

SOBRE MODELOS MATEMÁTICOS
DO PROJETO LÓGICO DE
SISTEMAS DE INFORMAÇÃO

MARIA DO CARMO NICOLETTI

DISSERTAÇÃO APRESENTADA AO
INSTITUTO DE CIÊNCIAS MATEMÁTICAS
DE SÃO CARLOS
DA UNIVERSIDADE DE SÃO PAULO
PARA OBTENÇÃO DO GRAU DE
MESTRE EM CIÊNCIAS DE COMPUTAÇÃO
ORIENTADOR: PROF. DR. ODELAR LEITE LINHARES

JANEIRO DE 1977

SÃO CARLOS

"Não sou nada

Nunca serei nada

Não posso querer ser nada

À parte isso, tenho em mim

todos os sonhos do mundo"

Fernando Pessoa

Para:

José

Dorvalina

Maria da Graça

Maria Aparecida

A G R A D E C I M E N T O S

Ao Prof. Dr. *Odelar Leite Linhares*, pela orientação segura e dedicada e pelo interesse contínuo, durante todas as fases do meu programa de mestrado, e ao Prof. Dr. *Fernão Stella de Rodrigues Germano*, pelo incentivo, disponibilidade e contribuições valiosas, sem os quais não teria sido possível a realização deste trabalho.

À *Célia Maria Finazzi de Andrade* e à *Maria Carolina Monard* de quem recebi total apoio e estímulo permanente.

Aos amigos: *Bernardina Hernandes*, *Euclides Braga Malheiros*, *Maria Clara Paçõ*, e *Rosely Sanches*, colegas de curso de quem recebi incentivos constantes.

À família Sanches (*Salvador, Rosalina, Suely, Rosely e Rosemary*) que me recebeu e tratou como filha.

Ao *Akiyoshi Mizukami* e *Luiz Augusto da Costa Ladeira*, pela colaboração quando da minha preparação para os Exames Gerais de Qualificação.

Aos professores, funcionários e colegas do Instituto de Ciências Matemáticas de São Carlos - USP, pelos ensinamentos, estímulo e companheirismo.

À *Maria da Graça do Prado Juliano* pela presteza e capricho com que executou o trabalho de datilografia.

Ao CNPq, FAPESP, CAPES e FINEP, pela constante ajuda ao ICMS-USP e, de modo particular, ao CNPq e FAPESP, pelo auxílio concedido, durante o cumprimento do meu programa de mestrado.

À Universidade Federal de São Carlos que propiciou a impressão deste trabalho.

R E S U M O

É apresentada a explicitação do modelo matemático implícito no desenvolvimento do Sistema PSL/PSA, do Projeto ISDOS, em sua versão plena. É evidenciado que este modelo é um hipergrafo dirigido, categorizado, com rótulos, e além disso, que se pode representá-lo através de dígrafos categorizados com rótulos. É discutido um trabalho de Durchholz sobre representação de relações por matrizes e por funções, no contexto de modelos matemáticos de Sistemas de Informação, que adaptam-se aos quatro enfoques abordados por esse autor.

A B S T R A C T

It is presented here, the explicitation of the mathematical model, implicit in the development of the PSL/PSA System of ISDOS Project, in its complete version. It is made evident that this model is a directed, categorized hipergraph with labels, and moreover, that one can represent it by categorized digraphs with labels. It is also discussed a paper by Durchholz about representation of relations by matrixes and by functions in the context of mathematical models of Information Systems that suit the four approaches mentioned by this author.

SUMARIO

CAPÍTULO I - Introdução.....	1
Apresentação.....	1
1.1. Considerações Gerais.....	2
1.2. Sistemas de Informação e Sistemas de Processamento da Informação..	7
1.3. Documentação.....	11
1.4. Conclusões.....	20
CAPÍTULO II - Conceitos necessários ao desenvolvimento deste trabalho..	21
2.1. Considerações Gerais.....	21
2.2. Sobre Teoria dos Grafos.....	22
2.3. Matrizes Esparsas e Representação de Dígrafos através de listas...	29
2.4. O Sistema de Base de Dados, usado no PSA.....	35
2.4.1. Preliminares.....	35
2.4.2. Esquema da Base de Dados.....	35
2.4.3. Diagramas da Base de Dados.....	50
2.5. Conclusões.....	53
CAPÍTULO III - O Modelo Matemático implícito na versão plena do Sistema PSL/PSA.....	54
3.1. Considerações Gerais.....	54
3.2. Explicitação do Modelo Matemático da Linguagem PSL, em sua versão plena.....	55
3.3. Representação do Modelo Matemático da Linguagem PSL em sua versão plena, através de dígrafos.....	63
3.4. Conclusões.....	84

CAPÍTULO IV - Alguns Modelos Matemáticos de Sistemas de Informação,relacionados ao do PSL/PSA.....	85
4.1. Considerações Gerais.....	85
4.2. Representações por matrizes e por funções.....	87
4.3. Sobre o Modelo de Lieberman.....	97
4.4. Conclusões.....	107
CAPÍTULO V - Conclusões finais e Sugestões para novas pesquisas.....	109
BIBLIOGRAFIA	111

CAPÍTULO I

INTRODUÇÃO

Apresentação

Neste capítulo, procuramos situar nossa área de trabalho (Projeto Lógico de Sistemas de Informação) e, para tanto, foi nosso objetivo, durante sua elaboração, dar ampla e geral visão a respeito de Sistemas, Sistemas de Informação, Sistema de Processamento da Informação, Projeto Lógico de Sistemas de Informação e elementos da linguagem PSL, de extrema relevância, no desenvolvimento do trabalho.

No Capítulo II, apresentamos de forma bastante concisa: elementos, conceitos e propriedades de Teoria dos Grafos e Hipergrafos, Matrizes Esparsas e sua representação, e também aspectos do Sistema de Base de Dados, usado no PSA, tópicos estes, que terão relevância, no trabalho que nos propusemos desenvolver.

O Capítulo III trata da explicitação do modelo matemático da linguagem PSL (Problem Statement Language), fortemente relacionado à Teoria dos Grafos e Hipergrafos, sendo uma extensão de [1].

No Capítulo IV, abordamos os quatro principais enfoques de modelos matemáticos de Sistema de Informação, baseados em estudo de Durchholz [2], evidenciando modelos que se adaptam a cada um dos quatro tipos abordados e sugerindo pesquisas futuras, tendo por base, o desenvolvimento feito nesse capítulo.

O Capítulo V traz as conclusões resultantes do estudo feito nos capítulos anteriores e sugestões para o prosseguimento da linha de pesquisa, em que desenvolvemos nosso trabalho.

1.1. Considerações Gerais

Desde o aparecimento e desenvolvimento dos computadores eletrônicos e de sua utilização em empresas, a ciência da administração vem sofrendo uma série de influências e mudanças que se tornam, cada dia, mais evidentes. Paralelamente ao desenvolvimento dos computadores, pode-se evidenciar também o dos equipamentos de comunicação que proporcionam meios necessários à recepção, processamento e transmissão de informações de forma cada vez mais rápida, eficiente e com alto grau de confiabilidade, o que favorece, de modo seguro, entre outros, o processo de tomada de decisões e o atendimento dos objetivos empresariais.

É evidente que a eficiência dos processos decisórios da administração de empresas é sempre função da qualidade da informação, bem como da forma pela qual ela é obtida e colocada à disposição da gerência, na empresa, donde surge a relevância do estudo e da busca de maior desenvolvimento dos sistemas de informação, principalmente daqueles que fazem uso do computador.

Segundo Churchman [3], "os sistemas são constituídos de conjuntos de componentes, que atuam juntos, na execução do objetivo global do todo", ou, citando Dias [4]: "Sistema é um conjunto de partes coordenadas que concorrem para a realização de um conjunto de atividades".

Quando falamos em enfoque sistêmico, estamos, simplesmente nos referindo a determinada forma de pensar a respeito desses sistemas e dos seus componentes. O alcance e a realização, por exemplo, de determinado objetivo proposto, requer especialistas de sistemas, para a análise e determinação das maneiras necessárias e disponíveis, visando, sempre, obter-se a solução ótima, ou pelo menos, a que mais dela se aproxime, levando em conta o propósito de conseguir uma eficiência tão grande quanto possível, aliada a um baixo custo, numa rede complexa de interações. Para isso, são exigidas técnicas altamente elaboradas e complicadas, bem como máquinas sofisticadas para executá-las, uma vez que o volume de cálculos, a velocidade com que devem ser realizados e a obten

ção de resultados, tornam impossível a execução dos mesmos, por uma pessoa, por mais qualificada que seja.

Ao estudarmos sistemas, de modo geral, podemos sempre evidenciar sua característica hierárquica inerente. Podemos, por exemplo [5] visualizar uma organização empresarial como uma série de grandes redes de informação, ligando as necessidades de informações de cada processo decisório, às fontes de dados e estendendo-se por todo o processo empresarial, incluindo as atividades daquelas que se utilizam das informações. Em grandes e complexas organizações empresariais, as diferentes operações de uma organização podem ser descritas como redes de informação separadas, com uma rede geral de informações, sobrepondo-se a cada uma daquelas.

Segundo Churchman [3], quando se pensa no significado de um sistema, devemos levar em conta cinco considerações básicas:

1. os objetivos totais do sistema e, mais especificamente, as medidas de rendimento do sistema inteiro;
2. o ambiente do sistema: as coações fixas;
3. os recursos do sistema;
4. os componentes do sistema, suas atividades, finalidades e medidas de rendimento;
5. a administração do sistema.

Um fato que podemos ressaltar aqui, é o do engano que, geralmente se comete, quando da explicitação dos objetivos do sistema. Muitas vezes, tal explicitação é feita da mais brilhante, atraente e completa forma possível, mesmo que bastante irreal, distorcida e até mesmo omissa, quanto aos aspectos essenciais do funcionamento do sistema, visando, com isso, unicamente, ao alcance de outros objetivos, que não aqueles para os quais o sistema foi projetado. Poderíamos, por exemplo, considerar o caso do diretor de certa empresa de grande projeção, que apresenta os objetivos de sua organização de tal maneira que con

siga, com a brilhante e falsa apresentação, maior número de acionistas.

Podemos, entretanto, acrescentar que tal distorção na declaração dos objetivos, pode, em alguns casos, contribuir para o alcance dos mesmos, tais como foram declarados, como acontece, por exemplo, com o chefe de Departamento de Química de uma Universidade, que deseja conseguir maior orçamento junto à reitoria, para a aquisição de material de laboratório mais sofisticado. Para tanto, comparece diante do reitor e expõe os objetivos do seu Departamento, projetos de pesquisa em andamento e os já realizados, de forma atrativa e brilhante, tendo em mente apenas agir de maneira astuta para conseguir a liberação de verba. Ele nada mais fez do que acrescentar brilho e realce aos objetivos do Departamento, visando meios que lhe possibilitassem, realmente, obter seu intento.

Devemos, por via das dúvidas, tomar sempre muito cuidado ao analisar os objetivos do sistema, pois, muitas vezes, não existe congruência entre os "objetivos declarados" e os "objetivos reais". A forma, talvez, de verificar se os objetivos declarados são, de fato, reais, seria a de determinar se a organização realiza ou não, tudo o que é necessário para o alcance dos objetivos declarados, mesmo em detrimento de quaisquer outras metas da organização. Caso isso aconteça de maneira expressiva, podemos tomar, como reais, os objetivos declarados pela organização.

Outro aspecto devemos considerar, em um sistema: o seu ambiente. Com sentido relativo, não é fácil definir o ambiente de um sistema e sua determinação é problema que envolve contínua e sistemática revisão. Poderíamos pensar em ambiente, como sendo o conjunto de elementos situado externamente ao sistema, considerando-se que o sistema pode fazer relativamente pouco a respeito de suas características ou de seu comportamento. Apesar de o ambiente ser alguma coisa que está fora do controle do sistema, ele é quem determina, em parte, seu funcionamento.

Ao considerarmos, agora, os recursos do sistema, podemos evidenciar que nada mais são do que os meios de que ele - sistema - necessita e dos quais

se utiliza, para desempenhar suas funções. Os recursos, ao contrário do ambiente, são controlados pelo sistema. A medida dos recursos é sempre feita baseada em dinheiro, equipamentos e horas-homem. Quando da determinação dos recursos, deve sempre ser enfatizada, de forma bastante relevante, a expansão tecnológica e os resultados dela advindos, que tornam exequível um aumento substancial dos recursos. Devemos nos concentrar não somente nos que existem, mas também procurar encontrar maneiras pelas quais esses recursos possam ser aumentados e incorporados ao sistema; devemos usar os atuais, visando também sua melhoria, no futuro. Para muitos sistemas, a componente que se relaciona com o desenvolvimento dos recursos é a de mais realce, no sistema.

A quarta consideração que devemos abordar é a relacionada aos componentes do sistema, suas atividades, fins e medidas de rendimento. Embora tenhamos a tendência de considerar departamentos, almoxarifados, repartições, seções, como sendo componentes do sistema, devemos notar que não são eles os componentes reais, embora como tal sejam rotulados. A idéia que geralmente temos é de que um departamento, por exemplo, o de "vendas", de certa firma industrial, tem como atividades únicas as relacionadas à venda e distribuição dos produtos manufaturados pela indústria, e podemos estar incorrendo em erro, uma vez que, em muitos casos, a distribuição dos produtos é função intimamente relacionada à componente de produção, e, para que a distribuição seja determinada, são imprescindíveis informações provenientes da componente produção. Por esse motivo, é que, ao tratar com um sistema, devemos pensar em termos de suas atividades básicas, e das informações a elas necessárias, tendo sempre em mente, a decomposição racional e sistemática das atividades que o sistema deve executar. Podemos, pois, fazer uma concepção dos "componentes" de um sistema, orientada pela idéia de missão.

O objetivo de se tratar com missões, ao invés de departamentos, seções, ou qualquer outra subdivisão hierárquica de determinada organização, é devido ao fato de que, ao analisar as missões, pode-se avaliar o rendimento de uma atividade para o sistema total, enquanto que a avaliação do rendimento de um

departamento é praticamente impossível, devido à interdependência existente entre os departamentos de uma organização e o partilhamento e a dependência de muitas de suas funções. Deve-se descobrir esses componentes (missões), cujas medidas de rendimento individuais estão relacionadas com a medida do rendimento global. Ainda mais, podemos supor que, se fixarmos todos os componentes de um sistema, menos um, e observarmos aumento na medida do rendimento total do sistema, é obvio que a medida do rendimento do componente não fixo aumentou, contribuindo para o aumento do rendimento total do sistema.

A última das cinco considerações, a respeito de sistemas, é relativa a sua administração. Ela é a responsável pela elaboração, implantação e controle dos planos que alocarão, aos diversos componentes do sistema, os recursos disponíveis, tendo por meta, o alcance dos objetivos do sistema, com o máximo rendimento possível, num mínimo de tempo.

O controle dos planos não se limita, simplesmente, a acompanhar seu desenvolvimento, visando execução à risca, de acordo com o idealizado e estabelecido; é antes, a forma dinâmica de atuação que acompanha o desenvolvimento dos planos, avalia-os e interfere, alterando-os sempre que necessário. A administração do sistema deve ser suficientemente aberta, de modo a poder reconsiderar, sob outros ângulos, seus conceitos e admitir medidas que visem alterações.

1.2. Sistemas de Informação e Sistemas de Processamento da Informação

"Sistema de Informação é qualquer sistema usado para prover informação (incluindo o seu processamento), qualquer que seja o uso feito dessa informação." [6]

O Sistema de Processamento de Dados (ou Sistema de Processamento da Informação - SPI) é um sub-sistema do Sistema de Informação, onde as operações com as informações são realizadas por equipamentos de processamento de dados. Fica a cargo do SPI, armazenar, manipular, recuperar ou organizar a informação, de modo a satisfazer as necessidades de uma particular organização. Como o SPI é voltado a cada uma em particular, seu projeto, é feito de forma específica.

Um Sistema de Processamento da Informação tem sempre um "ciclo de vida" bem definido. Por "ciclo de vida" de um SPI, entendemos a sequência de: concepção, construção, uso e, finalmente, obsolescência do sistema. A duração desse "ciclo de vida" é variável e depende principalmente, entre outros, do crescimento do Sistema de Informação de que faz parte.

O "ciclo de vida" de um SPI [7] se inicia quando se percebe a necessidade de seu estabelecimento, continua através da determinação dos requisitos do sistema (projeto lógico do sistema) e é seguido pelo projeto físico do sistema, projeto detalhado e construção, operação, alteração e manutenção e, finalmente, pelo encerramento da operação do sistema.

Paralelamente à percepção das necessidades, que podem surgir, por exemplo, ao se evidenciar um problema, cuja resolução abordada sob outro aspecto, que não o corrente, atinja padrões ótimos, fixados pela organização, devemos fazer um estudo da sua viabilidade; ou seja, estimar gastos e benefícios que o novo sistema (ou que as modificações, no antigo) vão trazer e através dos resultados desse estudo, concluir se a implantação do novo sistema (ou a introdução das alterações propostas) é vantajosa ou não; dependendo desses resultados, passar ao projeto lógico. O aspecto mais relevante, portanto, nessa fase de percepção das necessidades, está na relação gastos/benefícios. Dependendo de

la ser vantajosa ou não, seguir-se-á ou não para a próxima fase. Se, de início, esta relação não for suficientemente vantajosa, várias iterações podem ser tentadas, até se conseguir um resultado tal, que permita o prosseguimento do processo.

A segunda fase do processo de construção de um SPI é a do projeto lôgico do sistema. É durante o curso dessa fase, que é especificado o sistema que tem, como objetivo, melhorar o funcionamento da organização. Além disso, tal fase provê condições para a execução do projeto físico, permitindo à admnistração, tomar a decisão de continuar ou não com o processo, tendo como fator decisivo, a estimativa de viabilidade econômica e técnica.

Dias [4], decompõe, de maneira bastante objetiva, a fase do projeto lógico em duas sub-fases: a da análise do sistema existente e a do projeto do novo sistema, que interagem, de forma significativa, durante a especificação do sistema a ser construído.

Verificando-se, durante a análise do sistema existente se sua atuação corresponde, de fato, às expectativas da empresa, é possível, através dela, detectar pontos falhos, incompletos, ou que não correspondam ao esperado. Face a essa evidência de incorreções e falhas, detectadas durante a análise do sistema existente, o projeto do novo sistema deve ter como preocupação, entre outras, a de sanar as deficiências daquele sistema. Às vezes, o projeto do novo sistema consiste, unicamente, em modificar o existente, de tal forma que as deficiências evidenciadas durante sua análise, desapareçam.

O trabalho aqui realizado, volta-se para essa fase do ciclo de vida, onde, resumindo-se, é definido o que o SPI deve fazer, e, para sermos mais exatos, estaremos mais envolvidos na sub-fase de análise do sistema existente [8]. Já que, no projeto lógico, é definido o que o SPI deve fazer, a maneira de como fazer fica a cargo do projeto físico do sistema. É no projeto físico que são especificadas as configurações de "hardware" e de "hard-software". Ainda mais: verifica-se se as estimativas de custo estão coerentes com a análise de benefício/custo, feita durante a percepção das necessidades.

A fase de construção é a relacionada com a construção do SPI, propriamente dita. Refere-se, pois, à instalação de "hardware", codificação dos programas, construção dos arquivos, elaboração de manuais de operação, etc...

A quinta fase - a do teste, conversão e instalação - tem início quando os componentes do sistema, já testados e aceitos individualmente, podem ser agrupados para testes e aceitação, como sistema.

A seguinte fase de operação é orientada para se obter relatórios a partir de dados recebidos. A maior preocupação, no decorrer desse passo, está em verificar e assegurar que as instruções de operação estão sendo corretamente seguidas.

A última fase - modificação e manutenção - é relacionada às possíveis mudanças que deverão ser feitas, para manter o sistema viável; e à manutenção do sistema.

Deve ser característica do processo de construção de um SPI, a possibilidade de permitir divisões em partes menores, ou seja, em subunidades ou fases. Para cada uma delas, deve ainda haver uma especificação do que realizar, quais os "outputs" fornecidos e quais as informações necessárias à realização do pretendido. A estruturação do "ciclo de vida" de um SPI, por nós adotada, como já foi visto, é a fornecida por Teichroew [9]. A próxima figura (Figura 11) exhibe as técnicas usadas, no ciclo de vida de um sistema, classificadas pelas atividades, em suas diferentes fases.

Devemos ressaltar, ainda, que, durante a análise do sistema existente, grande parte do levantamento para ela realizado, serve e é de grande utilidade ao projeto do novo sistema, muito embora, durante a análise, o analista esteja realmente preocupado em conhecer o sistema atual e evidenciar todas as suas deficiências.

Fases	Outros "inputs" que são usados além dos resultados da fase anterior.	Principal Atividade	Habilidades e Conhecimentos adquiridos.	Principal Resultado
1. Percepção da necessidade	Plano Mestre Reconhecimento de uma oportunidade Percepção de um problema.	Definição de um problema e estimativa inicial de custos e benefícios.	Gerencial	Autorização para prosseguir com o projeto lógico de um sistema particular.
2. Projeto Lógico	Tecnologia de Sistemas de informação	Projeto "ótimo" de um SI para uma organização e a determinação de seus requisitos de informação.	Para: 1. Definir objetivos 2. Proj. Sist. Comerciais 3. Det. Requisitos de Inf.	O projeto funcional de um Sistema de Informação.
3. Projeto Físico	Disponibilidade e Tecnologia de "hardware" e "software".	Determinação da Configuração "ótima" de "Hardware /Software"	Conhecimento das disponibilidades tecnológicas e habilidades p/ aplicá-las	O projeto de um SPI para acompanhar a parte baseada em computador do SI.
4. Construção	Metodologia de construção e teste	Construção do SPI, isto é Programação e Construção de arquivo.	Programação	Construção e/ou aquisição dos componentes físicos do SPI
5. Teste, Conversão e instalação	Descrição do sistema proposto e plano de conversão	Teste e instalação do SPI	—	Testes dos componentes como um sistema e instalação.
6. Operação	—	Operação do SPI e controle de operação.	Gerencial	Produção de "outputs" de acordo com as especificações.
7. Modificação e Manutenção	Novas Oportunidades	Manutenção do SPI, monitoria do desempenho e investigação das possíveis mudanças.	Conhecimento das disponibilidades tecnológicas e habilidade p/ aplicá-las.	Modificação do Sistema como requisito de troca de SPI e para a redução de custos.

Fig. 1.1. Guias do Modelo do Processo Generalizado de Construção de Sistemas |9|

1.3. Documentação

Um dos relevantes problemas durante a análise de um sistema ou durante o projeto de outro, relaciona-se à dúvida que persiste a respeito de que tipo de documentação adotar. É evidente que alguma forma de documentação deve existir, em cada fase do ciclo de vida de um sistema, de maneira que as informações colhidas e evidenciadas numa delas, possam passar à subsequente, objetivando, sempre, maior eficiência e segurança, na tomada de decisões e continuidade do processo.

As vezes, informações servem especificamente a determinadas fases, mas isso não significa que as conclusões e resultados dizem respeito apenas àquelas fases, muito pelo contrário, devem elas ser passadas à fase seguinte, de forma a fornecer material e condições que possibilitem continuar o processo.

Na maioria das vezes, a documentação de que estamos falando, é feita através de uma descrição tão completa e clara quanto possível, usando a linguagem natural e procurando, sempre, desviar-se das possíveis e frequentes ambiguidades, que possam advir. Mesmo tomando todo o cuidado, e evitando sempre as ambiguidades, tal tipo de documentação apresenta muitos e incontestáveis inconvenientes. Poderíamos citar, por exemplo, a imprecisão da língua portuguesa, quando da especificação de uma série de informações que se fazem necessárias: palavras diferentes com o mesmo sentido, palavra com determinada acepção sendo interpretada com outra, dificuldade de expressão, dificuldade de manter a documentação em dia quando determinados trechos são trocados ou, quando, descobertos erros, torna-se necessário corrigi-los; é difícil descobrir os lugares onde eles incidem e com que frequência. Além de tudo isso, podemos acrescentar que o processo não pode ser automatizado, devido à dificuldade de se processar afirmações, em português.

Para substituir a descrição usando a linguagem natural, numeroso conjunto de técnicas está a disposição do analista, as quais se utilizam de formulários, tábuas de decisão, grafos de precedência, malhas, tabelas, matrizes,

fluxogramas e outros tipos de ajuda visual e de comunicação.

Mais usadas, até o momento, são as que utilizam fluxogramas, que nada mais são do que diagramas que exibem o fluxo das informações, dentro do sistema. Um dos mais sérios problemas, quando da elaboração de fluxogramas, é, segundo [10], a determinação do nível de concepção. Assim, existe concordância de que o diagrama de fluxo, por exemplo, é de mais utilidade se ele não for tão detalhado como (deve ser mais resumido do que) o programa que ele descreve. Para a criação de tal diagrama de fluxo, são requeridos: compreensão, condensação e eliminação de detalhes. Mas quais? E quantos? O problema do nível de detalhe afeta o rigor e o acabamento do fluxograma. Quando um fluxograma é preparado bem detalhadamente, tudo deve estar presente, no seu devido lugar, além de que tudo deve estar devidamente relacionado. Quando são suprimidos os detalhes dos fluxogramas, torna-se difícil determinar, a partir do próprio fluxograma, quando determinado processo ou operação está mostrado corretamente ou é essencial.

Apesar disso, são bem mais poderosos que a descrição natural, e até um nível de complexidade não muito elevado, eles têm uma utilidade realmente relevante, pois simplificam razoavelmente a descrição e, através de simples consulta a eles, tem-se visão global do fluxo de informação existente; isso tudo, ressaltamos novamente, quando se trata de um sistema de pequeno porte e volume de informações não grande, fluindo através dele. Para sistemas mais complexos, e com maior volume de informações, são de pouca valia, pois acabam se tornando tão ou mais complexos que os próprios sistemas e de difícil entendimento. Outra restrição ao uso de fluxogramas, que poderíamos ressaltar aqui, seria a dificuldade de podermos processá-los automaticamente.

Devido à imprecisão e à ineficiência dos métodos correntemente usados na descrição de sistemas de informação, é francamente conhecida a necessidade de se ter técnicas mais precisas e eficientes para documentar requisitos de Sistemas de Informação. Esse assunto foi tratado recentemente, com especial destaque, na 2nd Internacional Conference on Software Engineering, San Francis

co, Califórnia |11|.

Uma das técnicas que atualmente apresenta maior número de recursos, é bastante eficiente e é utilizada, com sucesso, na descrição de problemas complexos, é conhecida como "Problem Statement Language and Problem Statement Analyser - PSL/PSA" - desenvolvida pelo Projeto IDOS (Information Systems Design and Optimization Systems), sob a orientação do Prof. Daniel Teichroew, da Universidade de Michigan.

O PSL (Problem Statement Language) é uma linguagem independente de "hardware", usada para descrever sistemas de processamento da informação. Para a descrição de um SPI usando o PSL |12|, é necessária a identificação e a denominação de certo número de tipos de objetos e das relações existentes entre eles (Fig. 1.2.).

Cada objeto definido no sistema, recebe uma denominação e lhe é atribuído certo tipo. O PSL permite a definição de 22 tipos diferentes de objetos. A lista completa, em ordem alfabética, dos tipos permissíveis, está dada na Fig. 1.3. Propriedades de um objeto consistem de afirmações descrevendo-os e podemos ainda definir relações existentes entre objetos. Existem 55 diferentes tipos de relações, que podem ser usadas em PSL. A Figura 1.4 apresenta uma lista, em ordem alfabética, de todas as relações permitidas em PSL.

Voltando nossas considerações à Figura 1.2, devemos identificar, primeiramente, os objetos que estão situados na fronteira do SPI, isto é, que fornecem, recebem ou são responsáveis pela guarda de informações do SPI ("Real World Entities"). Devemos, também, identificar as unidades físicas nas quais os dados são transportados ou armazenados ("Inputs" e "Outputs"). Depois, as unidades de dados devem ser identificadas ("Sets", "Entities", "Groups" e "Elements"), bem como os processos que operam sobre tais dados ("Processes"). Quanto à parte dinâmica do sistema, é descrita em função de condições que deflagram eventos, sendo eles responsáveis pelo curso que as ações seguirão, posteriormente. Com "Events", "Conditions" e "Intervals", consegue-se uma dinamização da descrição, que permite, por exemplo, entre outros, estabelecer prioridades, por

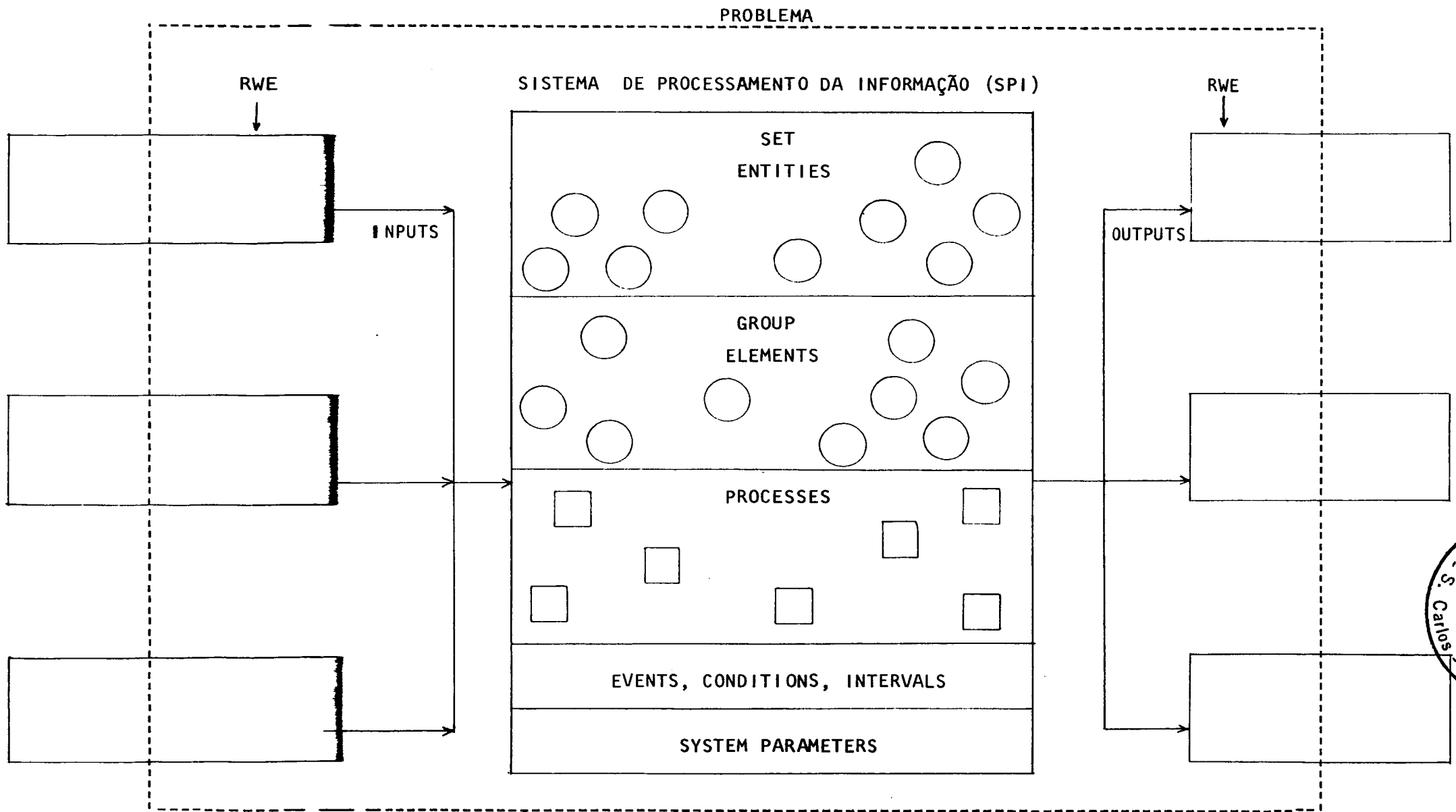


Fig. 1.2. Modelo do "Interior" de um S.P.I. Segundo o PSL/PSA |12|

TIPO DO OBJETO	ABREVIÇÃO
ATTRIBUTE	ATTR
ATTRIBUTE-VALUE	ATTV
CONDITION	COND
ELEMENT	ELE
ENTITY	ENT
EVENT	EVT
GROUP	GR
INPUT	INP
INTERFACE	INTF
INTERVAL	INT
KEYWORD	KEY
MAILBOX	BOX
MEMO	-
OUTPUT	OUT
PROBLEM-DEFINER	PD
PROCESS	PRC
RELATION	REL
SECURITY	SEC
SOURCE	SRC
SET	-
SUBSETTING-CRITERION	SSCN
SYNONYM	SYN
SYSTEM-PARAMETER	SYSP
UNDEFINED	-

Fig. 1.3. Tipos de Objeto e Abreviações | 7 |

RELAÇÃO	ABREVIACÃO	RELAÇÃO	ABREVIACÃO
APPLIES	APP	RELATED	REL
ASSOCIATED	ASOC	RESPONSIBLE	RESP
ASSOCIATED-DATA	ASOD	RESPONSIBLE-INTERFACE	RINT
ATTRIBUTES	ATTR	RESPONSIBLE-PROBLEM	
BECOMING	BEC	DEFINER	RPD
BETWEEN	BTWN	SECURITY	SEC
CARDINALITY	CARD	SEE-MEMO	SM
CONNECTIVITY	CONN	SOURCE	SRC
CONSISTS	CSTS	SUBPARTS	SUBP
CONTAINED	CNTD	SUBSET	SST
DERIVATION	DRVN	SUBSETS	SSTS
DERIVED	DRVD	SUBSETTING-CRITERIA	SSCA
DERIVES	DRVS	SUBSETTING-CRITERION	SSCN
DESCRIPTION	DESC	SYNONYM	SYN
GENERATED	GEND	TERMINATION	TERM
GENERATES	GENS	TERMINATION-CAUSES	TERC
HAPPENS	HAP	TRIGGERED	TRGD
IDENTIFIED	IDD	TRIGGERS	TRGS
IDENTIFIES	IDS	UPDATED	UPDD
INCEPTION	INCP	UPDATES	UPDS
INCEPTION-CAUSES	INCC	USED	-
KEYWORD	KEY	USES	-
MAILBOX	BOX	UTILIZED	UTLD
MAINTAINED	MNTD	UTILIZES	UTLS
MAINTAINS	MNTS	VALUES	VAL
PART	-	VOLATILITY	VOL
PROCEDURE	PRCD	VOLATILITY-MEMBER	VOLM
RECEIVED	RCVD	VOLATILITY-SET	VOLS
RECEIVES	RCVS	WHEN	-
		WHILE	WHL

Fig. 1.4. Tipos de Relação e Abreviações | 7 |

ocasião do curso dos processos. De acordo com [12], eventos ocorrem periodicamente, portanto, o método deve ser tal que permita a especificação do tempo. Em PSL, tal especificação é feita, definindo-se "Intervals". Finalmente, o tamanho do sistema é especificado através de parâmetros, que determinam a frequência, tamanho ou volume com que aparecem determinados objetos ("System Parameters").

