, deve ser projetado em função de características deste processo, como: a natureza do processo, a constante de tempo, etc, e não através de um esforço direcionado no sentido simplesmente de acumular todas as informações consideradas relevantes. Para isso, utilizam-se **MODELOS DE DADOS**, para formalizar clara e concisamente todas as características do Sistema de Controle que atendam as exigências das Aplicações para que os ABD's estruturem um Esquema de Dados adequado.

Numa empresa industrial, destacam-se três categorias de UD's (item 3.2.) - Gerencial, Supervisor e Processo - que executam atividades funcionalmente distintas e demandam informações de características diferentes (fig 3.19.).

NATUREZA DA INFORMAÇÃO EM RELAÇÃO A:	NÍVEL HIERÁR- QUICO		PROCESSO	SUPERVISOR	GERENCIAL
FONTE	Interna				externa
NÍVEL DE AGREGAÇÃO	detalhado				agregado
HORIZONTE DE VALIDADE	presente				futuro
IDADE	atual				antiga
PRECISÃO NORMALMENTE REQUERIDA (NUMÉRICA)	alta				baixa
FREQUÊNCIA DE ACESSO	alta				baixa

FIG 3.19: Características das informações dos três níveis do sistema de controle

O NÍVEL GERENCIAL envolve atividades referentes à determinação dos objetivos da empresa, dos recursos necessários para o alcance destes objetivos e da definição das estratégias a

serem seguidas para a obtenção e o emprego destes recursos. Isto requer da UD:

- .capacidade de análise de informações abstratas e subjetivas (pesquisa do mercado, descrição imprecisa das tecnologias disponíveis, situação real dos competidores, etc) originadas do meio externo à empresa;

- .intuição e criatividade para traçar táticas de produção de modo a tornar os produtos competitivos no mercado.

As decisões tomadas no nível gerencial são complexas e, normalmente, não se repetem ao longo do tempo, portanto, não existem ainda procedimentos padrões que gerem automaticamente soluções adequadas. Suas ações de controle podem envolver reestruturação de setores da empresa e levar um tempo muito grande para a sua completa efetivação, de modo que, a decisão pela execução destas ações deve ser cautelosa, para evitar resultados desastrosos e irremediáveis. Assim, as tomadas de decisão neste nível são caracterizadas por um grande horizonte temporal de existência e pela generalidade do seu conteúdo /STON 77//JABA 77/.

As decisões acima são implantadas no NÍVEL SUPERVISOR em forma de restrições e/ou objetivos. As decisões neste nível são consideradas como semi-estruturadas, pois, embora o problema requeira ainda a intuição e criatividade na sua solução, algumas decisões, tais como escalonamento do pessoal e alocação de equipamentos, podem ser automaticamente geradas, devido ao caráter rotineiro destas atividades /ROLI 83/.

No NÍVEL DE PROCESSO, os controladores atuam diretamente nos equipamentos físicos. Este nível, tem por objetivo a execução das tarefas específicas determinadas pelo Nível Supervisor e suas ações de controle são locais. No nível de Processo, a frequência de coleta de informações sobre o Sistema Físico é função das constantes de tempo do processo físico e, normalmente, muito maior que nos outros níveis (Gerencial e Supervisor), exigindo que as UD's gerem ações de controle num intervalo de tempo compatível com a constante de tempo do processo. Como o tipo de decisão envolvida neste nível é rotineiro e segue procedimentos padrões, as decisões são

classificadas como estruturadas ou programáveis e, na maioria dos casos, são executadas pelos computadores, sob a supervisão dos operadores /HQAG 77//LEFK 77/.

Como as informações envolvidas nestes três níveis e as exigências das ações de controle são bem diferentes, justifica-se a proposição de um sistema de informações integrado subdividido em três Sistemas de Banco de Dados, cada um projetado especificamente para atender as necessidades de cada nível (fig 3.20.).

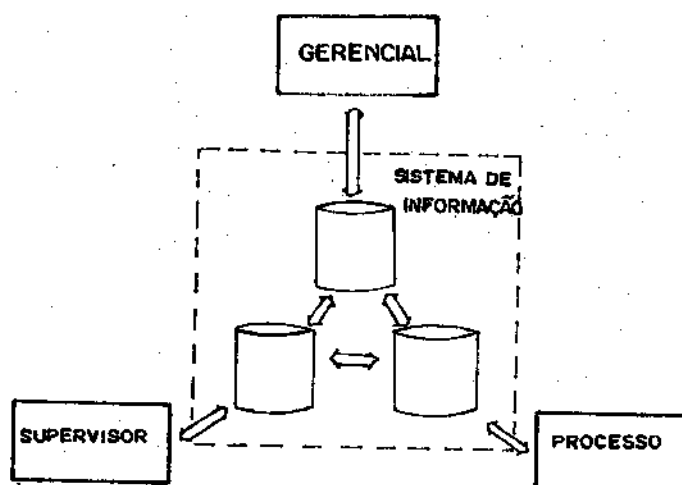


FIG 3.20: Sistema de informações como elemento integrador de três níveis: Processo, Supervisor e Gerencial

O objetivo básico deste trabalho é propor um Modelo de Dados que permita uma descrição natural e precisa das informações envolvidas no nível de Processo (cap.4). O sistema deve apresentar características para armazenar, acessar e atualizar estas informações, que podem ser:

- .informações atualizadas sobre os valores medidos pelos sensores;
- .informações sobre as condições dos componentes de controle (por exemplo, sensores, atuadores) e equipamentos físicos;
- .histórico de um período pré-estabelecido do andamento do processo;
- .informações sobre o sequenciamento de eventos,

quando o processo entra em estado perturbado;

.informações necessárias para tomadas de decisão, críticas no tempo devem ser de fácil acesso, e as informações, relevantes porém não de uso imediato, devem ficar disponíveis para consultas;

.informações indiretas ou derivadas de valores medidos e frequentemente consultadas, poderiam ser preprocesadas e armazenadas de antemão.

CAPITULO 4: MODELO ENTIDADE-RELACIONAMENTO BINARIO

4.1. INTRODUÇÃO

Modelo de Dados é um conjunto de conceitos e relações, através dos quais os usuários definem a sua concepção do Mundo Real. A descrição é denominada ESQUEMA e os dados definidos segundo o esquema constituem as OCORRENCIAS DO ESQUEMA. O projeto de um esquema é essencial para a utilização adequada da Base de Dados: esquema é um meio de comunicação entre o mundo real e o mundo computacional, devendo atender, por um lado, as exigências dos programas de aplicação e, por outro, a capacidade de abstração de dados do Sistema de Gerenciamento de Banco de Dados (SGBD) /BUBB 80/.

Em termos gerais as informações contidas em um esquema podem ser divididas em dois grupos /CHAM 76/:

.CONTROLE: estabelece restrições quanto à localidade, à existência lógica ou física, à forma de acesso aos dados, etc, para garantir a segurança, integridade, consistência dos dados, etc;

.DECLARAÇÃO DOS DADOS: define a natureza dos dados a serem armazenados na Base de Dados.

Uma propriedade muito importante para um Modelo de Dados é a sua capacidade de ABSTRAÇÃO /HAMM 76/. O nível de abstração dos dados está diretamente relacionado com o grau de independência física dos programas de aplicação dos seus respectivos dados. Aqueles com menor nível de abstração apresentam a vantagem dos usuários poderem ter o controle direto sobre o desempenho e a eficiência do sistema, e a desvantagem de que qualquer alteração na definição da visão particular de cada

usuário (SUBESQUEMA) poder vir a afetar a estrutura dos subesquemas de outros (item 2.4.).

Dos modelos existentes, são considerados como Modelos com um nível de abstração elevado aqueles que permitem aos usuários descrever o seu enfoque do Mundo Real numa linguagem natural e padronizada. O mapeamento preciso destas percepções em estruturas de dados convenientes ficam a cargo dos SGBD e/ou ABD'S, possibilitando assim uso mais natural do Sistema de Banco de Dados aos usuários menos especializados na área computacional /TAYL 76/ e aumentando a portatibilidade dos programas de aplicação.

No Capítulo 2, foi visto que existem essencialmente três classes de modelos de dados:

.modelos baseados em teoria relacional (por exemplo, MODELO RELACIONAL), onde as informações são representadas através de relações definidas sobre um conjunto de atributos que descrevem o Mundo Real. As relações necessárias para representar o Mundo Real são estabelecidas analisando-se as dependências entre os atributos por um processo conhecido como NORMALIZAÇÃO (por exemplo, dependência funcional, dependência multivalor, etc). A obtenção das informações requeridas é resultado de cálculos relacionais sobre as relações primárias existentes na Base de Dados;

.modelos baseados em teoria de conjunto (por exemplo, MODELO REDE) onde o Mundo Real é representado como conjuntos de entidades de propriedades semelhantes e relacionamentos que são definidos entre elas. Nestes Modelos, não há distinção semântica rigorosa entre as associações estabelecidas, isto é, qualquer tipo de relacionamento (funcional ou de dependência de existência ou de identificação entre as entidades) estabelecido entre dois conjuntos de entidades definem indistintamente os denominados TIPOS DE CONJUNTOS (Set Entity);

.modelos baseados em ambas as teorias (por exemplo, MODELO ENTIDADE-RELACIONAMENTO) onde são reunidas as propriedades favoráveis dos modelos baseados em teoria de conjunto e em teoria relacional. O Mundo Real é expresso através de dois primitivos, como nos modelos baseados em teoria de

conjunto: entidades e relacionamentos; porém, apresenta uma preocupação maior em definir com mais exatidão a semântica de cada conjunto de entidades/relacionamentos. Os relacionamentos, tais como os de dependência de existência ou de identificação entre as entidades, são explicitamente declarados, o que torna mais precisa a identificação entre o modelo e a realidade que se tem na mente dos usuários.

Uma situação real pode ser descrita utilizando-se qualquer uma dessas três classes de modelos. Conforme a riqueza semântica das linguagens de descrição do modelo utilizado, é possível obter um modelamento que se aproxima mais da concepção dos usuários. Por exemplo, a descrição dos diferentes valores que poderão ser atribuídos a uma variável de um forno é representada no modelo relacional como um conjunto de relacionamentos definidos pela tupla ((CHAVE DA VARIÁVEL), TEMPO, VALOR MEDIDO) e no Modelo Rede como um tipo de conjunto (set entity) que associa um tipo de registro proprietário (VARIÁVEL) ao tipo de registro dependente (VALOR).

Será visto no item 4.2 que, sob o ponto de vista semântico, a descrição do Mundo de Controle de Processo pelos especialistas em controle difere muito da forma como os usuários em aplicações convencionais (comerciais) percebem uma situação real, exigindo, assim, diferentes recursos ou notações para formalizar sua concepção. Por exemplo, uma das diferenças é que no primeiro caso existe a exigência de que, em um dado instante, sejam disponíveis para consulta os estados históricos de um objeto, enquanto no segundo, geralmente só o estado mais atualizado deste objeto é relevante. Os Modelos de Dados mais difundidos (Rede, Entity-set, Relacional, Hierárquico, etc) normalmente são mais adequados para modelar aplicações comerciais convencionais. O objetivo deste Capítulo é definir um Modelo de Dados que descreva de uma forma natural e precisa um processo físico, incluindo toda a formalização das necessidades dos usuários (operadores) - MODELO ENTIDADE-RELACIONAMENTO BINÁRIO (MERB). Este modelo é uma extensão do Modelo Entidade-Relacionamento /CHEN 76/ /NG 81/, adaptada às necessidades do Controle de Processos, como será apresentado no decorrer deste capítulo.

4.2. SISTEMAS DE BANCO DE DADOS CONVENCIONAL E PARA APLICAÇÕES EM CONTROLE DE PROCESSO

Um sistema convencional de Banco de Dados (utilizado em aplicações comerciais convencionais) atende as exigências de programas de aplicação em que o tempo de resposta não seja um fator crítico. Baseado nos dados armazenados na Base de Dados e os esquemas fornecidos pelos ABD's, o SGBD tem a capacidade de inferir precisamente as informações, mesmo que isto possa significar um consumo excessivo do processamento de dados "brutos" em informações "inteligíveis". Por exemplo, através da data de nascimento dos funcionários de uma empresa, é possível inferir a idade dos funcionários em qualquer dia do ano.

Nas aplicações em tempo real, principalmente no controle de processos, onde a constante de tempo do processo pode ser da ordem de (mili)segundos, o intervalo de tempo de acesso às informações no Sistema de Banco de Dados estabelecido, deve ser rigorosamente obedecido de forma a assegurar a compatibilidade das decisões tomadas sobre as ações de controle com a situação real do processo no momento de atuação. Para isso, deve-se reduzir ao mínimo o tempo de acesso aos dados na Base de Dados e o tempo de preprocessamento destes dados em informações compreensíveis /HOAG 77/.

Outra característica de um Sistema de Banco de Dados para aplicações em controle de processo é o seu compartilhamento por dois tipos de usuários de natureza bem distinta: sistema físico e operador humano. O sistema físico pode requisitar os pontos de referência das variáveis armazenados na Base de Dados e o operador pode solicitar os estados mais recentes do Sistema Físico para supervisionar o andamento do processo. Deve existir uma forma de tratamento de prioridade de atendimento destes dois tipos de solicitação, de acordo com as exigências para um bom desempenho do processo.

Além das restrições de tempo mínimo de acesso às informações e a distinção dos atendimentos dos usuários em

diferentes escalas de prioridade, existem outras particularidades de um Sistema de Banco de Dados para controle de processo que merecem ser destacadas:

num Sistema de Banco de Dados convencional, geralmente só é de interesse o armazenamento dos estados mais recentes das entidades e relacionamentos. Para não perder totalmente estas informações, são gerados relatórios em papel ou arquivos em fita, antes de removê-las da Base de Dados. Num Sistema de Banco de Dados para controle de processo os históricos dos estados de uma dada variável são frequentemente consultados para estimar os estados futuros da variável, a fim de auxiliar a tomada de decisão da ação de controle;

devido à natureza dinâmica dos processos, a frequência de atualização dos valores dos atributos de um primitivo (entidade/reacionamento) de um Sistema de Banco de Dados para controle de processo é muito maior que a de um Sistema de Banco de Dados convencional;

num Sistema de Banco de Dados convencional, a integridade dos dados pode ser garantida, através do bloqueio de acesso aos objetos que participam de uma transação. Num Sistema de Banco de Dados para controle de processo, esta espera pode ser indesejável, pois, para efeito de estimação e adaptação num controle de processo, é preferível ter o rápido acesso aos valores disponíveis das variáveis do que esperar pelos valores mais atualizados;

num Sistema de Banco de Dados para controle de processo, muitos dados são adquiridos diretamente do processo, via DDC ou dispositivos analógicos/digitais equivalentes (item 3.3.). Quando ocorrerem anormalidades nestes dispositivos ou nas linhas de transmissão, os dados armazenados serão irreais, o que poderá levar a tomada de decisão de ações de controle impróprias. Portanto, o controle de validação de dados tem um papel fundamental para assegurar a confiabilidade dos dados na Base de Dados.

Os problemas citados serão aqui tratados sob o ponto de vista de modelamento de dados, isto é, a nível descritivo e de formalização das exigências impostas pelos programas de aplicação num modo padronizado, permitindo assim que o SGBD e/ou ABD organize a Base de Dados eficientemente. Um

Modelo de Dados, que permite a formalização precisa de um Sistema Físico dentro das concepções dos usuários do Sistema de Controle, deve incorporar necessariamente:

- ..o conceito da caracterização temporal dos dados existentes na Base de Dados, isto é, a associação aos diferentes estados de uma entidade/relacionamento de um intervalo de tempo em que o estado seja válido;

- ..o conceito de agrupamento de dados funcionalmente dependentes para atender a um (ou mais) programas de aplicação em tempo real;

- ..o estabelecimento do domínio de valores permissíveis para descrição de um determinado estado do processo, para assegurar a validade dos dados armazenados;

- ..o conceito de caracterização existencial dos dados na Base de Dados para acompanhar a dinâmica dos processos, pois, conforme as estratégias de controle adotadas, a semântica das informações poderá variar, resultando na alteração dos dados existentes na Base de Dados.

4.3. PRIMITIVOS DO MODELO ENTIDADE-RELACIONAMENTO BINÁRIO (MERB)

De forma a atender aos requisitos necessários para o modelamento de dados relativos a aplicações em Controle de Processos, será definido a seguir o Modelo Entidade-Relacionamento Binário, o MERB, que resultou do Modelo Entidade-Relacionamento (MER) devidamente modificado e estendido (fig 4.1.).

é mais precisa e tem uma semântica mais clara que a dos modelos discutidos no item 2.5. As qualificações para um dado primitivo não são fixas em todos os instantes do espaço de tempo; portanto, podem estar associados a um qualificador diferentes valores (estados) correspondentes a diferentes intervalos de tempo. Por exemplo, os diferentes valores de temperatura de uma caldeira medidos em instantes distintos.

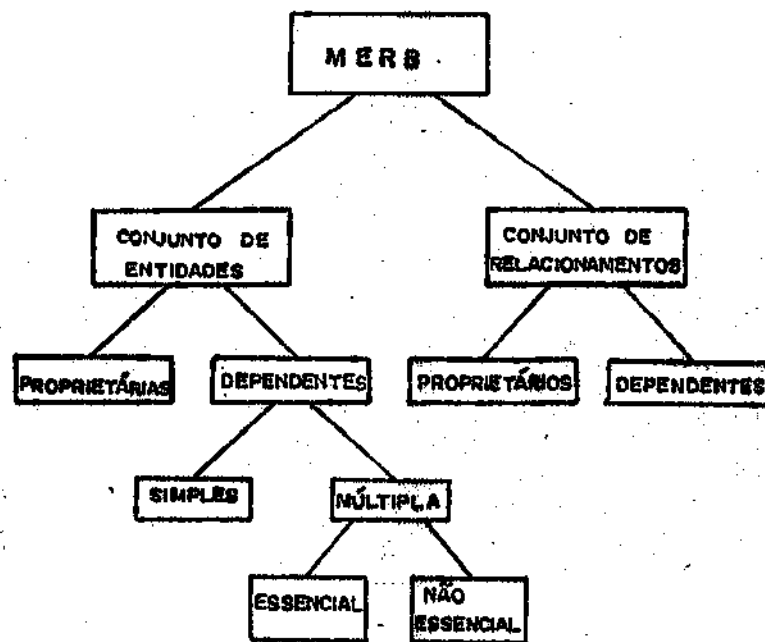


FIG 4.2: Elementos básicos na descrição do mundo real segundo o MERB

A seguir serão apresentados detalhadamente os dois primitivos.

4.3.1. ENTIDADE E CONJUNTO DE ENTIDADES

As entidades são objetos que têm existência própria e podem ser identificados univocamente no Mundo Real. Podem ser consideradas também como unidades atomizadas de informação, a partir das quais podem ser estabelecidas as associações e derivadas novas informações. Por exemplo, <FORNO DE

REAQUECIMENTO) e <TERMOMETRO> são dois objetos distintos, cuja associação através da lei de relação <-mede a temperatura de-> resulta uma terceira informação: temperatura do forno.

As entidades são classificadas em CONJUNTOS DE ENTIDADES, de acordo com as propriedades comuns entre elas, isto é, um conjunto de entidades tem como elementos as entidades que são caracterizadas pelo mesmo grupo de predicativos - ATRIBUTOS. Por exemplo, um forno de reaquecimento com o n.^o de série de fabricação X e outro, com o n.^o Y são dois objetos distintos, que podem ser caracterizados pelo mesmo conjunto de predicativos, tais como: n.^o de série de fabricação, capacidade máxima, temperatura máxima admissível, tipo, fabricante, etc.

Dentro do contexto de uma aplicação, existem entidades que têm a sua existência na Base de Dados dependente da existência de uma outra entidade ou de um grupo de entidades. Diz-se, então, que as entidades são subordinadas ou dependentes a uma entidade proprietária (dependência 1:1 ou DEPENDENCIA SIMPLES) ou a um grupo de entidades proprietárias (dependência 1:N ou DEPENDENCIA MULTIPLA) (nest entities) /WGIO 77/. Deste modo, distinguem-se os conjuntos de entidades como:

- .conjuntos de entidades dependentes;
- .conjuntos de entidades proprietárias ou não dependentes.