Para o sistema PSL/PSA, o objeto de documentação são todas as componentes e relações interiores ao retângulo pontilhado da Figura 1.2, que na terminologia do PSL/PSA denomina-se Problem Statement, significando um problema que deverá ser solucionado na fase do projeto Físico. Nessa fase, como já dissemos anteriormente, são selecionadas as melhores alternativas físicas de processamento, objetivando satisfazer aos requisitos propostos. Iremos chamar de interior do sistema de processamento da informação, a parte do problema relacionada com o processamento dos "inputs" e geração dos "outputs".

Além das informações sobre cada componente do sistema de processamento da informação, existem informações que se referem à especificação das múltiplas relações que surgem entre os componentes, sejam eles de mesmo tipo ou de tipos diferentes. Entre as relações mais importantes, temos as seguintes [13]:

- relações que se originam da dependência hierárquica - determinado componente é uma parte de outro, em nível hierárquico superior, ou de um componente fazem parte outros, de nível hierárquico inferior:

PROCESS : Cálculo-Preço-Total;

SUBPARTS : Cálculo-Preço-Mão-Obra;

Cálculo-Preço-Material;

Cálculo-Preço-Semi-Acabamento;

- relações que dizem respeito à pertinência - um componente consiste de outros tipos de componentes, ou um componente está contido em outro:

GROUP : Preço-Interno;

CONSISTS : Preço-Mão-Obra;

Preço-Material;

- relações que se originam do fluxo - um determinado componente recebe ou é recebido por outro; da mesma forma um componente pode gerar ou ser gerado por outro:

OUTPUT : Folha-Custo-Produto;

GENERATED BY : Obtenção-Folha-Custo;

- relações que dizem respeito à atualização:

SET : Arquivo-Confidencial;

UPDATED BY : Atualização-Arq-Confidencial;

- relações que se originam da troca de estados - uma troca de estado, em determinado componente, pode provocar ou causar a ocorrência de uma troca de estado, em outro componente:

CONDITION : Condição-De-Alteração;

BECOMING TRUE IS CALLED : Altera;

- relações que dizem respeito à responsabilidade - um componente pode ser responsável por outro:

RWE : Departamentos;

RESPONSIBLE FOR : Arquivo-Mestre;

- relações que se originam da identificação - um componente pode ser o identificador de outro:

ENTITY : Registro-Funcionário;

IDENTIFIED BY : Número-Funcionário;

- relações que se originam, de critérios de partição - um componente pode ser um critério de partição ou de classificação sobre outro:

SET : Arquivo-Mestre-Informações;

SUBSETTING-CRITERIA : Departamento;

ma PSL/PSA, é um "software" que processa a descrição do sistema, expressa em PSL. Ele submete as descrições em PSL a várias verificações sintáticas e permite ao analista guardá-las, em uma Base de Dados.

O PSA é de tal forma versátil, que permite obtermos, em qualquer fase do desenvolvimento do projeto lógico do sistema, uma série de relatórios, sobre o estado atual da definição do problema e, através de um estudo desses relatórios, poder introduzir novas definições e alterações, para obter melhor especificação do problema.

1.4. Conclusões

Tivemos sempre em mente, ao escrever este capítulo, situar nossa área de trabalho, de forma bastante objetiva, procurando não só defini-la e evidenciá-la, mas também dar uma visão geral de Sistemas de Informação, Sistemas de Processamento da Informação, Projeto Lógico de Sistemas de Informação, entre outros, tentando ser, ao mesmo tempo, concisos e abrangentes.

Introduzimos noções básicas da linguagem PSL, uma vez que dela é que iremos tratar, no desenvolvimento do trabalho. Nossa abordagem sobre a linguagem PSL é superficial, pois seria impossível tratar dela detalhadamente no presente trabalho, em virtude de sua extensão e também por não ser esse nosso objetivo.

Procuramos, na medida do possível, tornar essa dissertação bastante didática, visando a leitura da mesma por estudantes de graduação que já se voltam para essa área de pesquisa, no Departamento de Ciências de Computação e Estatística de São Carlos - USP.

CAPÍTULO II

CONCEITOS NECESSÁRIOS AO DESENVOLVIMENTO DESTE TRABALHO

2.1. Considerações Gerais

O objetivo principal desse capítulo é o de fornecer material e ferramentas necessários, de forma a possibilitar, de maneira natural, o desenvolvimento dos capítulos subsequentes, evitando que, no decorrer dos mesmos, haja necessidade de abertura de parêntese, para a introdução de conceitos e idéias necessários ao favorecimento de sua evolução. Iremos, ao desenvolver este capítulo, preocupar-nos principalmente, com aspectos de Teoria dos Grafos, Matrizes Esparsas e suas representações, e do Sistema de Base de Dados, usado no PSA, que, no fundo, constituem subsídios ao capítulo seguinte.

2.2. Sobre Teoria dos Grafos

Os conceitos de Teoria dos Grafos de que iremos tratar, foram compilados ou adaptados de documentos de autoria de Berge [14], Furtado [15] e Berztiss [16].

Definição 1:

Um dígrafo D é um par (X, U) , onde

- (1) X é um conjunto $\{x_1, x_2, \dots, x_n\}$ de elementos chamados nós.
- (2) U é uma família (u_1, u_2, \dots, u_m) de elementos do produto cartesiano $X \times X$, chamados arcos.

Observação:

Um arco D da forma (x, x) é chamado de laço.

Por exemplo, consideremos o par (R, S) , onde:

$$R = \{1, 2, 3, 4, 5, 6\}$$

$$S = \{(1, 2), (1, 3), (1, 6), (2, 4), (3, 3), (4, 2), (5, 6), (6, 2)\}$$

O par (R, S) é um dígrafo, cuja representação gráfica (representação sagital) apresentamos a seguir.

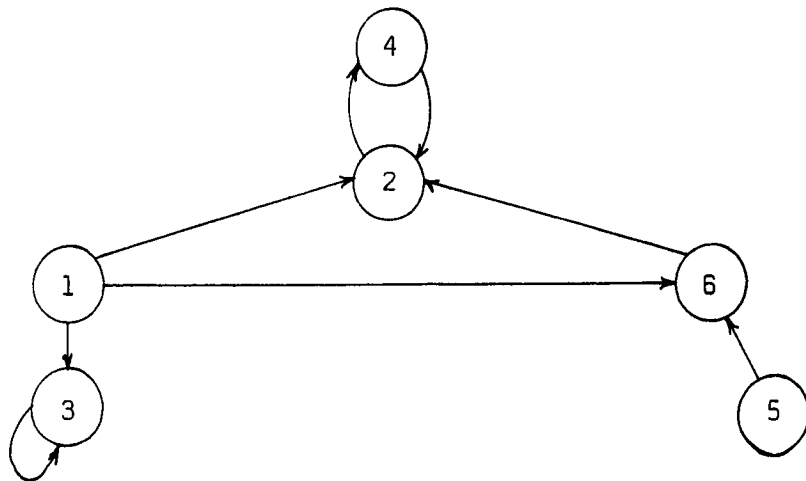


Figura 2.1.

Observação:

Outra representação de dígrafos, de importância relevante, no seu estudo, é a fornecida por sua matriz de adjacência. Se $D = (A, R)$ é um dígrafo

com $A = \{a_1, a_2, \dots, a_n\}$, a matriz adjacência X de D , é definida como:

$$x_{ij} = 1 \text{ se } (a_i, a_j) \in R$$

$$x_{ij} = 0 \text{ se } (a_i, a_j) \notin R$$

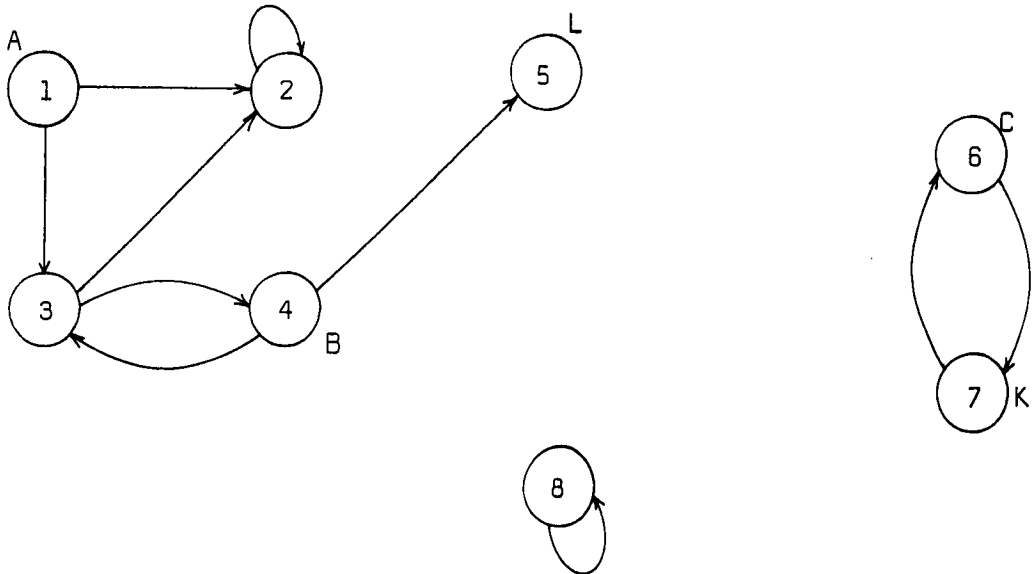
A matriz adjacência do dígrafo cuja representação gráfica é a da Figura 2.1 é:

$$\begin{bmatrix} 0 & 1 & 1 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 & 0 & 0 \end{bmatrix}$$

Definição 2:

Um dígrafo, no qual alguns ou todos os nós têm rótulos a eles as sociados (além dos nomes que os identificam), é chamado dígrafo com rótulos.

Por exemplo, o dígrafo cuja representação sagital é a que se se gue, é um dígrafo com rótulos.



O nó 4, além de ser identificado pelo nome (4), tem o rótulo B a ele associado. Da mesma forma, os nós 1, 5, 6, 7 também tem rótulos.

Definição 3:

Um dígrafo, no qual os arcos têm pesos a eles associados, é chamado de dígrafo ponderado.

Por exemplo, o dígrafo cuja representação sagital é a que se segue, é um dígrafo ponderado.

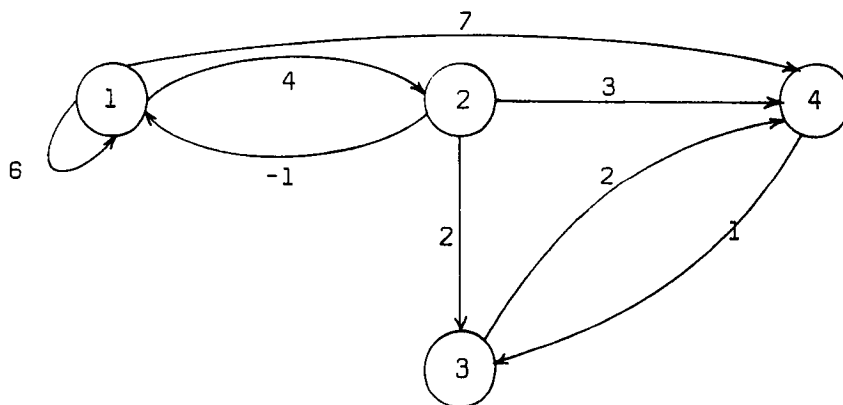


Figura 2.2.

Observação:

Se $D = (A, R)$ é um dígrafo ponderado e se w_{ij} é o peso associado com o arco (a_i, a_j) , então, assumindo que w_{ij} é zero se e somente se não existe o arco (a_i, a_j) , w_{ij} define a matriz W . A matriz W é chamada de matriz adjacência variável do dígrafo ponderado.

A matriz adjacência variável do dígrafo cuja representação gráfica é a da Figura 2.2 é:

$$\begin{pmatrix} 6 & 4 & 0 & 7 \\ -1 & 0 & 2 & 3 \\ 0 & 0 & 0 & 2 \\ 0 & 0 & 1 & 0 \end{pmatrix}$$

Devemos acrescentar que: ao dizermos, na definição 3, que os arcos têm pesos a eles associados, não devemos nos limitar a pensar em peso somente como medida e, portanto, um número associado ao arco. Muitas vezes, é nossa

intenção comunicar, através do peso, a idéia de tipo ou categoria. Quando isso acontece, podemos usar a expressão "dígrafo categorizado", como já foi adotada por Germano em [1], ao invés de "dígrafo ponderado".

Definição 4:

Seja $X = \{x_1, x_2, \dots, x_n\}$ um conjunto finito e seja $E = (E_i / i \in I)$, uma família de subconjuntos de X . Se:

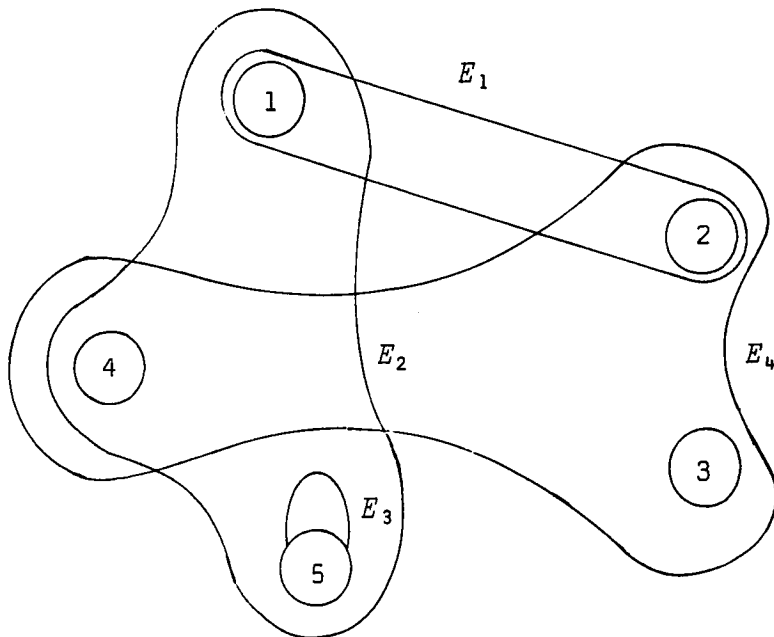
$$(1) E_i \neq \emptyset \quad (i \in I)$$

$$(2) \bigcup_{i \in I} E_i = X;$$

o par $H = (X, E)$ é chamado hipergrafo.

O número de elementos do conjunto X é chamado de ordem do hipergrafo. Os elementos x_1, x_2, \dots, x_n são chamados de nós e os conjuntos E_1, E_2, \dots, E_m são chamados de arcos do hipergrafo.

Por exemplo, o par (P, Q) , onde $P = \{1, 2, 3, 4, 5\}$ e onde $Q = \{\{1, 2\}, \{1, 4, 5\}, \{5\}, \{2, 3, 4\}\}$ é um hipergrafo, cuja representação gráfica, apresentamos a seguir.



Podemos ainda observar que:

- o conjunto E dos arcos, nada mais é do que um subconjunto do conjunto potência de X , ou seja, $E \subseteq 2^X$.

- se os arcos de E são todos distintos, o hipergrafo é chamado de hipergrafo simples.
- em um hipergrafo, dois nós são chamados adjacentes, se existe um arco de E que contém ambos os nós.
- em um hipergrafo, dois arcos são adjacentes se sua intersecção é não vazia.
- uma vez que num conjunto, um seu elemento não pode aparecer duas vezes (redundância), quaisquer que sejam dois arcos de E , eles não serão nunca idênticos, no sentido de possuírem os mesmos nós; surge, daí, a idéia de hipergrafos simples.
- um hipergrafo, no qual os arcos têm pesos a eles associados, é chamado de hipergrafo categorizado.
- a noção de direção será adicionada aos hipergrafos, exigindo-se que os arcos sejam conjuntos ordenados. Estaremos interessados em hipergrafos dirigidos, onde $E \subseteq X \cup X^2 \cup \dots \cup X^n$, para n inteiro.

Em hipergrafos dirigidos, iremos atribuir papéis aos nós, papéis esses, que serão indicados por números, de forma que com o uso dessa atribuição, possamos perfeitamente distinguir, em cada arco do hipergrafo, a sequência ordenada de nós que compõe aquele arco. Por exemplo, dos nós que aparecem na i -ésima posição dos conjuntos ordenados (arcos), dizemos estarem assumindo o papel i .

Para a representação gráfica, nós convencionaremos que se um nó v assume o papel i num arco E , então nós desempenhamos i linhas unindo v a E . Na Figura 2.3, mostramos essa convenção como um exemplo.

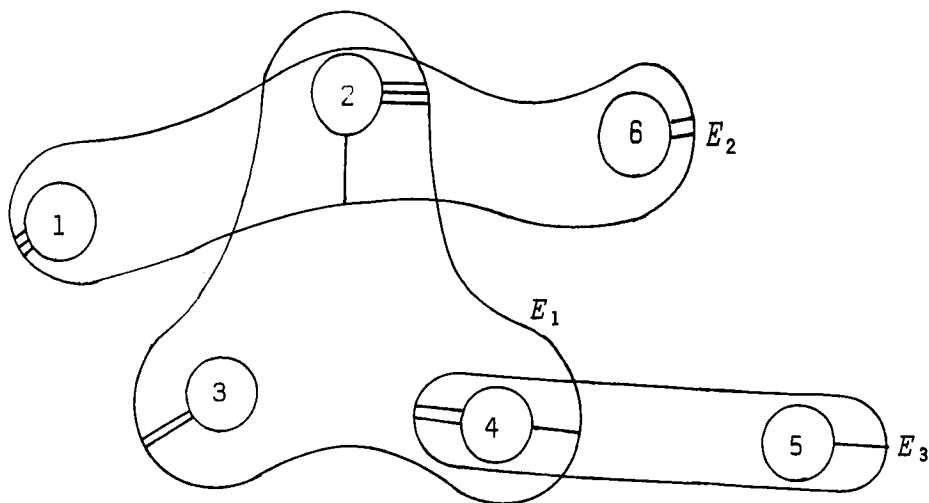


Figura 2.3.

O nó 2, por exemplo, assume o papel 1 no arco E_2 e o papel 3 no arco E_1 .

Definição 5:

A matriz de incidência de um hipergrafo $H = (X, E)$ é uma matriz $((a_j^i))$ com m linhas que representam os arcos de H e n colunas que representam os nós de H , tal que:

$$a_j^i \begin{cases} = 1 & \text{se } x_j \in E_i \\ = 0 & \text{se } x_j \notin E_i \end{cases}$$

Observação:

Cada matriz $(0,1)$ (matriz constituída somente de 0's e 1's) é matriz de incidência de um hipergrafo, se nenhuma linha ou coluna contém somente zeros.

Por exemplo, se tivermos um hipergrafo cuja representação gráfica é a seguinte:

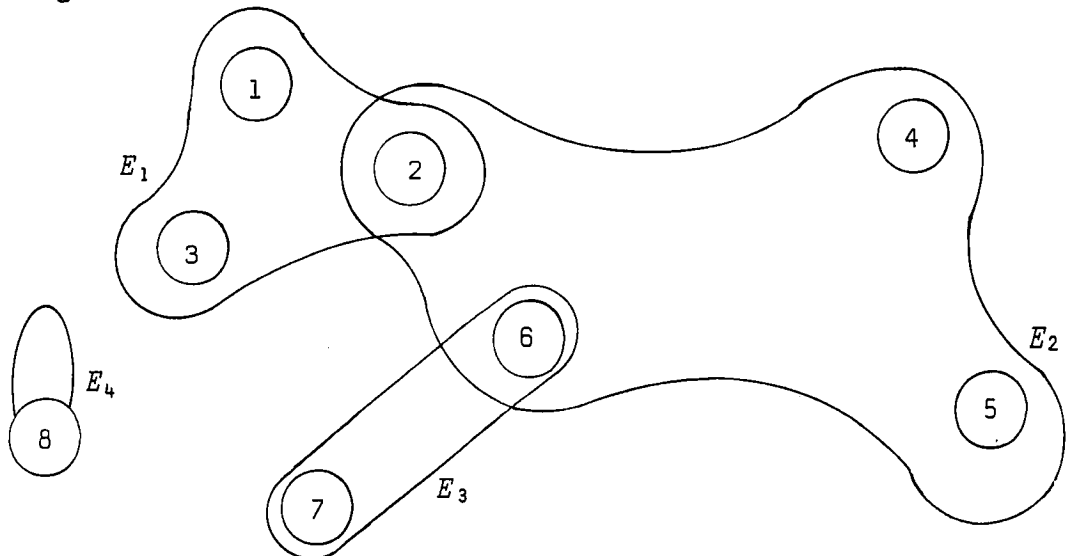


Figura 2.4.

sua matriz de incidência é:

$$\begin{pmatrix} 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 1 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{pmatrix}$$

Considerando que cada linha da matriz corresponde a um arco, podemos associar a cada linha, um tipo de relação, com o que daremos categoria a cada arco. O hipergrafo categorizado da Figura 2.4, seria então completamente representado pela matriz abaixo, onde B_1 , B_2 , B_3 e B_4 são as categorias associadas aos arcos E_1 , E_2 , E_3 e E_4 .

$$\begin{pmatrix} 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & B_1 \\ 0 & 1 & 0 & 1 & 1 & 1 & 0 & 0 & B_2 \\ 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & B_3 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & B_4 \end{pmatrix}$$

2.3. Matrizes Esparsas e Representação de Dígrafos Através de Listas

O nosso objetivo, ao introduzir esta seção, foi motivado pelo fato que as matrizes dos dígrafos correspondentes à descrições de especificações de Sistema de Informação, geralmente são esparsas, o que implica na conveniência de utilização de técnicas de representação e manipulação de matrizes esparsas.

Consideremos o dígrafo cuja representação gráfica é a seguinte:

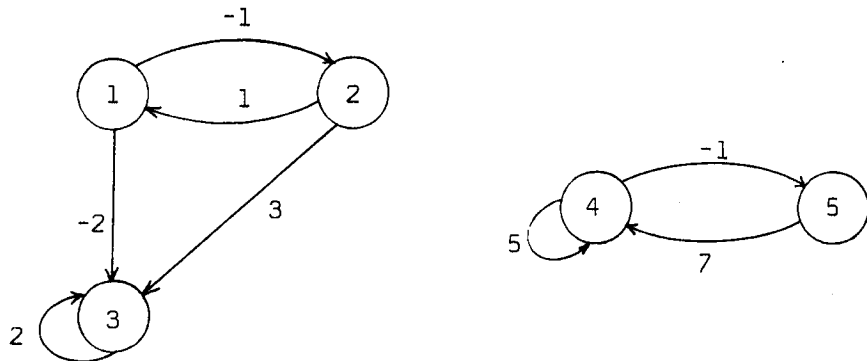


Figura 2.5.

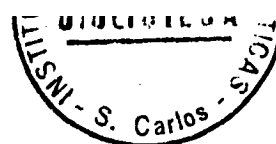
Sua matriz de adjacência variável é:

$$X = \begin{pmatrix} 0 & -1 & -2 & 0 & 0 \\ 1 & 0 & 3 & 0 & 0 \\ 0 & 0 & 2 & 0 & 0 \\ 0 & 0 & 0 & 5 & -1 \\ 0 & 0 & 0 & 7 & 0 \end{pmatrix}$$

Figura 2.6.

O número K de elementos não nulos nessa matriz é 7. Uma forma mais econômica de representá-la, seria pela matriz:

$$P = \begin{pmatrix} 1 & 1 & 2 & 2 & 3 & 4 & 4 & 5 \\ 2 & 3 & 1 & 3 & 3 & 4 & 5 & 4 \\ -1 & -2 & 1 & 3 & 2 & 5 & -1 & 7 \end{pmatrix}$$



uma vez que $24 < 25$. Os primeiros dois elementos de uma coluna em P , definem um arco, e o terceiro, o peso associado a esse arco; ou, traduzindo para a terminologia de matrizes, uma coluna armazena os índices e o valor de um elemento não nulo de X .

Generalizando, seja uma matriz X com N linhas e M colunas, e seja K o número de elementos não nulos em X . Normalmente, a matriz poderia ser armazenada em NM locações, mas quando $3K < NM$, uma matriz tal como P , dá uma representação mais econômica. Essas matrizes que têm muitos elementos iguais a zero, são chamadas matrizes esparsas.

Infelizmente, a representação proposta não é conveniente quando devemos realizar operações com as matrizes esparsas. É razoavelmente fácil encontrar todos os elementos pertencentes a uma dada linha de X , porque eles ocupam colunas adjacentes em P , mas quando se trata de encontrar elementos pertencentes a uma certa coluna, a busca já não é tão fácil, porque eles estão dispersos na matriz. Portanto, ao invés da representação da Figura 2.6, iremos considerar um dígrafo, no qual, para todo elemento não nulo da matriz esparsa, existe um nó correspondente, que tem como rótulo, a terna (número da linha, ou coluna, são representados, no dígrafo, por nós que estão unidos por uma seta. A próxima Figura, mostra a matriz do nosso exemplo, sob essa representação.

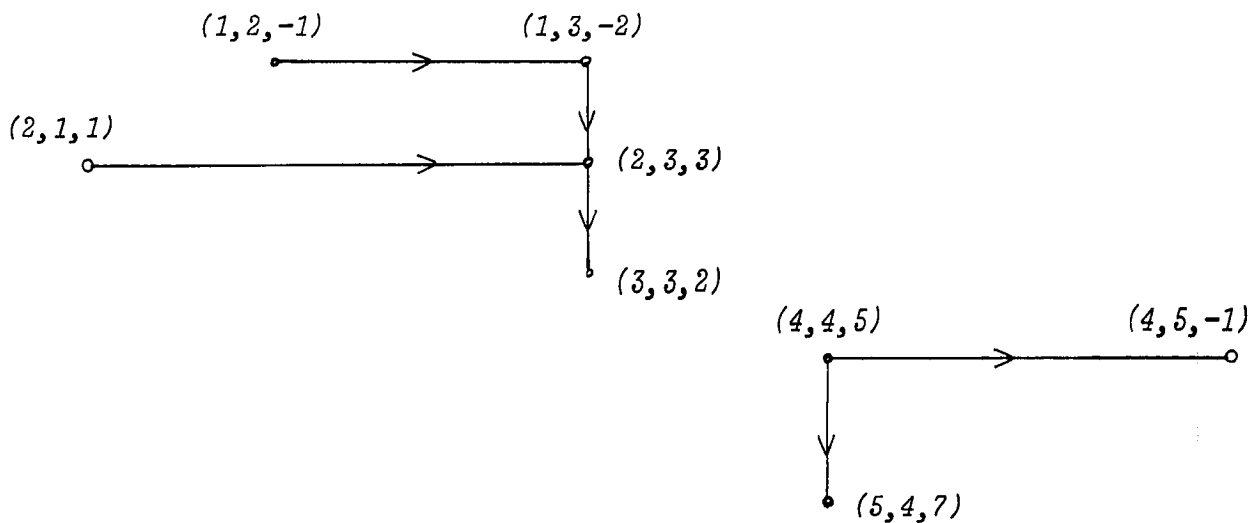


Figura 2.7.

Vamos agora representar a matriz X (ou o dígrafo da Figura 2.7), por

uma matriz X (ou o dígrafo da Figura 2.7), por uma matriz $4 \times 8, PP$, na qual, as três primeiras linhas são as mesmas de P , mas a quarta contém ponteiros ligando elementos que pertencem à mesma coluna em X .

$$PP = \begin{pmatrix} 1 & 1 & 2 & 2 & 3 & 4 & 4 & 5 \\ 2 & 3 & 1 & 3 & 3 & 4 & 5 & 4 \\ -1 & -2 & 1 & 3 & 2 & 5 & -1 & 7 \\ 0 & 4 & 0 & 5 & 0 & 8 & 0 & 0 \end{pmatrix}$$

Esta representação é chamada de representação em 4 linhas.

Por exemplo, vamos olhar para elementos que pertencem à terceira coluna de X . Esses elementos são identificados pelo número 3 na segunda linea de PP . Eles estão armazenados nas colunas 2, 4 e 5 dessa matriz. Os elementos da quarta linha de PP nos enviam da segunda coluna, para a quarta, e desta para a quinta. Como $PP(3,3) = 0$, isso indica que o elemento guardado nessa posição é o último diferente de zero pertencente à terceira coluna de X .

Iremos considerar, além da matriz PP , dois vetores: NC e NL . O valor $NC(J)$, aponta para a primeira coluna de PP que contém o primeiro elemento não nulo, da coluna J , em X . Temos pois que:

$$NC = (3, 1, 2, 6, 7)$$

Se todos os elementos da coluna J são zeros, então $NC(J) = 0$.

Já $NL(I)$, contém o número da coluna, de PP , que armazena o primeiro elemento não nulo da linha I em X ; e, desde que $NL(I + 1)$ contém um ponteiro para o primeiro elemento não nulo da linha $I + 1$, existem $NL(I + 1) - NL(I)$ colunas adjacentes armazenando elementos da linha I . Se todos os elementos da li nha I são zeros, então $NL(I) = NL(I + 1)$. Temos pois que:

$$NL = (1, 3, 5, 6, 8, 9)$$

Esta convenção implica na necessidade do vetor NL conter um elemento a mais do que o número de colunas de X . O último elemento desse vetor, contém um número que é igual ao número de colunas de PP , mais 1.

Vamos evidenciar o que acontece com a representação em quatro-linhas de uma matriz quando um elemento não nulo, torna-se zero, ou, quando zero, assume outro valor. No primeiro caso, fazemos uma simples troca do valor correspondente, na terceira linha da representação, por zero. No segundo caso, uma nova coluna deve ser adicionada à representação em quatro-linhas (como previsão de um acontecimento como esse, *PP* desde o início, deve ser dimensionada como tendo mais colunas do que o número de seus elementos distintos de zero). Como essa adição envolve uma operação relativamente trabalhosa, uma vez que as linhas não estão relacionadas por ponteiros e sim por posições adjacentes na matriz, esse problema é contornado, criando-se uma outra representação da matriz.

Vamos supor, por exemplo, que o elemento x_{22} assuma o valor -2. Para acrescentar esse elemento à matriz *PP*, vamos criar uma representação em 5-linhas, *PPP*. A quinta linha, nessa nova representação, consiste de ponteiros ligando elementos que pertencem à mesma linha na matriz original *X* (a quinta linha vai servir às linhas, de forma análoga à que a quarta linha de *PP* serve às colunas).

Vamos alterar o formato do vetor *NL*, objetivando tornar *NL* análogo a *NC*. Temos:

$$PPP = \begin{pmatrix} 1 & 1 & 2 & 2 & 3 & 4 & 4 & 5 \\ 2 & 3 & 1 & 3 & 3 & 4 & 5 & 4 \\ -1 & -2 & 1 & 3 & 2 & 5 & -1 & 7 \\ 0 & 4 & 0 & 5 & 0 & 8 & 0 & 0 \\ 2 & 0 & 4 & 0 & 0 & 7 & 0 & 0 \end{pmatrix}$$

$$NC = (3, 1, 2, 6, 7)$$

$$NL = (1, 3, 5, 6, 8)$$

Vejamos pois, como a matriz *PPP* é alterada, quando o elemento x_{22} passa a ser -2. Esse elemento é colocado na matriz *PPP* como última coluna, da esquerda para a direita e alteramos alguns ponteiros da quarta e da quinta li

nas. Com a introdução desse elemento, o ponteiro de colunas (quarta linha), que não apontava para nenhum elemento (uma vez que na segunda coluna da matriz, o único elemento distinto de zero era x_{12}), passa a apontar para o novo elemento. Já o ponteiro das linhas, que do elemento (2,1,1) apontava diretamente para o (2,3,3), passa a apontar para (2,2,-2), e esse por sua vez, para o (2,3,3). Temos então:

$$PPP = \begin{pmatrix} 1 & 1 & 2 & 2 & 3 & 4 & 4 & 5 & 2 \\ 2 & 3 & 1 & 3 & 3 & 4 & 5 & 4 & 2 \\ -1 & -2 & 1 & 3 & 2 & 5 & -1 & 7 & -2 \\ 9 & 4 & 0 & 5 & 0 & 8 & 0 & 0 & 0 \\ 2 & 0 & 9 & 0 & 0 & 7 & 0 & 0 & 4 \end{pmatrix}$$

A Figura 2.7, como já vimos, é um dígrafo representando a matriz esparsa X . Os nós, nessa figura, estão dispostos e unidos em linhas verticais e horizontais. As verticais, correspondem a colunas e as horizontais, a linhas. A representação da matriz, sugere que cada linha e cada coluna, seja representada por uma lista.

Considerando, então, cada linha sendo representada por uma lista, elas conterão, respectivamente, 2,2,1,2,1 elementos.

Consideremos, por exemplo, o elemento x_{44} . Ele poderá ser representado por elementos das listas da linha 4 e da coluna 4, as quais iremos notar por $NL(4)$ e $NC(4)$. Na lista $NL(4)$, o elemento de lista x_{44} poderá ser formado da terna (4,4,5) e de um ponteiro para o elemento de lista representando o x_{45} . Na lista $NC(4)$, o elemento de lista x_{44} poderá ser formado da terna (4,4,5), novamente, e de um ponteiro para o elemento de lista representando o x_{54} . Estaremos criando, assim, duas listas de elementos, separadas, para o x_{44} , onde as ternas associadas ao elemento, são iguais, e os valores dos ponteiros, diferentes. Da forma como colocada, essa abordagem poderá gastar um espaço razoável de memória de computador.