As ENTIDADES PROPRIETARIAS são aquelas cuja existência na Base de Dados independe de outras entidades, enquanto as ENTIDADES DEPENDENTES tem a sua existência vinculada à existência de outras entidades na Base de Dados. Por exemplo, os valores do consumo de um certo gás em diferentes unidades de um processo químico é dependente da existência deste gás no tanque de alimentação; portanto, pode-se modelar o gás consumido em cada unidade como entidade dependente de existência e o gás no tanque de alimentação como entidade proprietária.

Quando a DEPENDENCIA é SIMPLES, cada entidade dependente é associada a uma entidade proprietária, como ocorre no exemplo do parágrafo anterior. Muitas vezes, inclui-se a chave da entidade proprietária para a identificação das suas entidades

dependentes. Neste caso, diz-se que, além da dependência de existência, há DEPENDENCIA DE IDENTIFICAÇÃO /NG 81/.

Quando a DEPENDENCIA é MULTIPLA /SMIT 80/, uma entidade dependente pode estar associada a várias entidades proprietárias e, dependendo do contexto da aplicação, esta associação pode ser ESSENCIAL ou NÃO ESSENCIAL. Uma DEPENDENCIA MULTIPLA ESSENCIAL ocorre quando a existência de uma entidade depende da existência simultânea de todas as entidades a que ela está associada. Por exemplo, um objeto composto de carcaça, rotor, estator, etc, define um motor e a remoção de qualquer uma destas peças torna incompleta a sua composição; portanto, esta dependência pode ser considerada essencial. Uma DEPENDENCIA MULTIPLA NÃO ESSENCIAL acontece quando a existência da entidade depende da existência de uma das entidades a que ela está associada na Base de Dados. Por exemplo, uma linha de produção é caracterizada por um grupo de equipamentos em funcionamento e enquanto um destes equipamentos estiver em operação, a existência da linha de produção é garantida.

Resumindo, no MERB classificam-se os conjuntos de entidades nas seguintes categorias:

.Conjunto de Entidades Proprietárias;

.Conjunto de Entidades Dependentes:

.simples

.de existência

.de existência e de identificação

.múltipla

.essencial

.não essencial.

Nota-se que no MERB a classificação do conjunto de entidades é mais completa que a classificação proposta no Modelo Entidade-Relacionamento. No MER, não é claramente definido o tipo de existência que existe entre os conjuntos de entidades, tal como é proposto no MERB. Por exemplo, numa empresa, segundo o modelo MER, as diversas linhas de produção e as tarefas a serem executadas sequencialmente em cada linha de produção são modelados como conjuntos de entidades regulares (linhas de produção) e conjuntos de entidades fracos

(tarefas), isto é, a existência das entidades correspondentes as tarefas na Base de Dados depende da existência da linha de produção. Porém, não está clara a interdependência entre entidades referentes às linhas de produção e às tarefas que são comuns a mais de uma linha de produção, onde a remoção da entidade correspondente a uma linha de produção não deve implicar necessariamente a remoção de todas as tarefas associadas a elas, pois estas podem ser dependentes de outras linhas ainda não removidas. No modelo MERB, esta situação é elegantemente solucionada, considerando adequadamente o tipo de dependência de existência que há entre as tarefas e as linhas de produção. Neste exemplo particular, as linhas de produção são agrupadas em conjunto de entidades proprietário e as tarefas, em conjunto de entidades de dependência múltipla e não essencial do conjunto de entidades <LINHA DE PRODUÇÃO>, ou seja, a existência da tarefa é garantida se existe pelo menos uma linha associada a ela.

4.3.2. RELACIONAMENTO BINARIO E CONJUNTO DE RELACIONAMENTOS

Relacionamentos binários são associações que se estabelecem entre dois objetos definidos como entidades e/ou outros relacionamentos binários, através dos papéis (roles) que estes desempenham no Mundo Real. Por exemplo, uma lei de associação <-mede a temperatura de-> estabelece relacionamentos binários entre as entidades que desempenham papéis de MEDIDOR e OBJETO MEDIDO, respectivamente. Cada par de entidades associadas conforme acima é considerado um RELACIONAMENTO BINARIO e o conjunto de pares que se associam pela mesma lei é denominado CONJUNTO DE RELACIONAMENTOS BINARIOS. Formalmente, define-se um conjunto de relacionamentos binários R_i , como um par ordenado onde cada elemento pertence a um de dois conjuntos de entidades e/ou relacionamentos binários não necessariamente distintos:

$$R_i = \{ (e_1, e_2) / e_1 \in \{E\}_{i1} \wedge e_2 \in \{E\}_{i2} \}$$

onde: e_1, e_2 = entidades/relacionamentos binários;

$\{E\}_{i1}$ = conjunto de entidades/relacionamentos binários que desempenham o papel F_i ;

$\{E\}_{i2}$ = conjunto de entidades/relacionamentos binários que desempenham o papel F_2 ;

(e_1, e_2) = par ordenado de entidades/relacionamentos binários;

F_1, F_2 = papéis distintos desempenhados pelas entidades-relacionamentos binários.

Observe-se que o conjunto de relacionamentos estabelece a associação pelo papel (role) que o conjunto de entidades desempenha e não pelo conjunto de entidades, como no Modelo Rede (Network) (item 2.5.1.)

Os relacionamentos que estabelecem associações entre n ($n > 2$) entidades, como no Modelo Entidade-Relacionamento, podem ser sempre expressos como $(n-1)$ relacionamentos binários aninhados (nested). Por exemplo, o conjunto de relacionamentos $R(\text{MOTOR}, \text{CARGA}, \text{OPERADOR})$ pode ser desdobrado em dois relacionamentos binários $R_1(\text{OPERADOR}, R_2(\text{CARGA}, \text{MOTOR}))$. Por um lado, isto pode aumentar sobremaneira a quantidade de conjuntos de relacionamentos binários /CODD 70/, pois um conjunto de relacionamento de n tuplas passará a ser representado por, pelo menos, $(n-1)$ conjuntos de relacionamentos; por outro lado, a representação das associações em notação binária é semanticamente mais clara /SENK 73/. No exemplo acima, as associações entre o motor e a carga e entre o operador e a carga são indistintas no relacionamento $R(\text{MOTOR}, \text{CARGA}, \text{OPERADOR})$, tornando difícil definir qual a associação que represente o fato real, e nos relacionamentos $R_1(\text{OPERADOR}, R_2(\text{MOTOR}, \text{CARGA}))$ está explícita a associação direta entre o motor e a carga e entre o operador e o grupo motor-carga.

Autores como CODD /CODD 70/ argumentam que os relacionamentos binários tornam a obtenção de resposta às perguntas que envolvem mais de dois argumentos menos natural. Por exemplo, para obter o motor que acione uma carga X e o operador que supervisione o acionamento nos relacionamentos $R_1(\text{OPERADOR}, R_2(\text{MOTOR}, \text{CARGA}))$, deve-se seguir os seguintes passos:

- .obter todos os motores que acionem a carga X ;
- .obter os operadores que supervisionem o conjunto CARGA-MOTOR.

Porém, a resposta à pergunta acima seria

imediate se estivessem definidos os relacionamentos binários como R3(CARGA,R4(MOTOR,OPERADOR)). Assim, com a notação binária, é possível que os usuários definam, sob uma forma mais natural, todas as associações binárias básicas, tanto entre as entidades primárias como entre as entidades e outros relacionamentos binários, que sejam necessárias para obter com um mínimo tempo de acesso às informações complexas /BBLN 76/.

No MERB, um relacionamento pode ser interpretado como uma informação derivada das informações essenciais já existentes na Base de Dados - entidades e/ou de outros relacionamentos. Portanto, a sua existência é dependente da existência das entidades/relacionamentos que participaram na sua formação. Quanto à caracterização, os relacionamentos são descritos por um grupo de propriedades numa forma análoga à descrição das entidades. Apesar da semelhança na representação estrutural de relacionamentos e entidades, existe uma grande diferença entre eles quanto às suas existências na Base de Dados. A remoção de uma entidade implica sempre na remoção de todos os relacionamentos em cuja formação ela tomou parte e a remoção de um relacionamento não causa necessariamente a remoção das entidades associadas a ele.

Como consequência da classificação dos conjuntos de entidades em conjuntos de entidades dependentes e conjuntos de entidades proprietárias, os conjuntos de relacionamentos são também divididos em:

.conjunto de relacionamentos dependentes (corresponde ao conjunto de relacionamentos fracos do MER): quando uma das entidades que participam na associação é dependente;

.conjunto de relacionamentos proprietários (corresponde ao conjunto de relacionamentos fortes do MER): quando as entidades associadas são não dependentes.

4.4. DESCRIÇÃO ESTRUTURAL DO MERB

A descrição estrutural de um Modelo de Dados próximo à concepção do Mundo Real corresponde ao nível descritivo

(fig 2.7.) e é feita através da definição do esquema/subesquema. Em um esquema e/ou subesquema de dados, distinguem-se duas classes de informações:

- .declaração de dados;
- .controle.

4.4.1. DECLARAÇÃO DE DADOS

Consiste em um conjunto de formalizações com a finalidade de definir a natureza e características dos dados que compõem a Base de Dados (fig 4.7.).

4.4.1.1. ATRIBUTOS

Os relacionamentos e as entidades definidos no nível Conceitual são caracterizados utilizando a função ATRIBUTO, e descritos através de um conjunto de predicativos. Formalmente, define-se um ATRIBUTO como uma função que associa os valores de um DOMÍNIO DE VALORES a um conjunto de entidades e/ou conjunto de relacionamentos.

Tanto os tipos quanto o intervalo (range) dos valores em um domínio de valores dependem do atributo. Por exemplo, para o atributo <TEMPERATURA DO FORNO>, os valores permissíveis são do tipo real e o domínio deve cobrir o intervalo $[0^{\circ}, 150^{\circ}]$, enquanto para o atributo <ESTADO DE CHAVEAMENTO>, o domínio é definido usando-se apenas dois valores: LIGADO, DESLIGADO.

Os domínios dos valores sobre os quais são definidos os atributos não são necessariamente disjuntos. Por exemplo, um domínio de valores tipo caracteres, pode ser usado como o domínio dos atributos <NOME DO FABRICANTE>, <NOME DO OPERADOR>, <DESCRIÇÃO DA SEQUENCIA DE OPERAÇÕES>, etc.

Pela definição, cada relacionamento ou entidade é uma unidade distinta no conjunto ao qual ele pertence.

Isso significa que, através das qualificações atribuídas a eles, deve ser possível estabelecer uma correspondência biunívoca entre os primitivos (entidades/relacionamentos) e os valores de um subconjunto de qualificações definidas para eles. Este subconjunto de atributos (ATRIBUTOS PRIMÁRIOS) é denominado a CHAVE PRIMARIA do conjunto. Os outros atributos, não pertencentes às chaves, são designados ATRIBUTOS NÃO PRIMÁRIOS.

A seleção da chave primária entre os atributos é importante no projeto da estruturação de dados para evitar ambiguidade de acesso. No Modelo Entidade-Relacionamento Binário, esta seleção baseia-se nas seguintes regras:

.no conjunto de ENTIDADES: escolhe-se como a chave primária um subconjunto de atributos, cujos valores identificam univocamente as entidades no conjunto (fig 4.3.);

CONJUNTO DE ENTIDADES	ATRIBUTOS				
	PRIMÁRIOS	NÃO PRIMÁRIOS			
	CÓDIGO	CORRENTE NOMINAL	TENSÃO NOMINAL	NÚMERO DE POLOS	TIPO
DOMÍNIO DE VALORES					
SECCIONADOR	CÓDIGO	CORRENTE (A)	TENSÃO (V)	FASE	TIPO
1	3KE20	200	600	trípolar	rotativo
2	3KUI427	400	660	trípolar	rotativo
3	3TK113	16	440	trípolar	rotativo

FIG 4.3: Representação de um conjunto de entidades

.como cada RELACIONAMENTO caracteriza a associação estabelecida entre entidades, através dos papéis (roles) que estas desempenham, as chaves primárias do conjunto de relacionamentos serão representadas pelas próprias chaves primárias dos conjuntos de entidades associadas. Por exemplo, sejam dois conjuntos de entidades <ALARME> e <TERMOMETRO> associados pela relação <-sinaliza as condições anormais de ->, onde os dois conjuntos acima desempenhariam papéis de SINALIZADOR e MEDIDOR, respectivamente. Cada relacionamento poderia ser

identificado pelas chaves primarias de <ALARME> e <TERMOMETRO> (fig 4.4.);

CONJUNTO DE RELACIONAMENTOS	ATRIBUTOS		
	PRIMÁRIOS		NÃO PRIMÁRIOS
	PAPÉIS		NÚMERO DE MANOBRAS
	PROTETOR	EQUIPAMENTO	
	CONJUNTO DE ENTIDADES		DOMÍNIO DE VALORES
CHAVE/MOTOR	CÓDIGO	MOTOR	NÚMERO DE MANOBRAS
1	3KE20	M100HP	50
2	35A3820	M8HP	100

FIG 4.4: Estrutura de um conjunto de relacionamentos

nos conjuntos de ENTIDADES OU RELACIONAMENTOS DEPENDENTES NA IDENTIFICAÇÃO: cada unidade é identificada pela mesma chave selecionada para o seu conjunto de entidades proprietárias (fig 4.5.).

CONJUNTO DE ENTIDADES DEPENDENTES	ATRIBUTOS		
	PRIMÁRIOS		NÃO PRIMÁRIOS
	PROPRIETARIO	DEPENDENTE	APLICAÇÃO
	ROTOR	MOTOR	
	DOMÍNIO DE VALORES		
COMPONENTE	TIPO DE ROTOR	MOTOR	FINALIDADES
1	GAIOLA	M100HP	acionador de carga
2	BOBINADO	M80HP	bomba

FIG 4.5: Estrutura de um conjunto de entidades/relacionamentos dependentes

Os valores associados aos atributos de uma entidade/relacionamento podem ser SIMPLES (quando a associação é injetiva) ou MÚLTIPLOS (quando a função é sobrejetiva), como no

Modelo Entidade-Relacionamento. Esta característica torna o Modelo Entidade Relacionamento Binário (MERB) mais natural aos usuários em relação aos modelos Relacional e Rede. No Modelo Relacional, os valores múltiplos de um atributo de uma entidade são desdobrados em diversas relações e no Modelo Rede, são declarados como tipos de conjunto (set entity). Por exemplo, para a entidade <TERMOMETRO DO FORNO A>, os diversos valores correspondentes as diferentes medições do forno A são representados sob diferentes formas nos três modelos, conforme mostra a fig 4.6.

A) MODELO REDE

Tipo de registro proprietário: termômetro

Tipo de registro membro: valor de temperatura

Tipo de conjunto: termômetro/valor

B) MODELO RELACIONAL

TERMOMETRO	VALOR
A	1000°
A	985°

C) MODELO ENTIDADE - RELACIONAMENTO

TERMOMETRO	VALOR
A	1000° 985°

FIG 4.6: Diferentes representações dos valores múltiplos de um atributo

4.4.1.2. PAPEIS (ROLES)

O conceito de PAPEL aparece no Modelo Entidade-Relacionamento /CHEN 76/. No MERB, ele é considerado como um qualificador (atributo) especial e mapeia os objetos (definidos como entidades ou relacionamentos binários) em funções que eles desempenham numa associação estabelecida pelo conjunto de relacionamento. Um objeto pode desempenhar diversos papéis (quando ela pertence a vários conjuntos de relacionamentos) como objetos pertencentes aos conjuntos de entidades/relacionamentos binários distintos podem exercer um mesmo papel (quando o

conjunto de relacionamentos estabelece associação M:N entre os conjuntos de entidades/relacionamentos binários). Por exemplo, um conjunto de relacionamentos MEDIÇÃO (MEDIDOR, OBJETO) define não só os relacionamentos entre conjuntos de entidades <TERMOMETRO> e o <FORNO>, como também o relacionamento entre os conjuntos de entidades <PRESSOSTATO> e <CALDEIRA>.

4.4.1.3. VERSÕES / DOMÍNIO DE TEMPO

Na maioria das aplicações em controle de processo é imprescindível ter-se acesso a um conjunto de estados de um primitivo (entidade/relacionamento) sobre um intervalo de tempo /BUBB 80/. Estes estados constituem os históricos da evolução do processo e tem a seguinte utilidade:

- .relatórios mecanizados: fornecer os dados coletados numa forma estruturada;

- .correção: auxiliar o operador na decisão das medidas corretivas, quando ocorrer alguma falha no sistema;

- .otimização: auxiliar na avaliação do desempenho do sistema e otimizar as ações de controle;

- .prevenção: auxiliar os operadores na previsão do comportamento do sistema para evitar paradas de operações nas áreas prioritárias ou críticas.

O histórico tem a ver com os fatos ocorridos no espaço temporal, ou seja, podem existir diferentes valores associados a um primitivo através de uma mesma função ATRIBUTO, cada qual representando um estado do primitivo no domínio de tempo. Assim, só a caracterização dos primitivos através da função "atributo" não é suficiente para distinguir os diferentes estados destes primitivos. Introduz-se, então, a função VERSÃO, que mapeia um primitivo no domínio temporal para indicar os seus diferentes estados. Cada estado é mapeado no domínio de valores através da função "atributo" para qualificar cada estado (fig 4.7.). Quando só um estado é mantido na Base de Dados, ter-se-ia uma situação degenerada, em que o domínio de tempo é reduzido a um instante no espaço temporal.

4.4.1.4. DOMINIO DE VALORES

O DOMINIO DE VALORES é um conjunto de dados atomizados e derivados basicamente de dois domínios naturais: números e caracteres /LEOD 76/.

O Sistema de Gerenciamento de Banco de Dados inclui vários tipos de regras de integridade semântica, as quais definem com mais precisão os dados que compõem a Base de Dados. Um dos tipos de regra de integridade semântica útil para aplicações em controle de processo é a DEFINIÇÃO DO DOMÍNIO, que caracteriza os valores admissíveis no domínio de um certo atributo. Esta definição inclui a especificação dos tipos e o intervalo dos componentes dos valores, a ordem destes componentes para constituir um valor (por exemplo, valor <NOME> constituído pelos componentes PRENOME e SOBRENOME) e a definição de ações de controle, caso os valores a serem armazenados não satisfaçam as restrições impostas ao conteúdo da Base de Dados.

4.4.2. CONTROLE DE DADOS

Como o Modelo proposto visa a aplicações em tempo real, muitas funções do controle de dados, quanto à privacidade, validade, segurança, etc, tem a sua importância colocada em segundo plano, em favor do tempo de processamento de dados /SMIT 80/. O Modelo proposto oferece recursos (controladores abstratos) para que os usuários especifiquem a localidade funcional dos dados, definam caminhos lógicos de acesso auxiliares para melhorar o tempo de acesso às informações e estabeleçam as condições de existência destas informações na Base de Dados, conforme descrito a seguir (fig 4.7.).

4.4.2.1. MÓDULO

Conforme a constante de tempo do Processo, os dados funcionalmente dependentes para a execução de um procedimento podem ser agrupados em MÓDULOS. Por exemplo, todos os dados históricos necessários para estabelecer um novo ponto de referência para uma dada variável do processo são considerados dados funcionalmente dependentes.

A introdução deste conceito torna-se necessária nos sistemas de tempo real, onde o desempenho do sistema não é avaliado somente pelo acesso às informações corretas, mas também pela velocidade deste acesso. Para um Sistema de Gerenciamento de Banco de Dados, o acesso às informações inclui o acesso aos dados armazenados na Base de Dados (arquivos físicos externos) e o preprocessamento destes dados para a obtenção das informações requisitadas pelos programas de aplicação. Devido às limitações tanto nas tecnologias de software como de hardware, é ainda impossível estruturar uma Base de Dados que possa atender com a mesma eficiência todos os programas de aplicação que a compartilham; há, portanto, a necessidade de distinguir os dados em grupos que são críticos e não-críticos quanto ao tempo de acesso. Os grupos considerados críticos devem ter seus acessos otimizados para reduzir, ao mínimo, o tempo de acesso às informações por parte dos usuários.

Uma das formas de reduzir o tempo de acesso às informações é alocar todas as informações necessárias para a execução de uma tarefa à memória principal, antes da sua execução. Outra forma de reduzir o tempo é assegurar a localidade física dos dados, de modo que com um número mínimo de acessos, possa-se obter os dados necessários para processar as informações exigidas na execução de um programa de aplicação. Qualquer que seja a solução adotada, é necessário que o ABD tenha informações suficientes para organizar a estrutura de dados de acordo com as necessidades dos seus usuários.

A definição de MÓDULO é uma forma através da qual os usuários podem expressar a sua necessidade de localidade funcional no MERB.

Os módulos não são necessariamente disjuntos, porque um mesmo dado pode ser relevante para a execução de tarefas distintas. Por exemplo, o nível de gás para a combustão numa caldeira é um dado importante tanto para a tarefa de avaliação do consumo de energia como também para a tarefa de cálculo do rendimento da caldeira. Dependendo do intervalo de tempo pré-estabelecido para a execução dessas tarefas, pode-se definir o grau de velocidade de acesso às informações necessárias e, conseqüentemente, a prioridade do agrupamento lógico destes dados.

4.4.2.2. CHAVE SECUNDARIA

Em muitos programas de aplicação, um grupo de dados pode ser acessado pelos seus atributos, tais como: <obter todas as unidades que estão em manutenção> ou <obter todos os fornos de reaquecimento que têm capacidade acima de X>. A resposta a estes tipos de perguntas pode ser extremamente demorada, se forem consultados todos os registros e os valores dos seus atributos para selecionar aqueles que atendam as restrições impostas. A declaração explícita de CHAVE SECUNDARIA, através da qual os elementos são agrupados conforme um dado valor do seu atributo, é um outro recurso para aumentar a velocidade de acesso às informações.

4.4.2.3. EVENTO

Nos processos dinâmicos, os relacionamentos e as entidades definidos, são variantes no tempo, conforme a dinâmica do sistema. Por exemplo, um relacionamento ALARME(falha i, alarme X) só é inserido na Base de Dados no momento em que ocorrer o alarme, senão a associação não tem sentido físico. Para que o conteúdo da Base de Dados seja uma réplica do estado do processo em qualquer instante, é preciso formalizar a descrição do comportamento dinâmico em Modelo de Dados de modo que o ABD possa ter conhecimento de como o conteúdo da Base de Dados deve

acompanhar a evolução deste processo. Portanto, é introduzido no MERB o conceito de EVENTO, o qual estabelece as condições sob as quais o conteúdo da Base de Dados pode ser alterado.

Classifica-se um EVENTO em:

.AUTOMATICO: quando o acontecimento causa automaticamente a modificação do conteúdo da Base de Dados;

.MANUAL: quando o acontecimento demanda a inserção de novos dados impossíveis de serem derivados dos já existentes. Normalmente requer a interferência dos operadores humanos.

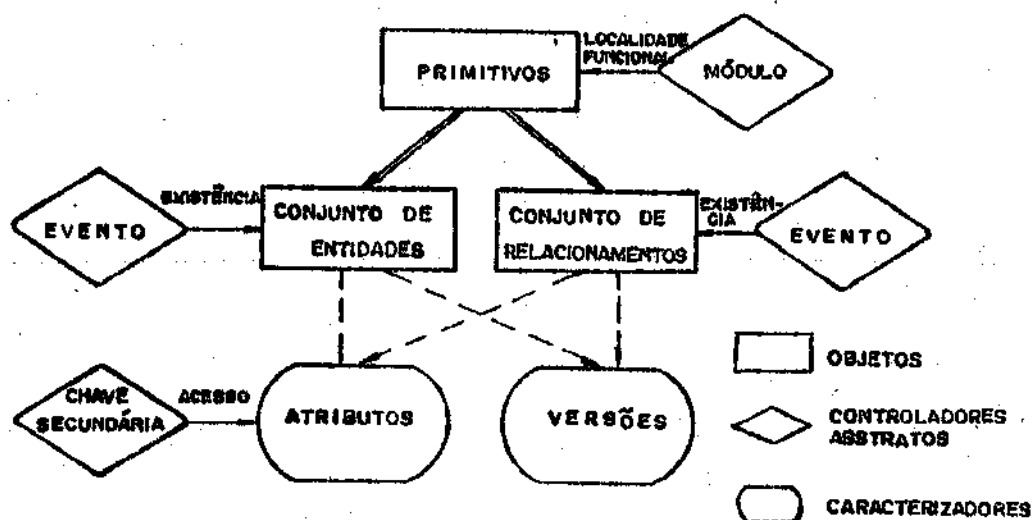


FIG 4.7: Componentes básicos para descrição do Mundo Real, segundo o Modelo MERB

4.5. SINTAXE DA DESCRIÇÃO ESTRUTURAL

A seguir é apresentada uma metalinguagem através da qual o MERB será formalizado. No item 5.4. será apresentada uma representação gráfica de todos os primitivos.

4.5.1. DESCRIÇÃO DA METALINGUAGEM

Para facilitar a discussão e a descrição formal da Linguagem de Definição de Dados do MERB, será utilizada uma linguagem simbólica (metalinguagem) segundo a notação BNF ('Backus-Naur Form'). A sintaxe da linguagem é descrita através dos seguintes símbolos metalinguísticos:

- . < > : indica uma variável metalinguística ou variável não-terminal, cuja definição é dada por uma expressão metalinguística;
- . ::= : define uma variável metalinguística, à sua esquerda, através de uma expressão metalinguística à direita. Uma expressão metalinguística pode ser composta por diferentes alternativas de definição. Por exemplo, <MOTOR> ::= <SINCRONO> ou <ASSINCRONO>;
- . | : separador das definições alternativas de uma variável metalinguística. No exemplo acima, a conjunção 'ou' pode ser substituída pelo símbolo metalinguístico '|';
- . { } : define a variável metalinguística por expressões ou frases escritas em linguagem natural;
- . [] : indica que o operador que segue o colchete se aplica a expressão metalinguística contida entre os colchetes;
- . * : operador de repetição, 0 ou mais vezes;
- . + : operador de repetição, 1 ou mais vezes.

Esta metalinguagem permite definir de forma concisa e completa a sintaxe da Linguagem de Descrição de Dados (LDD), pois, através dos diversos passos de refinamento, pode-se derivar a partir de uma fórmula metalinguística (composta de variável metalinguística e sua respectiva definição), um conjunto de símbolos básicos de LDD, denominados SIMBOLOS TERMINAIS. Para distinguir os símbolos metalinguísticos dos símbolos terminais da LDD é convencionado o uso de letras maiúsculas para representar os símbolos básicos e as letras minúsculas para os símbolos metalinguísticos.

Nem todas as fórmulas metalinguísticas definidas a seguir serão detalhadas até o nível de símbolos terminais, desde que isso não afete o objetivo dos itens seguintes, que é descrever formalmente o contexto de um esquema

considerando a sintaxe dos seus constituintes, de modo que através dele seja possível formalizar com precisão o Mundo Real visto pelos usuários.

4.5.2. DEFINIÇÃO DO ESQUEMA

Na estrutura de um esquema distinguem-se essencialmente dois blocos:

.a especificação dos dados contidos na Base de Dados, que, através da declaração dos diferentes estados coexistentes na Base de Dados e das qualificações de cada estado, define o tipo e a natureza dos primitivos identificados no Mundo Real pelos usuários;

.o corpo de esquema, que contém todas as informações adicionais para o SGBD poder controlar a localidade, a privacidade, a integridade, etc, dos elementos especificados conforme as condições estabelecidas pelos usuários.

```
.<esquema> ::=DEF ESQUEMA <identificador> <especificação>
                        <corpo>
.<identificador>::={nome do identificador}
.<especificação>::=<dados globais>
.<dados globais>::=<primitivo>
.<corpo>          ::=(<modulo>[<evento>][<lista>][<domínio de va-
                        lores>][<domínio de tempo>][combinação das 5
                        variáveis metalinguísticas anteriores
```

4.5.3. ESPECIFICAÇÃO DOS PRIMITIVOS

No Modelo MERB distinguem-se essencialmente dois primitivos, classificados em diferentes conjuntos:

.conjunto de entidades:

.proprietárias;

.dependentes:

.simples

.de existência;

.de existência e identificação;

.múltipla
 .essencial;
 .não essencial;
 .conjunto de relacionamentos:
 .proprietário;
 .dependente.

Ambos os elementos podem ter seus diferentes estados armazenados na Base de Dados, utilizando o mapeamento através da função VERSÃO. E cada estado é caracterizado por um conjunto de atributos primários e não-primários, onde o subconjunto de atributos não-primários pode ser um conjunto vazio. Os valores associados a cada elemento dos primitivos (conjunto de entidades ou conjunto de relacionamentos), através das funções VERSÃO e ATRIBUTO, pertencem ao DOMÍNIO DE TEMPO e DOMÍNIO DE VALORES, respectivamente.

```
.<primitivo> ::= <conj ent> [<conj rel>
.<conj ent> ::= <conj ent depend> [<conj ent propr>
.<conj ent propr> ::= DEF CONJ ENT PROPR <identificador> <quali-
    ficador> * <chave>
.<conj ent depend> ::= <conj ent mult> [<conj ent simples>
.<conj ent mult> ::= <conj ent essencial> [<conj ent não essen-
    cial>
.<conj ent simples> ::= DEF CONJ ENT DEP SIMP <identificador>
    { conjunto de entidades proprietária } <quali-
    ficador> * <chave>
.<conj ent essencial> ::= DEF CONJ ENT DEP ESSENCIAL <identifi-
    cador> <qualificador> * { lista de conjuntos de
    entidades proprietárias } <chave>
.<conj ent não essencial> ::= DEF CONJ ENT DEP NÃO ESSENCIAL <i-
    dentificador> <qualificador> * { conjuntos de
    entidades proprietárias } <tipo de dependên-
    cia> * <chave>
.<qualificador> ::= <versão> [<atr não primário>] <versão> <atr não
    primário>
.<atr não primário> ::= <atributo>
.<versão> ::= VERSÃO <identificador> { identificador do domínio
    de tempo }
.<chave> ::= <atributo> +
```

```

.<atributo> ::= ATRIBUTO <identificador> { identificador do
                                domínio de valores }
.<tipo de dependência> ::= ESSENCIAL | NÃO ESSENCIAL

```

A declaração dos conjuntos de relacionamentos difere um pouco da dos conjuntos de entidades. Como as associações entre as entidades são estabelecidas através das funções que estas desempenham e são identificadas pelas chaves dos conjuntos de entidades associadas, o seu atributo primário pode estar definido sobre diferentes domínios de valores, porque as funções desempenhadas pelos conjuntos de entidades não são necessariamente distintas.

```

.<conj rel> ::= <conj rel depend> * <conj rel propr>
.<conj rel propr> ::= DEFINE CONJ RELAC PROPR <identificador>
    <função> <atr não primário> *
.<função> ::= [<identificador><lista conj>] +
.<conj rel depend> ::= DEFINE CONJ RELAC DEPEND <identificador>
    <função> <atr não primário> *
.<lista conj> ::= { lista dos conjuntos de entidades que desem-
    penham a função considerada }

```

4.5.4. DEFINIÇÃO DO CORPO DO ESQUEMA

Além da identificação dos primitivos constituintes do Mundo Real, é necessário que sejam definidas a localidade funcional e os instantes da ocorrência destes primitivos numa situação real para melhorar não só o tempo de acesso aos dados como também o controle de integridade dos dados, atendendo assim as exigências de tempo real e o dinamismo de um Sistema de Controle.

4.5.4.1. DEFINIÇÃO DO MÓDULO, CHAVE SECUNDÁRIA E EVENTO

Agrupam-se os dados funcionalmente dependentes em MODULOS. A cada módulo atribui-se um grau de prioridade para

auxiliar o ABD a estabelecer critérios de atendimento das diferentes exigências entre vários usuários quanto à localidade funcional dos seus dados.

```
.<módulo> ::=DECL MODULO <identificador> <lista prim>
               <prioridade>
.<lista prim> ::= {lista das chaves das entidades/relacionamen-
               tos}
.<prioridade> ::= {número inteiro que define o grau de priori-
               dade funcional}
```

As CHAVES SECUNDARIAS constituem um recurso para acelerar a busca de informações. Através delas, os elementos dos primitivos (conjunto de entidades/relacionamentos) são agrupados conforme os valores dos seus atributos. Este agrupamento pode ser para cada valor distinto (por exemplo, todos os equipamentos que estão em estado de operação NORMAL), como para uma faixa de valores (por exemplo, todos os motores que apresentam um rendimento entre [0.85,0.90]).

```
.<chave secundária> ::=DECL CHAVE SECUND <identificador> {identi-
               ficador do atributo} {identificador do primi-
               tivo} <valor>
.<valor> ::= {estabelece os valores através dos quais são
               agrupados os elementos dos primitivos}
```

O EVENTO é um acontecimento que pode introduzir modificações no conteúdo de uma Base de Dados. As modificações podem demandar a intervenção dos usuários (modo MANUAL) ou ocorrem automaticamente sob o controle de SGBD (modo AUTOMATICO).

```
.<evento> ::=DECL EVENTO {especifica o tipo de evento}
               <modo de ação>
.<modo de ação> ::=AUTO | MANUAL
```

4.5.4.2. DEFINIÇÃO DO DOMÍNIO DE VALORES E DE TEMPO

O domínio de valores e o domínio de tempo impõem restrições sobre os valores permissíveis na Base de Dados para garantir a confiabilidade e a integridade semântica dos dados.

```
.<dom val> ::=DECL DOMINIO VALOR <identificador> <desc dom>
               <ordenação> <ação>
.<desc dom> ::= [<tipo> <restr dom>]*+
.<tipo> ::=REAL|CHARACTER[{outros tipos de dados suporta-
               dos pelo SGBD}]
.<restr dom> ::= <range> <validade>*
.<range> ::= {estabelece a faixa de dados válidos para um
               determinado tipo de dado, por exemplo,
               [0.0,30.0] e um range do tipo de dado real}
.<validade> ::= {estabelecem regras para as quais o range
               definido seja válido}
.<ordenação> ::= {estabelece a ordem dos tipos de dados que
               compõem o domínio de valores para constituir
               um valor do atributo, por exemplo, um valor
               como CA 1234 é formado por dois tipos de dados
               ordenados caracter-número inteiro}
.<ação> ::= {estabelece as ações de controle que devem ser
               atuadas quando os valores armazenados não
               pertencem ao domínio de valores, por exemplo,
               emissão de um sinal de alarme ao detetar um
               erro}

.<dom tempo> ::=DECL DOMINIO TEMPO <identificador> <período>
.<período> ::= {estabelece o intervalo de tempo cujos esta-
               dos dos primitivos devem estar na Base de
               Dados}
```

4.6. INTEGRIDADE E CONSISTENCIA ATRAVES DO MERB

Além de apresentar características favoráveis à descrição dos dados das aplicações em controle de processo, o modelo MERB inclui várias regras que possibilitam o controle de integridade semântica dos dados:

.Através da definição do tipo e do intervalo dos valores atribuídos às entidades ou aos relacionamentos tem-se maior garantia quanto à autenticidade dos valores a serem armazenados.

.Através da declaração dos conjuntos de entidades/relacionamentos dependentes, assegura-se que se as entidades proprietárias forem removidas, todas as entidades e relacionamentos dependentes também o serão. A mesma filosofia aplica-se aos relacionamentos definidos entre as entidades, quando uma das entidades que participa na sua formação for removida.

.Através da distinção de conjuntos e entidades de múltipla dependência em essencial e não essencial, permite-se modelar com precisão a interdependência de existência entre um objeto agregado /SMIT 80/ (entidade proprietária) e seus componentes. Quando a interdependência é essencial, a remoção de um componente implica na remoção do objeto, o que não acontece necessariamente quando a interdependência é não-essencial.

.Através da definição de EVENTO é fácil o controle do conteúdo da Base de Dados sob o ponto de vista da dinâmica de uma situação real, pois o ABD tem informações sobre como os dados devem ser modificados na ocorrência de um certo fato.

.O controle da integridade dos dados numa transação pode ser facilitado se for garantido que só é possível o acesso aos valores correspondentes aos estados dos primitivos anteriores ao intervalo de tempo em que ocorre a transação.

.O conjunto de relacionamentos binários é definido através dos papéis que as entidades desempenham. A inserção de um relacionamento só é admitido num conjunto de relacionamentos, se as entidades associadas pertencerem aos conjuntos de entidades que desempenham os papéis declarados.

CAPITULO 5: EXEMPLO DE APLICAÇÃO

5.1. INTRODUÇÃO

Como ilustração da utilidade do Modelo Entidade-Relacionamento Binário proposto no Capítulo 4, será apresentado neste capítulo um exemplo de sua aplicação em um processo real baseado no Sistema Metroviário de São Paulo METR 76/. O objetivo é mostrar a aplicabilidade do modelo para estruturação dos dados de um processo em que se deve levar em conta características de tempo real. A seguir será descrito sob o ponto de vista conceitual (item 2.4.1.1) o modelo operacional do processo metroviário.

O Sistema Metroviário é constituído de duas LINHAS que ligam diferentes pontos da cidade de São Paulo (atualmente o Metrô Paulistano tem as linhas: Leste-Oeste e Norte-Sul), ao longo das quais distribuem-se diversas ESTAÇÕES, com plataformas para embarque e desembarque dos passageiros. Em cada linha, existem duas VIAS PRINCIPAIS (ida e volta), que interligam as estações extremas. Embora, em operação normal, estas vias se destinem à circulação dos trens em sentidos pré-estabelecidos, há a possibilidade de desvio do tráfego de trens de uma via para outra através de PONTOS DE INTERLIGAÇÃO, permitindo assim garantir a continuidade de operação, mesmo quando um dos trechos da via for interrompido por trens defeituosos ou por problemas relacionados com a via. Cada parte da linha que apresenta pontos de interligação define uma ZONA DE INTERTRAVAMENTO. Existem ainda, ao longo das linhas, TERMINAIS onde os trens podem estacionar, fazer manobras de retorno, ou receber novas trajetórias de referência para o seu próximo percurso de ida-volta (anel de percurso) na linha. Numa das extremidades das vias de uma linha localiza-se o PATIO de manobras e oficinas onde os trens são recolhidos para manutenção,

limpeza, estacionamento e testes.

Em toda a sua extensão, uma linha não apresenta necessariamente a mesma geometria, podendo ter trechos com curvas acentuadas, com elevações ou com depressões. Para poder caracterizar com mais precisão as vias de uma linha, pode-se dividi-las em TRECHOS, de acordo com os acidentes topográficos (curvatura, altura, etc) e as condições ambientais (umidade, condição dos trilhos, etc). Atribui-se ainda a cada trecho um limitante superior de velocidade para garantir a segurança operacional do sistema. Este limitante superior é variante no tempo, pois ele depende não só dos acidentes topográficos como das condições ambientais sob as quais o trecho opera (por exemplo, todos os trens deverão ter a sua velocidade reduzida nos trechos a céu aberto em caso de chuva, ou quando trafegam na sombra dos trens antecessores. etc) (fig 5.1.).