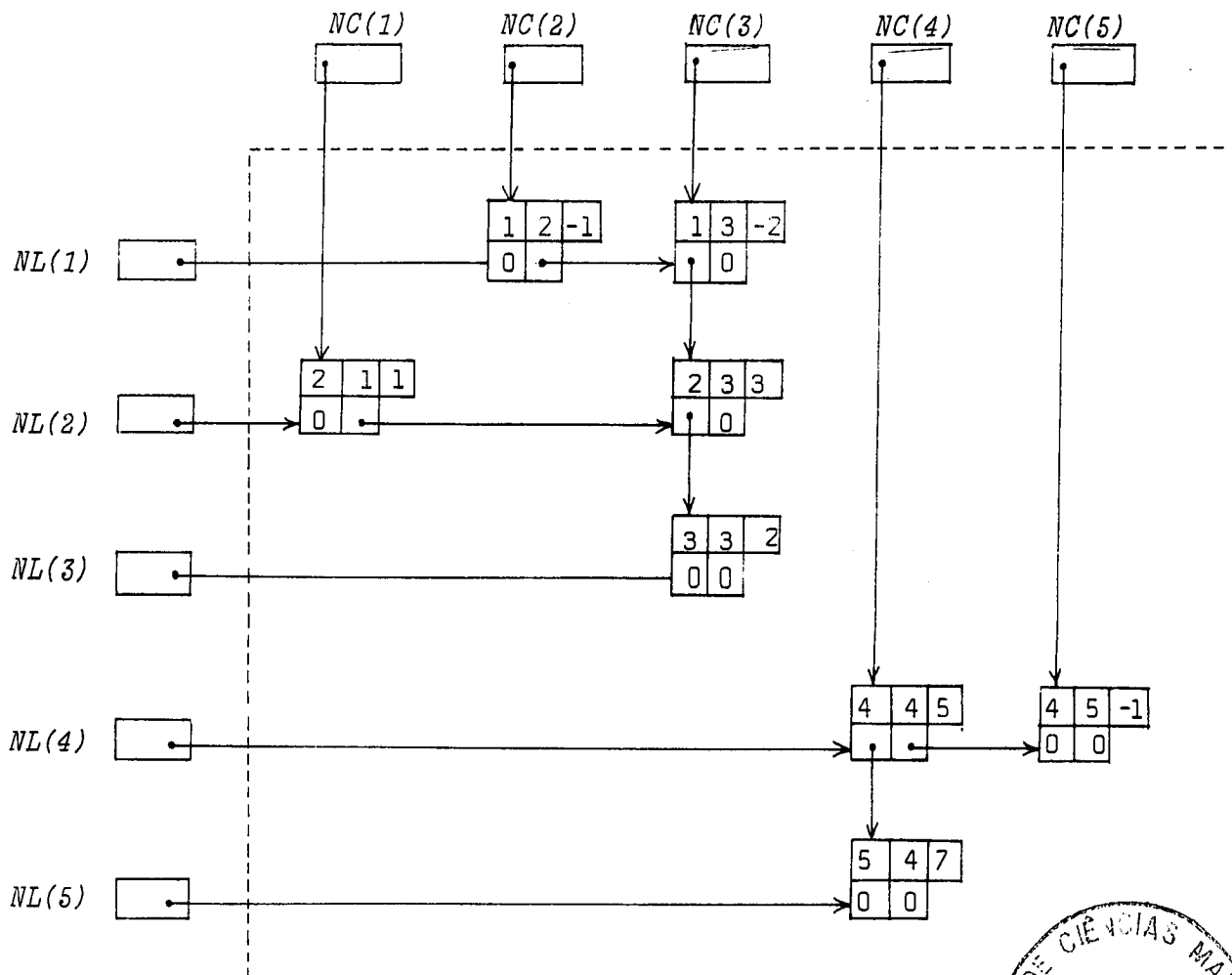
Uma forma de contornar o problema, é fazer com que todo elemento de

lista pertença a duas listas — uma de linhas e outra de colunas. O elemento de lista $x_{4,4}$, então, consistirá da terna (4,4,5) e de dois ponteiros, um relacionado à sua condição de membro da $NL(4)$ e o outro, da $NC(4)$.

Uma lista-linha, consistindo de k elementos, é assim "cruzada" por k listas-coluna, e isso corresponde precisamente ao que está mostrado na Figura 2.7.

Uma estrutura, na qual, cada elemento pertence a m listas, é chamada: estrutura de lista-cruzada de ordem m . Uma matriz esparsa pode ser representada por uma estrutura de lista cruzada de ordem 2. Já para representar uma matriz esparsa tri-dimensional, iremos necessitar de uma estrutura de lista-cruzada de ordem 3.

A representação em 5 linhas de uma matriz esparsa (PPP), é de fato uma estrutura de lista cruzada. Os vetores NL e NC contêm os nomes das listas linhas e listas-coluna. Uma representação esquemática da estrutura de lista cruzada que representa a matriz esparsa da Figura 2.6 é mostrada abaixo.



2.4. O Sistema de Base de Dados, usado no PSA

2.4.1. Preliminares

Esta seção, do presente estudo, baseia-se em [17] e tem por objetivo apresentar a estrutura da Base de Dados [18], usada pelo Problem Statement Analyser (PSA).

Inicialmente, apresenta a descrição do esquema utilizado para a Base de Dados, onde figuram: registros, itens e disposições e, a seguir, os diagramas dos tipos de conexão que compõem o esquema, fazendo referência à tabelas, que enumeram as conexões possíveis entre tipos de objetos, que correspondem aos tipos de relações, disponíveis em PSL.

2.4.2. Esquema da Base de Dados

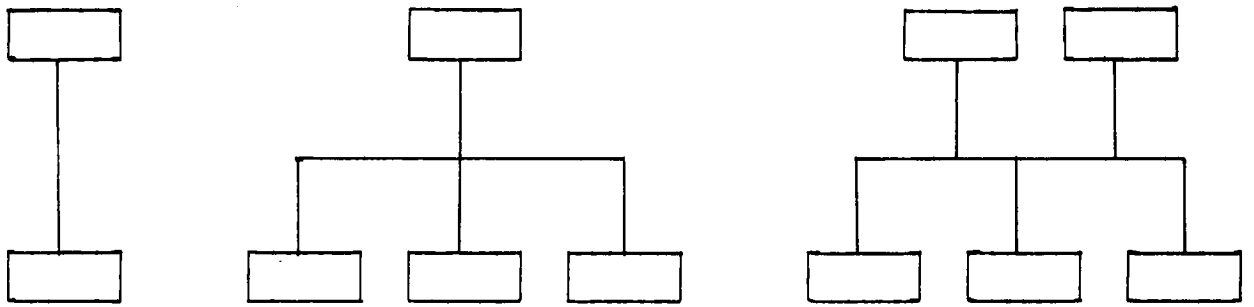
A Figura 2.8 apresenta a descrição na DDL, usada pelo ADBMS [19] (Sistema de Base de Dados do PSA), do esquema empregado na construção da Base de Dados, para armazenar descrições, em PSL.

A primeira coluna descreve os diferentes tipos de registros usados pelo analisador. A segunda, as disposições [18] ("sets" na nomenclatura da CO DASYL). No esquema da Base de Dados, existem nove tipos de diferentes registros e 8 de diferentes disposições. Procuraremos descrevê-los, bem como seus respectivos usos, citando exemplos para tornar a descrição mais clara e inteligível.

Devemos acrescentar que o Sistema de Base de Dados utilizado pelo PSA, em uma única disposição, admite definir-se mais de um registro mestre. As conexões a seguir, mostram exemplos de disposições que podem ser definidas no Sistema de Base de Dados que o PSA utiliza.

RECORD	NAMREC			SET	ALLNAM	SORTED	NAMEC
ITEM	NAMET	INTEG	16	OWNER	SYSTEM		
ITEM	NAMEC	CHAR	30	MEMBER	SYNREC		
RECORD	SYNREC			SET	RELA	SORTED	RELTYP
ITEM	NAMET	INTEG	16	OWNER	NAMREC		
ITEM	NAMEC	CHAR	30	OWNER	NUMREC		
RECORD	NUBA			MEMBER	NUBA		
ITEM	RELTYP	INTEG	16	MEMBER	NUBB		
RECORD	NUBB			MEMBER	NUBC		
ITEM	RELTYP	INTEG	16	MEMBER	COM		
RECORD	NUBC			SET	RELB	SORTED	RELTYP
ITEM	RELTYP	INTEG	16	OWNER	NAMREC		
RECORD	COM			OWNER	NUMREC		
ITEM	RELTYP	INTEG	16	MEMBER	NUBA		
RECORD	NUMREC			MEMBER	NUBB		
ITEM	NAMET	INTEG	16	MEMBER	NUBC		
ITEM	NAMEC	CHAR	30	SET	RELC	NEXT	
ITEM	FIL	CHAR	2	OWNER	NUBB		
ITEM	NUMVAL	INTEG	16	MEMBER	NUBC		
RECORD	COMLIN			SET	ALINE	FIFO	
ITEM	LINEC	CHAR	72	OWNER	COM		
RECORD	AREC			MEMBER	COMLIN		
ITEM	NMITEM	INTEG	16	SET	SYNFOR	SORTED	NAMEC
ITEM	DBKEYA	DBKEY		OWNER	NAMREC		
ITEM	DBKEYA	DBKEY		MEMBER	SYNREC		
ITEM	CHITEM	CHAR	30	SET	ALPHA	SORTED	CHITEM
				OWNER	SYSTEM		
				MEMBER	AREC		
				SET	BYVAL	SORTED	NMITEM
				OWNER	SYSTEM		
				MEMBER	AREC		

Fig. 2.8. Esquema empregado na Construção da Base de Dados, para Armazenar descrições, em PSL.



Das três conexões acima, as que atendem às especificações da CODASYL são as duas, à esquerda. Não é permitido a uma disposição tipo CODASYL possuir mais de um registro mestre, fato esse, como já falamos, admissível pelo sistema de Base de Dados, utilizado pelo PSA.

Observação:

Para o prosseguimento desta seção, juntaremos a cada tipo de objeto e de relação (uma relação e sua complementar pensadas como um tipo único), existente em PSL, o seu respectivo código.

TIPO DE OBJETO	CÓDIGO	TIPO DE OBJETO	CÓDIGO
ATTRIBUTE	1	MEMO	13
ATTRIBUTE-VALUE	2	OUTPUT	14
CONDITION	3	PROBLEM-DEFINER	15
ELEMENT	4	PROCESS	16
ENTITY	5	RELATION	17
EVENT	6	SECURITY	18
GROUP	7	SOURCE	19
INPUT	8	SET	20
INTERFACE	9	SUBSETTING-CRITERION	21
INTERVAL	10	SYSTEM-PARAMETER	22
KEYWORD	11		
MAILBOX	12		

Figura 2.9.

TIPO DE RELAÇÃO	CÓDIGO
ASSOCIATED-DATA IS / ASSOCIATED WITH	1
BECOMING TRUE IS CALLED / WHEN BECOMES TRUE	2
BECOMING FALSE IS CALLED / WHEN BECOMES FALSE	3
BETWEEN / RELATED TO VIA	4/5
CARDINALITY IS	6
CONNECTIVITY IS TO	7/8
CONTAINED IN / CONSISTS OF	9/10
DERIVATION	11
DERIVED BY USING / DERIVES USING / USED BY TO DERIVE / USES TO DERIVE	12/13
DESCRIPTION	14
SUBSET OF / SUBSETS ARE	15
GENERATED BY / GENERATES	16
HAPPENS TIMES-PER	17/18
GENERATED BY / RECEIVED BY / RESPONSIBLE-REAL-WORLD-ENTITY / RECEIVES / GENERATES / RESPONSIBLE FOR	19
IDENTIFIED BY/ IDENTIFIES	20
INCEPTION-CAUSES / ON INCEPTION OF	21
KEYWORDS ARE/ APPLIES TO	22
MALBOX / APPLIES TO	23
MAINTAINED BY / MAINTAINS	24
PART OF / SUBPARTS ARE	25
PROCEDURE	26
RECEIVED BY / RECEIVES	27
RESPONSIBLE FOR / RESPONSIBLE-PROBLEM-DEFINER	28
SECURITY IS / APPLIES TO	29
SEE-MEMO / APPLIES TO	30
SOURCE / APPLIES TO	31
SUBSETTING-CRITERIA ARE/ SUBSETTING-CRITERION FOR	32
TERMINATION-CAUSES / ON TERMINATION OF	33
TRIGGERED BY / TRIGGERS	34
UPDATE BY USING / UPDATES USING / USED BY TO UPDATES / USES TO UPDATE	35/36
USED BY / USES	37
VALUE IS	38/39

Figura 2.10

TIPO DE RELAÇÃO	CÓDIGO
VALUES ARE THRU	38/40
VOLATILITY	41
VOLATILITY-MEMBER	42
VOLATILITY-SET	43
TRUE WHILE	44
FALSE WHILE	45
UTILIZES / UTILIZED BY	46
ATTRIBUTES ARE	47/48
CONSISTS OF (Interval Section)	49/50
DESIGNATE AS A SYNONYM FOR / SYNONYMS ARE	-

Figura 2.10

Passemos agora, à descrição dos diferentes tipos de registro e de disposição:

A - Registro NAMREC

Cada nome definido em PSL (nome de objeto) é armazenado em um registro NAMREC, composto de dois itens: NAMET e NAMEC. O item NAMET contém um código numérico indicando o tipo do objeto (vide Figura 2.9), para a qual este registro está sendo definido. O item NAMEC contém o nome do objeto. Se tivéssemos, por exemplo:

INPUT : Cartão-de-Ponto;

esse objeto seria armazenado, da seguinte forma:

NAMREC	
NAMET	NAMEC
8	CARTÃO - DE - PONTO

Observação:

Tanto na implementação do sistema PSL/PSA, feita na Universidade de Michigan, quanto na do CCE - USP, o número máximo de caracteres permitidos, por nome de objeto, é trinta. Portanto, o item NAMEC, no máximo, conterá um nome de trinta caracteres.

B - Registro SYNREC

Cada vez que o usuário define um sinônimo, para um nome, fica ele armazenado em um registro SYNREC, que tal como o anterior, possui dois itens: NAMEC e NAMEC, usados de maneira análoga à do registro NAMEC. A única diferença é que o código do tipo do nome armazenado no NAMEC do registro SYNREC é o que está no item NAMEC do registro NAMEC, relativo ao nome, mais 100.

Consideremos o mesmo exemplo anterior, agora para definir um sinônimo aplicado ao nome *Cartão-de-Ponto*. O trecho referente a isso é:

INPUT : Cartão-de-Ponto;

SYNONYM : Cartão-Ponto;

Definido o sinônimo para o nome "Cartão-de-Ponto", deve ele ser armazenado, num registro SYNREC. Temos pois:

<i>SYNREC</i>	
<i>NAMEC</i>	<i>NAMEC</i>
108	CARTÃO - PONTO

Veremos mais adiante, (seção L), como se faz a correspondência entre um nome e seus sinônimos.

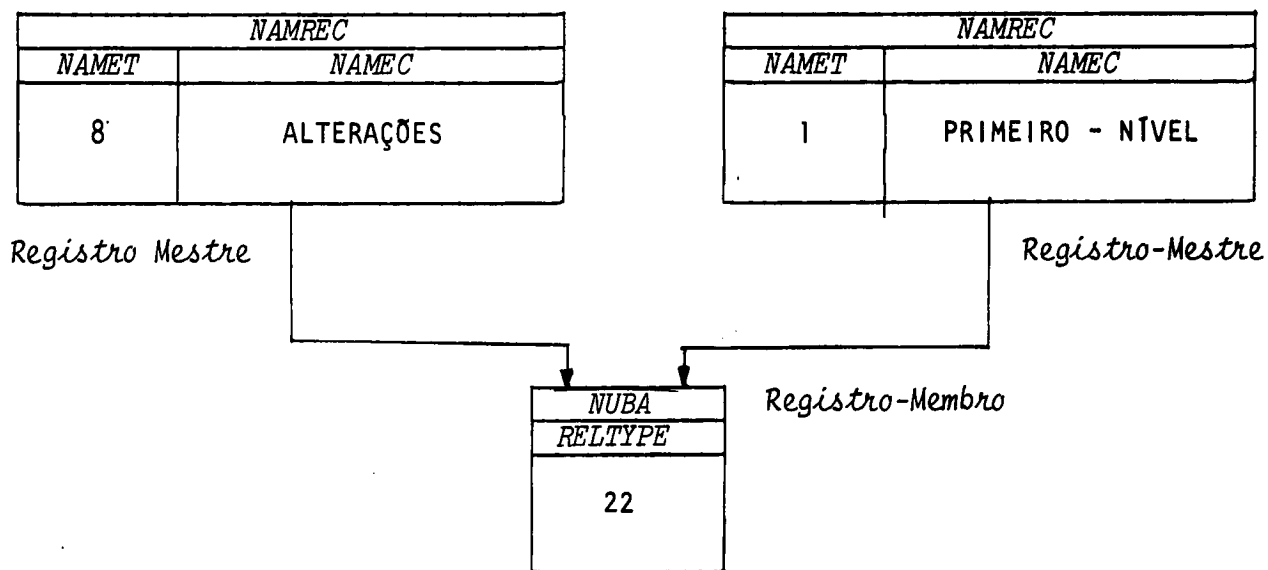
C - Registros NUBA, NUBB e NUBC

Os registros NUBA, NUBB e NUBC são usados para conectar objetos, ou

seja, para especificar relações que ocorrem entre registros NAMREC ou NUMREC. Os registros do tipo NUB são usados porque o sistema de Base de Dados não permite que sejam feitas diretamente, relações M a N. Também servem para indicar o tipo de relação que existe entre objetos. Por exemplo, havendo uma relação simples entre dois objetos, tal como *KEYWORD IS / KEYWORD FOR*, um registro NUBA é usado. Quando a relação é mais complexa, como por exemplo, na afirmação *ATTRIBUTES ARE*, empregam-se registros NUBB e NUBC. O registro NUB consiste de um único item - RELTYPE - contendo um código, que designa o tipo de relação representado no registro. Por exemplo:

PROCESS : Alterações;

KEYWORD : Primeiro-Nível;



Observação:

A disposição correspondente ao exemplo acima, será apresentada detalhadamente, mais adiante (seções H e I).

O item RELTYPE, no registro NUBA, contém o código 22, que designa a relação que ele representa. Isto significa que entre Alterações e Primeiro-Nível, há uma relação tipo 22, isto é, *KEYWORD IS / KEYWORD FOR*. Figura 2.10

D - Registros COMLIN e COM

Sempre que se quer armazenar um comentário na Base de Dados, as linhas que o compõem são armazenadas, em registros COMLIN. Somente as colunas de 1 a 72 do "input" são armazenadas.

Se a relação a ser representada for entre um objeto e as linhas de tipo comentário a eles relativas, far-se-á uma combinação de registros COM e COMLIN. Os registros COM têm um único item chamado RELTYP, contendo um código que indica o tipo da relação entre o registro NAMREC e o de entrada de comentário. Por exemplo:

ELEMENT : Número-Relativo-Componente;

DESCRIPTION ;

esse número fornece a posição da componente no produto final;

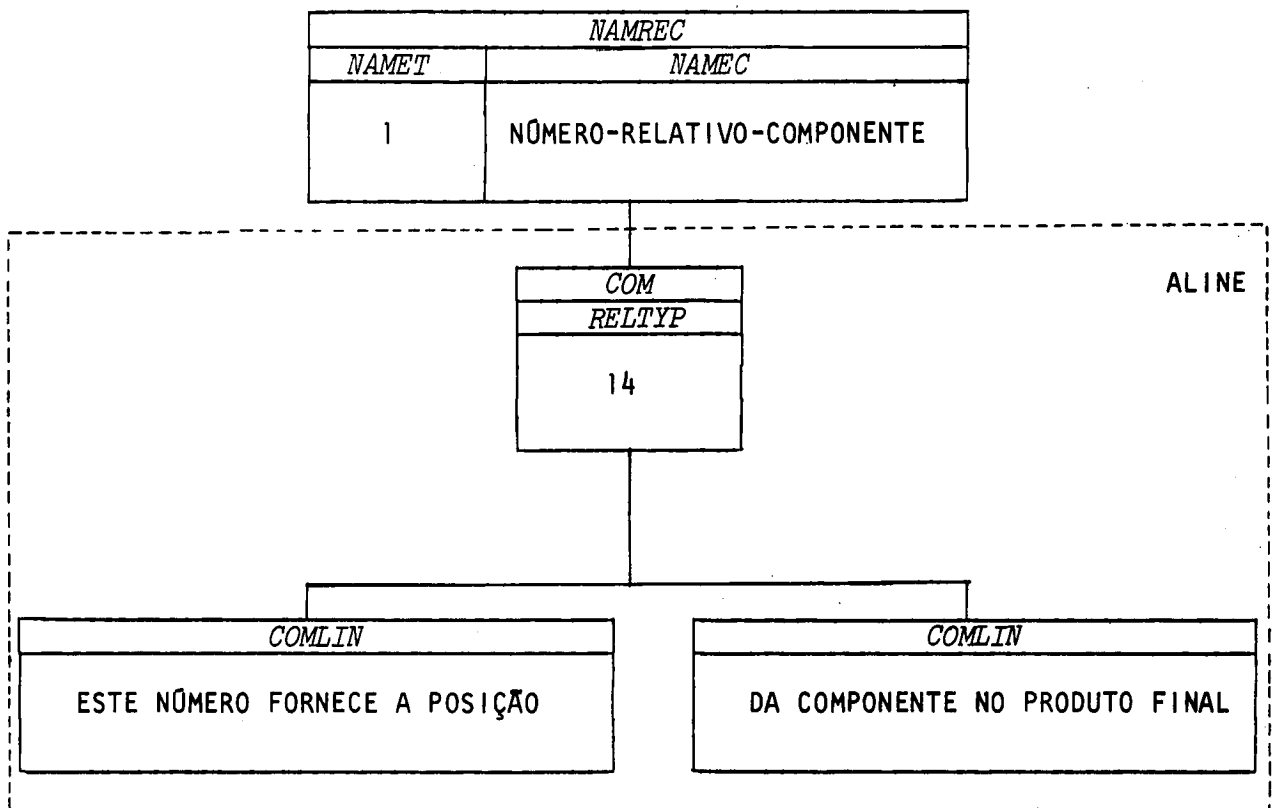


Figura 2.11

Observação:

A disposição correspondente ao exemplo acima (disposição ALINE), será apresentada detalhadamente, mais adiante (seção K).

E - Registro NUMREC

Sempre que se necessita armazenar um número inteiro na Base de Dados, usa-se um registro NUMREC. O tipo de registro NUMREC é análogo ao NAMREC, ou seja, tem os itens NAMET e NAMEC, onde são colocados, respectivamente, o tipo do número e seus caracteres e, além deles, os itens NUMVAL e FILL. O item NUMVAL põe-se a representação do número em binário; o item FILL é um enchimento (filler) necessário devido à implementação.

O System Parameter correspondente ao número inteiro 10, por exemplo, poderia ser armazenado na forma:

<i>NUMREC</i>			
<i>NAMET</i>	<i>NAMEC</i>	<i>NUMVAL</i>	<i>FILL</i>
22	DEZ	1010

F - Registro AREC

Esses registros são usados pelo analisador e não estão à disposição do usuário.

G - Disposição ALLNAM

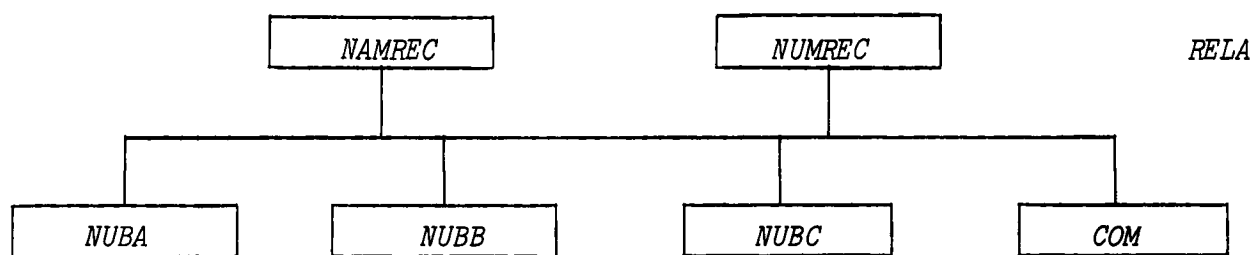
Esta disposição não é usada, como de praxe, para estabelecer relações entre registros mestres e registros membros, mas sim, para obter a ordenação

alfabética de todos os nomes e sinônimos que o usuário definiu (registros NAMREC e SYNREC). Utiliza-se esta disposição, para recuperar dados ou encontrar relações conectadas a determinado nome, na Base de Dados.

É conveccionado, nesta disposição, que o registro mestre é fictício, denominado SYSTEM, e os membros são NAMREC e SYNREC.

H - Disposição RELA

Esta disposição tem nomes e números, ou seja, registros NAMREC e NUMREC como mestres e registros NUB ou COM como membros. Serve para especificar o lado esquerdo de conexões simples e complexas. Quando a relação só pode existir entre dois nomes, a conexão é "simples" e somente envolve um registro NUBA entre eles. Envolvendo mais do que dois nomes, a conexão é "complexa" e requer dois registros NUB do tipo NUBB e NUBC. Se especificada com um comentário, somente um registro COM é usado como membro. O diagrama a seguir, mostra de uma forma geral, a disposição RELA.



Exemplos:

(1) - Conexões Simples

PROCESS : *Elaboração-Folha;*

GENERATES : *Folha-Pagamento;*

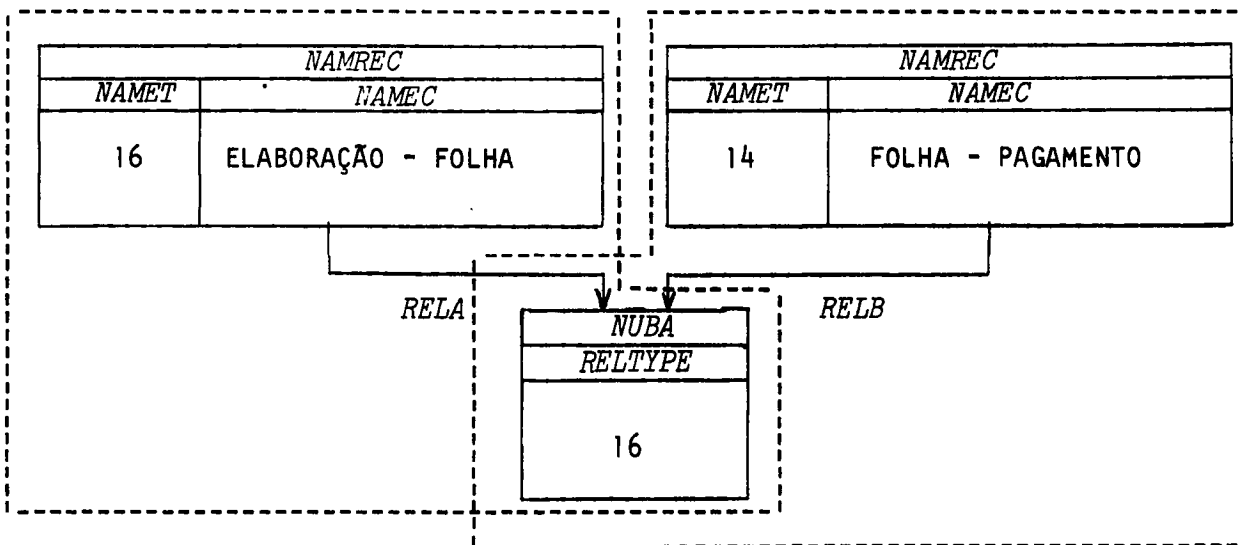


Figura 2.12

(2) - Conexões Complexas

RELATION : *Relação-Mat-Resumido-Peça;*

BETWEEN *Material-Resumido* AND *Informações-Peças;*

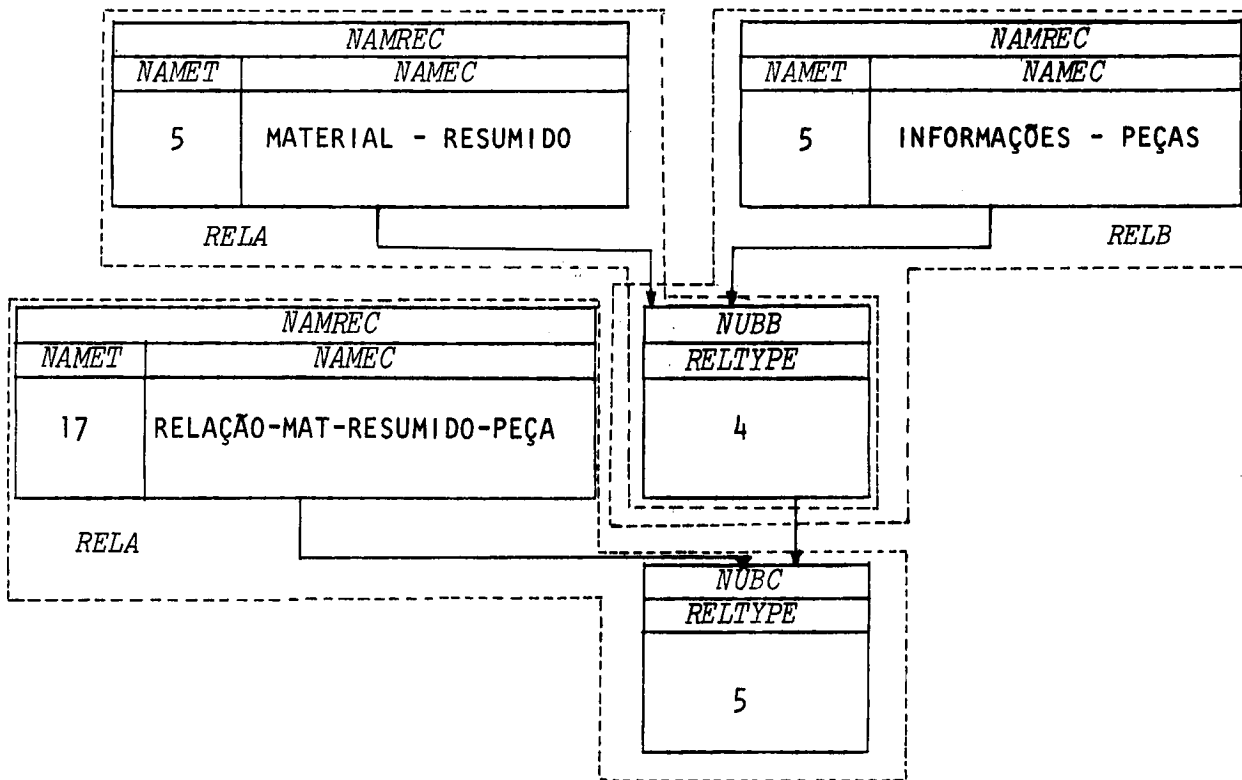


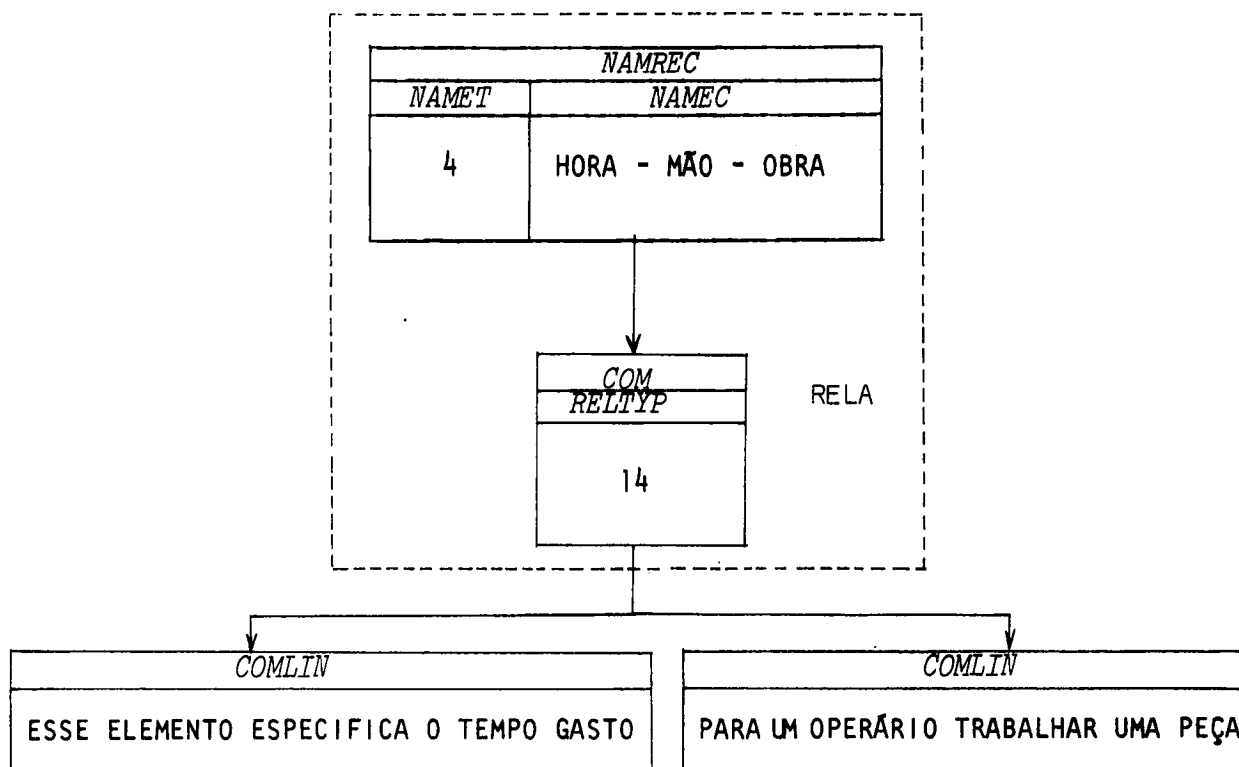
Figura 2.13

(3) - Conexão para um Comentário

ELEMENT : Hora-Mão-Obra;

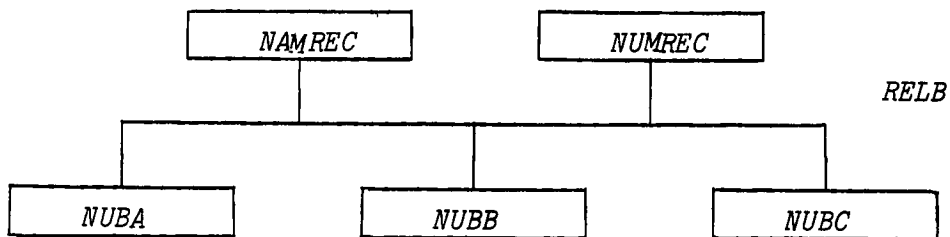
DESCRIPTION;

esse elemento especifica o tempo gasto
para um operário trabalhar uma peça;



1. Disposição RELB

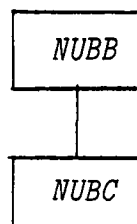
Esta disposição especifica o lado direito de relações entre objetos. Nela, NAMREC e NUMREC são registros mestres e os vários tipos de NUBs são registros membros. Devemos observar que os registros COM não podem ser membros desta disposição. Todas as relações envolvendo registros COM, devem ser definidas, através da disposição RELB. O esquema abaixo, mostra de forma geral a disposição RELB.