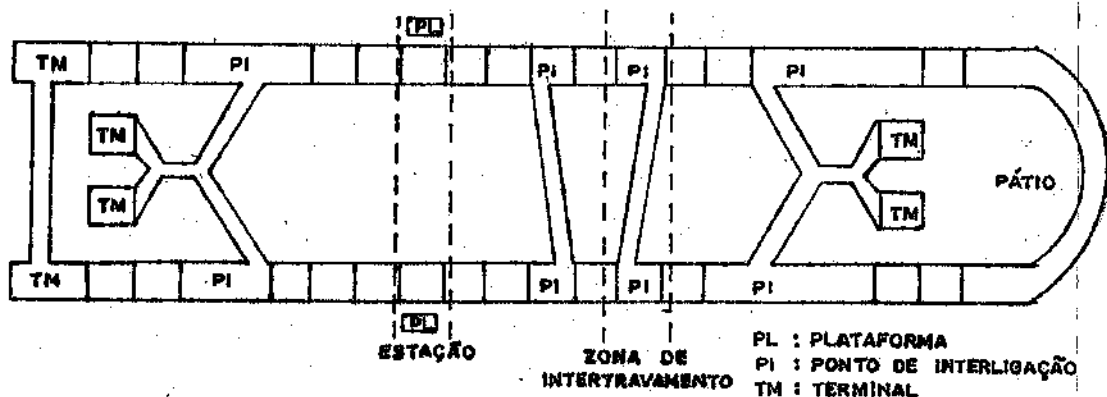


FIG 5.1: Esquema geral dos componentes básicos de uma linha

A operação dos trens nas linhas pode ser automatizada e a operação manual passa a ser necessária somente nos casos de mudança de sentido de tráfego ou em caso de ocorrência de defeitos na operação /METR 76/. Neste caso, os operadores que supervisionam ou monitoram o processo devem ter acesso aos dados sobre o comportamento do processo para introduzir correções operacionais que não são pré-programáveis..

Sob o ponto de vista de controle, distinguem-se no sistema de controle do Processo Metroviário as seguintes camadas hierárquicas (item 3.2.4.) (fig 5.2.) /MAGA 81/:

.otimização;

- . adaptação;
- . regulação;
- . implementação.

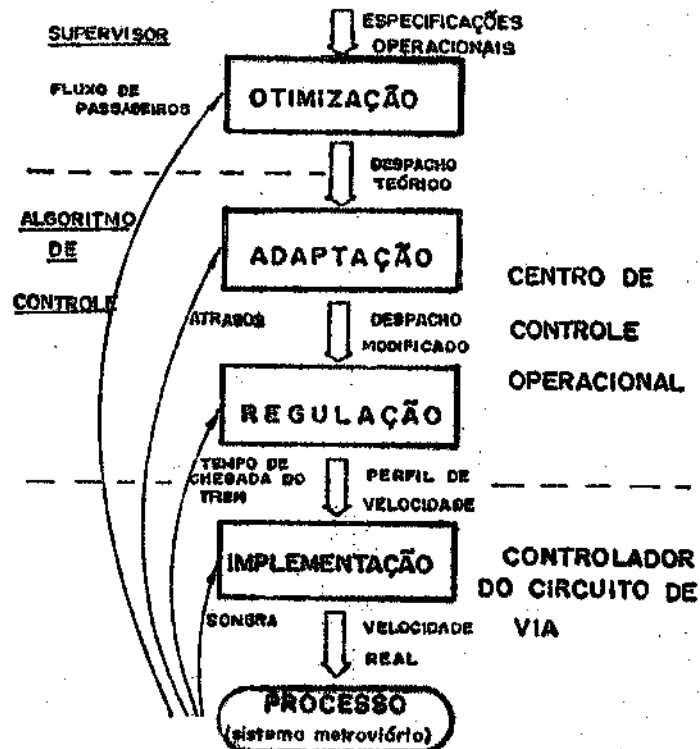


FIG 5.2: Camadas hierárquicas do sistema de controle metroviário

Na camada de IMPLEMENTAÇÃO, os controladores de via atuam em favor da segurança do sistema. No sistema metroviário existe uma especificação de segurança que estabelece a distância mínima entre dois trens consecutivos e esta restrição deve ser satisfeita através da definição de um limitante superior às velocidades máximas admissíveis nos trechos seguintes ao trecho ocupado. Deste modo, o perfil de velocidade de um trem, estabelecido nas estações ou terminais, pode sofrer modificações (causando atrasos), se ele encontra alguma 'sombra' deixada pelo trem antecessor no seu percurso até a próxima estação /METR 83/.

A camada de REGULAÇÃO corresponde principalmente à atuação de controle nas estações, sendo marcada por dois eventos bem definidos:

.CHEGADA DE UM TREM EM UMA ESTAÇÃO: quando é definido o tempo de parada e realizado o primeiro cálculo de seu novo nível de desempenho (perfil de velocidade) até a próxima

estação, de forma a minimizar o seu atraso em relação ao tempo de chegada operacional previsto para a próxima estação;

.PARTIDA DO TREM DE UMA ESTAÇÃO: se possível, é redefinido o nível de desempenho, levando em consideração o tempo real de parada e, portanto, as eventuais perturbações ocorridas devido ao fluxo de passageiros. Caso não seja possível o recálculo do novo perfil, o trem utiliza o valor já definido no momento da chegada do trem em estação e as perturbações introduzidas na estação serão propagadas para a próxima estação, onde serão introduzidas as devidas correções (fig 5.3.).

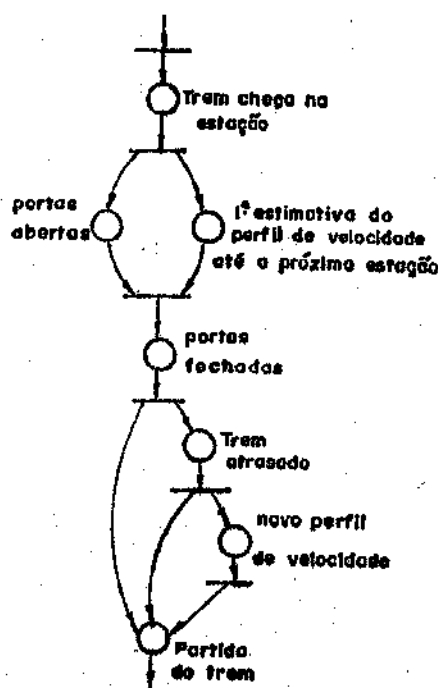


FIG 5.3: Sequência de eventos quando o trem chega a uma estação

A sequência de eventos mostrada na fig 5.3. pode sofrer pequenas variações dependendo do trecho compreendido entre as duas estações. Se os trechos pertencerem a uma zona de intertravamento, será necessário examinar as possíveis interferências que possam ocorrer entre dois ou mais trens com rotas convergentes e estabelecer a sequência de deslocamento destes trens, um por vez, na região, de forma que sejam minimizados os atrasos totais provocados por estas interferências (fig 5.4.) /METR 75/.

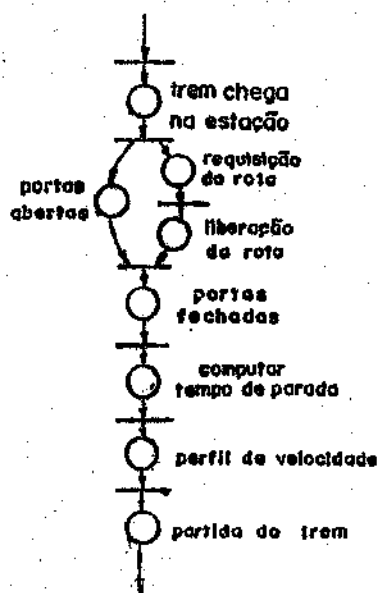


FIG 5.4: Sequência de eventos para o trem que chega em uma estação que precede uma Zona de Intertravamento.

Quando uma via é interrompida em algum ponto, há a necessidade de utilizar um ou mais trechos da outra via para a circulação dos trens nos dois sentidos, de modo que seja mantida a continuidade de tráfego dos trens em ambos os sentidos e evitado um acúmulo exagerado dos passageiros nas plataformas. Os trechos que servem temporariamente aos trens que percorrem em ambos os sentidos são denominados TRECHOS REVERSOS. Quando um trem chega a uma estação que precede os trechos reversos, as estratégias de controle também se modificam um pouco em relação às estações 'comuns', porque os trens passam a atravessar alternadamente, em grupos, os trechos reversos. Assim, todos os trens que cheguem na porta dos trechos reversos devem esperar pela sua vez de travessia. Uma vez liberada a rota, o deslocamento dos trens dentro da região de trechos reversos é idêntico ao movimento nos trechos em operação normal, com a diferença apenas no uso comum de uma plataforma para ambos os sentidos (fig 5.5) /METR 76/.

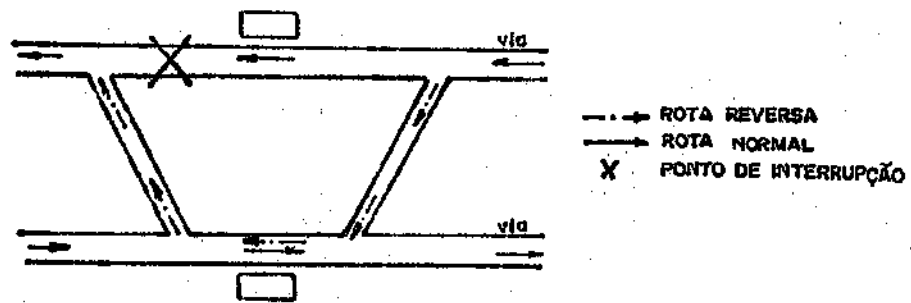


FIG 5.5: Configuração de trecho reverso

A camada de ADAPTAÇÃO engloba as estratégias de controle 'corretivas', que procuram adequar o referencial teórico estabelecido às situações reais perturbadas, através da definição de um novo referencial, garantindo a atuação normal das ações controladoras da camada de regulação. Estas estratégias se traduzem por /METR 75/:

.DEFASAGEM DO SISTEMA: deslocamento global dos horários pré-programados de todos os trens na linha;

.DISTRIBUIÇÃO DE INTERVALOS: deslocamento dos horários pré-programados dos trens que estão a frente de um trem com atraso, para eliminar os intervalos desiguais que existem entre dois trens consecutivos;

.DEFASAGEM DOS TRENS: deslocamento dos horários pré-programados dos trens que saem da zona de intertravamento fora da sequência escalada.

Na camada de OTIMIZAÇÃO são gerados os DESPACHOS TEÓRICOS, contendo a quantidade dos trens e os seus respectivos destinos numa linha, bem como o espaçamento mínimo entre dois trens consecutivos e outras restrições operacionais para atender satisfatoriamente a demanda dos passageiros. A partir do Despacho Teórico são definidas as trajetórias de referência teóricas EM UM ANEL DE PERCURSO para cada trem despachado do terminal. A determinação do Despacho Teórico leva em consideração o fluxo de passageiros nas estações e as restrições operacionais e técnicas, de modo a atender eficientemente a demanda global e manter um nível de conforto aceitável para os usuários /CURY 79//BEMH 82/.

Das quatro camadas de controle - Otimização,

Adaptação, Regulação e Implementação - somente a camada Implementação tem suas ações de controle geradas diretamente pelos controladores locais de via que reduzem automaticamente a velocidade dos trens em cada trecho a um nível admissível. As outras camadas onde decisões de controle mais sofisticadas são tomadas, os dados referentes ao comportamento do processo são transmitidos a um Centro de Controle Operacional (CCO) para serem geradas novas ações de controle, mantendo assim, sempre que possível, o nível de desempenho pré-estabelecido. Atualmente, os programas de aplicação (de controle) existentes no CCO são fortemente dependentes da estruturação dos dados, exigindo que os usuários destes dados (normalmente, especialistas em controle) tenham não só o conhecimento técnico dos problemas do processo em si, como também profundos conhecimentos em técnicas computacionais para organização interna destes dados conforme as exigências das aplicações em tempo real (item 2.2.). A utilização de um Sistema de Banco de Dados permitirá que um usuário especialista em Controle de Processos defina a natureza e a estrutura de dados necessários do seu ponto de vista e não do de especialistas em computação.

O requisito básico para uso adequado de um Sistema de Banco de Dados é que seja possível descrever com máxima clareza todos os aspectos relevantes do Mundo Real que se tem em mente do usuário (no caso, responsáveis pelo controle de processo) através do Modelo de Dados, obedecendo a semântica das informações.

No item 5.2, serão descritos os aspectos dinâmicos e estáticos do Sistema Metroviário para fazer um levantamento de informações relevantes do ponto de vista dos operadores que supervisionam o processo para atuação de controle em casos de emergência, conforme será mostrado no item 5.3. Estas informações se identificam com o Nível Conceitual na Hierarquia de Abstração dos Dados (item 2.4.1.1). Ressalta-se aqui que muitas aproximações e suposições são feitas para simplificar o exemplo.

No item 5.4 será apresentado um modelo do processo que corresponda ao Nível Descritivo (item 2.4.1.2.)

utilizando o MERB.

5.2. DESCRIÇÃO DO SISTEMA METROVIÁRIO

O Sistema Metroviário aqui analisado é composto dos seguintes recursos básicos que permitem a sua operação:

- .rede de trilhos ou linhas;
- .estação;
- .pátio;
- .zona de intertravamento;
- .trem

A seguir será analisada cada uma das camadas citadas no item 5.1 e serão mostradas as informações necessárias para tomadas de decisões por ações de controle mais convenientes.

5.2.1. CAMADA DE OTIMIZAÇÃO

A função da camada de Otimização é gerar um Despacho Teórico otimizado, respeitando as limitações impostas pela disponibilidade dos recursos e especificações operacionais definidas pela companhia de Metrô em função da segurança operacional, para atender as diferentes demandas dos passageiros em diferentes intervalos de tempo ao longo de um dia.

5.2.1.1. RECURSOS DISPONÍVEIS

No item 5.1, foi visto que a rede metroviária é constituída de uma ou mais linhas. Devido às variáveis condições físicas e operacionais ao longo de uma linha, cada linha é dividida em TRECHOS. Cada trecho pode ser caracterizado através das seguintes informações:

- .estado de operação;

- .geometria da via;
- .comprimento;
- .velocidade de segurança.

Para garantir a continuidade do serviço de transporte em ambos os sentidos mesmo com alguns trechos bloqueados, existem trechos que interligam duas vias em sentidos opostos através dos pontos de interligação (fig 5.6.). Estes pontos de interligação caracterizam as Zonas de Intertravamento. A descrição topológica de uma zona de intertravamento envolve os seguintes parâmetros /METR 80/:

- .trechos (plataformas) correspondentes à porta de entrada;
- .trechos (plataformas) correspondentes à porta de saída;
- .pontos de interligação;
- .estado de operação.

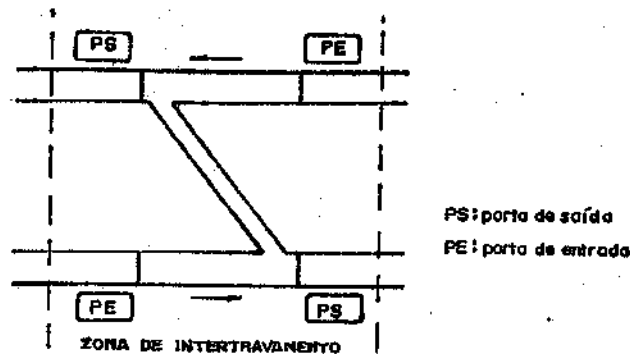


FIG 5.6: Zona de intertravamento definida por um conjunto de trechos

Em algumas Zonas de Intertravamento localizam-se os TERMINAIS onde os trens podem estacionar e receber trajetórias de referência para o próximo percurso no anel. Os terminais podem ser caracterizados pelas seguintes variáveis:

- .capacidade do terminal, isto é, a quantidade total de trens que podem se acomodar no terminal;
- .estado de operação.

A ESTAÇÃO é o local onde ocorre o embarque e desembarque dos passageiros. Normalmente, uma estação possui, no

mínimo, duas plataformas para embarque/desembarque dos passageiros. Quando há cruzamento de linhas numa estação, a quantidade de plataformas seria proporcional à quantidade de linhas que se cruzam.

Para o modelo do processo simplificado, considera-se que uma ESTAÇÃO pode ser completamente definida pelas seguintes variáveis:

- .tipo de estação (simples ou com cruzamento);
- .estado de operação;
- .capacidade de passageiros em cada plataforma.

O PATIO é o local onde os trens são recolhidos para manutenção, limpeza, estacionamento e testes. Por causa da variação de fluxo de passageiros nas estações ao longo de um dia, convém dividir o dia em períodos distintos, para um dos quais é solicitada uma certa quantidade de trens na linha. Os outros trens não escalados para serviço ficam estacionados, nos terminais ou no pátio, até a ordem do seu despacho. A variável que os operadores precisarão para decidir quantos trens poderão ser mantidos no pátio é a capacidade de alojamento dos trens.

A configuração física da rede metroviária pode ser integralmente definida a partir das informações anteriores, se forem estabelecidas as localizações relativas entre as diferentes unidades (pátio, terminais, estações, trechos, plataformas, etc).

Os TRENS são meios transportadores de passageiros que circulam ao longo das linhas metroviárias. Os principais atributos dos trens são /PAA 76//CURY 79/:

- .quantidade de vagões que o compõem;
- .comprimento médio dos vagões;
- .capacidade máxima dos passageiros;
- .aceleração máxima;
- .razão de frenagem;
- .máxima frenagem de emergência;
- .velocidade máxima;
- .estado de operação (por exemplo, em manutenção corretiva, em manutenção preventiva, etc.);

.local de estacionamento dos trens que não circulam na linha.

.nível de conforto máximo (COFMAX), isto é, a quantidade mínima de passageiros dentro do trem;

.nível de conforto mínimo (COFMIN), ou seja, a quantidade máxima de passageiros dentro do trem.

5.2.1.2. ESPECIFICAÇÕES OPERACIONAIS

São impostas certas restrições operacionais, de forma a assegurar a operação do Sistema Metropolitano dentro da faixa de segurança e a oferecer um nível de conforto aceitável aos usuários. Do total de restrições impostas ao sistema, serão consideradas neste modelo simplificado somente as seguintes:

.espaçamento máximo entre trens consecutivos (HDMAX);

.espaçamento mínimo entre trens consecutivos (HDMIN);

.defasagem máxima do sistema (ISYMAX);

.defasagem mínima do sistema (ISYMIN);

.defasagem máxima dos trens (TRMAX);

.defasagem mínima dos trens (TRMIN);

.tempo morto máximo (TMMAX), isto é, o tempo de parada máximo nos terminais;

.tempo morto mínimo (TMMIN), isto é, o tempo de parada mínimo nos terminais;

.tempo de parada máximo nas estações (DWMAX);

.tempo de parada mínimo nas estações (DWMIN);

5.2.1.3. FLUXO DE PASSAGEIROS

A determinação do Despacho Teórico será feita levando-se em consideração o histórico do fluxo de passageiros nas estações, de modo que seja atendida a demanda global dos passageiros, sem violar as restrições impostas pelas especificações operacionais e disponibilidade dos recursos /BEMH 82/.

A variação de fluxo de passageiros nas estações ao longo do dia dá-se de modo contínuo e portanto, para efeito de geração de despachos teóricos otimizados segundo um critério de manutenção de nível de conforto adequado, é necessário que seja estimado o número médio de passageiros por hora (taxa de passageiros), que afluem a uma estação com destino a qualquer outra em intervalos de uma hora ou de quinze minutos (em horas de pico de demanda). Estas informações são resultados de levantamentos estatísticos usando o fluxo de embarque/desembarque de passageiros em cada estação. Para o modelo, será representado o fluxo de passageiros como:

.taxa de passageiros entre duas estações (origem e destino) nos diferentes intervalos de tempo ao longo de um dia.

5.2.1.4. DESPACHO TEÓRICO

A partir do fluxo de passageiros, são alocados os trens e definidas as trajetórias de trens ao longo das linhas de modo a obter o nível de conforto desejado com um mínimo custo operacional, através do DESPACHO TEÓRICO. Entre as variáveis do Despacho Teórico, citam-se:

.tempo de parada nominal em cada estação EST (ou plataforma) - DWNOM(EST);

.velocidade nominal entre duas estações consecutivas (equivalente ao tempo de percurso para uma dada velocidade nominal entre duas estações) - PLNOM(EST);

.quantidade total de trens a serem alocados n linha - NTREM;

.quais os trens a serem alocados para linha;

.tempo de despacho (TDESP) e o respectivo destino (DEST) em cada terminal (TM);

.tempo morto no terminal TM, isto é, tempo de manobra do trem - TMRT(TM);

.estabelecimento de trechos reversos, se for necessário;

.definição de quantas e quais as Zonas de Intertravamento a serem utilizadas.