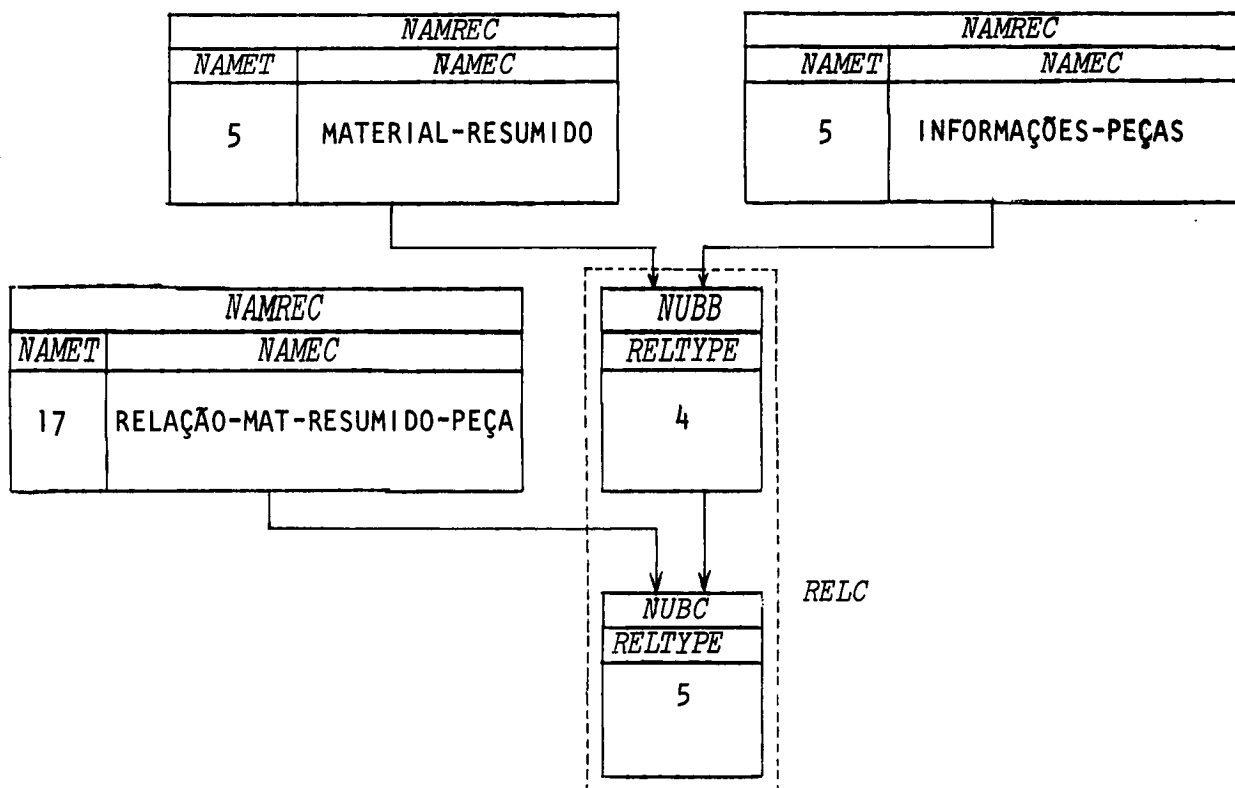
Exemplos deste tipo de disposição, estão nas Figuras 2.12 e 2.13.

J - Disposição RELC

Essa disposição é empregada na especificação de relações envolvendo mais do que dois objetos e relaciona um particular registro NUBB a um registro NUBC. O esquema geral da disposição RELC é:

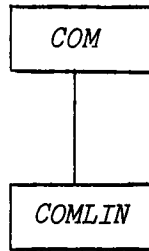


Como exemplo deste tipo de disposição, consideremos, novamente, o relativo à figura 2.13.



K - Disposição ALINE

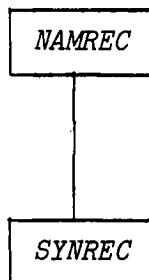
Especifica a relação entre um registro COM e seus associados COMLIN. Devemos observar que esta disposição é FIFO |20|, para assegurar que o comentário seja guardado, na mesma ordem em que foi lido. O esquema geral da disposição ALINE é:



Como exemplo, consideremos o mostrado na Figura 2.11.

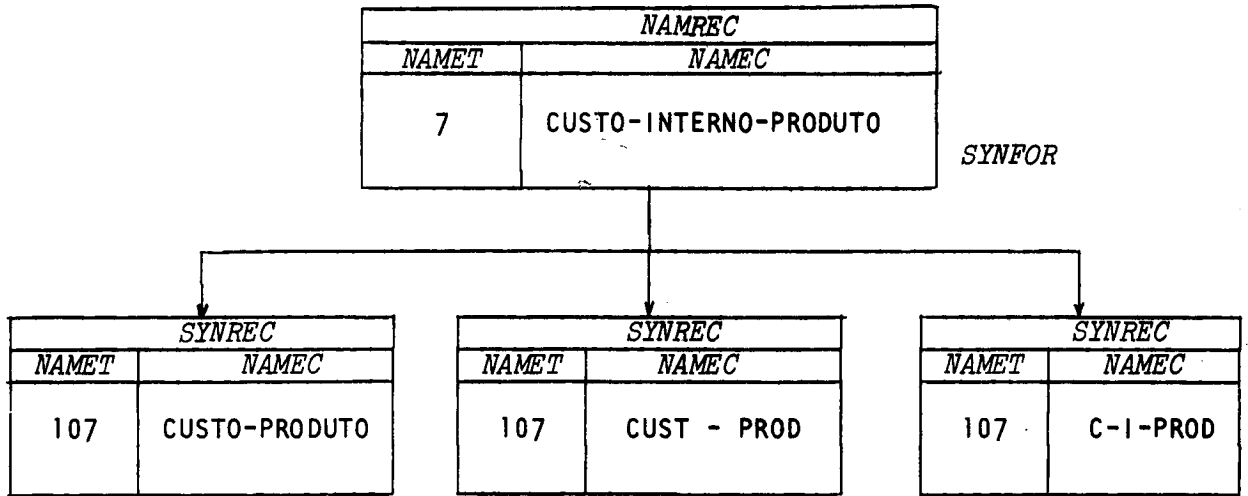
L - Disposição SYNFOR

Cada nome definido pelo usuário, pode ter um número arbitrário de sinônimos, que os registros SYNREC especificam. A relação entre o nome e os seus sinônimos é especificada pela disposição SYNFOR, onde o nome (registro NAMREC) é o mestre e os sinônimos (registros SYNREC) são membros. O esquema geral da disposição SYNFOR é:



Exemplo:

```
GROUP :    Custo-Interno-Produto;  
SYNONYMS :    Custo-Produto,  
              Cust-Prod,  
              C - I-Prod ;
```



M - Disposição ALPHA e BYNAL

As disposições ALPHA e BYNAL são conexões temporárias, utilizadas pelo analisador e o usuário não tem acesso a elas.

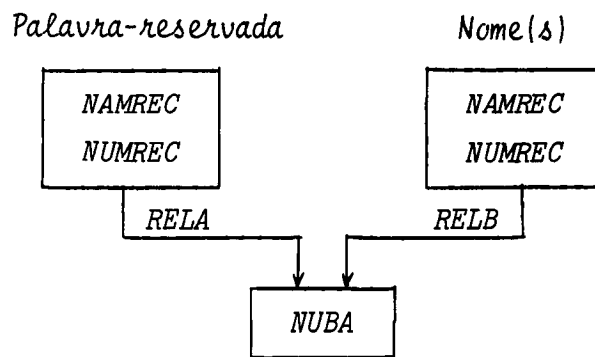


2.4.3. Diagramas da Base de Dados

Existem basicamente, quatro diferentes tipos de conexões, especificadas na Base de Dados:

(1) - Conexão Simples

Representa uma relação entre dois registros NAMREC/NUMREC e requer o uso do registro NUBA, para especificar esta relação. Emprega-se este tipo de conexão nas afirmações na Linguagem PSL da forma:



A Tabela 3.6, no Capítulo III, nas suas quatro primeiras colunas, mostra este tipo de conexão, interpretado sob seus vários aspectos.

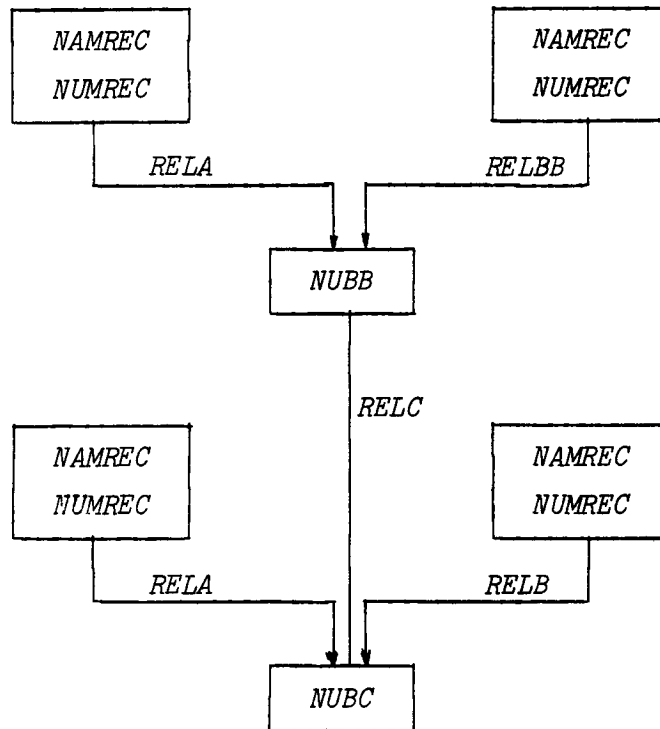
(2) - Conexão Complexa

Representa uma relação envolvendo mais do que dois registros NAMREC e/ou NUMREC. Requer uma forma (ou algum subconjunto dessa forma), tal como a mostrada logo abaixo. Em qualquer caso, essa conexão exige o uso de registros NUBB e NUBC, para conectar registros NAMREC e/ou NUMREC. Esse tipo de conexão é usado nas afirmações, na Linguagem PSL, da forma:

palavra-reservada *nome* *palavra reservada* *nome;*

palavra-reservada [*nome*] *nome;*

palavra-reservada *nomes(s)* *nomes(s);*



A Tabela 3.7, no Capítulo III, nas suas sete primeiras colunas, mostra este tipo de conexão, interpretado sob seus vários aspectos.

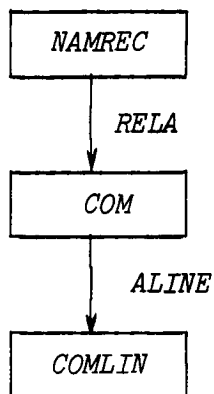
(3) - Comentários

A maneira pela qual um nome é relacionado ao comentário correspondente, é mostrada, na próxima figura. Esse tipo de conexão requer o uso de apenas um registro COM, para relacionar um registro NAMREC a registros COMLIN. Emprega-se nas afirmações, em Linguagem PSL, da forma:

palavra-reservada;

{texto}

Cada linha do texto escrito, pelo usuário, é armazenado num registro COMLIN.



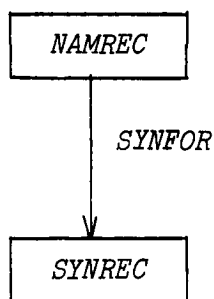
A Tabela 3.8, no Capítulo III, nas suas três primeiras colunas, mostra este tipo de conexão, interpretado sob seus vários aspectos.

(4) - Sinônimos

Sinônimos são relacionados a um nome (registro NAMREC) através da conexão mostrada abaixo. Não são necessários registros NUBs para formar essa conexão.

Emprega-se tal tipo de conexão nas afirmações em Linguagem PSL, da forma:

<i>synonym(s)</i>	<i>name(s);</i>	<i>e</i>	
<i>designate</i>	<i>name</i>	<i>synonym</i>	<i>name;</i>



2.5. Conclusões

Introduzimos o presente capítulo, em nosso estudo, visando, unicamente, fornecer material para o desenvolvimento do capítulo seguinte.

Queremos salientar que, além da introdução de conceitos de Teoria dos Grafos e Hipergrafos, Matrizes Esparsas e sua manipulação, fizemos a reformulação de [17], totalmente voltada para os objetivos do Capítulo III; procurando, em nossa abordagem, através de exemplos, tornar tal documento mais acessível às pessoas envolvidas com Sistemas de Informação e, particularmente, com o Sistema PSL/PSA.

CAPÍTULO III

O MODELO MATEMÁTICO IMPLÍCITO NA VERSÃO PLENA DO SISTEMA PSL/PSA

3.1. Considerações Gerais

Neste capítulo, nós nos preocuparemos com a explicitação do modelo matemático da linguagem PSL plena, usando os recursos que a Teoria dos Grafos e dos Hipergrafos nos proporciona. Germano [1], evidenciou para um subconjunto da linguagem PSL, isto é, a linguagem PSL desprovida de uma série de recursos, tais como alguns tipos de relações e alguns tipos de nós, que uma descrição de um Sistema de Informação, pode ser considerada como sendo um dígrafo categorizado com rótulos, e mostrou como poderia ser feita a extensão desses estudos para a linguagem PSL plena, usando Teoria de Hipergrafos. Nós, além de evidenciarmos que o modelo matemático da linguagem PSL plena é um hipergrafo dirigido, mostraremos que se pode representá-lo através de dígrafos.

3.2. Explicitação do Modelo Matemático da Linguagem PSL Plena

A versão simplificada da linguagem PSL não permite, entre outros, a utilização de relações que não binárias entre os objetos, relações essas, que são de importância significativa, se tentarmos fazer uma descrição bastante detalhada de um Sistema de Informação. Exemplificando, poderíamos considerar a situação:

Numa fábrica de compressores, o tempo gasto, na fabricação de um produto final, é calculado, tendo como base, o tempo gasto na fabricação de cada componente, que entra na confecção do produto final-compressor.

Poderíamos, descrever essa situação usando o PSL, da seguinte maneira:

Tempo-Fabricação-Componente USED BY Cálculo-Tempo-Total TO DERIVE Tempo-Total-Componente;

Isto significa que *Tempo-Fabricação-Componente* é *USED BY Cálculo-Tempo-Total* e que esse mesmo *Cálculo-Tempo-Total* *DERIVES Tempo-Total-Componente*. Além disso, acrescenta: é somente com o uso de *Tempo-Fabricação-Componente* que o *Cálculo-Tempo-Total* deriva *Tempo-Total-Componente*.

Poderíamos, perfeitamente, admitir a possibilidade de que esse mesmo processo *Cálculo-Tempo-Total* usasse outros objetos e derivasse outros mais; entretanto o objeto *Tempo-Total-Componente* é derivado por *Cálculo-Tempo-Total* com o uso de *Tempo-Fabricação-Componente*. Germano [1], analisa a modelação de uma situação desse tipo, em termos de dígrafos, da seguinte forma:

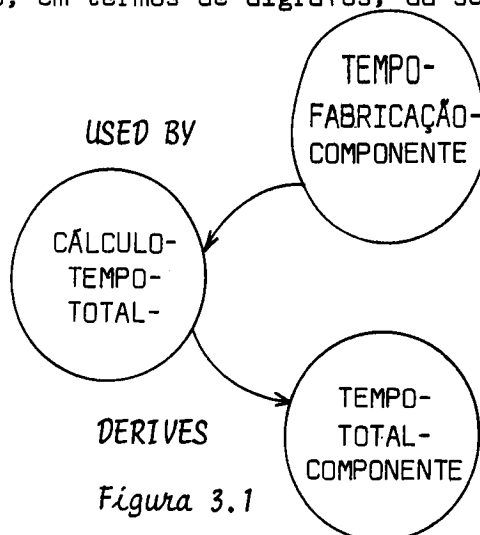


Figura 3.1

Conclui também que esse tipo de modelação, em nada satisfaz o pretendido, ou seja, olhando para a figura acima, a única conclusão a que poderíamos chegar é que:

Tempo-Fabricação-Componente USED BY Cálculo-Tempo-Total

e

Cálculo-Tempo-Total DERIVES Tempo-Total-Componente

Nada, porém, indica que *Tempo-Total-Componente* está sendo derivado usando *Tempo-Fabricação-Componente*.

Se tivéssemos uma situação um pouco mais complexa, por exemplo, considerando-se que *Tempo-Montagem-Componente* é *USED BY Cálculo-Tempo-Total TO DERIVE Tempo-Total-Montagem*, poder-se-ia pensar numa modelação, como mostra a Figura 3.2:

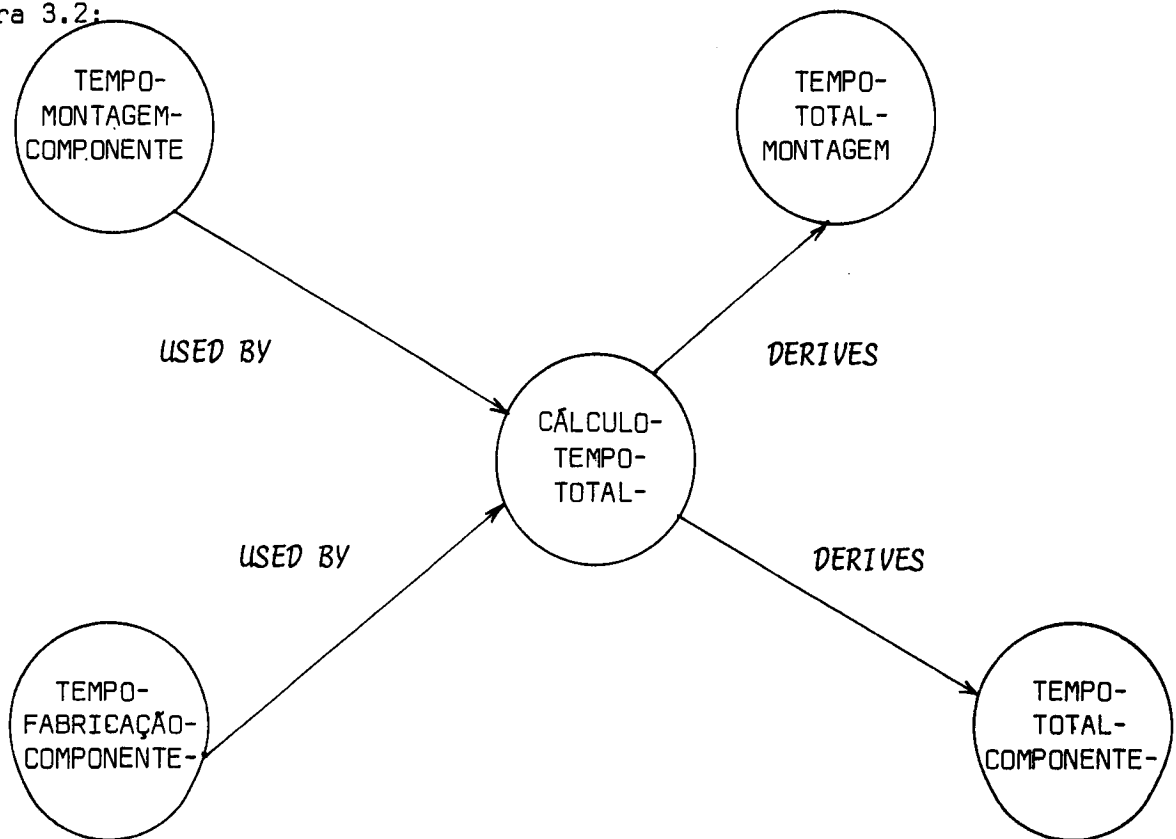


Figura 3.2

Olhando para essa representação, poderíamos concluir, por exemplo, que *Tempo-Montagem-Componente* é *USED BY Cálculo-Tempo-Total TO DERIVE Tempo-Total-Montagem*, ou que *Tempo-Fabricação-Componente* é *USED BY Cálculo-Tempo-Total TO DERIVE Tempo-Total-Componente*.

tal TO DERIVE Tempo-Total-Montagem, o que, em ambos os casos, nos conduziria a um erro.

Sabemos já, por Germano, que quando no PSL tratamos de relações entre dois objetos (relações binárias), o modelo: dígrafo categorizado com rótulos satisfaz inteiramente o pretendido, ou seja, é modelo que se adapta, perfeitamente, ao PSL simplificado; pode-se, portanto, para esse PSL, usar resultados de Teoria dos Grafos.

Já ao tratarmos de relações mais complexas entre objetos, por exemplo, relações que envolvem maior número deles (relações ternárias), vimos que o modelo de dígrafo categorizado com rótulos, como foi proposto, não corresponde às nossas expectativas, pois dá margem a interpretações completamente erradas, o que não contribui, em nada, para o pretendido.

Partimos, pois, como sugere Germano [1], para uma generalização do conceito de grafo, que seria o hipergrafo, numa tentativa de explicitação do modelo matemático para a linguagem PSL, na sua versão completa, como o implantado na Universidade de Michigan e no Centro de Computação Eletrônica da Universidade de São Paulo - SP.

Se nos concentrarmos, novamente, no exemplo a que se refere à Figura 3.2 e, para maior simplicidade chamarmos:

<i>Cálculo-Tempo-Total</i>	nó 1
<i>Tempo-Montagem-Componente</i>	nó 2
<i>Tempo-Total-Montagem</i>	nó 3
<i>Tempo-Fabricação-Componente</i>	nó 4
<i>Tempo-Total-Componente</i>	nó 5
<i>Relação USED BY</i>	R1
<i>Relação DERIVES</i>	R2

representando-a ainda como um dígrafo categorizado com rótulos, teremos a seguinte configuração:

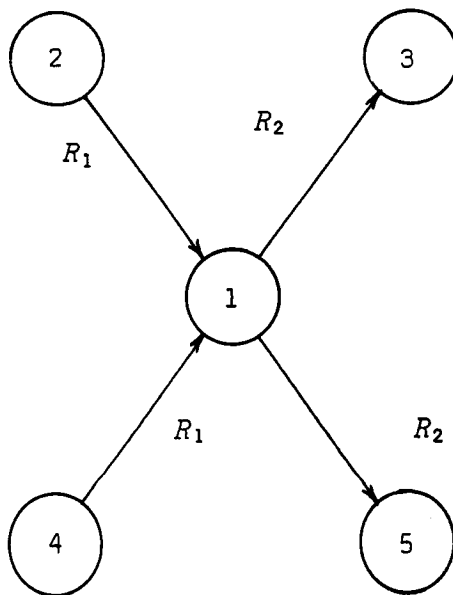


Figura 3.3.

Como já evidenciamos, esta representação está falha. Baseado nos conceitos de hipergrafos categorizados, apresentados no Capítulo II, a título de pré-requisito e informação necessária para melhor compreensão do nosso trabalho, este mesmo exemplo (Figura 3.3), olhado como um hipergrafo, teria a seguinte representação gráfica:

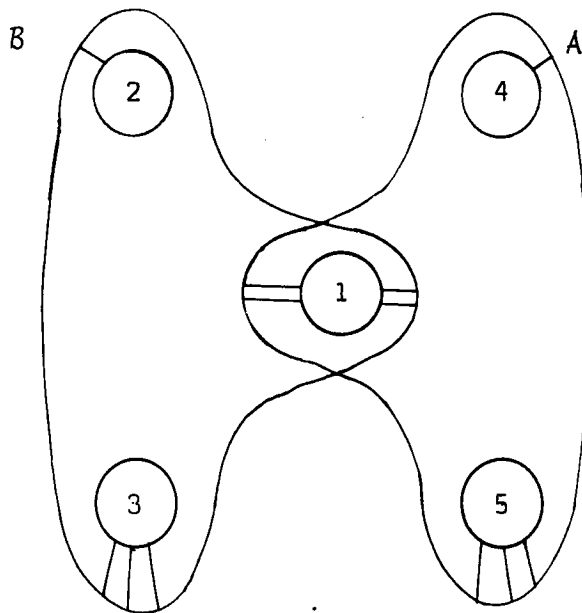


Figura 3.4.

Devemos lembrar, antes de prosseguir, que o sistema PSL/PSA, sempre que reconhece uma relação entre dois objetos, automaticamente define, também, entre eles, a sua complementar. Assim, ao especificar que *Tempo-Montagem-Componente* é USED BY *Cálculo-Tempo-Total*, automaticamente a relação *Cálculo-Tempo-Total* USES *Tempo-Montagem-Componente* é definida.

Olhando para um dos arcos do hipergrafo da Figura 3.4, por exemplo, o arco A, veremos que o nó 4 desempenha o papel 1, o nó 1, o papel 2 e o nó 5, o papel 3. Esses papéis foram atribuídos, tendo como base as relações R_1 e R_2 , anteriormente definidas. Como $R_1 = USED\ BY$ e $R_2 = DERIVES$, no arco A do hiperarco, teremos:

- papel 1 : desempenhado pelo nó que é "usado".*
- papel 2 : desempenhado pelo nó que "usa".*
- papel 3 : desempenhado pelo nó que é "derivado",*

uma vez que queremos relacionar os três nós, de forma que um deles use o outro, para derivar o terceiro.

As relações *USED BY* e *DERIVES*, representadas em termos de hipergrafos, dão origem ao aparecimento da categoria do arco do hipergrafo (*USED BY, DERIVES*) e não apenas a essa, mas também a mais três:

(*DERIVED BY, USING*), (*USES, TO DERIVE*), (*DERIVES, USING*).

Uma vez que os papéis dos nós estão fixos, no fundo as quatro categorias, que o arco A possui, têm o mesmo significado. Fazendo, então, o arco A assumir uma a uma das categorias evidenciadas, teremos as seguintes situações:

- (1) $\bar{n}o\ 4$ USED BY $\bar{n}o\ 1$ TO DERIVE $\bar{n}o\ 5$
 (papel 1) (papel 2) (papel 3)

- (2) $\bar{n}o\ 5$ DERIVED BY $\bar{n}o\ 1$ USING $\bar{n}o\ 4$
 (papel 3) (papel 2) (papel 1)

- (3) $\bar{n}o\ 1$ USES $\bar{n}o\ 4$ TO DERIVE $\bar{n}o\ 5$
 (papel 2) (papel 1) (papel 3)

(4)	nó 1	DERIVES	nó 5	USING	nó 4
	(papel 2)		(papel 3)		(papel 1)

Em todas as situações acima, o nó 4 é o que desempenha o papel 1, ou seja, é o nó usado; o nó 1 é o que desempenha o papel 2, ou seja, é o que faz uso e, finalmente, o nó 5 é o que desempenha o papel 3, ou seja, é o nó derivado.

De maneira análoga ao acontecido com o arco A do hipergrafo da Figura 3.4, acontece com o arco B, com a diferença que os nós do hipergrafo, envolvidos no arco B são: 1, 2 e 3.

Resta-nos, ainda, um pequeno problema. Como atribuir, aos arcos, essas categorias tratadas anteriormente? Ou melhor, como formalizar a atribuição dessas categorias?

Consideremos o hipergrafo (V, E) , onde V é o conjunto de nós e E é o conjunto de arcos, ou seja, E é um conjunto de subconjuntos de V , satisfazendo (Definição 4 - Capítulo II). A cada um dos elementos de E , uma categoria é atribuída, ou deve ser atribuída.

Se chamarmos de K o conjunto de todas as possíveis relações entre objetos, existentes em PSL (vide Figura 1.4), o conjunto Q , das categorias possíveis, poderia ser definido como um subconjunto de $K \cup K^2$.

É evidente que todos os pares ordenados de K^2 não podem estar presentes em Q . Por exemplo, se tomássemos o par ordenado $(DERIVES, PART OF)$ (*) como elemento de Q , isto significaria a possibilidade de acontecer que um arco (A, B, C) tivesse a categoria $(DERIVES, PART OF)$. Traduzindo para o PSL : objeto A DERIVES objeto B PART OF objeto C , o que exige a introdução de relação ternária, que pode ser expressa como duas relações binárias.

No fundo, o conjunto K^2 deveria sofrer ainda uma restrição no seguinte sentido: embora alguns elementos desse conjunto sejam completamente diferentes (por exemplo, $(DERIVES, USING)$, $(DERIVED BY, USING)$, $(USES, TO DERIVE)$, $(USED BY, TO DERIVE)$), uma vez que os papéis dos nós estejam definidos, qualquer um deles, atribuído como categoria a um arco, traduz a mesma informação,

e portanto, bastaria um único deles estar presente. Podemos então, pensar em tomar o subconjunto $T \subset K^2$, onde categorias diferentes, mas com o mesmo significado, fossem compensadas numa única representante, e onde categorias como (*), fossem eliminadas.

O hipergrafo poderia então ser notado por $H = (V, S)$, onde V é o conjunto de nós, e $S \subset E \times Q$, onde $Q \subset K \cup T$.

Se considerarmos, por exemplo, a seguinte descrição em PSL:

```
PROCESS : A;
RECEIVES : I;
GENERATES : O;
SUBPARTS : P,
           Q;
INPUT : I;
CONSISTS : G1,
          G2;
OUTPUT : O;
CONSISTS : G3;
PROCESS : P;
        USES G2 TO DERIVE G3;
GROUP : G1,
       G2,
       G3;
PROCESS : Q;
```

O hipergrafo dirigido, correspondente à descrição, é $H = (V, E)$, onde:

$$V = \{A, I, O, P, Q, G1, G2, G3\}$$

$$E = \{(A, I), (A, O), (A, P), (A, Q), (I, G1), (I, G2), (O, G3), (P, G2, G3)\}$$

Vamos agora, atribuir as categorias aos arcos, de tal forma a obter o hipergrafo dirigido categorizado $H = (V, S)$, correspondente à descrição. No

nosso caso, as categorias de K e T que nos interessam são:

RECEIVES / RECEIVED BY	Código 27
GENERATED BY / GENERATES	Código 16
PART OF / SUBPARTS ARE	Código 25
CONSISTS OF / CONTAINED	Código 9/10
USES TO DERIVE / DERIVED BY USING /	Código 12/13
DERIVES USING / USED BY TO DERIVE	

Portanto, no nosso exemplo,

$$Q = \{ 27, 16, 25, 9/10, 12/13 \}$$

$$S \subset E \times Q, S = \{ ((A,I), 27), ((A,O), 16), ((A,P), 25), ((A,Q), 25), ((I,G1), 9/10), ((I,G2), 9/10), ((O,G3), 9/10), ((P,G2,G3), 12/13) \}$$

Podemos ainda construir o hipergrafo dirigido categorizado com rótulos, relativo à descrição anterior. Para tanto, basta rotular cada nó que representa cada objeto, com o seu tipo.

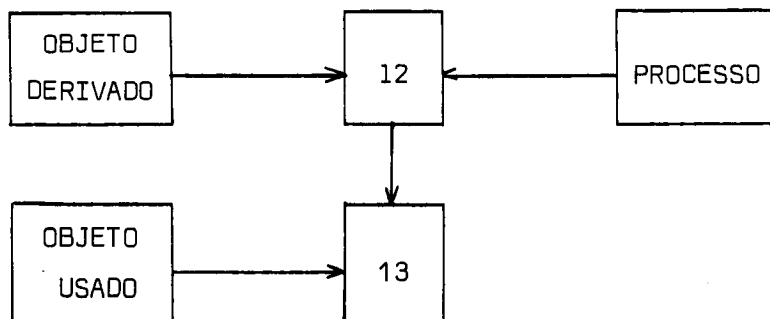
A lista completa dos tipos de objetos admissíveis pelo PSL encontra-se na Figura 2.9 do Capítulo II (seção 2.4.2).

Teríamos pois o hipergrafo $H = (F, S)$, onde: $F \subset E \times C$. No nosso exemplo, então:

$$F = \{ (A,16), (I,8), (O,14), (P,16), (9,16), (G1,7), (G2,7), (G3,7) \}$$

3.3. Representação do Modelo Matemático da Linguagem PSL, através de dígrafos

O Sistema PSL/PSA em sua versão plena, consegue representar a situação proposta na Figura 3.1, de uma forma completa. Faz isso, lançando mão da seguinte conexão:



A conexão acima, colocada em termos de Teoria dos Grafos, corresponderia ao estabelecimento de um arco e um nó, como mostra a Figura abaixo:

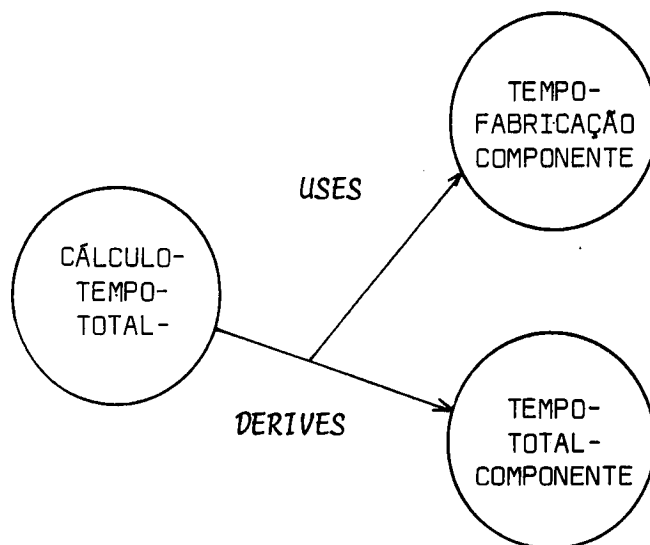


Figura 3.5.

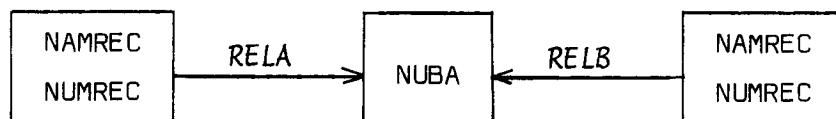
Não se consegue representar essa situação diretamente através de dígrafos, ou seja, em Teoria dos Grafos, relações são estabelecidas entre nós, mas não entre arcos e nós.

Como o Sistema PSL/PSA conseguiu a representação da Figura 3.5, isso nos permite antecipar a possibilidade da obtenção do modelo matemático de um sistema de informação descrito em PSL, tendo por base, a representação proposta pelo Sistema PSL/PSA. Em outras palavras, o nosso objetivo agora é o de

tentar exprimir o hipergrafo, anteriormente explicitado como modelo matemático do PSL completo, usando dígrafos categorizados com rótulos, e, para tanto, vamos recorrer às indicações que constam do Capítulo II.

Vamos analisar, separadamente, e com o auxílio de exemplos, os vários tipos possíveis de conexões:

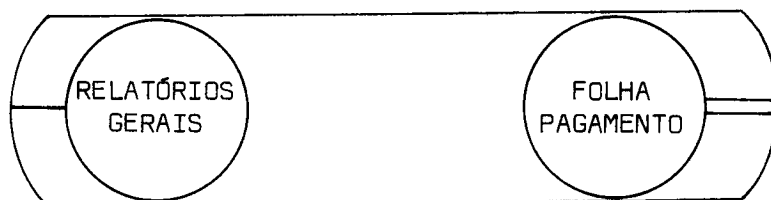
(1) - *Conexão Simples*



Exemplo:

- *OUTPUT* : *Relatórios-Gerais;*
 SUBPARTS : *Folha-Pagamento;*
- *OUTPUT* : *Folha-Pagamento;*
 PART OF : *Relatórios-Gerais;*

A representação, por hipergrafos, dessa descrição é:



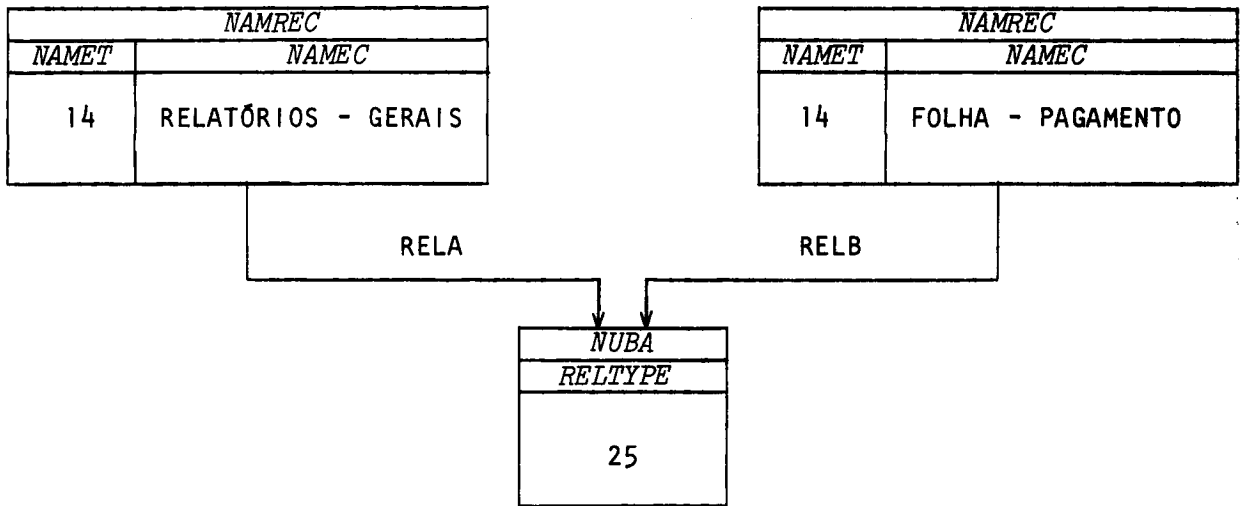
SUBPARTS ARE / PART OF
Categoria 25

onde:

papel 1 : desempenhado pelo nó do qual se está especificando uma parte.

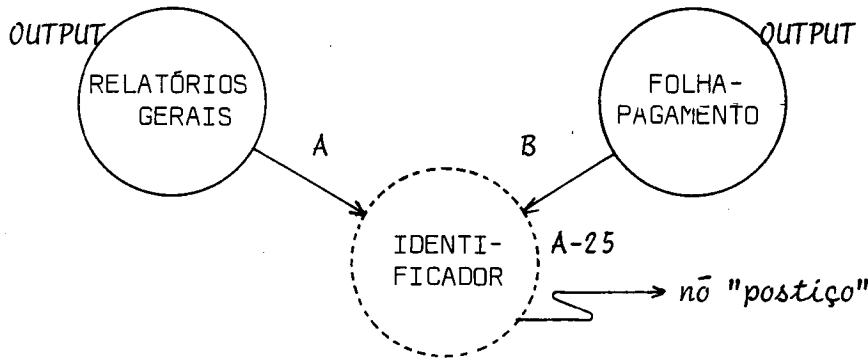
papel 2 : desempenhado pelo nó que é parte do outro.

Em termos de representação por diagramas da Base de Dados utilizada pelo PSA, temos:



O item *RELTYPE* do registro NUBA, contém o código 25, que especifica a relação *SUBPARTS ARE/ PART OF*.



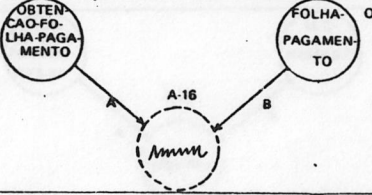
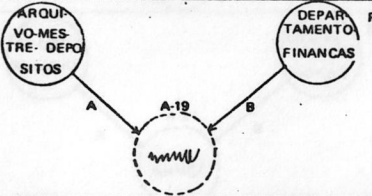
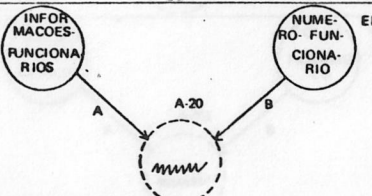
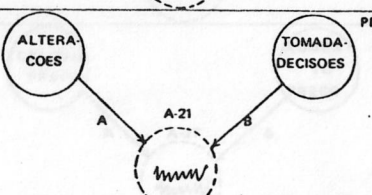
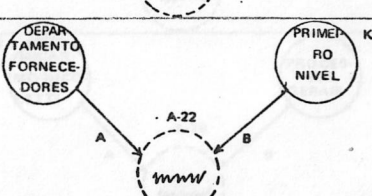
A partir do diagrama acima, passamos, então, à representação da descrição através de dígrafo categorizado com rótulos.



Na próxima tabela (Figura 3.6), damos uma listagem exhaustiva de todas as possibilidades de conexões simples, entre os vários tipos de objetos que podem ser estabelecidas em PSL.

Fig. 3.6 - Listagem exhaustiva de todas as possibilidades de conexões simples, entre os vários tipos de objetos, que podem ser estabelecidos em PSL.

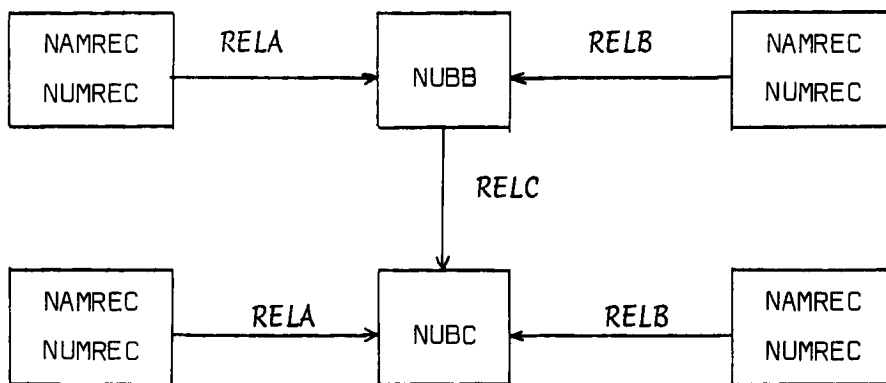
NOMENCLATURA DO NOME DO REGISTRO OU NÚMERO DE REGISTRO LIGADO A NUBANCA POR RELACIONAMENTO	NOMENCLATURA DO NOME DO REGISTRO OU NÚMERO DE REGISTRO LIGADO A NUBANCA POR RELACIONAMENTO	RELAÇÃO DO NUBANCA	EXEMPLOS	REPRESENTAÇÃO EM HIPERGRAFOS CATEGORIZADOS COM ATRIBUIÇÃO DE PAPEIS							REPRESENTAÇÃO EM DÍGRAFOS CATEGORIZADOS COM RÓTULOS
				FUNÇÃO DA CATEGORIA DO HIPERARCO	NÓ QUE DESEMPENHA PAPEL 1			NÓ QUE DESEMPENHA PAPEL 2			
					IDENTIFICADOR	RÓTULO	PAPEL 1	IDENTIFICADOR	RÓTULO	PAPEL 2	
17 (RELATION)	7 (GROUP) 4 (ELEMENT)	1	<p>OBSERVAÇÃO: OS EXEMPLOS SÃO FORMULADOS EM PSL E MOSTRAM COMO A CONEXÃO SERIA INTERPRETADA SOB SEUS VÁRIOS ASPECTOS, TAL COMO FIGURARIAM NO RELATÓRIO FORMATTED PROBLEM STATEMENT</p> <p>RELATION : RELACAO - PECA - PRODUTO; ASSOCIATED - DATA IS NOME - PECA; ELEMENT : NOME - PECA; ASSOCIATED WITH RELACAO - PECA - PRODUTO;</p>	ESPECIFICAR OS OBJETOS (GROUPS OU ELEMENTS) QUE CONTÉM INFORMAÇÃO ESPECÍFICA SOBRE A "RELATION".	RELACAO-PECA-PRODUTO	17 (RELATION)	"RELATION"	NOME-PECA	4 (ELEMENT)	OBJETO ASSOCIADO A "RELATION"	
3 (CONDITION)	6 (EVENT)	2	<p>CONDITION : CONDICAO - DE - ALTERACAO; BECOMING TRUE IS CALLED ALTERA; EVENT : ALTERA ; WHEN CONDICAO - DE - ALTERACAO BECOMES TRUE;</p>	ESPECIFICAR A OCORRÊNCIA DE UM "EVENT" QUANDO UMA "CONDITION" ASSUME O VALOR "TRUE" (VERDADEI).	CONDICAO-DE-ALTERACAO	3 (CONDITION)	"CONDITION"	ALTERA	6 (EVENT)	"EVENT"	
3 (CONDITION)	6 (EVENT)	3	<p>CONDITION : CONDICAO - DE - ALTERACAO; BECOMING FALSE IS CALLED NAO - ALTERA; EVENT : NAO - ALTERA; WHEN CONDICAO - DE - ALTERACAO BECOMES FALSE;</p>	ESPECIFICAR A OCORRÊNCIA DE UM "EVENT" QUANDO UMA "CONDITION" ASSUME O VALOR "FALSE" (FALSO).	CONDICAO-DE-ALTERACAO	3 (CONDITION)	"CONDITION"	NAO-ALTERA	6 (EVENT)	"EVENT"	
5 (ENTITY) 20 (SET) 17 (RELATION)	22 (SYSTEM - PARAMETER) NUMERO	6	<p>SET : ARQUIVO - FUNCIONARIOS; CARDINALITY IS NUMERO - DE - FUNCIONARIOS; DEFINE NUMERO - DE - FUNCIONARIOS AS A SYSTEM PARAMETER /*CARDINALITY OF: ARQUIVO - FUNCIONARIOS*/</p>	ESPECIFICAR A QUANTIDADE DE INSTÂNCIAS DE DETERMINADO OBJETO.	ARQUIVO-FUNCIONARIOS	20 (SET)	OBJETO	NUMERO-DE-FUNCIONARIOS	22 (SYSPAR)	"SYSTEM-PARAMETER QUE ESPECIFICA A QUANTIDADE DE INSTÂNCIAS DO OBJETO"	
20 (SET)	20 (SET)	15	<p>SET : ARQUIVO - FUNCIONARIOS; SUBSETS ARE : ARQUIVO-FUNCIONARIOS-HORISTAS; SET : ARQUIVO - FUNCIONARIOS - HORISTAS; SUBSET OF : ARQUIVO - FUNCIONARIOS;</p>	ESPECIFICAR OS "SUBSETS" DE UM "SET".	ARQUIVO-FUNCIONARIOS	20 (SET)	"SET"	ARQUIVO-FUNCIONARIOS-HORISTAS	20 (SET)	"SET" QUE É "SUBSET"	

NOMET DO NAM REC OU NUMREC LIGADO A NUBA POR RELB	NOMET DO NAM REC OU NUMREC LIGADO A NUBA POR RELB	RETYPE DO NUBA	EXEMPLOS	REPRESENTAÇÃO EM HIPERGRAFOS CATEGORIZADOS COM ATRIBUIÇÃO DE PAPEIS						REPRESENTAÇÃO EM DIGRAFOS CATEGORIZADOS COM RÓTULOS			
				FUNÇÃO DA CATEGORIA DO HIPERARCO	NÓ QUE DESEM PENHA PAPEL 1			NÓ QUE DESEM PENHA PAPEL 2					
													
						IDENTIFICADOR	RÓTULO	PAPEL 1	IDENTIFICADOR	RÓTULO	PAPEL 2		
16 (PROCESS)	14 (OUTPUT)	16	OBSERVAÇÃO: OS EXEMPLOS SÃO FORMULADOS EM PSL E MOSTRAM COMO A CONEXÃO SERIA INTERPRETADA SOB SEUS VÁRIOS ASPECTOS, TAL COMO FIGURARIAM NO RELATÓRIO FORMATTED PROBLEM STATEMENT PROCESS : OBTENCAO - FOLHA - PAGAMENTO; GENERATES : FOLHA - PAGAMENTO; OUTPUT : FOLHA - PAGAMENTO; GENERATED BY : OBTENCAO-FOLHA-PAGAMENTO;	ESPECIFICAR QUE UM "OUTPUT" É GERADO POR UM "PROCESS".	OBTENCAO-FOLHA-PAGAMENTO	16 (PROCESS)	"PROCESS"	FOLHA-PAGAMENTO	14 (OUTPUT)	"OUTPUT" QUE É GERADO PELO "PROCESS"	PROCESS 		
20 (SET)	9 (RWE)	19	SET: ARQUIVO - MESTRE - DEPOSITOS; RESPONSIBLE - REAL - WORLD - ENTITY IS DEPARTAMENTO - FINANÇAS; RWE : DEPARTAMENTO - FINANÇAS; RESPONSIBLE FOR : ARQUIVO - MESTRE - DEPOSITOS;	INFORMAR DA RESPONSABILIDADE DE UMA "INTERFACE" SOBRE UM "SET".	ARQUIVO-MESTRE-DEPOSITOS	20 (SET)	"SET" QUE ESTA SOB A RESPONSABILIDADE DE UMA "INTERFACE".	DEPARTAMENTO - FINANÇAS	9 (RWE)	"INTERFACE"	SET 		
5 (ENTITY)	7(GROUP) 4 (ELEMENT)	20	ENTITY : INFORMACOES - FUNCIONARIO; IDENTIFIED BY: NUMERO - FUNCIONARIO; ELEMENT : NUMERO - FUNCIONARIO; IDENTIFIES : INFORMACOES - FUNCIONARIOS;	INFORMAR A RESPEITO DO OBJETO (GROUP OU ELEMENT) QUE IDENTIFICA "ENTITY".	INFORMACOES-FUNCIONARIO	5 (ENTITY)	"ENTITY"	NUMERO-FUNCIONARIO	4 (ELEMENT)	OBJETO QUE IDENTIFICA A "ENTITY"	ENTITY 		
6 (EVENT)	16 (PROCESS)	21	PROCESS: TOMADA - DECISAO; INCEPTION - CAUSES : ALTERACAO; EVENT : ALTERACAO; ON INCEPTION OF TOMADA - DECISAO;	ESPECIFICAR O "EVENT" E O "PROCESS" QUE ORIGINA O INICIO DA QUELE.	ALTERACOES	6 (EVENT)	"EVENT"	TOMADA DECISOES	16 (PROCESS)	"PROCESS" QUE AO SE INICIAR, CAUSA O PRINCIPIO DO "EVENT"	EVENT 		
QUALQUER OBJETO EXCETO KEYWORD	11 (KEYWORD)	22	RWE : DEPARTAMENTOS - E - FORNECEDORES; KEYWORDS ARE: PRIMEIRO NIVEL; DEFINE PRIMEIRO NIVEL AS A KEYWORD ; APPLIES TO : DEPARTAMENTOS - E - FORNECEDORES;	RELACIONAR UM "KEYWORD" A UM OBJETO, VISANDO SELEÇÃO E ANÁLISE.	DEPARTAMENTOS-E-FORNECEDORES	9 (RWE)	OBJETO A QUE ESTA RELACIONADO UM "KEYWORD".	PRIMEIRO-NIVEL	11 (KEYWORD)	"KEYWORD"	RWE 		

NOMENCLATURA DO NOME DO REGISTRO OU NÚMERO DE LIGAÇÃO À NUBA POR RELACIONAMENTO	NOMENCLATURA DO NOME DO REGISTRO OU NÚMERO DE LIGAÇÃO À NUBA POR RELACIONAMENTO	TIPO DE NUBA	EXEMPLOS	REPRESENTAÇÃO EM HIPERGRAFOS CATEGORIZADOS COM ATRIBUIÇÃO DE PAPEIS						REPRESENTAÇÃO EM DÍGRAFOS CATEGORIZADOS COM RÓTULOS
				FUNÇÃO DA CATEGORIA DO HIPERARCO	NÓ QUE DESEMPEÑA PAPEL 1			NÓ QUE DESEMPEÑA PAPEL 2		
				IDENTIFICADOR	RÓTULO	PAPEL 1	IDENTIFICADOR	RÓTULO	PAPEL 2	
			OBSERVAÇÃO: OS EXEMPLOS SÃO FORMULADOS EM PSL E MOSTRAM COMO A CONEXÃO SERIA INTERPRETADA SOB SEUS VÁRIOS ASPECTOS, TAL COMO FIGURARIAM NO RELATÓRIO FORMATTED PROBLEM STATEMENT							
12 (MAILBOX)	15 (PROBLEM-DEFINER)	23	PROBLEM DEFINER: M-C-NICOLETTI; MAILBOX: PROJETO - COMPRESSORES - SP; DEFINE PROJETO - COMPRESSORES - SP AS A MAILBOX; APPLIES TO: M-C-NICOLETTI;	PROJETO-COMPRESSORES - SP	12 (MAILBOX)	"MAILBOX"	M-C-NICOLETTI	15 (PROBLEM-DEFINER)	"PROBLEM-DEFINER"	MAILBOX PROJETO-COMPRESSORES-SP A-23 M-C-NICOLETTI B PROBLEM-DEFINER
16 (PROCESS)	17 (RELATION) 21 (SUBSETTING-CRITERIA)	24	PROCESS: CALCULO - SALARIO; MAINTAINS: RELACAO - INFORMACOES - HORAS; RELATION: RELACAO - INFORMACOES - HORAS; MAINTAINED BY: CALCULO - SALARIO;	CALCULO SALARIO	16 (PROCESS)	PROCESSO QUE MANTEM A "RELATION"	RELACAO-INFORMACAO-HORAS	17 (RELATION)	"RELATION"	PROCESS CALCULO SALARIO A-24 RELACAO-INFORMACAO-HORAS B RELATION
9 (RWE) 16 (PROCESS) 8 (INPUT) 14 (OUTPUT)	9 (RWE) 16 (PROCESS) 8 (INPUT) 14 (OUTPUT)	25	OUTPUT : RELATORIOS - GERAIS ; SUBPARTS : FOLHA - PAGAMENTO; OUTPUT : FOLHA - PAGAMENTO; PART OF : RELATORIOS - GERAIS;	RELATORIOS-GERAIS	14 (OUTPUT)	OBJETO QUE SE ESPECIFICA AS PARTES	FOLHA DE PAGAMENTO	14 (OUTPUT)	OBJETO QUE É PARTE DO OUTRO	OUTPUT RELATORIOS-GERAIS A-25 FOLHA PAGAMENTO B OUTPUT
16 (PROCESS)	8 (INPUT)	27	INPUT: AUMENTO - PRECO ; RECEIVED BY: CALCULO - PRECO; PROCESS: CALCULO - PRECO; RECEIVES: AUMENTO - PRECO;	CALCULO-PRECO	16 (PROCESS)	"PROCESS"	AUMENTO-PRECO	8 (INPUT)	"INPUT" QUE É RECEBIDO PELO "PROCESS"	PROCESS CALCULO PRECO A-27 AUMENTO PRECO B INPUT
15 (PROBLEM-DEFINER)	QUALQUER OBJETO EXCETO PROBLEM-DEFINER	28	PROBLEM - DEFINER: M-CARMO-NICOLETTI; RESPONSIBLE - FOR: PROCESSOS - GERAIS; PROCESS: PROCESSOS GERAIS; RESPONSIBLE-PROBLEM-DEFINER S. M-CARMO-NICOLETTI;	M-C-NICOLETTI	15 (PROBLEM-DEFINER)	"PROBLEM-DEFINER"	PROCESSOS GERAIS	16 (PROCESS)	OBJETO SOB A RESPONSABILIDADE DO "PROBLEM-DEFINER"	PROBLEM-DEFINER M-CARMO-NICOLETTI A-28 PROCESSOS GERAIS B PROCESS

NOME DO NOME REC OU NUMREC LIGADO A NUBA POR RELA	NOME DO NOME REC OU NUMREC LIGADO A NUBA POR RELB	RETYPE DO NUBA	EXEMPLOS	REPRESENTAÇÃO EM HIPERGRAFOS CATEGORIZADOS COM ATRIBUIÇÃO DE PAPEIS						REPRESENTAÇÃO EM DÍGRAFOS CATEGORIZADOS COM RÓTULOS		
				FUNÇÃO DA CATEGORIA DO HIPERARCO	NÓ QUE DESEM PENHA PAPEL 1		NÓ QUE DESEM PENHA PAPEL 2					
			OBSERVAÇÃO: OS EXEMPLOS SÃO FORMULADOS EM PSL E MOSTRAM COMO A CONEXÃO SERIA INTERPRETADA SOB SEUS VÁRIOS ASPECTOS, TAL COMO FIGURARIAM NO RELATÓRIO FORMATTED PROBLEM STATEMENT									
IDENTIFICADOR	RÓTULO	PAPEL 1	IDENTIFICADOR	RÓTULO	PAPEL 2							
QUALQUER OBJETO EXCETO SECURITY	18 (SECURITY)	29	<p>PROCESS: DETERMINAÇÃO-PRÉCO-VENDA;</p> <p>SECURITY : CHEFE - DEPTO - VENDAS;</p> <p>DEFINE CHEFE-DEPTO-VENDAS AS A SECURITY;</p> <p>APPLIES TO: DETERMINAÇÃO - PRÉCO - VENDA;</p>	ESPECIFICAR O GRAU DE SEGURANÇA ASSOCIADO À DESCRIÇÃO DE UM OBJETO EM PSL	DETERMINAÇÃO-PRÉCO-VENDA	16 (PROCESS)	OBJETO	CHEFE-DEPTO-VENDAS	18 (SECURITY)	NÍVEL DE SEGURANÇA QUE SERÁ ATRIBUÍDO A UM OBJETO		PROCESS SECURITY
QUALQUER OBJETO EXCETO MEMO	13 (MEMO)	30	<p>PROCESS: CÁLCULO - PAGAMENTO;</p> <p>SEE-MEMO: MEMO - PROCESSO;</p> <p>MEMO: MEMO - PROCESSO;</p> <p>APPLIES TO: CÁLCULO - PAGAMENTO;</p>	ESPECIFICAR A DESCRIÇÃO DE UM OBJETO, RELACIONANDO-O A UM "MEMO".	CÁLCULO-PAGAMENTO	16 (PROCESS)	OBJETO QUE SERÁ ASSOCIADO A UM "MEMO"	MEMO-PROCESSO	13 (MEMO)	"MEMO"		PROCESS MEMO
QUALQUER OBJETO EXCETO SOURCE	19 (SOURCE)	31	<p>PROCESS: DECISÃO - DE - ALTERAÇÃO;</p> <p>SOURCE: ENGENHARIA - SUELY - SANCHES;</p> <p>DEFINE ENGENHARIA - SUELY - SANCHES AS A SOURCE;</p> <p>APPLIES TO: DECISÃO - DE - ALTERAÇÃO;</p>	IDENTIFICAR INFORMAÇÕES NÃO CONTIDAS NA DOCUMENTAÇÃO DO SISTEMA E QUE SÃO FUNDAMENTAIS PARA SEU ENTENDIMENTO	DECISÃO-DE-ALTERAÇÃO	16 (PROCESS)	OBJETO	ENGENHARIA-SUELY-SANCHES	19 (SOURCE)	"SOURCE" QUE PODE FORNECER INFORMAÇÕES EXTRAS SOBRE UM OBJETO		PROCESS SOURCE
20 (SET)	7 (GROUP) 4 (ELEMENT)	32	<p>SET: ARQUIVO - INFORMAÇÕES - FUNCIONÁRIOS;</p> <p>SUBSETTING-CRITERIA ARE: CÓDIGO-DE-TRABALHO</p> <p>ELEMENT: CÓDIGO-DE-TRABALHO;</p> <p>SUBSETTING-CRITERION FOR ARQUIVO-INFORMAÇÕES - FUNCIONÁRIOS;</p>	ESPECIFICAR O OBJETO QUE DETERMINA COMO O "SET" DEVE SER DIVIDIDO EM "SUBSETS"	ARQUIVO-INFORMAÇÕES-FUNCIONÁRIOS	20 (SET)	"SET"	CÓDIGO-DE-TRABALHO	4 (ELEMENT)	OBJETO QUE FORNECE CRITÉRIO PARA A DIVISÃO DO "SET" EM "SUBSETS"		SET ELEMENT
6 (EVENT)	16 (PROCESS)	33	<p>PROCESS: MANUTENÇÃO-ARQUIVO-ACUMULADO;</p> <p>TERMINATION - CAUSES: INÍCIO - DA - TOMADA DECISÃO;</p> <p>EVENT: INÍCIO - DA - TOMADA - DECISÃO;</p> <p>ON TERMINATION OF: MANUTENÇÃO -ARQUIVO ACUMULADO;</p>	ESPECIFICAR O "EVENT" QUE RESULTA DO TÉRMINO DO PROCESSO	INÍCIO-DA-TOMADA-DECISÃO	6 (EVENT)	"EVENT"	MANUTENÇÃO-ARQUIVO-ACUMULADO	16 (PROCESS)	"PROCESS" CUJO TÉRMINO DÁ INÍCIO A UM "EVENT"		EVENT PROCESS

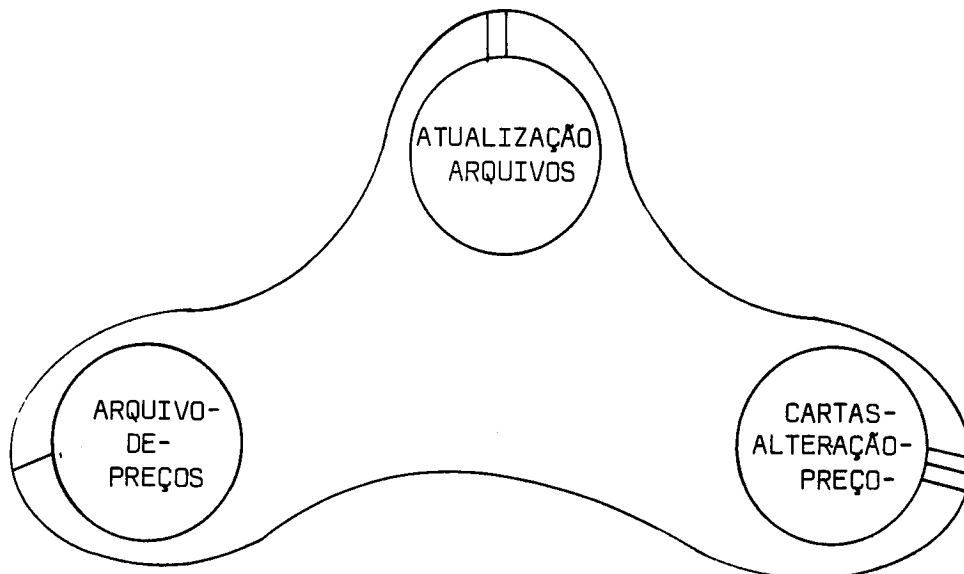
(2) - Conexão Complexa



Exemplo:

- PROCESS : *Atualização-Arquivos;*
 USES *Cartas-Alteração-Preço* TO UPDATE *Arquivo-de-Preço;*
- SET : *Arquivo-de-Preços;*
 UPDATE BY *Atualização-Arquivos* USING *Cartas-Alteração-Preço;*
- INPUT : *Cartas-Alteração-Preço;*
 USED BY *Atualização-Arquivos* TO UPDATE *Arquivo-de-Preços;*
- PROCESS : *Atualização-Arquivos;*
 UPDATES *Arquivo -de-Preços* USING *Cartas-Alteração-Preço;*

A representação, por hipergrafos, dessa descrição, é:



USES TO UPDATE / UPDATE BY USING / USED BY TO UPDATE / UPDATES USING

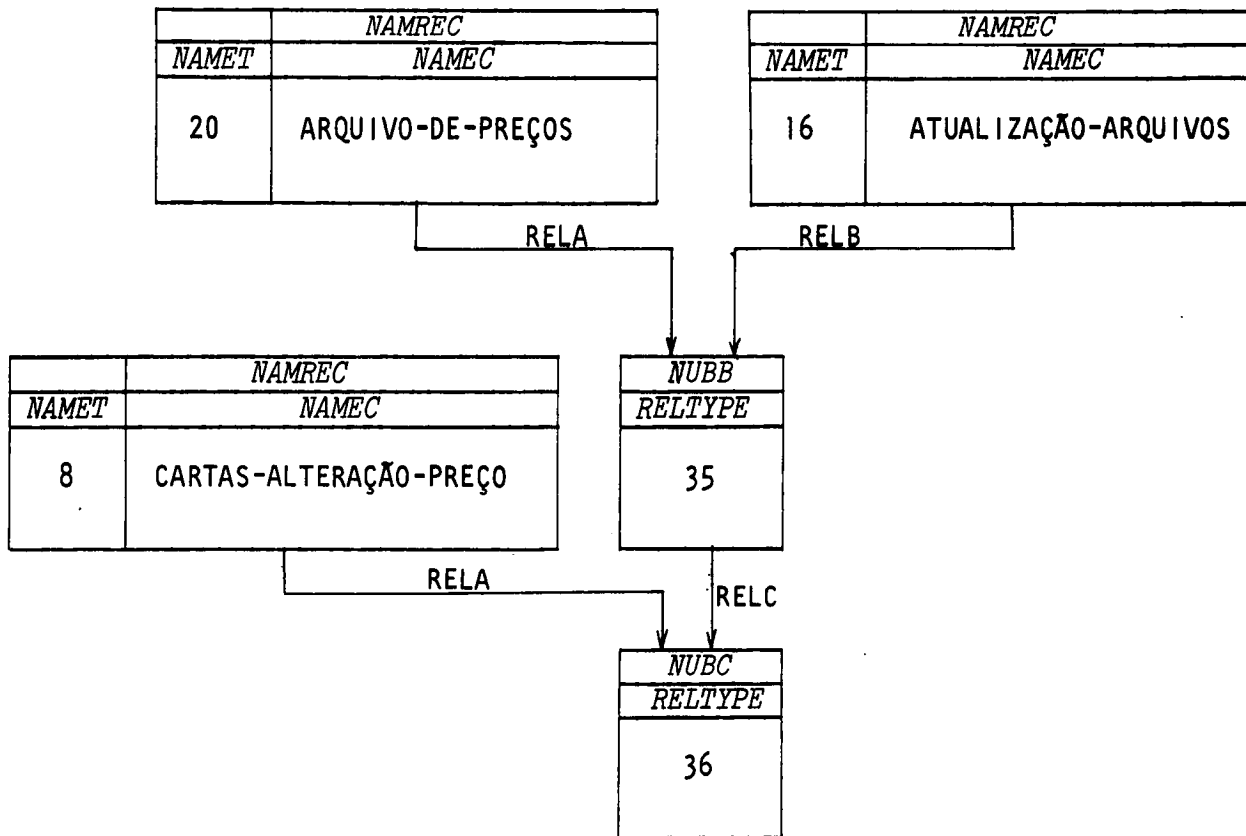
onde:

papel 1 : desempenhado pelo nó que será atualizado

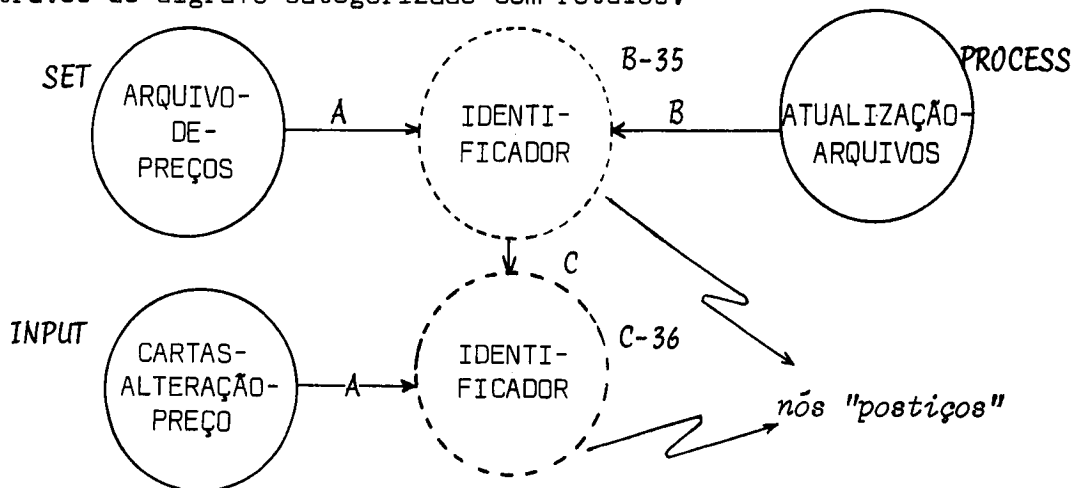
papel 2 : desempenhado pelo nó de rótulo process

papel 3 : desempenhado pelo nó é usado

Em termos de representação por diagramas da Base de Dados utilizada pelo PSA, temos:

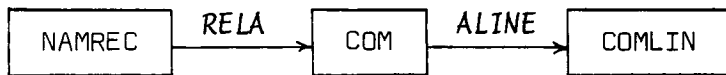


A partir do diagrama acima, passamos, então, à representação da descrição, através de dígrafo categorizado com rótulos.