A partir das informações básicas acima, muitas informações úteis poderão ser derivadas para ajudar os operadores nas tomadas de decisão:

.ALINHAMENTO DE ROTA: para as linhas que apresentam uma ou mais zonas de intertravamento ativadas, o trem, ao entrar na porta da região, deve ter a sua rota alinhada, de modo a conduzi-lo ao seu destino correto. Em cada Zona de Intertravamento, distinguem-se basicamente três tipos possíveis de rotas para os trens que a atravessam (fig 5.7.):

.reto: quando o trem não se desvia da via principal;

.diagonal: quando o trem entra no terminal (se existir terminais na Zona de Intertravamento considerada);

.curvo: quando o trem passa de uma via a outra.

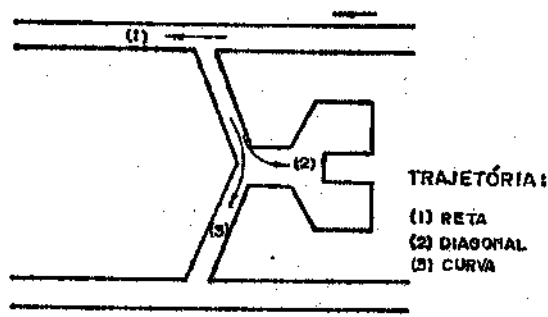


FIG 5.7: Diferentes rotas numa zona de intertravamento

.SEÇÕES: por causa das estratégias de controle adotadas (item 5.2.2.), é útil distinguir uma linha em operação em várias SEÇÕES intercaladas entre as Zonas de Intertravamento. A seção se caracteriza pela existência de uma única rota de percurso para os trens e é composta de uma sequência de trechos operantes (fig 5.8.). O bloqueio de qualquer um destes trechos implica necessariamente na desativação da seção, requerendo a definição de novas trajetórias de controle (fig 5.8.). O modelo apresentado considera que uma seção é caracterizada pelas seguintes variáveis:

.trecho (plataforma) correspondente à porta de entrada;

- .trecho (plataforma) correspondente à porta de saída;
- .tipo de operação, sob o ponto de vista do sentido de tráfego dos trens sobre ela (unidirecional ou bidirecional);
- .quantidade de estações operantes.

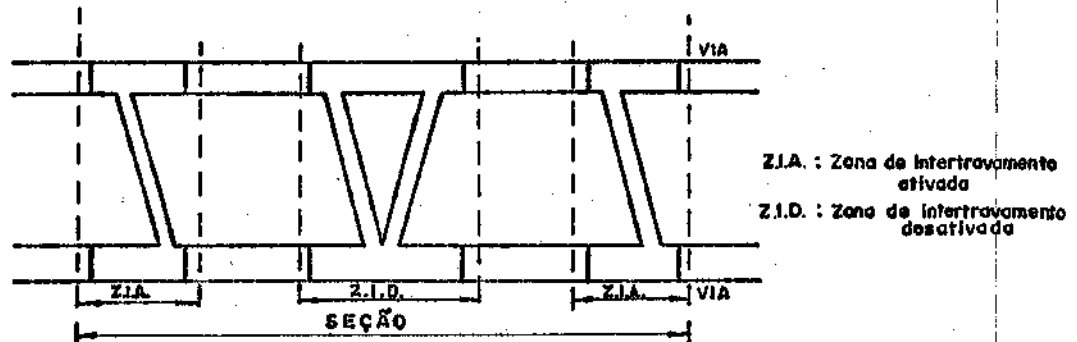


FIG 5.8: Divisão lógica da linha em seção e zona de intertravamento

.TIPOS DE PLATAFORMAS: segundo as estratégias de controle utilizadas, as ações de regulação só atuam nas plataformas (item 5.1) e a forma como elas atuam (a sequência de procedimentos) depende das regiões que sucedem as plataformas (fig 5.3, fig 5.4). Portanto, é útil diferenciar as plataformas quanto às regiões que elas precedem:

- .plataforma que precede uma seção reversa (bidirecional);
- .plataforma que precede uma zona de intertravamento;
- .plataforma comum.

.DISTANCIA ENTRE ESTAÇÕES E TERMINAIS: esta informação é importante para estimar o tempo de percurso médio de um trem entre dois pontos operacionais consecutivos para uma determinada velocidade de operação.

O Despacho Teórico gerado no nível de Otimização é utilizado como trajetória de controle referencial (trajetória teórica) para as estratégias de controle aplicadas nas outras camadas (Adaptação e Regulação). As informações que estas estratégias de controle necessitam são as relacionadas aos recursos efetivamente usados na operação do sistema metroviário.

Portanto, sob o ponto de vista operacional, é interessante ainda distinguir os recursos que participem na operação ativamente e os que fiquem temporariamente ociosos.

O fluxo de dados envolvido na Camada de Otimização é esquematizado na fig 5.9.

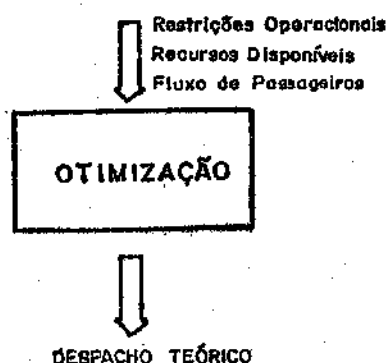


FIG 5.9: Fluxo de dados na camada de Otimização

5.2.2. CAMADA DE ADAPTAÇÃO

Será visto no item 5.2.3 que, na camada de Regulação, os controladores atuam no sentido de ajustar o nível de desempenho (velocidade) e o tempo de parada de cada trem em cada plataforma para que o desvio entre a trajetória real e a trajetória teórica gerada no Despacho seja mínimo. Esta regulação só apresenta um desempenho satisfatório em situação ideal ou com pequenas perturbações. Em situações fortemente perturbadas, devem-se introduzir as correções nas trajetórias teóricas de modo a atender as condições de momento para permitir que o regulador funcione de maneira normal. Se, mesmo com as correções feitas, o sistema não consegue retornar à sua operação normal modificada, os operadores devem ser notificados e, possivelmente, será gerado um novo Despacho Teórico.

Os quatro principais indicadores das perturbações que ocorrem numa linha são:

.ATRASSO TEÓRICO (ITHLAT) de um trem NTR, definido como:

$$ITHLAT(NTR) = TCH(NTR) - [TCT(I, NTR) + SYSOFF + TROFF(NTR)] \quad (5.1)$$

.ATRASSO OPERACIONAL (IOPLAT) de um trem NTR, definido como:

$$IOPLAT(NTR) = TCH(NTR) - [TCM(I, NTR) + SYSOFF + TROFF(NTR)] \quad (5.2)$$

.ATRASSO RECUPERAVEL (IARL) de um trem NTR em cada seção, definido como:

$$IARL(NTR) = \sum_{K=EST}^{IFS} [DW(K, NTR) - DWMIN] + \sum_{K=EST}^{IFS} [TPE(K, PLN) - TPE(K, PL1)] \quad (5.3)$$

.ATRASSO IRRECUPERAVEL (IATIR) de um trem NTR em cada seção, definido como:

$$IATIR(NTR) = ITHLAT(NTR) - IARL(NTR) \quad (5.4)$$

em que,

TCH(NTR), tempo de chegada real do trem NTR na estação;

TCT(I, NTR), tempo de chegada teórico do trem NTR na estação I;

TCM(I, NTR), tempo de chegada operacional do trem NTR na estação I;

SYSOFF, defasagem de tempo atual do sistema;

TROFF(NTR), defasagem de tempo atual do trem NTR;

DW(I, NTR), tempo de parada na estação I do trem NTR;

DWMIN, tempo de parada mínimo imposto pelo sistema;

TPE(I, PLN), tempo de percurso da estação I até I+1 em velocidade nominal;

TPE(I, PL1), tempo de percurso da estação I até I+1 em velocidade máxima permissível;

IFS, última estação dentro de uma seção;

EST, estação em que se encontra o trem NTR.

As três estratégias de controle citadas no item 5.1 têm como objetivo uniformizar o fluxo de trens na linha, através da distribuição de atrasos ou aplicar um offset a todo o sistema para acelerar a recuperação de uma determinada perturbação, como se mostra a seguir /METR 75/:

.ESTRATEGIA DE DEFASAGEM DO SISTEMA: aplica-se um atraso global na linha (SYSOFF), obtido a partir da média dos atrasos irrecuperáveis de todos os trens numa seção e de todas as seções numa linha (fig 5.10);

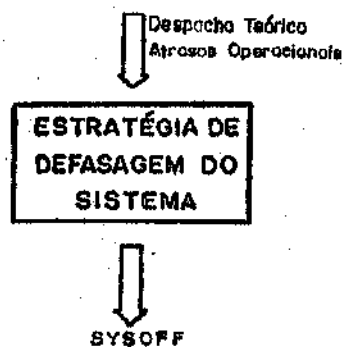


FIG 5.10: Fluxo de dados na estratégia de Defasagem do Sistema

.ESTRATEGIA DE DISTRIBUIÇÃO DE INTERVALOS: todos os trens com atraso operacional em cada seção são detectados e procura-se distribuir este atraso entre os trens que estão a sua frente e que ocupam a mesma seção, através da modificação dos tempos de chegada teóricos destes trens em cada estação. Os novos horários de chegada são conhecidos como TEMPOS DE CHEGADA OPERACIONAIS (fig 5.11).



FIG 5.11: Fluxo de dados na estratégia de distribuição de intervalos

.ESTRATEGIA DE DEFASAGEM DO TREM: tem por finalidade corrigir os horários teóricos de trens que saem de uma Zona de Intertravamento fora da sequência escalada, resultante da ação de estratégia de sequenciamento do movimento dos trens que concorram no uso de uma mesma Zona de Intertravamento. Esta correção é introduzida nas equações através da variável TROFF (fig 5.12).

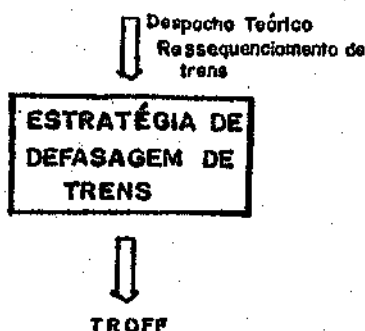


FIG 5.12: Fluxo de dados na estratégia de defasagem de trem

Vale ressaltar aqui que as estratégias de controle acima, conhecidas também como Estratégias de controle de alto nível /METR 75/ /METR 83/, só se aplicam em operações normais, ou seja, quando não são envolvidos percursos em seções reversas. Nas operações com rotas variantes, não faz sentido usar estas estratégias, uma vez que a própria natureza de operação não permite que se tenha uma distribuição regular dos trens nas linhas.

5.2.3. CAMADA DE REGULAÇÃO

As estratégias de controle nesta camada têm a finalidade de definir o tempo de parada (DWELL) de um trem NTR e o seu perfil de velocidade (PL) até a próxima estação, de forma a minimizar a diferença entre o tempo previsto de chegada na próxima estação e o tempo de chegada operacional pré-estabelecido, isto é:

$$\underset{\substack{\text{Min } T \\ \text{DWELL} \\ \text{PL}}}{T} = TCH(I) + DWELL(I) + TPS(I, PL) - TCM(I+1, NTR) + SYSOFF + TROFF(NTR) \quad (5.5)$$

onde,

TCH(I) é o tempo de chegada real do trem a estação I;

DWELL(I) é o tempo de parada na estação I;

TPS(I, PL) é o tempo de percurso até a próxima estação I+1, com uma velocidade PL;

$TCM(I+1, NTR)$ é o tempo de chegada operacional do trem NTR na estação $I+1$;

$SYSOFF$ é o valor de defasagem do sistema;

$TROFF$ é o valor de defasagem do trem NTR.

A equação (5.5) só é válida se forem levadas em consideração as restrições operacionais do sistema, tais como:

.antes de entrar numa Zona de Intertravamento, o trem deve esperar pela liberação da rota, que pode provocar um tempo de parada adicional inevitável;

.antes de entrar numa seção reversa, o trem também deve esperar pela liberação da rota que, normalmente, exige um tempo muito maior que o tempo de parada para embarque/desembarque dos passageiros, etc.

A figura 5.13 mostra a sequência de eventos como consequência da entrada dos trens nas plataformas de diferentes tipos - comum, que antecede zona de intertravamento, que antecede seção reversa.

muito grande de ações de operadores pode ocorrer. Enquanto o nível de atuação dos operadores depende diretamente do grau de automatização do processo, a qualidade das ações depende, principalmente, da experiência dos operadores e do seu conhecimento do estado global, no instante em que for necessária a sua atuação.

O modelo usado como exemplo considera que os operadores atuem normalmente na Camada de Otimização, através da modificação do Despacho Teórico, e supervisionem os controladores das camadas inferiores, para assegurar a operação correta destes. Em casos de emergência, como operações com rotas variantes, exige-se uma atuação e monitoração mais frequente dos operadores. A seguir, serão descritas sucintamente as principais funções das ações dos operadores que serão levadas em consideração /METR 80//MAGA 81/.

5.3.1. DESPACHOS EXTRAS

Entre os despachos extras, distinguem-se:

.despacho extra do pátio: quando se deseja incluir um ou mais trens, a partir do pátio, na linha que está em operação normal;

.despachos extras dos terminais: quando se deseja inserir novos valores na tabela de despacho de cada terminal.

5.3.2. MODIFICAÇÃO NOS DESPACHOS

Neste caso, tem-se como objetivo a modificação da tabela de despacho pré-programado para um determinado terminal.

5.3.3. CANCELAMENTO DE DESPACHO

Neste caso tem-se como finalidade o cancelamento de um valor da tabela de despacho pré-programado para um terminal.

5.3.4. MODIFICAÇÃO NO TEMPO DE PARADA

Neste caso, deseja-se que o tempo de parada para um determinado trem, ou para vários trens, seja modificado, conforme um dos seguintes tipos de modificação:

- .modificar o tempo de parada de um trem em uma determinada estação;

- .modificar o tempo de parada de um trem a partir de uma estação;

- .modificar o tempo de parada de todos os trens a partir de um trem específico numa estação.

5.3.5. MODIFICAÇÃO NO NÍVEL DE DESEMPENHO

Esta ação permite que o nível de desempenho (velocidade) definido para um determinado trem, ou para vários trens, seja alterado, através de:

- .modificação do nível de desempenho de um trem em uma determinada estação;

- .modificação do nível de desempenho de um trem a partir de uma determinada estação;

- .modificação do nível de desempenho dos trens que passam, a partir de um determinado trem, numa estação especificada.

5.3.6. HORARIO DEFASA

Através desta ação o operador poderá controlar os limites do SYSOFF (defasagem do sistema) ou colocá-lo num

valor pré-fixado.

5.3.7. TREM DIRETO

Para melhor distribuir os trens nas vias após alguma perturbação, ou para evitar um atraso maior de um ou mais trens, usa-se, muitas vezes, o recurso de trem direto, isto é, cancelar a parada do trem pré-programada numa certa estação, através da alteração da trajetória de referência dos trens pelos operadores.

5.4. MODELAMENTO DO SISTEMA METROVIARIO ATRAVES DO MERB

Os itens 5.1 a 5.3 descreveram o Processo Metropolitano, apresentando todas as informações relevantes referentes ao aspecto operacional; portanto, o processo está definido sob o ponto de vista conceitual (fig 2.8.). Assim, utilizando os primitivos definidos no MERB, o processo será descrito formalmente, tendo por objetivo estruturar as informações necessárias às tomadas de ações de controle, do ponto de vista de um operador situado na camada de Otimização de Nível de Processo.

Essa descrição formal (Nível Descritivo) será exposta usando os símbolos gráficos mostrados na fig 5.14.

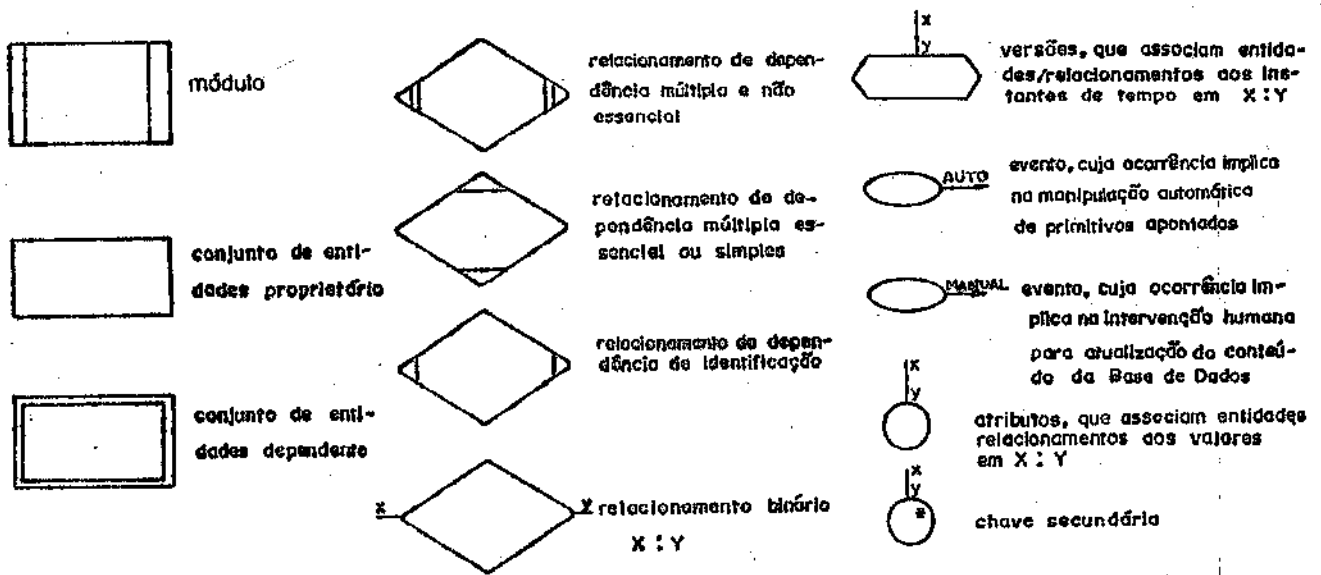


FIG 5.14: Símbolos gráficos do MERB

5.4.1. INFORMAÇÕES DO PROCESSO FÍSICO

Para o modelamento, os seguintes domínios de valores e de tempo serão utilizados conforme fig 5.16. a fig 5.27:

$A = \{\text{operante, inoperante}\}$

$B =$ conjunto de descrições topográficas e geométricas dos trechos

$C = \{x/x \in R^+\}$

$D =$ conjunto de identificadores de plataforma

$E =$ conjunto de identificadores de estação

$F = \{x/x \in N\}$

$G = \{\text{comum, com cruzamento de linhas}\}$

$H =$ conjunto de identificadores de terminal

$J =$ conjunto de identificadores de trecho

$L =$ conjunto de identificadores de pátio

$M =$ conjunto de identificadores de linha

$N =$ conjunto de identificadores de Zona de Intertravamento

$A' =$ conjunto de identificadores de trem

$C' = \{x/x \in R\}$

D.A. = domínio temporal, abrangendo todos os instantes correspondentes a um dia

D.B. = domínio temporal, abrangendo todos os instantes correspondentes a uma hora

5.4.1.1. RECURSOS DISPONIVEIS

Estas informações são importantes para que os operadores tenham um conhecimento global da capacidade da rede metroviária. É suposto que a informação semântica: <todos os recursos que estão operantes> seja de consulta frequente, o que justifica a existência de acesso direto pelo estado <OPERANTE> (item 5.2.1.1) (fig 5.15).

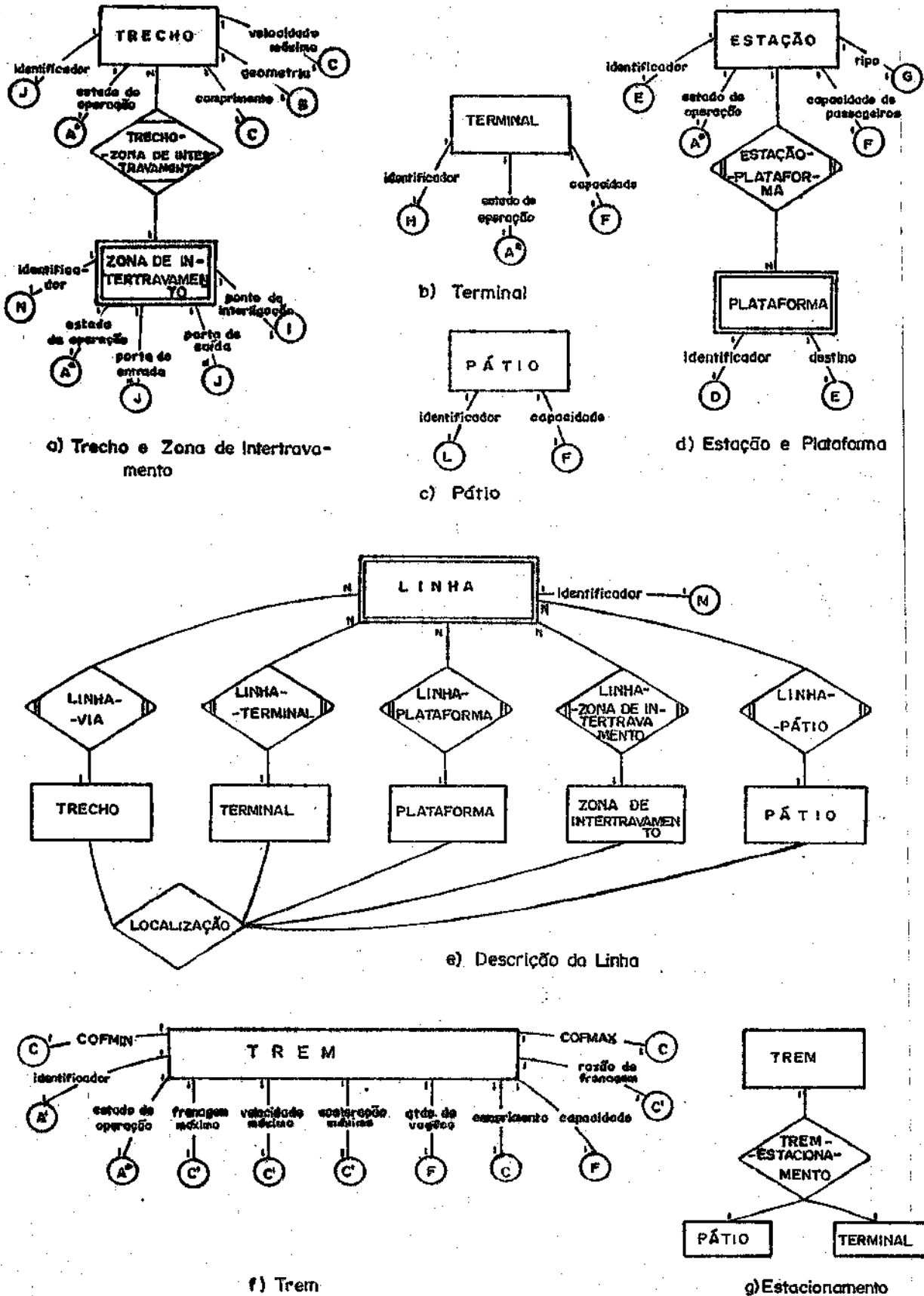


FIG 5.15: Recursos Disponíveis

5.4.1.2. ESPECIFICAÇÕES OPERACIONAIS

A rede metroviária é constituída por duas linhas e para assegurar as especificações operacionais, uma série de restrições sobre o movimento dos trens é imposta (item 5.2.1.2.) (fig 5.16).

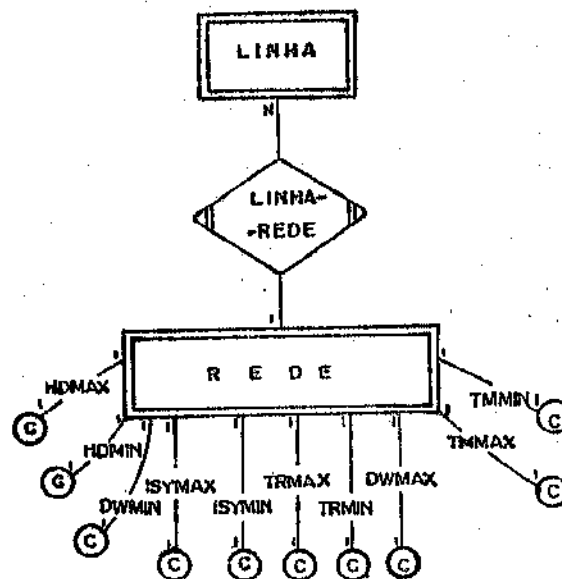


FIG 5.16: Especificações Operacionais

5.4.2. INFORMAÇÕES SOBRE FLUXO DE PASSAGEIROS

O fluxo de passageiros será definido como um relacionamento que se estabelece entre duas plataformas (estações), onde uma desempenha o papel de <DESTINO> e a outra, o papel de <ORIGEM>. Há a necessidade do registro de fluxo de passageiros correspondente a diferentes intervalos de tempo em um dia, isto é, é preciso guardar os diferentes estados do relacionamento (ORIGEM, DESTINO) em um dia para auxiliar a geração do Despacho Teórico (item 5.2.1.3) (fig 5.17).



FIG 5.17: Fluxo de Passageiros

5.4.3. INFORMAÇÕES SOBRE CONDIÇÕES OPERACIONAIS DO PROCESSO

O Despacho Teórico estabelece as condições sob as quais o sistema deve funcionar, de forma a assegurar a otimalidade da solução da trajetória de controle, definida com base no desempenho desejado pré-estabelecido (no processo, conforme o modelo adotado, o desempenho é estabelecido através do nível de conforto). Os trechos, as estações, os terminais, o pátio, os trens e as plataformas escalados para operação serão considerados RECURSOS OPERANTES, cujos estados devem ser supervisionados ou monitorados pelos operadores, que podem alterar o estado operacional através de despachos extras.

Nota-se que são definidos PLNOM (velocidade nominal) e DWNOM (tempo de parada nominal) para os trechos e estações operantes, respectivamente, na ocasião do despacho (item 5.2.1.4.) (fig 5.18.).

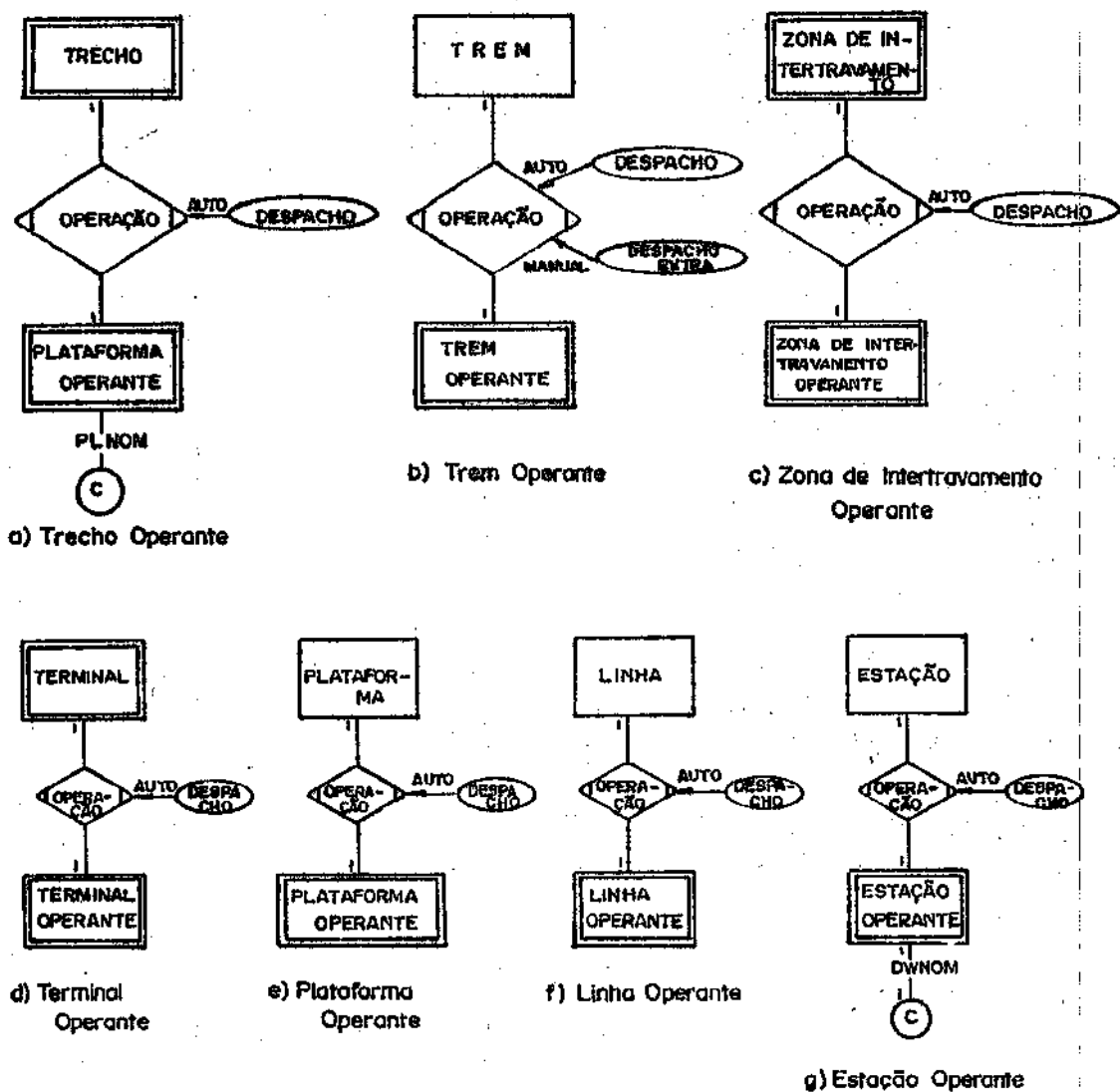


FIG 5.18: Recursos Operantes

É gerada também no Despacho Teórico, uma tabela de despacho dos trens em cada terminal, que leva em consideração o fluxo de passageiros, as restrições de segurança e o nível de conforto pré-estabelecido. Segundo estes horários é que os trens serão despachados dos terminais, depois de definidas suas trajetórias de referência (uma tabela de tempos de chegada teóricos em cada estação) para o seu próximo percurso. Além disso, são definidos os tempos de parada nominais em cada estação operante e a velocidade nominal em cada trecho operante (fig 5.19).

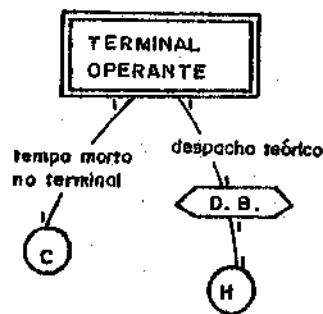


FIG 5.19: Horário de chegada em cada estação

Foi visto que, para as estratégias de controle adotadas, é vantajoso distinguir uma linha operante em seções e zonas de intertravamento. Tendo definidos os trechos operantes e as zonas de intertravamento, é imediata a derivação dessa informação. Cada seção guarda com os seus trechos componentes uma dependência essencial, pois a interrupção de um trecho (inativação de um trecho) causa sempre a definição de uma nova estratégia de controle e portanto, geração de novo Despacho Teórico (fig 5.20).

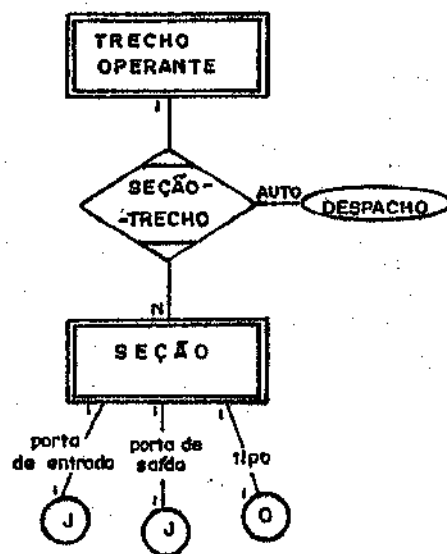


FIG 5.20: Seções

A Zona de Intertravamento é caracterizada pela existência de duas ou mais rotas, portanto é importante ter a

seguinte informação semântica: <para cada destino (terminal), existe uma rota na zona de intertravamento que leva o trem a ele> (fig 5.21).

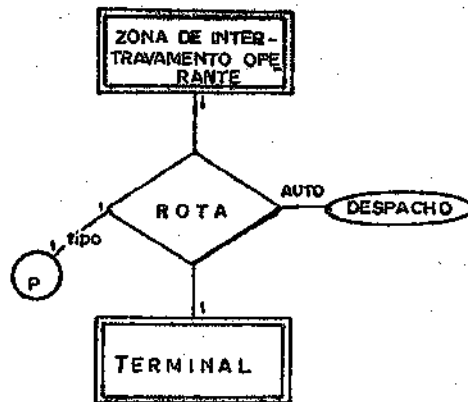


FIG 5.21: Descrição da Zona de Intertravamento

A distância entre duas estações operantes consecutivas numa seção e a distância entre a estação operante e os terminais podem ser modeladas através de um relacionamento <DISTANCIA> entre as entidades acima (fig 5.22).

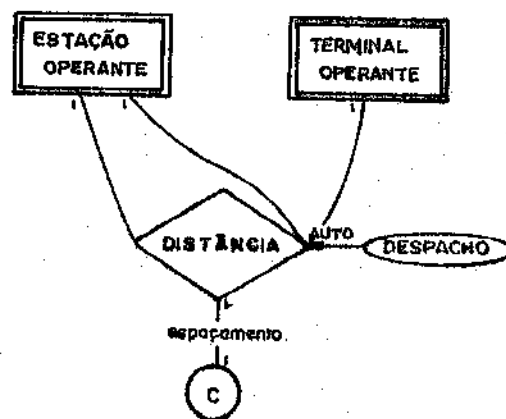


FIG 5.22: Distância

5.4.4. INFORMAÇÕES SOBRE OPERAÇÃO DO PROCESSO

A informação básica, para que as unidades de decisão possam acompanhar o comportamento dinâmico do processo, é

a localização dos trens ao longo das linhas operantes (item 5.2.4) (fig 5.23).

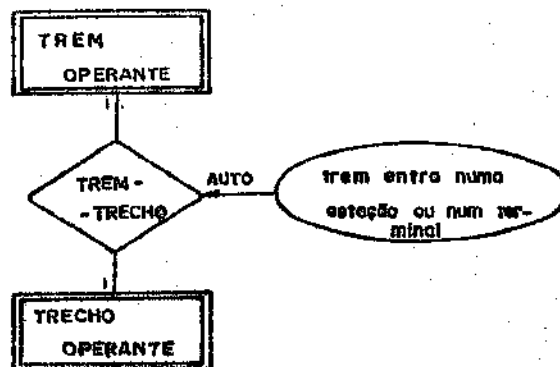


FIG 5.23: Localização de cada trem operante na via

As trajetórias de referência teórica (TCT), operacional (TCM) e real (TC) dos trens consistem em um conjunto de tempos de chegada destes trens em cada estação. A comparação entre estas trajetórias permite que os operadores avaliem a qualidade das ações de controle (item 5.2.3) (fig 5.24).

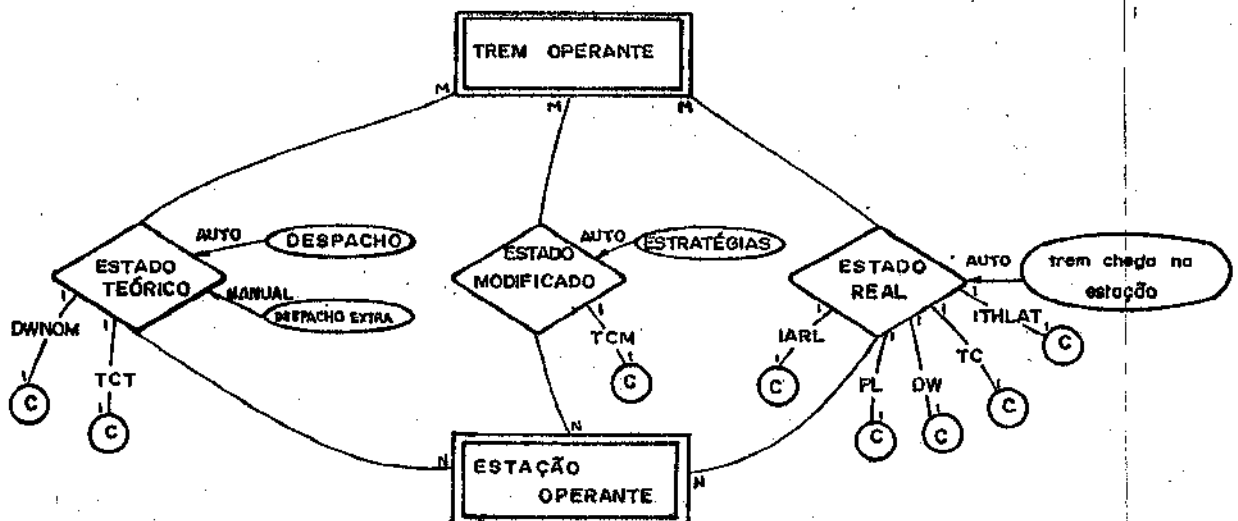


FIG 5.24: TCT, TCM e TC de cada trem operante

O tratamento do movimento de trem antes de entrar na Zona de Intertravamento é especial, pois há a necessidade de requisição de rotas antes de entrar nela, para evitar colisão entre os trens que apresentem rotas convergentes. Deste modo, a informação semântica: <os trens que requisitam o

uso de uma determinada Zona de Intertravamento) é importante e pode ser modelada como (item 5.2.3) (fig 5.25):

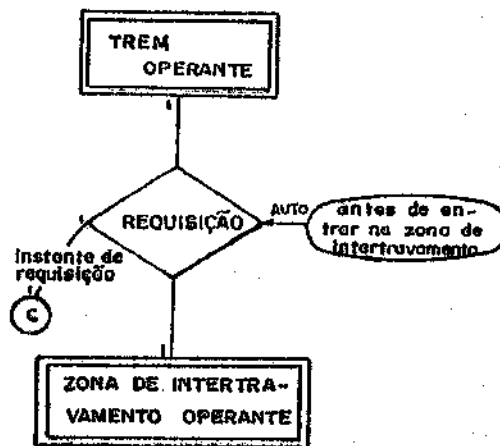


FIG 5.25: Requisição de rota

No item 5.2.2 foram descritas as estratégias de controle de alto nível que introduzem correções em torno do referencial ótimo gerado pelo Despacho Teórico, para que as ações reguladoras possam atuar normalmente. Estas técnicas geram os valores das correções e entre eles, citam-se o valor de defasagem do sistema (SYSOFF) e o valor de defasagem de trem (TROFF). O primeiro valor é uma característica de cada linha e o segundo, a característica de cada trem. Sob o ponto de vista dos operadores, é importante manter o histórico dos valores de SYSOFF para ter uma idéia do comportamento global do sistema (fig 5.26.).