Na próxima Tabela (Figura 3.7), damos uma listagem exaustiva de todas as possibilidades de conexões complexas, entre os vários tipos de objetos, que podem ser estabelecidas em PSL.

(3) - Conexão Para Comentário



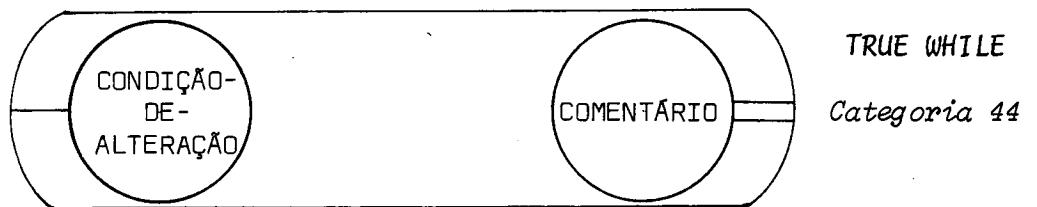
Exemplo:

- *CONDITION* : *Condição-de-Alteração*;

TRUE WHILE;

O elemento Código-Decisão tiver valor 1;

A representação, por hipergrafos, dessa descrição é:



onde:

papel 1 : desempenhado pelo nó de rótulo *condition*.

papel 2 : desempenhado pelo nó do comentário.

Em termos de representação por diagramas da Base de Dados utilizada pelo PSA, temos:

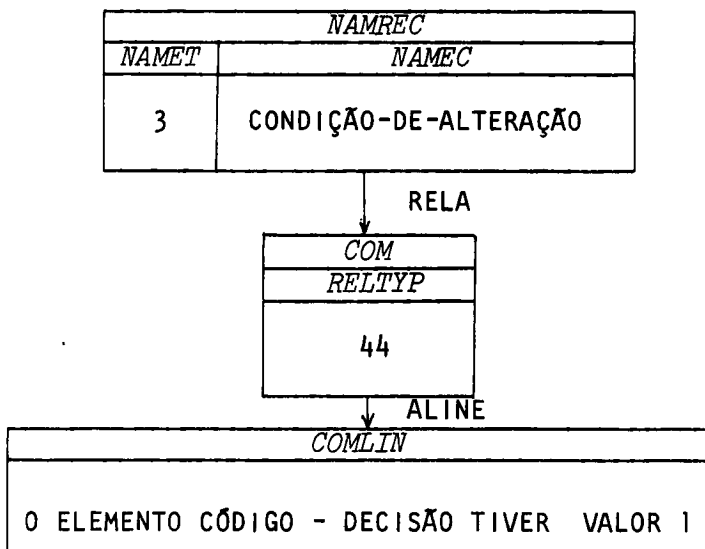


Fig. 3.7 - Listagem exhaustiva de todas as possibilidades de conexões complexas, entre os vários tipos de objetos, que podem ser estabelecidos em PSL.

NOMENCLATURA DO REGISTRO OU NÚMERO DE REC LIGADOS A NUB B POR RELA	NOMENCLATURA DO REGISTRO OU NÚMERO DE REC LIGADOS A NUB B POR RELB	NOMENCLATURA DO REGISTRO OU NÚMERO DE REC LIGADOS A NUB C POR RELA	NOMENCLATURA DO REGISTRO OU NÚMERO DE REC LIGADOS A NUB C POR RELB	RELAÇÃO DO NUB B	RELAÇÃO DO NUB C	EXEMPLOS	REPRESENTAÇÃO EM FUNÇÃO DA CATEGORIA DO HIPERARCO	HIPERGRAFO CATEGORIZADO COM ATRIBUIÇÃO DE PAPEIS												REPRESENTAÇÃO EM DIGRAFO CATEGORIZADO COM RÓTULOS
								NÓ QUE DESEM PENHA PAPEL 1			NÓ QUE DESEM PENHA PAPEL 2			NÓ QUE DESEM PENHA PAPEL 3			NÓ QUE DESEM PENHA PAPEL 4			
IDENTIFICADOR	RÓTULO	PAPEL 1	IDENTIFICADOR	RÓTULO	PAPEL 2	IDENTIFICADOR	RÓTULO	PAPEL 3	IDENTIFICADOR	RÓTULO	PAPEL 4									
5 (ENTITY)	5 (ENTITY)	17 (RELATION)	---	4	8	<p>RELAÇÃO: RELAÇÃO - MAT - RESUMIDO - COMPO; BETWEEN: MATERIAL - RESUMIDO - CUSTO - DATA AND INFORMAÇÕES COMPONENTES;</p> <p>ENTITY: MATERIAL - RESUMIDO - CUSTO - DATA; RELATED TO: INFORMAÇÕES - COMPONENTES VIA RELAÇÃO - MAT - RESUMIDO - COMPO</p> <p>ENTITY: INFORMAÇÕES - COMPONENTES; RELATED TO: MATERIAL - RESUMIDO - CUSTO - DATA VIA RELAÇÃO - MAT-RESUMIDO-COMPO;</p>	ESPECIFICAR QUAIS "ENTITIES" UMA "RELATION" RELACIONA	MATERIAL-RESUMIDO-CUSTO-DATA.	5 (ENTITY)	ENTITY	INFORMAÇÕES COMPONENTES	5 (ENTITY)	ENTITY	RELAÇÃO (RELATION)	17 (RELATION)	"RELATION"	---	---	---	<p>ENTITY MAT. RES. CUSTO - DATA</p> <p>RELATION A</p> <p>ENTITY B-4</p> <p>INFORMAÇÕES COMPONENTES</p> <p>RELATION C</p> <p>ENTITY C-6</p> <p>RELAÇÃO MATERIAL RESUMIDO COMPO</p>
22 (SYSTEM-PARAMETER)	22 (SYSTEM-PARAMETER)	17 (RELATION)	---	7	8	<p>RELAÇÃO: RELAÇÃO - MAT - RESUMIDO - COMPO; CONNECTIVITY IS UMA TO VÁRIAS;</p> <p>DEFINE UMA AS A SYSTEM-PARAMETER; /* LEFT CONNECTIVITY OF: RELAÇÃO - MAT - RESUMIDO - COMPO */</p> <p>DEFINE VÁRIAS AS A SYSTEM-PARAMETER; /* RIGHT CONNECTIVITY OF: RELAÇÃO - MAT - RESUMIDO - COMPO */</p>	ESPECIFICAR O TIPO DE CORRESPONDÊNCIA QUE UMA "RELATION" ESTABELECE ENTRE DUAS "ENTITIES"	UMA	22 (SYSTEM-PARAMETER)	"SYSTEM-PARAMETER"	VÁRIAS	22 (SYSTEM-PARAMETER)	"SYSTEM-PARAMETER"	RELAÇÃO (RELATION)	17 (RELATION)	"RELATION"	---	---	---	<p>SYS- PARA- UMA</p> <p>RELATION A</p> <p>SYS- PARA- VÁRIAS</p> <p>RELATION B-7</p> <p>RELAÇÃO MATERIAL-RESUMIDO-COMPO</p> <p>RELATION C</p> <p>ENTITY C-8</p>
20 (SET)	8 (INPUT)	14 (OUTPUT)	5 (ENTITY)	7 (GROUP)	4 (ELEMENT)	<p>SET: ARQUIVO - OPERÁRIOS; CONSISTS OF: NÚMERO - DE - OPERÁRIOS REGISTRO - DE - OPERÁRIOS</p> <p>ENTITY: REGISTRO - DE - OPERÁRIOS; CONTAINED IN; ARQUIVO - DE - OPERÁRIOS;</p> <p>DEFINE NÚMERO - DE - OPERÁRIOS AS A SYSTEM-PARAMETER; /* ARQUIVO - DE - OPERÁRIOS CONSISTS OF NÚMERO - DE - OPERÁRIOS REGISTRO - DE - OPERÁRIOS */</p>	ESPECIFICAR OS OBJETOS DE QUE DETERMINADO OBJETO CONSISTE, PODENDO CONTER INFORMAÇÃO QUANTO À QUANTIDADE DE CADA OBJETO QUE ENTRA NESTA CONSTITUIÇÃO	ARQUIVO - OPERÁRIOS	20 (SET)	OBJETO SOBRE CUA CONSTITUIÇÃO ESTÁ FORNEC. INFORM.	---	---	---	NÚMERO DE OPERÁRIOS	22 (SYSTEM-PARAMETER)	REGISTRO DE OPERÁRIOS	8 (ENTITY)	OBJETO QUE ENTRA NA CONSTITUIÇÃO DE OUTRO	<p>SET ARQUIVO-OPERÁRIOS</p> <p>RELATION A</p> <p>ENTITY B-9</p> <p>OBJETO QUE ENTRA NA CONSTITUIÇÃO DE OUTRO</p> <p>RELATION C</p> <p>SYS- PARA- NÚMERO-DE-OPERÁRIOS</p> <p>RELATION A</p> <p>ENTITY C-10</p> <p>REGISTRO DE OPERÁRIOS</p>	
20 (SET)	14 (OUTPUT)	6 (ENTITY)	7 (GROUP)	4 (ELEMENT)	16 (PROCESS)	<p>PROCESS: DETERMINAÇÃO - FAIXA - FUNCIONÁRIO; USES INFORMAÇÕES - FUNCIONÁRIO TO DERIVE NÚMERO - FAIXA;</p> <p>ELEMENT: NÚMERO - FAIXA DERIVED BY: DETERMINAÇÃO - FAIXA - FUNCIONÁRIO USING INFORMAÇÕES - FUNCIONÁRIO</p> <p>ENTITY: INFORMAÇÕES - FUNCIONÁRIO; USED BY: DETERMINAÇÃO - FAIXA - FUNCIONÁRIO TO DERIVE NÚMERO - FAIXA</p> <p>PROCESS: DETERMINAÇÃO - FAIXA - FUNCIONÁRIO; DERIVES NÚMERO - FAIXA USING INFORMAÇÕES - FUNCIONÁRIOS;</p>	ESPECIFICAR QUE DETERMINADO PROCESSO DERIVA UM OBJETO - PODENDO CONTER INFORMAÇÃO QUANTO AO OBJETO USADO PARA ESSA DERIVAÇÃO.	NÚMERO FAIXA	14 (ELEMENT)	OBJETO DERIVADO PELO "PROCESS"	DETERMINAÇÃO DO FAIXA FUNCIONÁRIO	16 (PROCESS)	"PROCESS"	INFORMAÇÕES FUNCIONÁRIO	6 (ENTITY)	OBJETO USADO PELO "PROCESS"	---	---	---	<p>ELEMENT NÚMERO FAIXA</p> <p>RELATION A</p> <p>ENTITY B-12</p> <p>DETERMINAÇÃO FAIXA FUNCIONÁRIO</p> <p>RELATION C</p> <p>ENTITY INFORMAÇÕES FUNCIONÁRIOS</p>
6 (EVENT)	8 (INPUT)	16 (PROCESS)	14 (OUTPUT)	10 (INTERVAL)	17	18	<p>OUTPUT: FOLHA - PAGAMENTO - SEMANAL; HAPPENS UMA TIMES - PER SEMANA;</p> <p>DEFINE UMA AS A SYSTEM - PARAMETER; /* FOLHA - PAGAMENTO - SEMANAL HAPPENS UMA TIMES - PER SEMANA */</p> <p>INTERVAL SEMANA; /* FOLHA - PAGAMENTO - SEMANAL HAPPENS UMA TIMES - PER SEMANA */</p>	ESPECIFICAR O ACONTECIMENTO OU PRODUÇÃO DE DETERMINADO OBJETO EM CERTO INTERVALO DE TEMPO.	FOLHA PAGAMENTO-PES-SOAL	14 (OUTPUT)	OBJETO QUE É PRODUZIDO OU ACONTECE	SEMANA	10 (INTERVAL)	PERÍODO DE TEMPO	22 (SYSTEM-PARAMETER)	"SYSTEM-PARAMETER que informa sobre o no. de instâncias de determinado obj. ocorre"	---	---	---	<p>OUTPUT FOLHA-PAGAMENTO-SEMANAL</p> <p>RELATION A</p> <p>ENTITY B-17</p> <p>SEMANA</p> <p>RELATION C</p> <p>SYS- PARA- UMA</p> <p>RELATION A</p> <p>ENTITY C-18</p>

NOMET DO NAM REC OU NUM REC LIGADOS A NUB B POR RELA	NOMET DO NAM REC OU NUM REC LIGADOS A NUB B POR RELB	NOMET DO NAM REC OU NUM REC LIGADO A NUB C POR RELA	NOMET DO NAM REC OU NUM REC LIGADO A NUB C POR RELB	RELTYPE DO NUB B	RELTYPE DO NUB C	EXEMPLOS	REPRESENTAÇÃO EM FUNÇÃO DA CATEGORIA DO HIPERARCO	HIPERGRAFO CATEGORIZADO COM ATRIBUIÇÃO DE PAPEIS												REPRESENTAÇÃO EM DÍGRAFO CATEGORIZADO COM RÓTULOS
								NÓ QUE DESEM PENHA PAPEL 1			NÓ QUE DESEM PENHA PAPEL 2			NÓ QUE DESEM PENHA PAPEL 3			NÓ QUE DESEM PENHA PAPEL 4			
IDENTIFICAÇÃO			RÓTU LO	PAPEL 1	IDENTIFICAÇÃO			RÓTU LO	PAPEL 2	IDENTIFICAÇÃO			RÓTU LO	PAPEL 3	IDENTIFICAÇÃO			RÓTU LO	PAPEL 4	
						<p>OBSERVAÇÃO: OS EXEMPLOS SÃO FORMULADOS EM PSL E MOSTRAM COMO A CONEXÃO SERIA INTERPRETADA SOB SEUS VÁRIOS ASPECTOS TAL COMO FIGURARIAM NO RELATÓRIO FORMATTED PROBLEM STATEMENT.</p>	<p>PROCESS: ATUALIZAÇÃO - ARQUIVOS; USES CARTAS-ALTERAÇÃO-PRECOS TO UPDATE ARQUIVO-DE-PRECO; SET: ARQUIVO-DE-PRECO; UPDATE BY ATUALIZAÇÃO-ARQUIVOS USING CARTAS-ALTERAÇÃO-PRECOS; INPUT: CARTAS-ALTERAÇÃO-PRECOS; USED BY ATUALIZAÇÃO-ARQUIVOS TO UPDATE ARQUIVO-DE-PRECO;</p> <p>PROCESS: ATUALIZAÇÃO - ARQUIVOS; UPDATES ARQUIVO - DE - PRECOS USING CARTAS - ALTERAÇÃO - PRECO;</p>	<p>ESPECIFICAR QUE DETERMINADO PROCESSO ATUALIZA UM OBJETO PODENDO TAMBÉM CONTER INFORMAÇÃO QUANTO AO OBJETO USADO PARA ESSA ATUALIZAÇÃO.</p>					<p>ARQUIVO-DE-PRECO</p> <p>20 (SET)</p> <p>OBJETO ATUALIZADO - PELO "PROCESS"</p>	<p>ATUALIZAÇÃO-ARQUIVOS</p> <p>18 "PROCESS"</p> <p>"PROCESS"</p>	<p>CARTAS-ALTERAÇÃO-PRECOS</p> <p>8 (INPUT)</p> <p>OBJETO USADO PELO "PROCESS"</p>	<p>SET</p> <p>ARQUIVO-DE-PRECO</p> <p>ATUALIZAÇÃO-ARQUIVOS</p> <p>CARTAS-ALTERAÇÃO-PRECOS</p>	<p>PROCESS</p> <p>ATUALIZAÇÃO-ARQUIVOS</p>			
<p>20 (SET)</p> <p>8 (ENTITY)</p> <p>7 (GROUP)</p> <p>4 (ELEMENT)</p>	<p>18 (PROCESS)</p>	<p>20 (SET)</p> <p>8 (INPUT)</p> <p>8 (ENTITY)</p> <p>7 (GROUP)</p>	<p>38</p>	<p>38</p>	<p>PROCESS: ATUALIZAÇÃO - ARQUIVOS; UPDATES ARQUIVO - DE - PRECOS USING CARTAS - ALTERAÇÃO - PRECO;</p>	<p>DEFINIR VALOR NUMÉRICO A UM OBJETO.</p>	<p>NÚMERO OCORRÊNCIAS</p> <p>22 (SYSTEM-PARAMETER)</p> <p>OBJETO AO QUAL SE ESTÁ ATRIBUINDO VALOR NUMÉRICO</p>	<p>1</p>	<p>1</p>	<p>1</p>	<p>1</p>	<p>VALOR NUMÉRICO</p>	<p>SYSPAR</p> <p>UMA</p> <p>VALOR NUMÉRICO</p>	<p>1</p>	<p>1</p>	<p>1</p>	<p>VALOR NUMÉRICO</p>			
<p>22 (SYSTEM-PARAMETER)</p> <p>4 (ELEMENT)</p> <p>2 (ATTRIBUTE-VALUE)</p>	<p>NÚMERO</p>	<p>NÚMERO</p>	<p>38</p>	<p>40</p>	<p>DEFINIR VALOR NUMÉRICO A UM OBJETO.</p>	<p>ESTABELECE LIMITES DE VARIAÇÃO QUE DETERMINADO OBJETO PODE ASSUMIR</p>	<p>NÚMERO-FUNÇÃO-HORISTAS</p> <p>22 (SYSTEM-PARAMETER)</p> <p>OBJETO SOBRE O QUAL SE DEFINE LIMITES DE VARIAÇÃO</p>	<p>300</p>	<p>300</p>	<p>1000</p>	<p>1000</p>	<p>VALOR NUMÉRICO QUE INDICA O LIMITE MÍNIMO.</p>	<p>SYSPAR</p> <p>300</p> <p>VALOR NUMÉRICO QUE INDICA O LIMITE MÁXIMO.</p>	<p>1000</p>	<p>1000</p>	<p>1000</p>	<p>VALOR NUMÉRICO QUE INDICA O LIMITE MÁXIMO.</p>			
<p>QUALQUER OBJETO EXCETO ATRIBUTO OU ATTRIBUTE-VALUE</p>	<p>1 (ATTRIBUTE)</p>	<p>2 (ATTRIBUTE-VALUE) OU NÚMERO</p>	<p>47</p>	<p>48</p>	<p>OUTPUT: FOLHA - PAGAMENTO - CONTRATADOS; ATTRIBUTES ARE: COPIAS DUAS;</p> <p>DEFINE COPIAS AS AN ATTRIBUTE; /* VALUES ARE: GUAS FOR FOLHA - PAGAMENTO - CONTRATADOS */</p> <p>DEFINE DUAS AS /*N ATTRIBUTE - VALUE;</p>	<p>ESPECIFICAR O ATRIBUTO ASSOCIADO A UM OBJETO E O VALOR QUE ESSE ATRIBUTO ASSUME.</p>	<p>FOLHA-PAGAMENTO-CONTRATADOS</p> <p>14 (OUTPUT)</p> <p>OBJETO SOBRE O QUAL SE ESPECIFICA UM ATRIBUTO</p>	<p>1</p>	<p>1</p>	<p>1</p>	<p>1</p>	<p>VALOR NUMÉRICO QUE INDICA O LIMITE MÍNIMO.</p>	<p>OUTPUT</p> <p>COPIAS</p> <p>ATTRIBUTE-VALUE</p>	<p>1</p>	<p>1</p>	<p>1</p>	<p>VALOR NUMÉRICO QUE INDICA O LIMITE MÁXIMO.</p>			
<p>10 (INTERVAL)</p>	<p>22 (SYSTEM-PARAMETER) OU NÚMERO</p>	<p>10 (INTERVAL)</p>	<p>49</p>	<p>50</p>	<p>INTERVAL: ANO; CONSISTS OF 12 MES;</p> <p>INTERVAL: MÊS; /*ANO CONSISTS OF 12 MES*/</p>	<p>ESPECIFICAR O INTERVALO CONSTITUINTE DE OUTRO PODENDO CONTER INFORMAÇÕES QUANTO A QUANTIDADE DESSE INTERVALO CONSTITUINTE.</p>	<p>ANO</p> <p>10 (INTERVAL)</p> <p>INTERVALO DE TEMPO QUE ESTÁ ESPECIFICANDO EM TERMOS DE INTERV. CONST.</p>	<p>12</p>	<p>12</p>	<p>12</p>	<p>12</p>	<p>VALOR NUMÉRICO QUE EXPRESSA A QUANTIDADE DE INTERVALOS CONSTITUINTE</p>	<p>INTERVAL</p> <p>ANO</p> <p>INTERVALO DE TEMPO CONSTITUINTE</p>	<p>12</p>	<p>12</p>	<p>12</p>	<p>VALOR NUMÉRICO QUE EXPRESSA A QUANTIDADE DE INTERVALOS CONSTITUINTE</p>			

A partir do diagrama anterior, passamos, então à representação da descrição, através de dígrafo categorizado com rótulos.



Na próxima Tabela (Figura 3.8), damos uma listagem exhaustiva de to das as possibilidades de conexões com comentário, dos vários tipos de objetose comentário, que podem ser estabelecidas em PSL.

O seguinte tipo de conexão (conexão com sinônimos) é reconhecidamente um dígrafo categorizado e com rótulos, pois a disposição *SYNFOR* consiste de um registro mestre do tipo *NAMREC* e de registros membros do tipo *SYNREC*.

Apresentamos, a seguir, um exemplo prático de uma descrição em PSL, de um sistema bem simples, com o intuito de expormos as representações de uma descrição em PSL, evidenciadas nesse trabalho.



Fig. 3.8. - Listagem exhaustiva de todas as possibilidades de conexões com comentário, dos vários tipos de objetos e comentário, que podem ser estabelecidas em PSL.

UNIVERSIDADE DE SÃO PAULO/CCE

A S - I S S O U R C E L I S T I N G

PARAMETERS FOR: SYNU

SOURCE XREF UPDATE DBREF

LINE	S T M T	ID FIELD
1	>PROCESS:	CLASSIFICACAO-FUNCIONARIOS;
2	>	SYNONYM: CLAS-FUNC;
3	>	RECEIVES: FOLHA-PREENCHIDA-DADOS;
4	>	GENERATES: CODIGO-NIVEL-FUNCIONARIO;
5	>	SUBPARTS: DETERMINACAO-GRAU-APTIDAO,
6	>	DETERMINACAO-GRAU-POTENCIALID;
7	>INPUT	FOLHA-PREENCHIDA-DADOS;
8	>	CONSISTS: DADOS-Pessoais,
9	>	DADOS-PROFISSIONAIS;
10	>OUTPUT:	CODIGO-NIVEL-FUNCIONARIO;
11	>	DESCRIPTION;
12	>	COM ESSE RELATORIO SERAO CAL-
13	>	CULADAS AS COMISSOES SEMES -
14	>	TRAI DOS FUNCIONARIOS;
15	>	CONSISTS: INFORMACOES-NIVEL;
16	>PROCESS:	DETERMINACAO-GRAU-APTIDAO;
17	>	USES DADOS-PROFISSIONAIS TO DERIVE INFORMACOES-NIVEL;
18	>GROUP:	DADOS PESSOAIS,
19	>	DADOS PROFISSIOAIS,
20	>	INFORMACOES-NIVEL;
21	>PROCESS:	DETERMINACAO-GRAU-POTENCIALID;
22	>EOF	

UNIVERSIDADE DE SÃO PAULO/CCE

CROSS REFERENCE

SEQ N A M E	T Y P E
1 CLAS-FUNC	2 SYNONYM FOR CLASSIFICACAO-FUNCIONARIOS
2 CLASSIFICACAO-FUNCIONARIOS	1 PROCESS
3 CODIGO-NIVEL-FUNCIONARIO	4 OUTPUT 10
4 DADOS-PESSOAIS	8 GROUP 18
5 DADOS-PROFISSIONAIS	9 GROUP 17 19
6 DETERMINACAO-GRAU-APTIDAO	5 PROCESS 16
7 DETERMINACAO-GRAU-POTENCIALIDAD	6 PROCESS 21
8 FOLHA-PREENCHIDA-DADOS	3 INPUT 7
9 INFORMACOES-NIVEL	15 GROUP 17 20

NOME DO NOME REC OU NUM REC LIG. AO REGISTRO COM POR RELA	RELTYP DO REGISTRO COM	EXEMPLOS	REPRESENTAÇÃO EM	HIPERGRAFOS CATEGORIZADOS COM ATRIB.. DE PAPEIS						REPRESENTAÇÃO EM DÍGRAFO CATEGORIZADO COM RÓTULOS		
			FUNÇÃO DA CATEGORIA DO HIPERARCO	NÓ QUE DESEMPENHA PAPEL 1			NÓ QUE DESEMPENHA PAPEL 2					
				IDENTIFICADOR	RÓTULO	PAPEL 1	IDENTIFICADOR	RÓTULO	PAPEL 2			
		OBSERVAÇÃO: OS EXEMPLOS SÃO FORMULADOS EM PSL E MOSTRAM COMO A COEÇÃO FIGURARIA NO RELATÓRIO FORMATTED PROBLEM STATEMENT										
17 (RELATION) 20 (SET)	11	SET : ARQUIVO-FUNCIONARIOS; DERIVATION; O PROCESSAMENTO - NOVOS - FUNCIONARIOS ADICIONA MEMBROS A ESTE SET E O PROCESSAMENTO - FUNCIONARIOS - DEMITIDOS RETIRA MEMBROS (QUE SAO REGISTROS-DE-FUNCIONARIOS).	ESPECIFICAR AS REGRAS PARA A DERIVAÇÃO DE OCORRÊNCIA DE DADOS. EM UM SET .	ARQUIVO-FUNCIONARIOS	20 (SET)	OBJETO				COMENTARIO		
QUALQUER OBJETO	14	ELEMENT: PRECO-COMPONENTE; DESCRIPTION; ESSE ELEMENTO REFERE-SE AQUELAS COMPONENTES QUE SAO COMPRADAS PRONTAS, POR EXEMPLO, PARAFUSOS COM PORCAS;	ESPECIFICAR INFORMAÇÕES SOBRE UM OBJETO.	PRECO-COMPONENTE	4 ELEMENT	OBJETO					COMENTARIO	
16 (PROCESS)	26	PROCESS : PROCESSAMENTO-NOVOS-FUNCIONARIO PROCEDURE; 1.ADICIONE NOVO REGISTRO FUNCIONARIO 2.INCREMENTE O NUMERO DE FUNCIONARIOS DO DEPARTAMENTO CORRESPONDENTE. 3. ESPECIFIQUE A RELAÇÃO ENTRE O REGISTRO-DE-FUNCIONARIO E O DE DEPARTAMENTO. 4. INICIALIZA TODOS OS CAMPOS APROPRIADOS NO REGISTRO - DE - EMPREGADOS.	ESPECIFICAR, COM DETALHES SUFICIENTES PARA A IMPLEMENTAÇÃO, AS REGRAS PARA A EXECUÇÃO DE UM "PROCESS".	PROCESSAMENTO-NOVOS-FUNCIONARIOS	16 (PROCESS)	"PROCESS"					COMENTARIO	
5 (ENTITY)	41	ENTITY: REGISTRO-FUNCIONARIOS-HORISTAS; VOLATILITY; ESSE REGISTRO MUDA CERCA DE UMA VEZ POR SEMANA;	ESPECIFICAR A MANEIRA PELA QUAL UMA "ENTITY" MUDA, AO LONGO DO TEMPO.	REGISTRO-FUNCIONARIOS-HORISTAS	5 (ENTITY)	"ENTITY"					COMENTARIO	
20 (SET)	42	SET: ARQUIVO-FUNCIONARIOS-HORISTAS; VOLATILITY-MEMBER; OS MEMBROS DESTE "SET" SÃO MODIFICADOS CERCA DE UMA VEZ POR SEMANA;	ESPECIFICAR A MANEIRA PELA QUAL OS MEMBROS DE UM "SET" MUDAM, AO LONGO DO TEMPO.	ARQUIVO-FUNCIONARIOS-HORISTAS	20 (SET)	"SET"					COMENTARIO	

UNIVERSIDADE DE SÃO PAULO/CCE

NAME GEN

PARAMETERS FOR: NG

ATTRIBUTE ATTRIBUTE-VALUE CONDITION ELEMENT ENTITY EVENT GROUP INPUT INTERVAL KEYWORD MAILBOX
MEMO OUTPUT PROBLEM-DEFINER PROCESS INTERFACE RELATION SECURITY SET SOURCE
SUBSETTING-CRITERION SYSTEM-PARAMETER NOUNDEFINED NOSYNONYMS BASIC ORDER=ALPHA PUNCH PRINT

1	CLASSIFICACAO-FUNCIONARIOS	PROCESS
2	CODIGO-NIVEL-FUNCIONARIO	OUTPUT
3	DADOS-PESSOAIS	GROUP
4	DADOS-PROFISSIONAIS	GROUP
5	DETERMINACAO-GRAU-APTIDAO	PROCESS
6	DETERMINACAO-GRAU-POTENCIALID	PROCESS
7	FOLHA-PREENCHIDA-DADOS	INPUT
8	INFORMACOES-NIVEL	GROUP

UNIVERSIDADE DE SÃO PAULO/CCE

FORMATTED PROBLEM STATEMENT

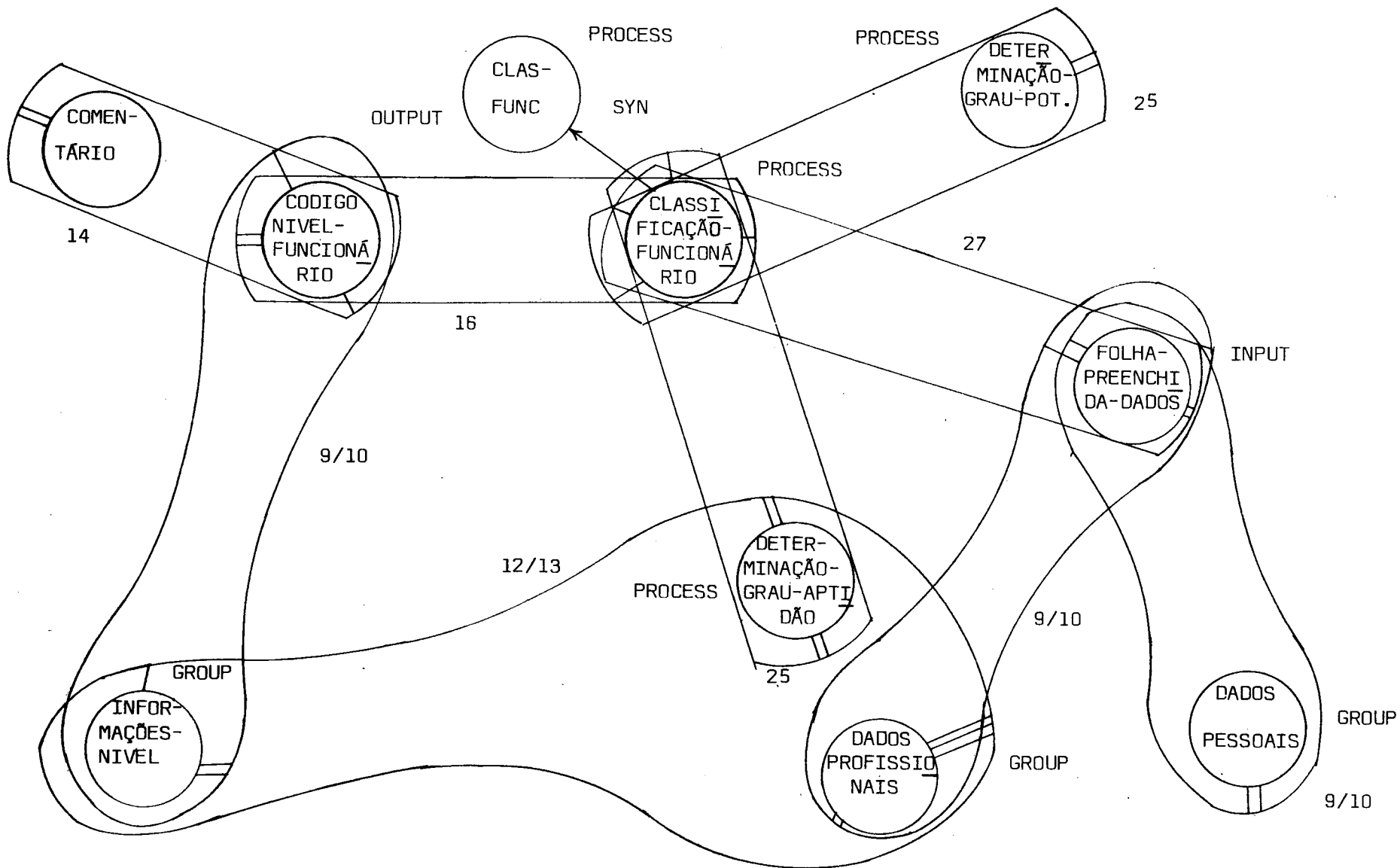
PARAMETERS FOR: FPS

FILE NOINDEX PRINT NOPUNCH SMARG=5 NMARG=20 AMARG=10 BMARG=25 RNMARG=70 CMARG=1 HMARG=40 DESG
ONE-PER-LINE DEFINE COMMENT NONEW-PAGE NONEW-LINE