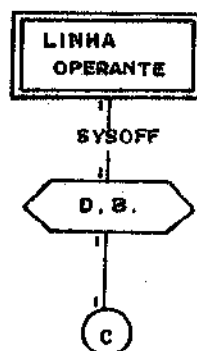


FIG 5.26: SYSOFF

5.4.5. INFORMAÇÕES SOBRE A LOCALIDADE FUNCIONAL DE DADOS

Foi afirmado no Capítulo 4, que o MERB permite que os usuários definam com maior precisão o tempo de acesso a um conjunto de dados de que eles necessitem. Com isso, o Sistema de Banco de Dados terá condições de organizar os dados, de modo que a sua recuperação e/ou atualização sempre ocorra em um intervalo de tempo que corresponda, aproximadamente, às exigências dos usuários. Existem três formas de informar ao sistema as características funcionais dos dados:

.MÓDULO: é utilizado para descrever um grupo de entidades/relacionamentos que são funcionalmente dependentes, isto é, o êxito da execução completa de uma tarefa em tempo real pode depender do tempo total de acesso parcial ou total destes dados;

.CHAVE SECUNDARIA: é utilizada para definir quais conjuntos de entidades/relacionamentos devem ter suas entidades/relacionamentos endereçados através dos valores de seus atributos, de modo a aumentar a velocidade de recuperação de um conjunto de dados que apresentem uma determinada característica comum;

.EVENTO: é utilizado para caracterizar um acontecimento que pode resultar em uma alteração, automática ou manual, do conteúdo da Base de Dados.

A definição dos EVENTOS e a sua interferência no conteúdo da Base de Dados e a utilidade de CHAVE SECUNDARIA já foram oportunamente mostradas nos itens anteriores (Itens 5.4.1 a 5.4.4). O MÓDULO estabelece a localidade funcional dos elementos da Base de Dados do ponto de vista da execução de uma determinada tarefa e sua definição é função do intervalo de tempo para execução desta tarefa. Serão tomadas as atividades dos operadores (item 5.3.) como base para modularizar as informações semânticas descritas nos itens anteriores:

Para a alteração dos despachos em cada terminal, os operadores precisam saber a disponibilidade dos trens nos estacionamentos (pátio ou terminais) e ter acesso às tabelas de despacho (fig 5.27).

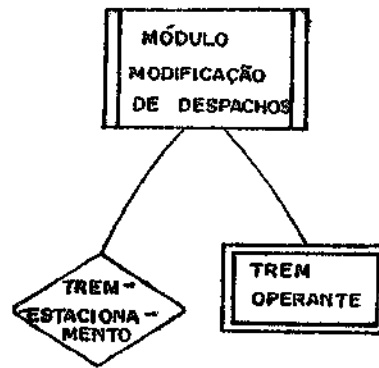


FIG 5.27: Modificação dos Despachos

Para a alteração da trajetória de referência de cada trem, é útil que os operadores tenham informações sobre a localização relativa dos trens nas linhas (fig 5.28).

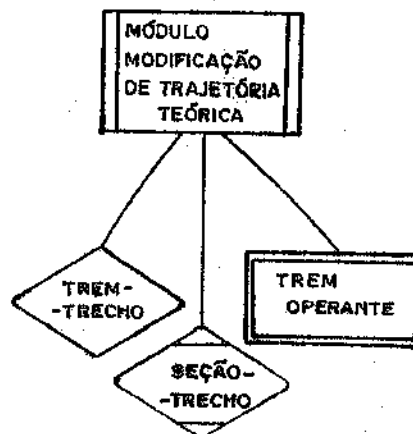


FIG 5.28: Modificação das Trajetórias Teóricas

E, finalmente, para a alteração das restrições técnicas e operacionais pré-estabelecidas, os dados do sistema metroviário funcionalmente dependentes são como mostrados na fig 5.29.



FIG 5.29: Modificação de restrições operacionais

5.5. CONCLUSÃO

Ao longo deste capítulo, foi descrito o controle de um processo metroviário e foram destacados cinco tipos de informações utilizadas para esta finalidade:

- . informações sobre processo físico;
- . informações sobre fluxo de passageiros;
- . informações sobre condições operacionais do processo;
- . informações sobre operação do processo;
- . informações sobre a localidade funcional de dados.

Estas informações incluem a descrição física e da dinâmica operacional do processo físico e os dados necessários aos operadores humanos para monitorar o processo. Uma das características do processo metroviário é que ele apresenta uma constante de tempo de operação da ordem de segundos, de modo que o tempo máximo admissível para manipulação dessas informações e geração de ações de controle, deve estar dentro desta ordem de grandeza, para assegurar a confiabilidade das tomadas de decisão.

Utilizando os primitivos do modelo MERB, foi possível descrever, precisamente, a interdependência entre estas informações, garantindo não só a integridade do conteúdo da Base de Dados (item 4.6.) como a estruturação mais apropriada destas informações, de forma a atender as exigências do processo através

da formalização de fatos como: a interdependência de existência entre <TRECHO> e <SEÇÃO>, a necessidade de acesso direto pelo conteúdo do <ESTADO DE OPERAÇÃO>, a inserção de novas entidades como consequência do evento <DESPACHO> ou <DESPACHO EXTRA>, o agrupamento do conjunto de informações necessárias para tomada de decisão por uma ação controladora sob o ponto de vista dos operadores humanos (por exemplo, despachos extras, modificação de despachos, etc).

Observe-se que os modelos ditos convencionais (e até mesmo o Modelo Entidade-Relacionamento) não oferecem recursos para modelar as exigências citadas no parágrafo anterior (fig 4.1.), visto que são destinados a aplicações comerciais, onde o tempo não é fator crítico. Não há preocupação, nestes modelos, em formalizar rigorosamente as necessidades dos usuários quanto ao tempo de manipulação (acesso, atualização, etc) do conteúdo da Base de Dados. Da mesma forma, o tratamento e a manutenção quanto à integridade e consistência das informações são apoiados nos recursos especiais, como a definição de dependência de existência múltipla entre primitivos e versões no domínio de tempo.

CAPITULO 6: CONCLUSÕES

O Modelo de Dados proposto neste trabalho - Modelo Entidade-Relacionamento Binário (MERB) permite uma formalização da concepção dos usuários de Aplicações em Controle de Processo, conforme exemplificado no Capítulo 5 (processo metroviário). Com isso, mostra-se a viabilidade da utilização de Sistemas de Banco de Dados para tratamento de informações em Controle de Processo, visto que um dos requisitos básicos deste tipo de aplicação é a transmissão concisa e precisa do modelo do processo que tem na mente do usuário ao Sistema de Banco de Dados; desta forma, o ABD (Administrador de Banco de Dados) e/ou o SGBD (Sistema de Gerenciamento de Banco de Dados) podem estruturar adequadamente os dados considerados relevantes aos objetivos da Aplicação.

O Modelo proposto é uma extensão adaptada do Modelo Entidade-Relacionamento (MER), proposto por CHEN /CHEN 76/ sobre o qual podem-se apresentar as seguintes conclusões:

1) Eficiência do Modelo para descrição de aplicações em tempo real

São introduzidos no MERB três primitivos relacionados ao Controle de localidade funcional dos dados: CHAVE SECUNDARIA, EVENTO e MÓDULO. Isso permite que os usuários definam o conjunto de informações necessárias para a execução de uma tarefa em tempo real, de modo que elas sejam organizadas na Base de Dados para que o tempo de acesso (e/ou manipulação) a elas seja minimizado.

2) Eficiência do Modelo sob o ponto de vista de integridade e consistência

A classificação dos primitivos em: CONJUNTO DE ENTIDADES/RELACIONAMENTOS PROPRIETARIOS e DEPENDENTES (simples, múltiplo essencial e múltiplo não essencial), permite modelar com

precisão a interdependência de existência entre as entidades/relacionamentos, garantindo a integridade e consistência nas operações sobre o conteúdo da Base de Dados.

A integridade é também garantida com a definição do DOMÍNIO DE TEMPO e DOMÍNIO DE VALORES, para evitar que dados incorretos, dentro do contexto de uma Aplicação, sejam armazenados, sem a notificação dos operadores, pois, conforme mostrado no Capítulo 3, em Sistemas de Controle de Processo, a maioria dos dados são originados diretamente do Processo Físico (como por exemplo, por meio de sensores acoplados diretamente aos microprocessadores).

Além disso, é introduzido no modelo proposto o conceito VERSÃO que permite descrever os diferentes valores associados a um mesmo atributo de uma entidade (por exemplo, valores medidos de uma variável do processo), correspondentes aos diferentes pontos do espaço temporal numa forma mais natural para os usuários (itens 4.4.1.1. e 4.4.1.3.); este conceito também facilita o controle de integridade numa transação específica através do bloqueio de acesso aos valores da versão correspondente ao intervalo de tempo da transação.

Paralelamente aos estudos deste trabalho, foi implementado um núcleo de gerenciamento de dados, baseado no Sistema CORAS, originalmente desenvolvido na Universidade Darmstadt, Alemanha, objetivando a utilizá-lo como suporte ao modelo proposto.

No estágio atual de implementação, o processo de mapeamento entre os primitivos do MERB e os primitivos suportados pelo CORAS não é imediato, pois, muitas funções devem ser adicionadas e o conteúdo da Base de Dados deve ser estendido, conforme mostra o Apêndice 2.

Os testes feitos mostraram que o CORAS apresenta grande flexibilidade na manipulação dos dados, satisfazendo muitas exigências que devem ser atendidas em Aplicações de Controle de Processo. Talvez, o fator limitante de maior importância seja a velocidade máxima de manipulação dos dados que se consegue, já que esta velocidade depende não só do software, como também, do hardware.

Finalmente, são sugeridos como trabalhos futuros dentro da linha de pesquisa deste trabalho, os seguintes temas:

1)desenvolvimento e aperfeiçoamento do modelo MERB para descrição de Aplicações em tempo real genéricas;

2)desenvolvimento e implementação de uma interface entre o MERB e o CORAS;

3)estudo da viabilidade do uso do CORAS como núcleo de gerenciamento de dados para Aplicações em tempo real.

APENDICE 1: CORAS - CORE SYSTEM FOR ASSOCIATIVE STORAGE

CORAS /WST 83//VIWU 82//KLOS 81/ /KLNE 80/ é um sistema de software básico que, simulando memória associativa, permite a recuperação eficiente de informações, cujos nomes, posições ou associações com outras informações são conhecidas, viabilizando assim a sua utilização no suporte a gerenciamento de dados. Coras resultou de aprimoramentos do Sistema DATAS /ENGL 72/, sendo adequado para uso em computadores de médio porte e contém as seguintes funções básicas:

- .gerência: criação/cancelamento/estruturação de dados;

- .recuperação: operações matemáticas de conjunto/cópia/busca de dados;

- .outras utilidades: desativação/ativação do sistema/expansão da Base de Dados, etc.

Sob o ponto de vista lógico, o Sistema CORAS distingue dois elementos básicos:

- .entidades;

- .triadas.

ENTIDADE é o menor item de informação endereçável em Coras, podendo ser qualificado por um ou mais atributos. O atributo, cujos valores identificam univocamente as entidades na Base de Dados, é denominado CHAVE ou NOME, pois, ele poderá ser utilizado como o 'endereço lógico' para acesso às entidades. Os dados referentes aos outros atributos são arbitrários, seu significado não influi diretamente na manipulação do conteúdo da Base de Dados e seus valores podem apresentar tamanho variável.

Distinguem-se quatro classes de entidades no CORAS (fig. A1.1):

.entidade simples: corresponde à caracterização de um objeto unitário;

.entidade lista: agrupa em um conjunto ordenado as entidades simples e/ou lista, existentes na Base de Dados;

.entidade conjunto: agrupa logicamente as entidades existentes na Base de Dados. O termo 'conjunto' tem sentido equivalente ao conjunto matemático e são válidas todas as regras de operações sobre conjuntos, como união, intersecção, diferença, etc, entre as entidades desta classe. Uma entidade conjunto pode ter como elementos: entidade simples, entidade lista, entidade conjunto e entidade relação;

.entidade relação: estabelece relações binárias entre entidades existentes na Base de Dados através da definição de TRIADAS.

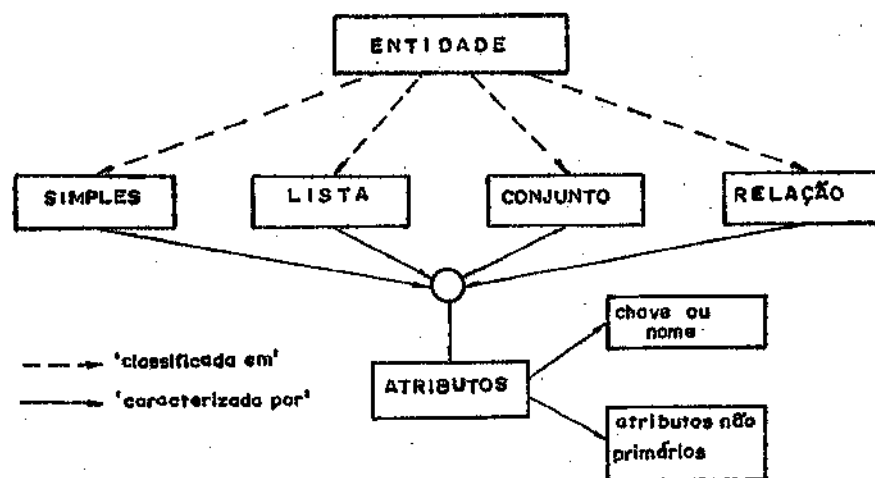


FIG A1.1: Quatro classes de entidades

O interrelacionamento entre estas quatro classes de entidades é mostrado na fig. A1.2.

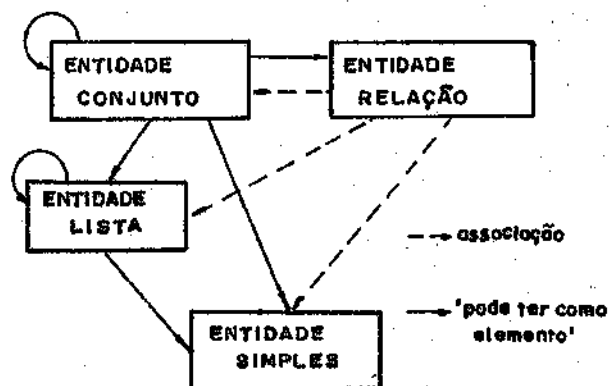


FIG A1.2: Interrelacionamento entre as entidades

TRIADA é a unidade elementar das associações lógicas estabelecidas entre as entidades. Consiste em uma tripla de entidades, onde uma entidade relação estabelece a relação binária entre duas outras, não pertencentes à classe de relação. Através dela, é possível acessar um dado, conhecendo suas possíveis associações com outros dados.

Para cada tripla de entidades associadas, existem três possíveis formas de acesso (através de cada uma das três entidades) e todas são implementadas fisicamente no CORAS numa estrutura de anel, obtendo assim as características de simetria na recuperação de qualquer tipo de informação, via associação. Estas diferentes formas de entrada correspondem a diferentes PERMUTAÇÕES de uma mesma triada. Uma triada ordenada em ABC significa que se poderá ter acessos diretos às entidades B e C, conhecendo-se A. A definição de outras permutações como BCA aumentará a flexibilidade de busca, pois, a existência de uma tripla BCA indicaria que, pela entidade B, é possível acessar as entidades C e A e assim por diante.

No CORAS, a técnica utilizada para acesso pela chave é o acesso por tabela hash, ou seja, através da função hashing, obtém-se a posição da chave na tabela hash, onde está o endereço físico de todos os dados qualificadores referentes à entidade. A alocação do espaço necessário para a tabela hash é estática, pois, o algoritmo de hashing é dependente da quantidade total de posições existentes para alojar as chaves na tabela. E

a alocação do espaço necessário para armazenar as triadas e os valores não pertencentes à chave é dinâmica, dependendo da quantidade de associações estabelecidas (no caso das triadas) e do tamanho do string que caracteriza a entidade (no caso dos valores). Assim, subdivide-se a Base de Dados em três partes, quanto à estruturação de dados (fig. A1.3):

.domínio de NOMES: corresponde à tabela hash, onde estão os nomes das entidades e os endereços físicos dos dados referentes a elas;

.domínio de DADOS: contém, além dos dados que qualificam as entidades, todas as outras informações adicionais que facilitem a recuperação e a manipulação dos dados e a manutenção de consistência na Base de Dados;

.domínio de TRIADAS: contém as triadas definidas e organizadas em forma de anel, para garantir a recuperação quasi-simétrica das entidades a partir das suas associações com outras entidades.

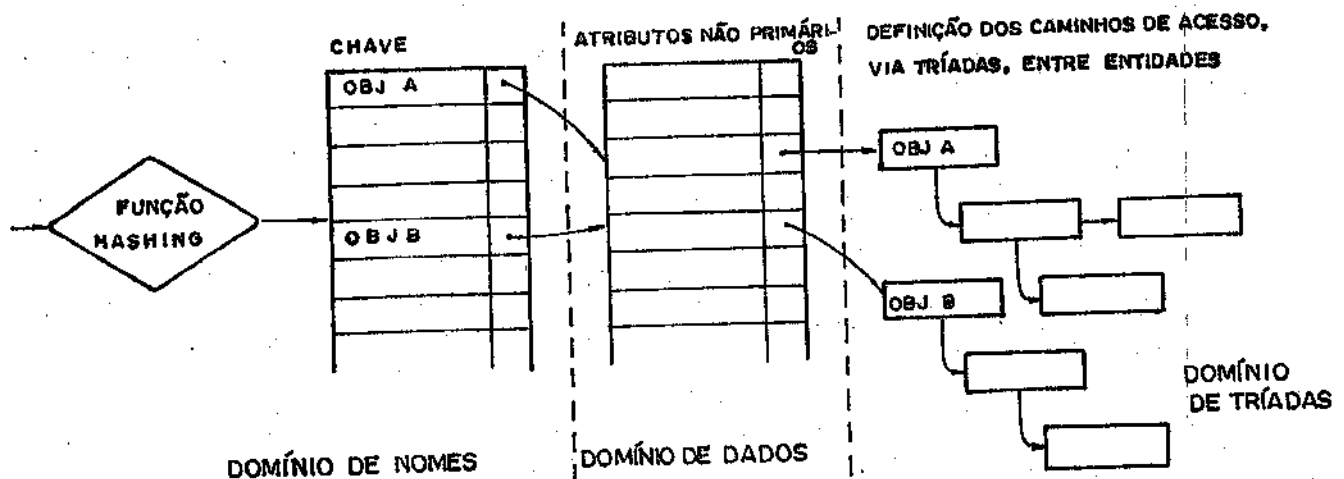


FIG A1.3: Três domínios distintos no CORAS.

Cada um destes domínios é subdividido em páginas de tamanho correspondente ao do buffer de E/S, de modo a otimizar o acesso aos dados armazenados na memória secundária (disco). Na memória principal, é mantida uma certa quantidade de páginas de nome, páginas de dados e páginas de triadas, conforme a sua capacidade. O transporte de página é feito automaticamente por um sistema de paginação, de maneira transparente aos usuários, durante a busca de uma informação. Dois parâmetros são levados em consideração no processo de paginação:

.fator de utilização: quando a memória

principal já está carregada com o número máximo de páginas admissíveis e uma nova página deve ser lida do disco, a página a ser substituída é aquela que tiver o menor fator de utilização, isto é, a que for menos utilizada;

„localidade das páginas“ devido à estrutura encadeada (blocos indivisíveis interligados conforme o espaço necessário para armazenar as informações de tamanho variável), em certas operações, é interessante manter na memória principal, as páginas logicamente sequenciais.

Para acelerar o transporte das páginas, é apenas atualizada aquela que sofreu modificação durante sua permanência na memória principal. Vale citar aqui que, enquanto uma página sofre alterações na memória principal, é sempre mantida a sua versão antiga intacta na memória secundária e a atualização só ocorre se for confirmada a modificação. Isto protege o sistema de manipulações impróprias.

E, permitido, também no CORAS que o próprio usuário carregue uma página que ele necessite num certo momento, desde que seja fornecido adequadamente o endereço físico da página.

Um outro aspecto interessante é que CORAS é provido de funções através das quais os usuários podem especificar a proximidade física de um conjunto de dados funcionalmente dependentes em Aplicações onde o tempo é fator crítico /DEPP 82/. Estas funções garantem que uma página contenha somente os dados funcionalmente dependentes cuja velocidade de acesso influi no desempenho do sistema de Aplicação, de modo a aumentar a probabilidade de encontrar a informação requerida pelos usuários, na memória principal, no contexto de uma Aplicação específica.

CORAS foi inicialmente desenvolvido e implementado em linguagem de tempo real, PEARL, na Universidade de Darmstadt na Alemanha. Baseada na sua filosofia, foi desenvolvida, paralelamente aos trabalhos desta tese, uma outra versão (CORAS-UNICAMP) no Departamento de Engenharia Elétrica da Faculdade de Engenharia na Unicamp. Esta nova versão /WST 83/ é

implementada em FORTRAN IV plus, conforme as especificações do compilador do Sistema Operacional RSX-11M/pdp 11/45. O conjunto de programas que compõem o CORAS-UNICAMP é modulado conforme as suas funções. A interface entre estes módulos é muito nítida, o que permite que um módulo seja modificado sem afetar as características e o desempenho dos outros:

- .módulo de gerenciamento de E/S;

- .módulo de gerenciamento de dados na memória principal;

- .módulo de interface com os usuários, onde são incluídas todas as funções de manipulação acessíveis por estes.

No estágio atual de trabalho, já se encontram implementadas todas as funções dos três módulos, com exceção do agrupamento de dados funcionalmente dependentes. Conforme a finalidade do Sistema de Banco de Dados, é possível adicionar outras funções, tais como: de proteção, de recuperação, de privacidade, etc, de modo a aumentar os seus recursos e torná-lo um Sistema de Gerenciamento de Banco de Dados mais completo.

APENDICE 2: MAPEAMENTO ENTRE CORAS-UNICAMP E MERB

Os diferentes níveis de abstração de dados podem ser organizados numa estrutura hierárquica, apresentando os seguintes níveis: (item 2.4)

- Nível Conceitual;
- Nível Descritivo;
- Nível Lógico;
- Nível Físico.

Sistemas de Gerenciamento de Banco de Dados - SGBD (item 2.1) contém funções que provêm meios para o mapeamento entre estes diferentes níveis e a garantia de univocidade da correspondência entre os dados armazenados na memória secundária e a informação modelada pelos usuários de Aplicação. CORAS é um subconjunto destas funções que atua como interface entre o Nível Lógico e o Nível Físico, pois, os seus procedimentos têm como a finalidade principal de organizar, estruturar e gerenciar a manipulação dos dados contidos na Base de Dados. Ele tanto pode ser utilizado diretamente como gerenciador da estruturação de dados que se classificam em dois primitivos (Apêndice 1) - entidades e triadas - como também, pode ser expandido a um Sistema de Gerenciamento de Banco de Dados mais complexo que suporte Modelos de Dados mais próximos à concepção do Mundo Real (fig. A2.1).

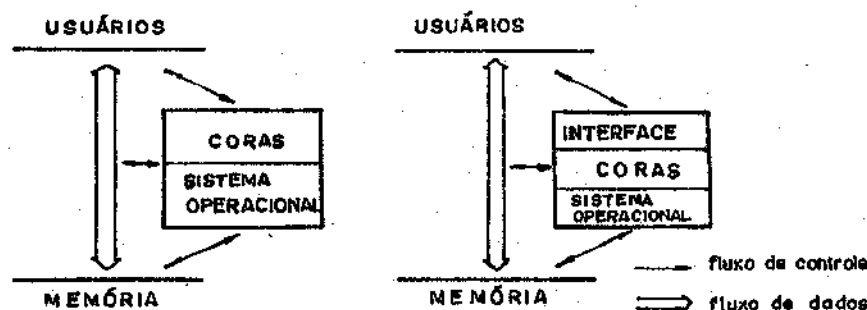


FIG. A2.1: CORAS num SGBD

CORAS foi projetado visando principalmente a otimizar o tempo de manipulação dos dados armazenados na Base de Dados, dentro da capacidade do computador (Apêndice 1). Portanto, suas funções de manipulação e estruturação de dados são adequadas para um Sistema de Gerenciamento de Banco de Dados para Aplicações em tempo real, em que se enquadrariam as aplicações em Controle de Processo. A expansão do CORAS a um SGBD que suporte modelo de dados para Aplicações em Controle de Processo proposto neste trabalho, não é trivial. Em termos gerais, podem-se fazer as seguintes correspondências entre os primitivos do CORAS e os primitivos do MERB:

.entidades proprietárias $\langle = \rangle$ entidades-conjunto: possibilita o estabelecimento de dependência entre a entidade-proprietária e seus dependentes;

.entidades dependentes $\langle = \rangle$ entidades-simples;

.relacionamentos binários $\langle = \rangle$ entidades-relação e triadas: cada associação binária pode ser definida como uma triada no CORAS;

.chaves secundárias $\langle = \rangle$ entidades-conjunto: utilizando-se a estrutura da entidade-conjunto, podem-se considerar todas as entidades associadas à chave secundária como elementos da entidade-conjunto;

.módulos $\langle = \rangle$ agrupamentos físicos de dados;

.versão, atributos $\langle = \rangle$ lista de atributos das entidades.

Note-se que as correspondências entre primitivos listados acima não são biunívocas, havendo a necessidade de acrescentar funções, para distinguir precisamente as diferentes correspondências. Entre as principais funções a serem adicionadas, citam-se:

1) funções de conversão de tipos de valores suportados pelo MERB, para o tipo normalizado de representação interna manipulável pelo CORAS (tipo INTEGER*2);

2) funções que interpretam os papéis desempenhados pelos conjuntos de entidade, para armazenar os possíveis tipos de conjuntos de entidades associados num determinado con-

junto de relacionamentos, garantindo assim a consistência da Base de Dados na inserção de novas ocorrências do conjunto de relacionamentos;

3) funções que estruturam e armazenem a estrutura de dados correspondente à descrição dos campos de registros (entidades/relacionamentos) pertencentes a cada conjunto de entidades/relacionamentos, permitindo assim decodificar apropriadamente o string de bits referente a um registro que descreva uma entidade;

4) funções que interpretem a classe dos conjuntos de entidades (proprietário ou dependente) em classes de entidades suportadas pelo CORAS. É necessário adicionar ainda um campo de informações na classe de entidades-simples, para distinguir os diferentes tipos de entidades dependentes, facilitando o controle interno de manipulação dos dados, pois, para cada tipo de dependência existem regras específicas de existência na Base de Dados;

5) funções que manipulem e estruturam as chaves secundárias, associando-as às entidades-conjunto e garantindo que a inserção de uma nova entidade de um conjunto de entidades sempre implique no teste da existência de alguma chave secundária, associada àquele conjunto, e na inserção desta entidade à chave, caso existir;

6) funções que manipulem e estruturam os relacionamentos binários dos conjuntos de relacionamentos, associando-as às entidades-relação;

7) funções para temporização que ativem, ou desativem, um determinado evento e que definam as diferentes versões de uma mesma entidade;

8) funções de controle de evento;

9) funções de controle de integridade dos dados, através da comparação dos valores inseridos com os valores definidos no domínio de valores;

10) funções que permitam o mapeamento do módulo ao agrupamento físico de dados suportado pelo CORAS.

Além disso, sob o ponto de vista do conteúdo da Base de Dados, é necessário incluir informações sobre:

1) estrutura de cada conjunto de entidades/relacionamentos: tamanho de cada atributo; tipo dos

atributos;

2) tabela de índices de entrada às chaves secundárias associadas a cada conjunto de entidades/relacionamentos;

3) papéis desempenhados pelos conjuntos de entidades e os papéis associados através dos conjuntos de relacionamentos;

4) domínio de valores;

5) estabelecimento interno do vínculo existente entre cada entidade e o seu respectivo conjunto de entidades/relacionamentos;

6) estabelecimento interno das ligações existentes entre os conjuntos de entidades e os módulos a que eles pertencem;

7) outras informações que auxiliem a construção e a manutenção da consistência dos esquemas definidos pela Aplicação, conforme a conveniência de cada problema.

E, finalmente, devem-se levar em consideração os requisitos das Aplicações em tempo real no projeto de software para expansão do CORAS como suporte do modelo MERB para que o desempenho do projeto final seja realmente compatível com a sua finalidade.

BIBLIOGRAFIA

CAPITULO 2:

- /BACH 69/Bachman,C.W., Data Structure Diagrams, Database 1,2, 1969
- /BLAS 79/Blasgen M.W. et al, Sistema R - An Architectural Update, Report Computer Science, IBM, 1979
- /CARV 82/Carvalho,M.A. & Golendziner,L.G., Um Sistema Operacional para Suporte a Bancos de Dados, IV Congresso Nacional de Informatica SUCEJU, Rio de Janeiro, Outubro, 1982
- /CHAM 76/Chamberlin,D.D, Relational Data Base Management Systems, Computing Survey, Vol. 8, n^o 1, March, 1976
- /CHEN 76/Chen,P.P.S., The Entity Relationship Model - Toward a unified view of Data, Transaction on Data Base Systems, Vol 1, n.1, March, 1976
- /CODD 70/Codd,E.F., Relational Model of Data for Large Shared Data Banks, Communication of the ACM, Vol.13, n^o 6, June, 1970
- /CODD 71/Codd,E.F., Further Normalization of the Data Base Relational Model, Information Technology, August, 1971
- /DATE 74/Date,C.J, An Introduction to Database, Addison Wesley Publishing Company, 1974
- /DBAS 81/ - , DBASE II - Assembly-Language Relational Database Management System, Ashton-Tate, California, 1981
- /DELO 80/Delobel,C., An Overview of the Relational Data Theory, Information Processing 80, IFIP, 1980
- /EICH 79/Eichhorn J., Datenbanksysteme erfolgreich eingesetzt, Datascope, 30, 1979
- /FAGI 77/Fagin,R., Multivalued Dependencies and a New Normal Form for Relational Database Systems, Vol. 2, n^o 3, Sept, 1977
- /FRY 76/Fry,J.P., Sibley,E.H., Evolution of Data Base Management Systems, Computing Survey, Vol. 8, n^o 1, March, 1976

- /FUSA 79/Furtado, A.L., Santos, C.S., Organização de Bancos de Dados, Editora Campus, Rio de Janeiro, 1979
- /HAMM 76/Hammer, M., Data Abstractions for Data Base, Bulletin of ACM SIGMOD, Vol. 8, n^o 2, 1976
- /KAHN 80/Kahn, B.K., LDDM - A Structured Logical Database Design Methodology, Tutorial on Software Design Techniques, 3rd Edition, 1980
- /KNUT 73/Knuth, D.E., The Art of Computer Programming - Sorting and Searching, Addison-Wesley Publishing, Massachusetts, 1973
- /MART 77/Martin, J., Computer Data Base Organization, Prentice-Hall, 1977
- /NEUM 82/Neumann, T., Hornung, C., Consistency and Transactions in CAD Database - Proceedings of Conference on VLDB, Mexico, Sept, 1982
- /TAYL 76/Taylor, R.W., Frank, R.L., Codasyl - Database Management Systems, Computing Surveys, Vol. 8, n^o 1, March, 1976
- /TSIC 77/Tsichritzis, D.C., Lochovsky, F.H., Database Management Systems, Academic Press, New York, 1977
- /ULLM 80/Ullman, J.D., Principles of Database Systems, Computer Science Press, 1980
- /WECK 74/Weck, G., SAD - Ein Modell einer Strukturum abhaengigen associativen Datenstruktur, Doktor-Ingenieurs Dissertation, Darmstadt Universitat, Deutschland, 1974
- /WEDE 74/Wedekind, H., Datebank Systeme I, Reihe Informatik, 16, 1974
- /WGIO 77/Wiederhold, G., Database Design, McGraw-Hill, 1977

CAPITULO 3:

- /ANDR 82/Andrade, C.A.M., O Impacto da microeletrônica na Automação Industrial, 1^o CONAI, São Paulo, Julho, 1983
- /AOKI 78/Aoki, M., Large ecentralized Control Problems with imperfect information, 7th Triennial World Congress, IFAC, Finland, June, 1978
- /ATHA 78/Athans, M., Advances and Open problems in the Control of Large Scale Systems, 7th Triennial World Congress, Finland, June, 1978
- /AZHO 78/D'Azzo, J.J., Houpis, C.H., Análise e Projeto de Sistemas

- de Controle Lineares, Guanabara Dois, Rio de Janeiro, 1978
- /COKO 78/Coughanowr, D.R., Koppel, L.B., Análise e Controle de Processo, Guanabara Dois, Rio de Janeiro, 1978
- /CTSG 70/Chen, C.T., Introduction to Linear System Theory, Holt, Rinehart and Winston, Inc, 1970
- /CSGM 83/Copellovitch, S. et al., Automação de Laminadores Contínuos de Aços não Planos, 1^o CONAI, São Paulo, Julho 1983
- /HOAG 77/Hoag, J.W., ADP in Manufacturing - the Process Industries, Automatic Data Processing Handbook, McGraw-Hill, 1977
- /HUNT 78/Hunter, R.P., Automated Process Control Systems, Prentice Hall, 1978
- /IMTO 83/Imoto, J.I., Automação de Plantas Industriais e Distribuição de inteligência: Princípios e Práticas, 1^o CONAI, São Paulo, Julho, 1983
- /JABA 75/James, F.B., Operational-level Subsystems Design, Automatic Data Processing Handbook, McGraw-Hill, 1977
- /JLFL 83/Junior, C.G.A., Lima, R.R., Fleury, A.C.C., Sistemas de Apoio à Decisão em projetos de Engenharia, 1^o CONAI, São Paulo, Julho, 1983
- /JUZE 83/Junior, M.M., Zerbini, R.C., Controle de Processos, 1^o CONAI, São Paulo, Julho, 1983
- /KOMP 81/Kompass, E.J., A long Perspective on Integrated Process of Control Systems, Control Engineering, August, 1981
- /LEFK 77/Lefkowitz, I., Integrated Control of Industrial Systems, Phil Trans. R.Soc.Lond.A.287, Great Britain, 1977
- /LERO 79/Kahne, S., Lefkowitz, I., Rose, C., Automatic Control by Distributed Intelligence, Scientific American, Vol. 240, n^o 6, June, 1979
- /LUWI 81/Lukas, M.P., An Advanced System Architecture for Distributed Control, Control Engineering, August, 1981
- /MMTA 70/Mesarovic, M.D., Macko, D., Takchara, Y. Theory of Hierarchical Multi-level Systems, Academic Press, New York, 1970
- /PAA 76/Pun, L., Aracil, J., Abattet, J.L., Integrated Automation Practice, North-Holland Publishing Company, 1976
- /ROLI 83/Rozenfeld, H., Lirani, J., Desenvolvimento de Software Aplicativo de Planejamento de Produção utilizando

- minicomputadores, 1^o CONAI, São Paulo, Julho, 1983
- /SANT 80/Santos, J.J.H., Automação Industrial,
- /SAVA 65/Savas, E.S., Computer Control of Industrial Process,
McGraw-Hill, 1965
- /SCHW 80/Schwarz, G., Computadores em tempo real, COPPE, Rio de
Janeiro, 1980
- /SILJ 77/Siljak, D.D., Large Scale Dynamic Systems - Stability and
Structure, System Science and Engineering, North-
Holland, 1977
- /STON 75/Stone, M.M., Planning-level MIS Design, Automatic Data
Processing Handbook, McGraw-Hill, 1977
- /STRE 78/Streznova, Z., Control Problems in Management Systems,
7th Triennial World Congress, IFAC, Finland, June, 1978
- /WILL 74/Williams, T.J., Interface Problems for Process Control,
4th IFIC/IFAP Part III, Zuerich, March, 1974
- /WILL 78/Williams, T.J., Hierarchical Control for Large Scale
Systems - A Survey, IFAC, Finland, June, 1978
- /ZEMU 75/Zimmermann, R., Etschberger, K., Mukli, W., Aufgaben des
Mensch in Modernen Prozesswarten, Regelstechnische
Prexis, Heft 7/8, 1975

CAPITULO 4:

- /BUBB 76/Bubbenko, J.A., Berild, S., Ohlin, E.L., Nachmens, S., From
Information Requirements to DBTG - Data Structures,
Proceedings of Conference on Data, Salt Lake City,
March, 1976
- /BUBB 80/Bubbenko, J.A., Information Modeling in the context of
System Development, Information Processing, IFIP, 1980
- /CHAM 76/Chamberlin, D.D., Relational Data Base Management
Systems, Computing Surveys, Vol. 8, n^o 1, March, 1976
- /CHEN 76/Chen, P.P.S., The Entity-Relationship Model - toward a
unified view of data transaction on Data Base Systems,
Vol. 1, n^o 1, March, 1976
- /Codd 70/Codd, E.F., Relational Model of Data of Large Shared Data
Banks, Communication of the ACM, Vol. 13, n^o 6, June,
1970
- /HOAG 77/Hoag, J.W., ADP in Manufacturin - the Process Industries,
Automatic Data Processing Handbook, McGraw-Hill, 1977

- /LEOD 76/McLeod,D.J., High Level Domain Definition in a Relational Data Base System, Proceedings of Conference on Data, Salt Lake City, March, 1976
- /NG 81/Ng,P.A., Further Analysis of the Entity-Relationship Approach to Database Design, IEEE Transaction Software Engineering, Vol SE-7, n^o 1, 1981
- /SENK 73/Senko,M.E.,Altman,E.B.,Astrahn M.M.,Fehder P.L., Data Structures and Accessing in Data Base Systems, IBM Systems Journal, Vol. 12, n^o 1, 1973
- /SMIT 80/Smith D.C.P.,Smith,J.M., Conceptual Database Design, Tutorial on Software Design Techniques, 3rd Edition, 1980
- /WGIO 77/Wiederhold,G., Database Design, McGraw-Hill, 1977

CAPITULO 5:

- /BEMH 82/Bergamaschi,F.A.,Milani,B.E.A.,Hsin,T.C., Geração de Horários de Despacho de Trens em linhas de transporte metroferroviário, 4^o CBA, São Paulo, Setembro, 1982
- /CURY 79/Cury,J.E.R., Metodologia para Geração Automática de Programa-Horário otimizado para linha metroviária de São Paulo, tese de mestrado, UNICAMP, Março, 1979
- /MAGA 81/Magalhães,L.P., Graphisch-Interaktive Mensch-Maschine Kommunikation in der Prozessautomatisierung:Problemanalyse und Hardware/Software Loesung, Dissertation, TH Darmstadt, 1981
- /METR 75/ - , Análise dos Programas de Controle de Trens ao Nível de Estratégias, relatório da tarefa 5 do Convênio Metrô-Unicamp, Maio, 1975
- /METR 76/ - , Estudo de Métodos Operacionais para situações anormais da linha Norte-Sul, Relatório final do Convênio Metrô-Unicamp, Setembro, 1976
- /METR 80/ - , Sistema Básico para Treinamento de operadores do CCO, 2^o Relatório parcial da subtarefa 1.1, Convênio Metrô-Unicamp, Março, 1980
- /METR 83/Martino,J.M., 2^o Relatório de Atividades - Fapesp/Unicamp, Julho, 1983
- /PAA 76/Pun,L.,Aracil,J.,Abattet,J.L., Integrated Automation Practice, North-Holland Publishing Company, 1976

APENDICE 1:

- /ENGL 72/Encarnação J., Glatzer U., DATAS - Datenstrukturen in Assoziativer Speicherung, Angewandte Informatik, Applied Informatics, Heft 9/72, 1972
- /KLNE 80/Klos, W.F., Numan T., GRIFOP - 1979/80 - Zwischenbericht zur GRIFOP - Entwicklung, Jahres-Projekt-Bericht fuer 1979, TH. Darmstadt, FG Graphische-Interaktive Systeme, GRIS 80-3, 1980
- /KLOS 81/Klos, W., GRIFOP - Graphische Informations System fuer Prozessanwendungen zu erscheinen ericht in der serie - Bericht Kernforschungszentrum Karlsruhe, 1981
- /VIWU 82/Vieira, M.S., Wu, S.T., CORAS, relatório interno do setor de Automação da Faculdade de Engenharia da UNICAMP, 1982
- /WST 83/Wu, S.T., Implementação de um Núcleo de Banco de Dados baseado na filosofia de CORAS - Documentação Preliminar, relatório interno do setor de Automação da Faculdade de Engenharia da UNICAMP, 1983