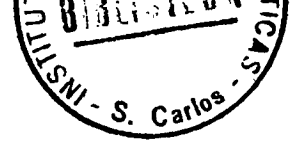
```
1 PROCESS                                CLASSIFICACAO-FUNCIONARIOS;
2     SYNONYMS ARE:  CLAS-FUNC;
3     SUBPARTS ARE:  DETERMINACAO-GRAU-APTIDAO,
4                   DETERMINACAO-GRAU-POTENCIALID;
5     RECEIVES:     FOLHA-PREENCHIDA-DADOS;
6     GENERATES:    CODIGO-NIVEL-FUNCIONARIO;
7
8 OUTPUT                                CODIGO-NIVEL-FUNCIONARIO;
9     DESCRIPTION;
10
11     COM ESSE RELATORIO SERAO CAL-
12     CULADAS AS COMISSOES SEMES -
13     TRAIS DOS FUNCIONARIOS;
14
15     CONSISTS OF:
16
17     INFORMACOES-NIVEL;
18     GENERATED BY: CLASSIFICACAO-FUNCIONARIOS;
19
20 GROUP                                DADOS-PESSOAIS;
21     CONTAINED IN: FOLHA-PREENCHIDA-DADOS;
22
23 GROUP                                DADOS-PROFISSIONAIS;
24     CONTAINED IN: FOLHA-PREENCHIDA-DADOS;
25     USED BY:
26     DETERMINACAO-GRAU-APTIDAO
27     TO DERIVE     INFORMACOES-NIVEL;
28
29 PROCESS                                DETERMINACAO-GRAU-APTIDAO;
30     PART OF:      CLASSIFICACAO-FUNCIONARIOS;
31     DERIVES:      INFORMACOES-NIVEL
32     USING:        DADOS-PROFISSIONAIS;
```

FORMATTED PROBLEM STATEMENT

```
30
31 PROCESS          DETERMINAÇÃO-GRAU-POTENCIALID;
32   PART OF:      CLASSIFICAÇÃO-FUNCIONARIOS;
33
34 INPUT          FOLHA-PREENCHIDA-DADOS;
35   CONSISTS OF:
36               DADOS-PESSOAIS,
37               DADOS-PROFISSIONAIS;
38   RECEIVED BY:  CLASSIFICAÇÃO-FUNCIONARIOS:
39
40 GROUP          INFORMACOES-NIVEL;
41   CONTAINED IN: CODIGO-NIVEL-FUNCIONARIO;
42   DERIVED BY:   DETERMINAÇÃO-GRAU-APTIDAO
43   USING:        DADOS-PROFISSIONAIS;
44
45 EOF EOF EOF EOF EOF
```

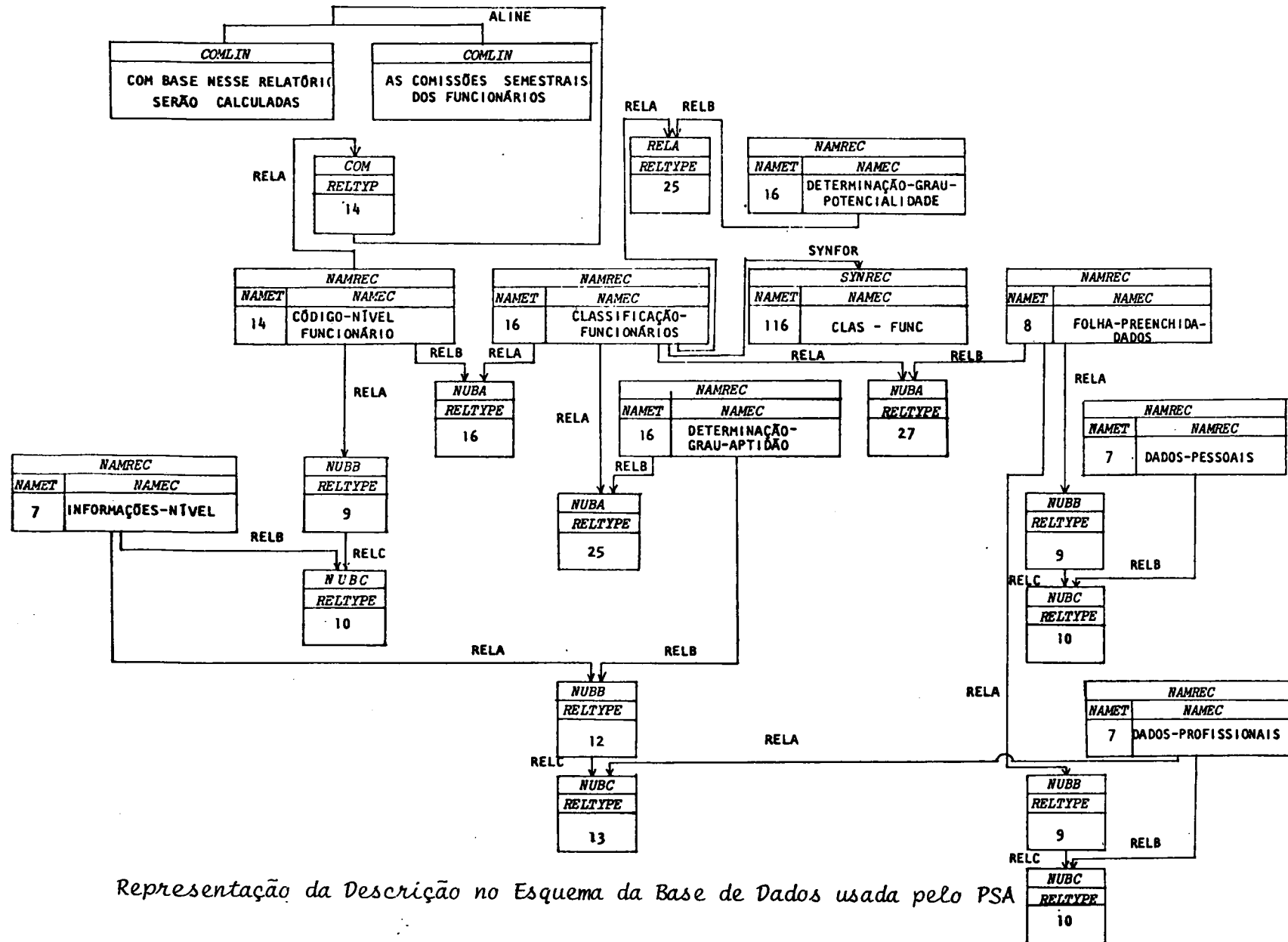


Representação por Hipergrafos da Descrição Anterior

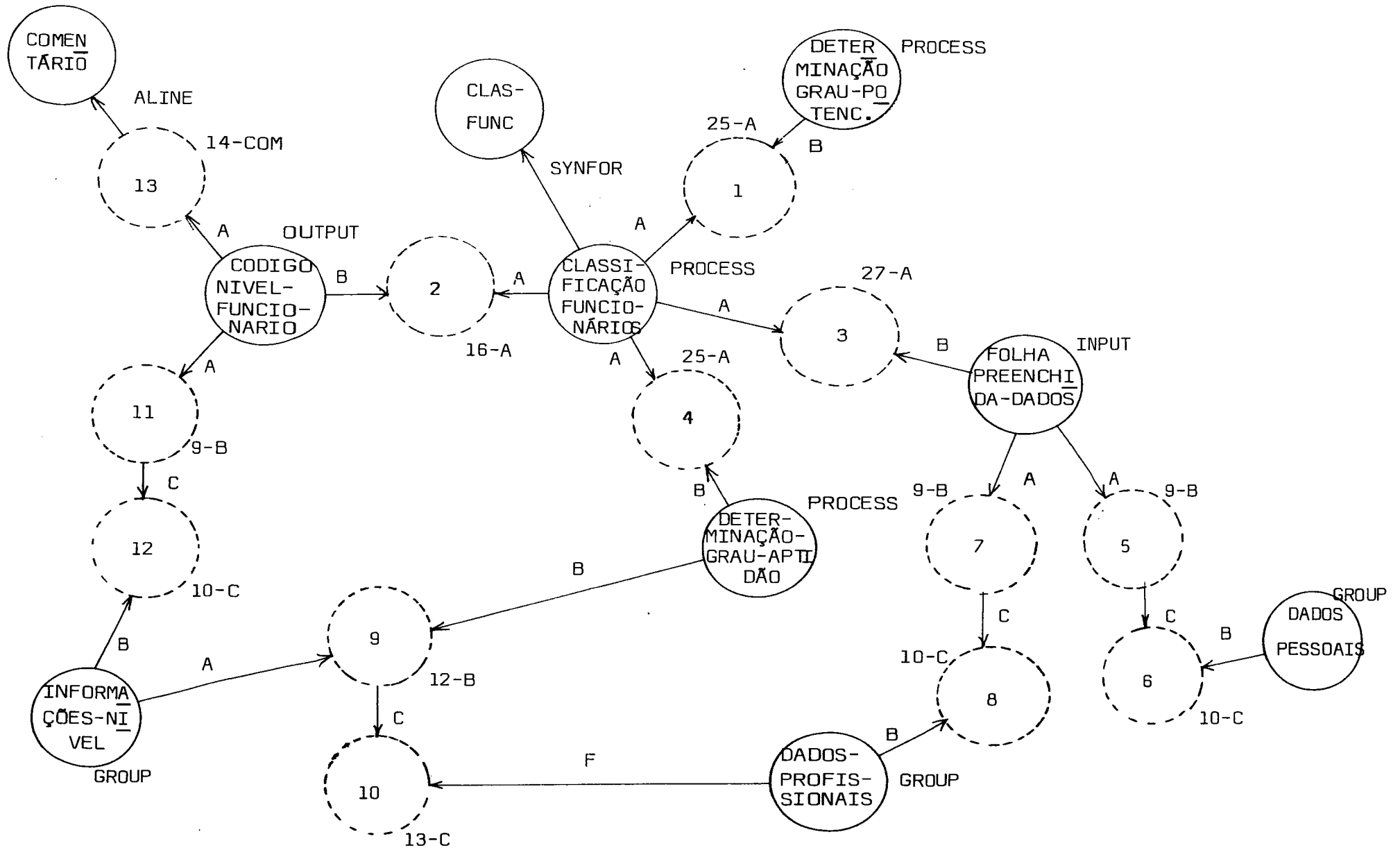


Determinação grau-potencialidade	Folha-Preenchida-Dados	Dados-Pessoais	Dados-Profissionais	Determinação-Grau-Aptidão	Classificação-Funcionários	Código-Nível-Funcionários	Class-Func.	Comentário	Informações-Nível	Categoria do arco
	1	2								9/10
	1		2							9/10
						1			2	9/10
				2	1					25
2					1					25
	2				1					27
					1	2				16
						1		2		14
			3	2					1	12/13
					1		1			-

Matriz de Incidência Correspondente à Descrição das Especificações.



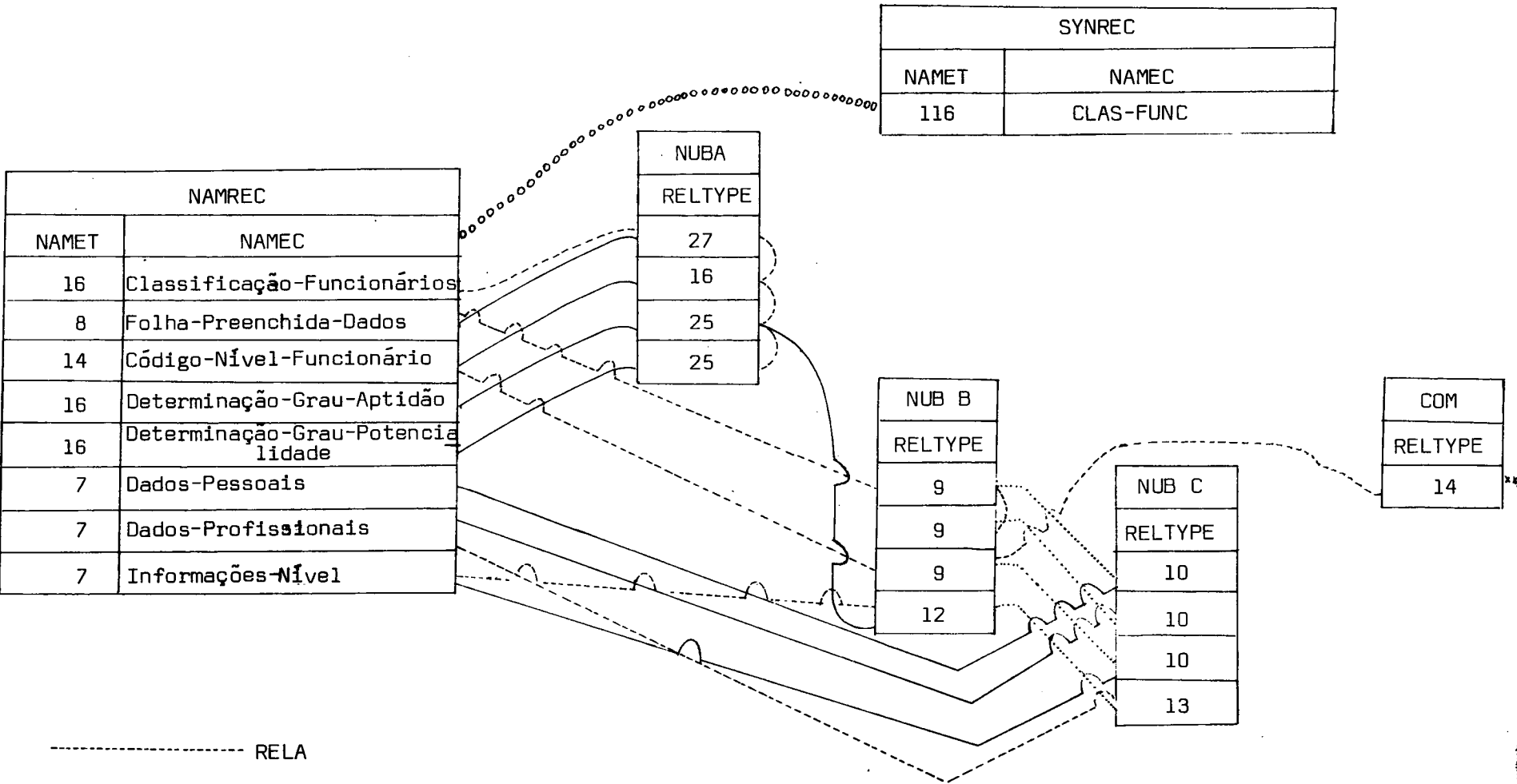
Representação da Descrição no Esquema da Base de Dados usada pelo PSA



Representação da Descrição usando Dígrafo Categorizado com Rótulos.

	código-nível-funcionário	informações-nível	comentário	determinação-grau-aptidão	classificação-funcionários	clas-func	determinação-grau-potencialidade	folha-preenchida-dados	dados-profissionais	dados-pessoais	1	2	3	4	5	6	7	8	9	10	11	12	13
código-nível-funcionamento												B									A		A
Informações-nível																			A			B	
comentários																							
determinação grau-aptidão														B					B				
classificação funcionários											A	A	A	A									
clas-func																							
determinação grau-potencialidade							B																
folha-preenchida-dados								B					A				A						
dados-profissionais																		B		A			
dados-pessoais														B									
1																							
2																							
3																							
4																							
5																C							
6																							
7																		C					
8																							
9																					C		
10																							
11																						C	
12																							
13		AL																					

Matriz de adjacência do Dígrafo Categorizado com Rótulos, da Figura Anterior.



- RELA
- RELB
- RELC
-○..... SYNFOR
- ***** ALINE

COMLIN
Com esse relatório serão
calculadas as comissões
semestrais dos funcionários.

Representação na Descrição da Base de Dados do Sistema PSL/PSA, utilizando Listas, como evidenciado em [1]

3.4. Conclusões

Neste capítulo, explicitamos o modelo matemático da linguagem PSL plena, tal como implementado na Universidade de Michigan. Mostramos também, durante seu desenvolvimento, a maneira de representar o modelo explicitado (hipergrafo), através de dígrafos, utilizando, para tanto, o esquema da Base de Dados, usada pela PSA.

Temos certeza de que a explicitação, enfocando principalmente os papéis dos nós, como os introduzimos, irá favorecer, sobremaneira, o entendimento do Sistema, bem como proporcionar sólidas bases para um desenvolvimento futuro.

A explicitação do modelo matemático, para a linguagem PSL plena, terá grande valia, quando da implementação de testes que verificam a consistência, na descrição das Especificações de um Sistema de Informação, como foi realizado por Germano [1], para um subconjunto da Linguagem PSL.

CAPÍTULO IV

ALGUNS MODELOS MATEMÁTICOS DE SISTEMAS DE INFORMAÇÃO, RELACIONADOS AOS DO PSL/PSA

4.1. Considerações Gerais

Iniciamos este capítulo, apresentando um trabalho de grande relevância, realizado por Durchholz [2], sobre a representação de relações por matrizes e por funções. Esse autor prendeu-se ao estudo de Sistemas de Base de Dados, preocupando-se em evidenciar que a visão relacional de uma Base de Dados, não difere, em essência, da funcional. Enfoca, de forma objetiva, quatro das principais abordagens de modelos de Sistemas de Informação: por pontos, tabular (matricial), relacional e funcional, e mostra que, na essência, têm o mesmo significado, uma vez que cada uma delas, pode ser "traduzida" para qualquer das três restantes.

Já sabemos que para a linguagem PSL, aplicam-se tanto o enfoque relacional (hipergrafo), quanto o funcional (Base de Dados usada pelo PSA). Germano [21], mostra que na Álgebra da Informação da CODASYL, o modelo matemático tem um enfoque por pontos; resta-nos, portanto, a apresentação de um modelo matricial, de forma que os quatro enfoques, apresentados por Durchholz sejam enfatizados. Apresentaremos, pois, a seguir, o modelo matemático de Lieberman, [22] que é de extrema importância, uma vez que foi um dos pioneiros na área de Modelos Matemáticos de Sistemas de Informação. Como veremos, o conceito de precedência entre conjuntos de informação enfatizado por Langefors [23], já estava presente, no trabalho de Lieberman, embora de maneira informal e implícita. O modelo proposto por Lieberman é baseado em matrizes e em operações sobre elas realizadas.

Concluimos o capítulo, sugerindo uma interessante linha de pesquisa

futura, que teria como objetivo final, a integração dos trabalhos de Durchholz, Teichroem, Lieberman e Bosak (este último, estendido, posteriormente pelo grupo CODASYL).

4.2. Representação de Relações por Matrizes e por Funções

Durcholz publicou um trabalho [2], sobre a representação de relações por matrizes e por funções, onde mostra como uma representação matricial pode ser transformada numa representação funcional e vice-versa. Faz uso da representação da relação como um conjunto de pontos em um espaço, buscando com isso, maior facilidade quando da representação matricial e funcional da relação.

Seja, por exemplo, a relação "fornecedor (S)/peças fornecidas (P)", e a sua representação como conjunto de pontos, mostrada na Figura 4.1.

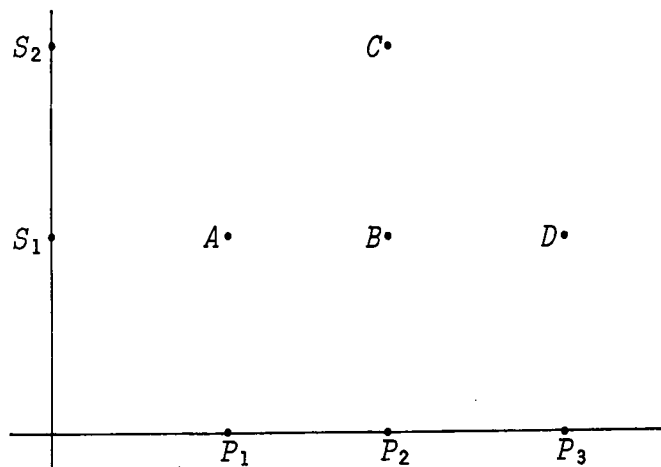


Figura 4.1

O conjunto de pontos mostra que o fornecedor S_1 fornece todas as três peças e o fornecedor S_2 somente a peça P_2 . Consideremos agora a relação ternária: "fornecedor Sfornece as peças Ppara o projeto S". A Figura 4.2 dá uma ilustração para essa relação.

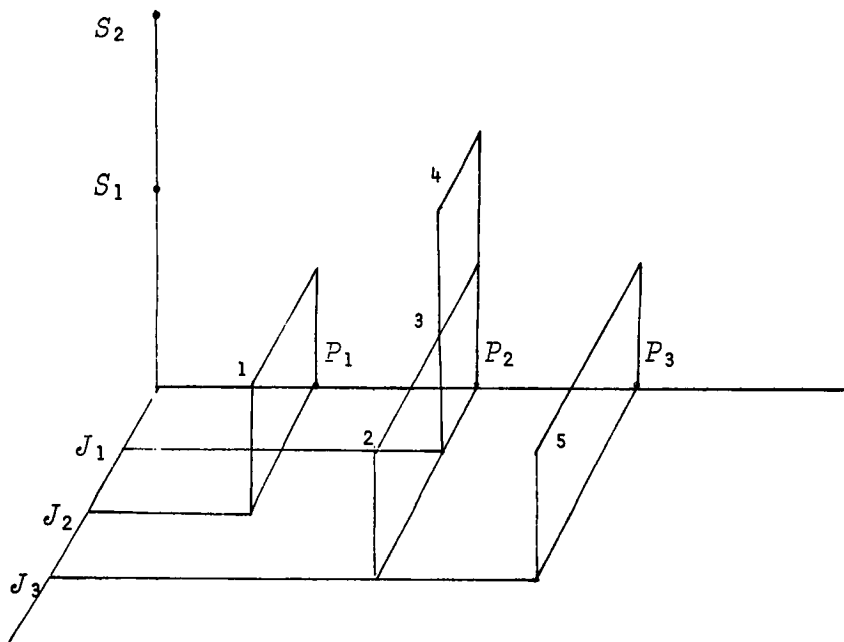


Figura 4.2

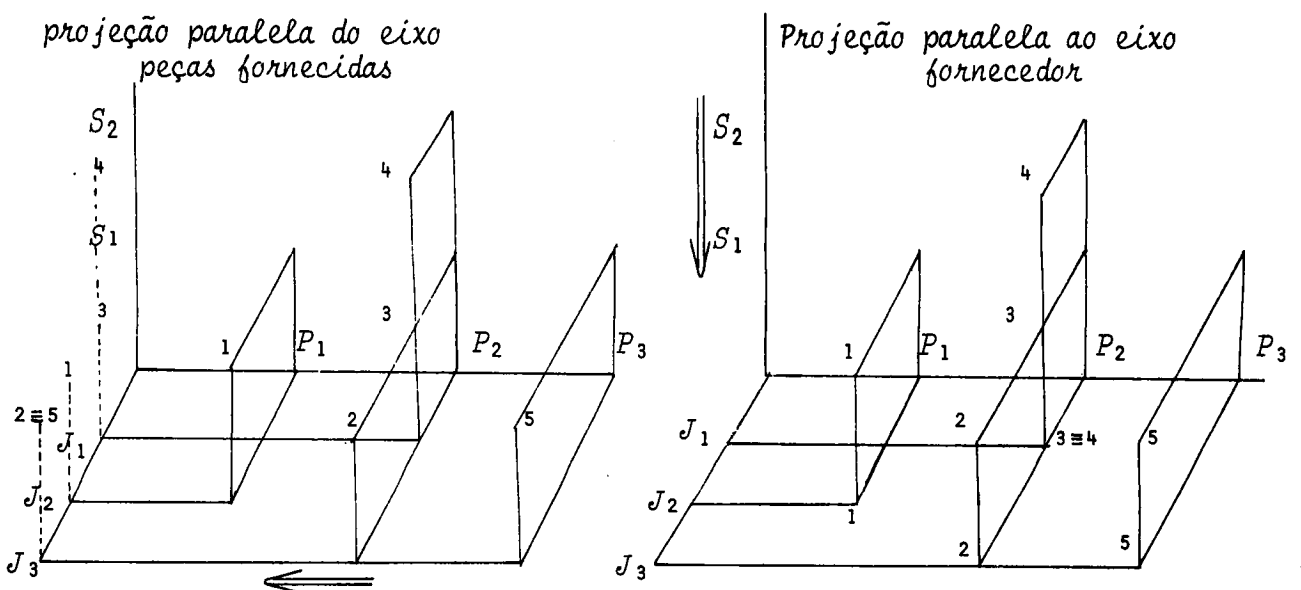
Na Figura 4.2, podemos notar, por exemplo, que o projeto J_1 usa somente a peça P_2 , que é fornecida por S_1 e S_2 , e que os projetos J_2 e J_3 recebem apenas peças do fornecedor S_1 , sendo que a peça recebida para o projeto J_2 (peça P_1) é distinta das recebidas para o projeto J_3 (peças P_2 e P_3).

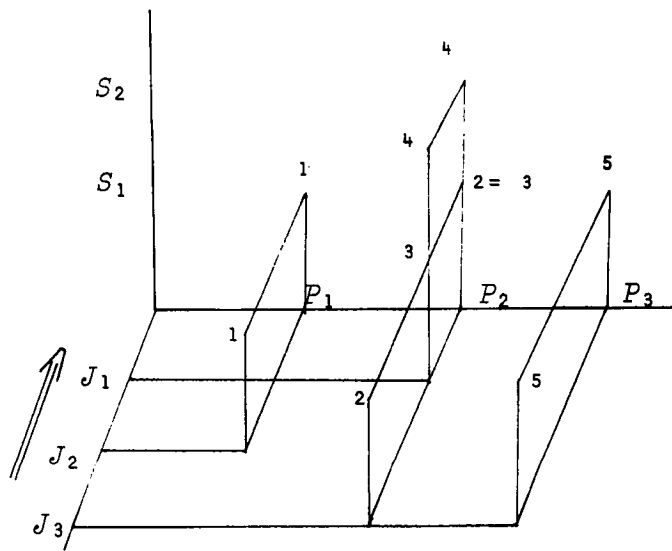
O espaço, onde são alocados os pontos que representam uma relação, é o espaço euclidiano n-dimensional. Os eixos do espaço representam "atributos". Em cada eixo, "valores de atributo" são atribuídos aos pontos. Voltando ao exemplo da Figura 4.1, teríamos como atributos: fornecedor e peças fornecidas e, como valores de atributos: S_1, S_2, P_1, P_2, P_3 .

Diz-se que um ponto terá um "valor de atributo", com relação a um atributo, se ele tiver as mesmas coordenadas que o valor do atributo, relativamente a este atributo, ou, em outras palavras, se estiver em um hiperplano dado pela condição "coordenada relativa ao atributo = valor do atributo".

Se um conjunto de pontos pode ser projetado, paralelamente, a um eixo, tal que dois pontos nunca coincidam, este eixo (atributo), é chamado de funcionalmente dependente, com relação ao conjunto de pontos. Neste caso, a dimensão do espaço de atributos poderá ser reduzida, se admitirmos a possibilidade de se associar aos pontos, valores de atributos.

A Figura 4.2 não pode ser usada como exemplo, porque nenhum dos três atributos nela representados, é funcionalmente dependente, em relação ao conjunto de pontos. Ao fazer a projeção paralelamente a qualquer dos eixos, evidenciamos que sempre haverá coincidência de dois pontos. De fato:





Projeção paralela ao eixo projeto

Se retirarmos o ponto $3 = (S_1, P_2, J_1)$, poderemos notar que o atributo "projeto" se torna funcionalmente dependente em relação ao conjunto de pontos, uma vez que, ao projetarmos o conjunto de pontos paralelamente a esse eixo (atributo), não acontecerá superposição de pontos.

A Figura 4.3 mostra o resultado dessa projeção, com a associação dos correspondentes valores de atributos, aos pontos.

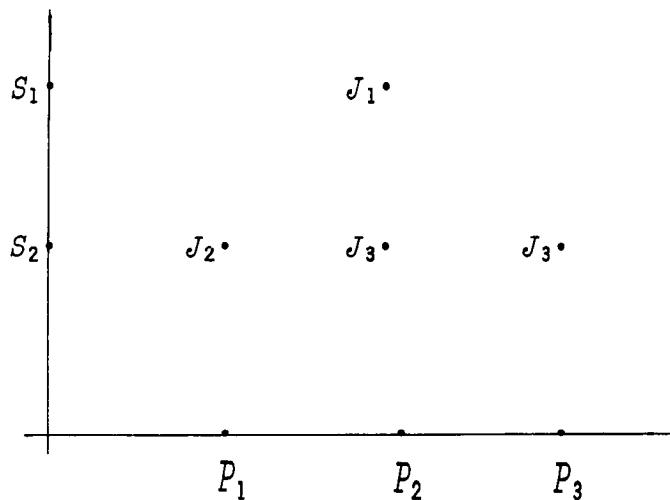


Figura 4.3

Essa redução de dimensão pode ser repetida (mas não neste exemplo, uma vez que nenhum dos atributos restantes é funcionalmente dependente).

Valores de atributos de "atributos projetados", serão, então, anexados àqueles já associados aos pontos. Todos os valores associados a um ponto,

devem ter uma ordem, de tal forma a permitir sua correspondência com os atributos, dos quais são tomados; ou seja, à medida que projeções vão sendo feitas, vamos associando valores aos pontos, ordenadamente. Com isso, conseguimos associar a cada ponto, um vetor, que tem relacionado a cada sua coordenada, um "valor de atributo", do que foi projetado. Dessa forma, podemos construir uma matriz de valores de atributos, onde, cada linha corresponderá ao vetor de um ponto, sendo que todos eles terão o mesmo número de elementos.

Se chegarmos a um determinado estágio em que nenhuma projeção posterior será mais possível, ou seja, se fosse feita, pontos coincidiriam (não existe mais dependência funcional), então, os atributos restantes são ainda suficientes para identificar cada ponto do conjunto, mas qualquer subconjunto do conjunto de atributos, não identificará todos os pontos. Esse conjunto minimal de atributos que identifica qualquer dos pontos, é conhecido como "espaço de coordenadas discriminatório" em [24].

Notemos que existe a possibilidade de se ter vários de tais "conjuntos minimais" de atributos, para uma relação e que eles podem ter um número diferente de atributos. Isso é consequência da perda da dependência funcional, quando projeções são feitas relativamente a atributos diferentes. Como exemplo dessa situação, consideremos as próximas quatro Figuras [4.4(a,b,c,d)].

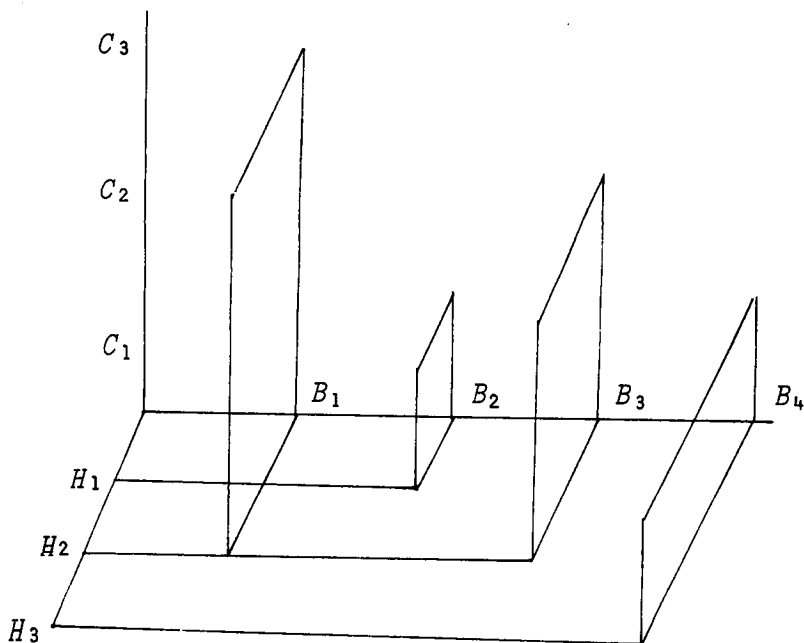


Figura 4.4.a

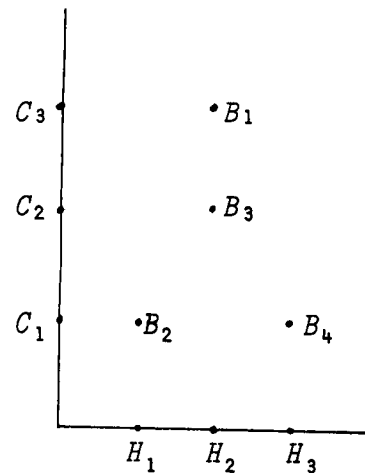


Figura 4.4.b

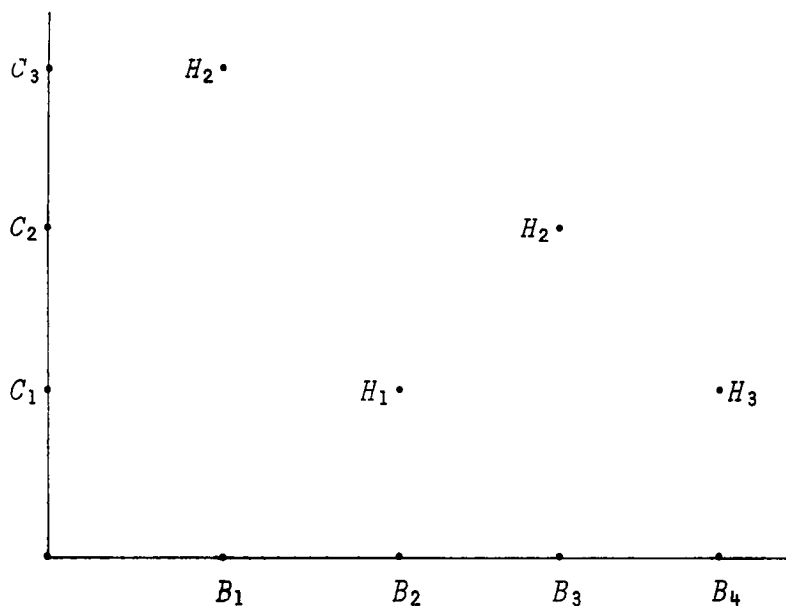


Figura 4.4.c

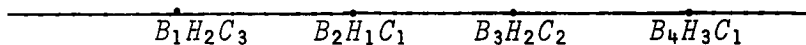


Figura 4.4.d

A Figura 4.4.b é obtida a partir da 4.4.a projetando-se o conjunto de pontos paralelamente ao eixo B . Na Figura 4.4.b, nenhum dos atributos é funcionalmente dependente, logo, não se pode mais projetar o conjunto de pontos paralelamente a qualquer dos eixos, sem que haja coincidência de pontos. Chegamos, pois, ao conjunto minimal de atributos $\{H, C\}$.

Se, por outro lado, iniciarmos, a partir da Figura 4.4.a, a projeção dos pontos, paralelamente ao eixo H , obteremos, depois da projeção, a Figura 4.4.c. Ainda nesta Figura, poderemos notar que o atributo C é funcionalmente dependente, ou seja, se projetarmos o conjunto de pontos, paralelamente a esse eixo, ponto algum irá sobrepor-se a outro. Obteremos, então, a Figura 4.4.d. Iniciando, pois, a projeção do conjunto de pontos, paralelamente ao eixo H , chegamos ao conjunto minimal $\{B\}$, que difere do obtido, quando a primeira projeção é feita, é a paralela ao eixo B .

Passemos agora ao enfoque matricial, que considera uma relação sendo

representada por uma matriz, onde a ordem das linhas é indiferente. Cada linha da matriz evidencia uma situação, a que a relação se aplica. Exemplificando: a relação expressa como um conjunto de pontos, na Figura 4.2, tem, como representação matricial, a mostrada na Figura 4.5.

$$\begin{pmatrix} S_1 & P_1 & J_2 \\ S_1 & P_2 & J_1 \\ S_1 & P_2 & J_3 \\ S_2 & P_2 & J_1 \\ S_1 & P_3 & J_3 \end{pmatrix}$$

Figura 4.5

A construção da matriz, a partir do conjunto de pontos, é trivial. Os atributos de um conjunto de pontos, são ordenados de forma arbitrária e os valores dos atributos de um ponto, na ordem fixada, correspondem a uma linha da matriz.

Se considerarmos um conjunto de pontos com dimensão reduzida, o vetor de valores de um ponto é simplesmente anexado à sequência de atributos. A sequência composta corresponde, então, a uma linha da matriz.

Na abordagem matricial, uma projeção paralela a um atributo é traduzida delindo-se a coluna a que aquele atributo corresponde e, por subsequente remoção de todas as múltiplas ocorrências de linhas. Quando uma coluna é removida, pode ocorrer o aparecimento de linhas repetidas - isso acontece se as informações existentes, na coluna removida, forem necessárias para distinguir linhas. Neste caso, o conjunto de pontos projetado terá menos pontos que o original e a matriz resultante (com a eliminação das ocorrências repetidas de linhas) terá menos linhas.

Uma vez que a preocupação é relativa à Base de Dados, serão inversas de funções que terão relevância na sequência do desenvolvimento do trabalho. Como consequência disso, na representação gráfica apresentada, as flexas apontam

para argumentos e não para os valores da função.

A Figura 4.6 mostra uma função que associa cidades a países.

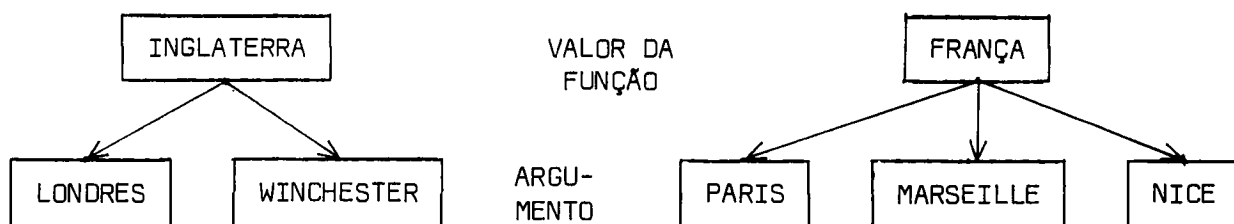


Figura 4.6

Para nossos propósitos, no entanto, outra representação gráfica, muito usada na discussão de disposições, é mais conveniente. Ao invés da introdução de um conjunto de flexas, para unir valor de função a argumentos, eles se se rão unidos numa sequência de flexas ou em uma rede de flexas (por razões de cl reza gráfica). Como ao valor da função se segue uma sequência de argumentos, as pontas das flexas são omitidas. Levando-se em conta essas considerações, a função da Figura 4.6 pode ser representada pela Figura 4.7.

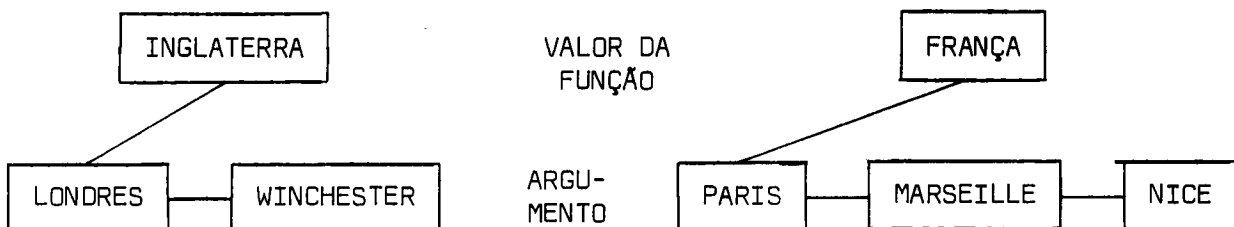


Figura 4.7.

Devemos acrescentar, que esta espécie de representação tem duas defi ciências. Uma é que ela sugere uma ordenação dos argumentos, o que não é deseja do. Outra é que a ligação entre o valor da função e o argumento ao qual ele está diretamente ligado, parece ter a mesma natureza que as ligações entre ar gumentos, o que não é verdade.

A Figura 4.8 ilustra o método usado para representar uma relação, por funções. Consideremos, para essa ilustração a relação expressa na Figura 4.8.

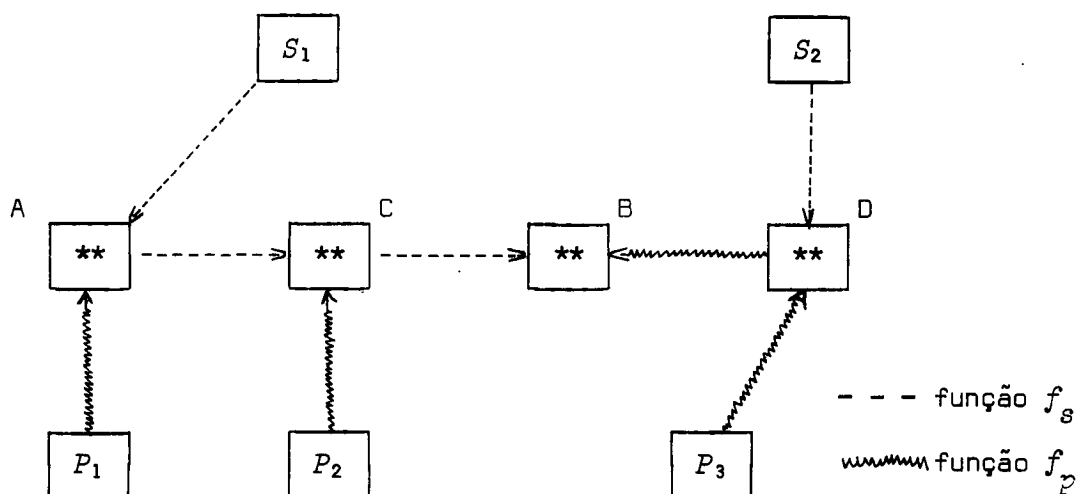


Figura 4.8

Essa representação é interpretada da seguinte forma: para se obter as peças do fornecedor S_1 , primeiro devemos encontrar todos os registros $**$, que são ligados a S_1 por $\text{-----}\rightarrow$. A cada um dos registros $**$ assim relacionado, devemos encontrar o registro da peça, o qual é ligado por $\text{~~~~~}\rightarrow$. A mesma regra aplica-se a S_2 . Os fornecedores das peças são obtidos de maneira análoga.

Em geral, para construir-se uma representação funcional a partir de um conjunto de pontos, procede-se de acordo com as seguintes regras:

- R_1 : crie um registro ("registro de ligação"), para cada ponto do conjunto de pontos. A única exigência quanto a esses registros é que eles devem ser distintos.
- R_2 : Para cada valor de atributo, crie um registro ("registro de valor") que contém o valor do atributo.
- R_3 : Para cada atributo, crie uma função que associe cada registro de ligação com um registro de valor, criado para esse atributo. A função é tal que o valor correspondente a um registro de ligação, é o do atributo correspondente ao registro de valor, que está associado ao registro de ligação.

Para o exemplo da Figura 4.1, depois da aplicação das regras R_1 e R_2 ,

obtemos a Figura 4.8. Quando da aplicação da R_3 , obtemos as funções:

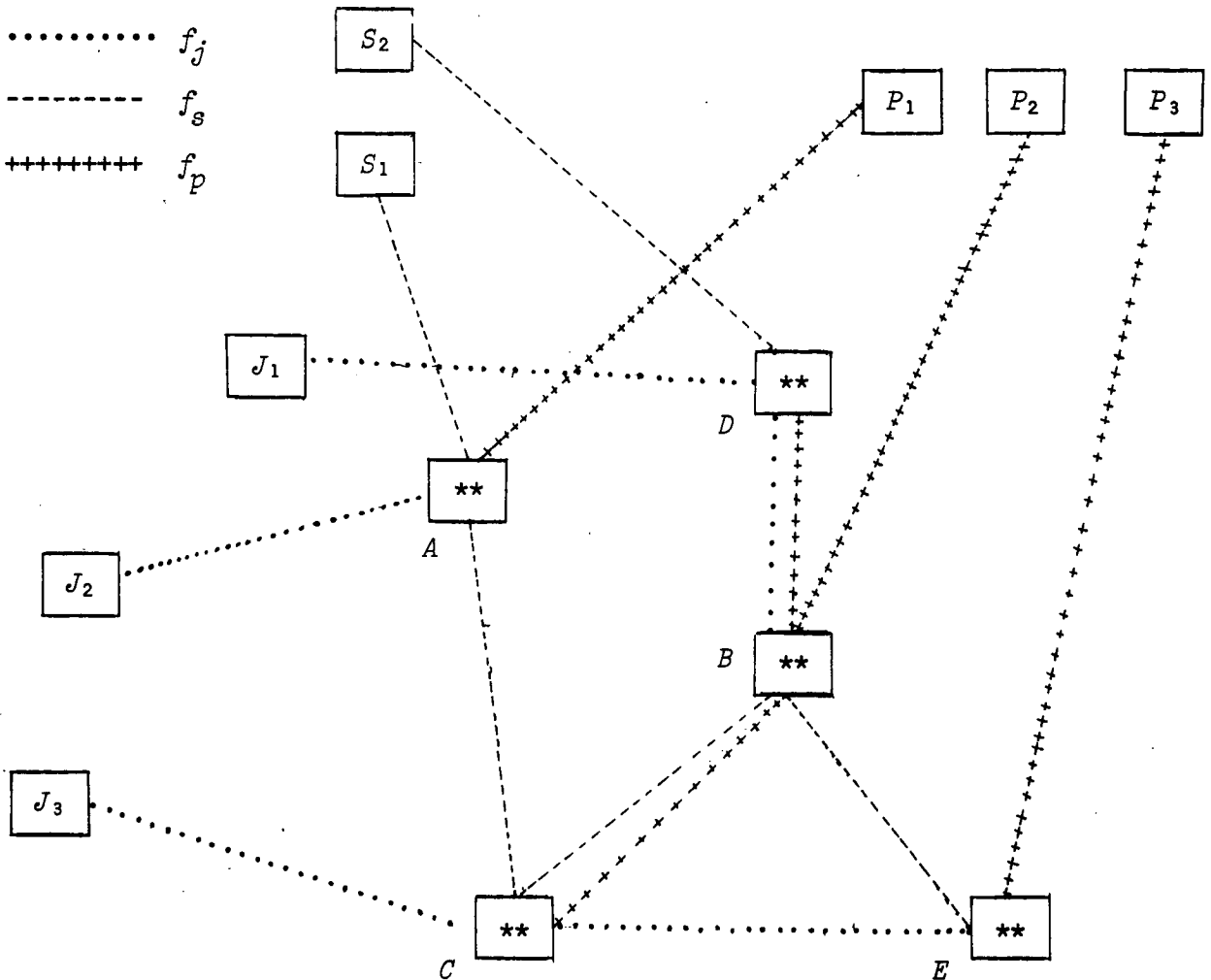
$$f_s = \begin{pmatrix} A & B & C & D \\ S_1 & S_1 & S_2 & S_1 \end{pmatrix}$$

$$f_p = \begin{pmatrix} A & B & C & D \\ P_1 & P_2 & P_2 & P_3 \end{pmatrix}$$

Passando, agora, ao exemplo da Figura 4.2, a relação ternária, lá explicitada, é representável pela matriz:

$$\begin{pmatrix} S_1 & P_1 & J_2 \\ S_1 & P_2 & J_1 \\ S_1 & P_2 & J_3 \\ S_2 & P_2 & J_1 \\ S_1 & P_3 & J_3 \end{pmatrix}$$

Aplicando as regras R_1 , R_2 , e R_3 , obtemos:



Evidenciamos, pois, as seguintes funções:

$$S \rightarrow f_s = \begin{pmatrix} A & B & C & D & E \\ S_1 & S_1 & S_1 & S_2 & S_1 \end{pmatrix}$$

$$J \rightarrow f_j = \begin{pmatrix} A & B & C & D & E \\ J_2 & J_1 & J_3 & J_1 & J_3 \end{pmatrix}$$

$$P \rightarrow f_p = \begin{pmatrix} A & B & C & D & E \\ P_1 & P_2 & P_2 & P_2 & P_3 \end{pmatrix}$$

4.3. Sobre o Modelo de Lieberman

Uma forma de pensar em Sistemas de Informação, é imaginá-los como uma rede de comunicações. Informações que estão disponíveis em um ponto, são necessárias a outro e devem, portanto, ser passíveis de transmissão de um a outro ponto do sistema.

Dessa maneira, um operário, por exemplo, pode dar origem a uma informação (número de horas extras semanais), que é transmitida ao departamento de contabilidade, pelo chefe de seção. Desse departamento, por sua vez, a informação é disseminada a vários outros pontos da organização, onde se faz necessária. A organização pode, assim, ser vista em termos de rede de comunicações ou fluxo de informações.

O que Lieberman propõe é uma técnica matemática para a investigação do fluxo de informação, num Sistema de Informação, técnica essa, fundamentada na noção matemática de matriz.

Essa técnica permite a construção de matrizes que evidenciam detalhes do fluxo de dados. Através da manipulação e operações dessas matrizes, podemos analisar, pelas matrizes resultantes, a disponibilidade dos dados, sua utilidade e redundância, bem como sua não utilização pelas funções comerciais.

O modelo de Lieberman tem como componentes:

- (1) - *Função Comercial (Business Function)*: é um conjunto de atividades administrativas que deve ser realizado por um grupo, de acordo com o tipo de tarefa que realiza. Como exemplo de funções comerciais, podemos citar: compra, contabilidade, controle de estoque, orçamento, etc....
- (2) - *Classe de Informação (Class of Information)*: consiste num ou mais elementos de informação que têm todas as características em comum. Uma classe de informação pode ter qualquer número de membros. Deve-se ressaltar a diferença entre classe de informação e sua medida quantitativa. Por exemplo: nome, data, quantidade em estoque, horas extras, etc., são todas di

ferentes classes de informação; já, por exemplo, o número de horas-extras trabalhadas, é medida quantitativa daquela particular classe de informação (horas-extras-trabalhadas).

Além disso, as classes de informação podem ser de tipo identificador ou quantitativo.

- (1) - *Classes de Informação de Tipo Identificador (i)*: essas classes descrevem ou identificam um documento. Por exemplo: data, número da ordem de compra, nome do funcionário, endereço, etc...
- (2) - *Classes de Informação de Tipo Quantitativo (q)*: tais classes de informação fornecem a medida quantitativa da informação. Por exemplo: número de horas trabalhadas, número de peças fabricadas, etc...

A informação sempre entra, no sistema, sob a forma de documento ou formulário. Lieberman os distingue, dizendo que os documentos são instâncias dos formulários. Poderíamos ter, então, o formulário-cartão de ponto-e os documentos (instâncias do formulário): cartão de ponto do José e cartão de ponto da Maria, documentos esses, diferentes.

Além disso os formulários são classificados em:

- (1) - *Formulários de Dados (Source Data Forms)*: são aqueles nos quais a informação é registrada pela primeira vez e não são derivados de nenhum outro existente.
- (2) - *Formulários de Relatórios (Report Forms)*: são aqueles que são feitos a partir de operação ou combinação de operações, realizadas nos Formulários de Dados.

Com esses conceitos, Lieberman mostra como podem ser construídas várias matrizes, que exprimem como se pode obter Relatórios, a partir de outros intermediários e como os Relatórios Finais são usados pelas Funções Comerciais.

A primeira matriz construída, chamada M_0 [25], mostra, a cada Formulário de Dados, quais são as classes de informações usadas.

O que é de interesse, é saber se uma classe de informação é ou não usada, em um Formulário de Dados. Isso se evidencia, na primeira matriz construída.

Para tanto, basta examinar a célula correspondente à intersecção da linha, associada àquela classe de informação, com a coluna associada ao referido Formulário de Dados. Ela conterá 1 se a informação foi usada no Formulário e zero, caso contrário.

Se existirem n classes de informação e m formulários. então a matriz M_0 terá n linhas e m colunas.

Homer [25], introduziu uma notação para o modelo de Lieberman, que adotamos.

Seja:

d_i : o i -ésimo item de dados (classe de informação de tipo identificador ou de tipo quantitativo) no sistema, $i = 1, \dots, m$.

$R_{k(j)}$: o k -ésimo Formulário de Relatório do j -ésimo nível, onde:
 $j = 1, \dots, n$ e $k = 1, \dots, p_j$.

Em particular, seja:

$R_{k(1)}$: o k -ésimo Formulário de Dados.

B_r : a r -ésima função comercial, $r = 1, \dots, q$.

M_{j-1} : a matriz que representa os componentes necessários à preparação dos Formulários de Relatório, no j -ésimo nível.

Em particular seja:

M_0 : a matriz que representa quais itens de dados aparecem, em cada Formulário de Dados.

M_n : matriz que representa quais Formulários de Relatório, no n -ésimo nível, são usados no desempenho das várias funções comerciais (isso implica que as funções comerciais são executadas no nível $n + 1$).

As matrizes M_{j-1} e M_n são formadas da seguinte maneira:

- (1) - Construa uma coluna para cada $R_{k(j)}$ (ou para cada B_r , no caso de M_n).
- (2) - Construa uma linha para cada $R_{k(j-1)}$ (ou para cada d_i , no caso de M_0).
- (3) - Faça o elemento da matriz igual a 1, se o correspondente $R_{k(j-1)}$ ou (d_i) for usado para produzir o correspondente $R_{k(j)}$ (ou B_r). Faça todos os outros elementos iguais a zero.

Consideremos, a título de exemplo, uma organização que tem por funções comerciais:

- (1) - Administração do Pessoal.
- (2) - Controle da Produção.

Seus Formulários de Dados são:

CARTÃO DE PONTO ($R_{1(1)}$)	BOLETIM DE SERVIÇO ($R_{2(1)}$)
<i>Data</i>	<i>Data</i>
<i>Departamento</i>	<i>Departamento</i>
<i>Seção</i>	<i>Seção</i>
<i>Nome-Funcionário</i>	<i>Nome-Funcionário</i>
<i>Número-Funcionário</i>	<i>Número-Funcionário</i>
<i>Turno</i>	<i>Turno</i>
<i>Horas-Normais</i>	<i>Número-da-Máquina</i>
<i>Horas-Extras</i>	<i>Número-Peças-De feitas</i>
<i>Salário-Hora</i>	<i>Número-Peças-Perfeitas</i>
<i>Código-Trabalho</i>	

A matriz M_0 , que representa que classes de informação ou item de dados aparece, em cada Formulário de Dados, é:

		$R_1(1)$	$R_2(1)$
d_1	Data	1	1
d_2	Departamento	1	1
d_3	Seção	1	1
d_4	Nome-Funcionário	1	1
d_5	Número-Funcionário	1	1
d_6	Turno	1	1
d_7	Horas-Normais	1	0
d_8	Horas-Extras	1	0
d_9	Salário-Hora	1	0
d_{10}	Código-Trabalho	1	0
d_{11}	Número-da-Máquina	0	1
d_{12}	Número-Peças-Perfeitas	0	1
d_{13}	Número-Peças-Defeituosas	0	1

A partir das informações contidas nos Formulários de Dados, são construídos os seguintes Formulários de Relatório:

FOLHA SALÁRIO INTEGRAL ($R_{1(2)}$)

Data

Departamento

Seção

Número-Funcionário

Turno

Salário-Integral-Diário

FOLHA DESCONTOS DIÁRIOS ($R_{2(2)}$)

Data

Departamento

Seção

Nome-Funcionário

Total-a-Descontar

Número-Peças-Defeituosas

Número-Peças-Perfeitas

A matriz M_1 , a seguir, mostra quais Formulários de Dados são usados

para construir cada um dos Formulários, no segundo nível.

$$M_1 = \begin{array}{c|cc} & R_1(2) & R_2(2) \\ \hline R_1(1) & 1 & 1 \\ \hline R_2(1) & 0 & 1 \\ \hline \end{array}$$

A partir das informações contidas nos Formulários de Relatórios (segundo nível), são construídos outros Formulários de Relatórios (terceiro nível), que são:

FOLHA SALÁRIO DESCONTADO ($R_1(3)$)

RELATÓRIO PEÇAS DEFEITUOSAS POR SEÇÃO
($R_2(3)$)

Data

Data

Departamento

Departamento

Seção

Seção

Turno

Número-Peças-Defeituosas

Nome-Funcionário

Número-Funcionário

RELATÓRIO PEÇAS PERFEITAS POR SEÇÃO

Salário-Descontado

($R_3(3)$)

Data

Departamento

Seção

Número-Peças-Perfeitas

A matriz M_2 mostra a seguir, quais os Formulários de Relatório, em segundo nível, são usados para construir cada um dos Formulários de Relatório, em terceiro nível.

$$M_2 = \begin{array}{c|ccc} & R_1(3) & R_2(3) & R_3(3) \\ \hline R_1(2) & 1 & 0 & 0 \\ \hline R_2(2) & 1 & 1 & 1 \\ \hline \end{array}$$

A próxima matriz M_3 , mostrará quais Formulários de Relatório, em terceiro nível, são usados para o desempenho de cada uma das duas funções comerciais (B_1 e B_2).

	B_1	B_2
$R_1(3)$	1	0
$R_2(3)$	0	1
$R_3(3)$	0	1

Tais matrizes, naturalmente, podem ser manipuladas. Neste modelo, os elementos que encabeçam cada linha de qualquer uma das matrizes, por exemplo, a M_j , são idênticos aos que encabeçam as colunas, na matriz precedente M_{j-1} . Dessa forma, podemos fazer multiplicações sucessivas de matrizes. Seja:

$$M_{a,b} = M_a M_{a+1} M_{a+2} \dots M_{b-1} M_b; 0 \leq a < b \leq j$$

Os elementos, que encabeçam a linha em $M_{a,b}$, são os mesmos que encabeçam as linhas em M_a , digamos, $R_{k(a)}$, e os elementos que encabeçam as colunas em $M_{a,b}$ são os mesmos que encabeçam as colunas em M_b , $R_{k(b)}$. Particularmente, a matriz $M_{0,n}$, representa o número de vezes que cada d_i está disponível a cada B_r . Cada elemento de $M_{a,b}$ pode ser interpretado como o número de caminhos através dos quais, cada $R_{k(a)}$ atinge o correspondente $R_{k(b)}$.

O produto das quatro matrizes, do exemplo anterior é:

	B_1	B_2
d_1	3	4
d_2	3	4
d_3	3	4
d_4	3	4
d_5	3	4
d_6	3	4

d_7	2	2
d_8	2	2
d_9	2	2
d_{10}	2	2
d_{11}	1	2
d_{12}	1	2
d_{13}	1	2

Isso indica, por exemplo, que o item de dado d_{13} , está disponível à função comercial B_1 , através de um único caminho: $(R_{2(1)} - R_{2(2)} - R_{1(3)} - B_1)$, enquanto que está disponível à B_2 através de dois caminhos: $(R_{2(1)} - R_{2(2)} - R_{2(3)} - B_2)$ e $(R_{2(1)} - R_{2(2)} - R_{3(3)} - B_2)$.

Como Lieberman sugere, podemos estabelecer um modelo, digamos, $A_{o,n}$, que evidência as exigências de dados ideal, a cada função comercial.

Então, a matriz:

$$C_{o,n} = M_{o,n} - A_{o,n}$$

dá indicação do excesso de incidência de dados, ou da falta dos mesmos, no sistema.

O modelo proposto por Lieberman [26], considera que: o nível de cada relatório é único, que todas as funções comerciais são realizadas em apenas um nível e em nenhum outro, que os dados entram no sistema no nível $j = 1$ e que nenhum relatório final é produzido em outro nível, que não o $j = n$.

Modificações no modelo, visando permitir que informações (itens de dados) possam entrar no sistema, em níveis que não o primeiro, e que relatórios finais possam ser produzidos, antes do último nível, dão origem a uma série de matrizes incompatíveis para a multiplicação.

Homer [25], propõe um algoritmo para superar essa dificuldade, mas omite tanto a definição de "produto" incompatível de matrizes, quando a demonstração matemática da validade do algoritmo.

Como já vimos, nesse modelo estão implícitas as noções de caminho em dígrafos, bem como noções de precedência de conjuntos de informação, apresentadas, posteriormente, por Langefors [23]. Segundo este autor, dentro de determinada organização, existem diferentes funções tendo por finalidade o cumprimento de determinados objetivos. Na maioria dos casos, pode-se definir tais funções a partir dos objetivos básicos da organização, ou seja, conhecidos os objetivos da organização, definem-se funções que, mediante determinadas entradas, alcançam, plenamente, os objetivos.

Além disso, supõe sempre possível, definir todas as entradas necessárias para que determinado objetivo seja alcançado, uma vez que qualquer informação pode ser definida, em termos de informações mais elementares.

Langefors, através da definição de precedência entre conjuntos de informação, consegue associar a um Sistema de Informação, uma matriz, por ele denominada de - matriz de precedência - que espelha as relações de precedência entre os conjuntos de informação relativos ao sistema.

O estudo que Lagefors faz, baseado em Teoria dos Conjuntos, para a ordenação, por níveis, dos conjuntos de informação, visando uma disposição ótima das colunas da matriz de precedência, para maior facilidade, por ocasião de elaborar o dígrafo correspondente, nada mais é do que, em terminologia de Lieberman, o estabelecimento dos níveis, entre os Formulários, ou seja, o estabelecimento dos Formulários de Dados (primeiro nível) e dos Formulários de Relatórios (em vários níveis).

Uma vez que, no modelo de Lieberman, os Formulários já estão classificados, por níveis, podemos estabelecer analogia entre este modelo e o de Langefors e construir, para o exemplo anterior, o seu grafo de precedência, como mostrado na Figura. 4.9.

CLASSES DE INFORMAÇÃO

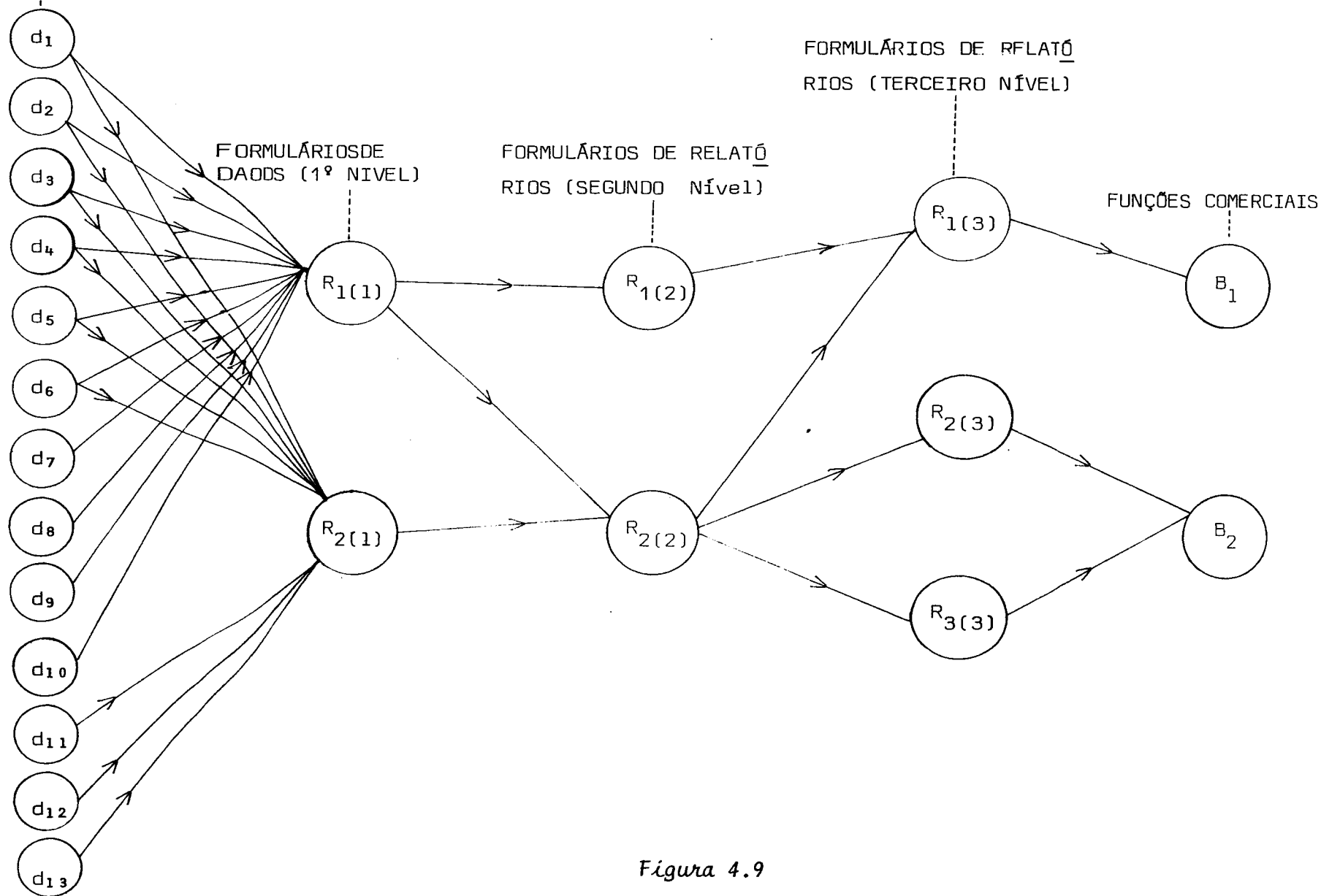


Figura 4.9

4.4. Conclusões

Neste capítulo, foi apresentado um trabalho de Durchholz, onde se enfocou as principais abordagens de Modelos de Sistemas de Informação : relacional, funcional, matricial e por pontos.

Sabemos que uma descrição em PSL, quando explicitada em termos de hipergrafos, adapta-se ao enfoque relacional e quando em termos de disposições da Base de Dados, usada pelo PSA (as que atendem às especificações da CODASYL), adapta-se ao funcional (disposições podem ser pensadas como inversas de funções).

Já o modelo matemático da Álgebra da Informação da CODASYL, é um modelo por pontos. Uma das mais importantes características dessa Álgebra, é a de nos dar um modo de fazer referência e manipular conjuntos de pontos, em um espaço de propriedades. Não realçamos a Álgebra da Informação neste capítulo, em virtude do trabalho bastante abrangente e completo, realizado por Germano, em [21].

Como o modelo matemático proposto por Lieberman é matricial, resolvemos apresentá-lo. Quando da apresentação do modelo desse autor, fizemos uma pequena analogia com o proposto por Langefors, uma vez que esses dois autores adotaram tipos de abordagem semelhantes. Completamos, dessa forma a apresentação de quatro modelos enfatizados por Durchholz. Além disso, introduzimos as idéias deste autor, quanto à dependência funcional de atributos, que estão intimamente relacionados com as Formas Normais de Codd [27], intencionado, com essa introdução, fornecer material para o desenvolvimento de pesquisas que envolvam comparações entre os trabalhos de Codd e os de Bosak (estendidos pela CODASYL).

Além disso, uma atraente linha de pesquisas que evidenciamos, foi a da busca da integração, entre os trabalhos de Teichroew, Lieberman e Bosak, apoiados no de Durchholz. Sugerimos, pois, como pesquisa futura, evidenciar cada um deles, de forma a dar os enfoques propostos por Durchholz, utilizando a

técnica que este autor colocou à disposição, ao expressar relações como matrizes e funções, usando, como auxílio, a abordagem por pontos.

CAPÍTULO V

CONCLUSÕES FINAIS E SUGESTÕES PARA NOVAS PESQUISAS

O objetivo principal do trabalho que nos propusemos realizar, foi o da explicitação do modelo matemático que estava implícito no desenvolvimento do Sistema PSL/PSA pleno. Germano [1] fez semelhante estudo, para um subconjunto da linguagem PSL; evidenciou que o modelo poderia ser considerado como sendo um dígrafo categorizado com rótulos e mostrou como poderia ser feita a extensão desses estudos para a linguagem PSL plena, através do conceito de hipergrafos.

Mostramos, pois, que o modelo matemático da linguagem PSL plena é um hipergrafo dirigido e, além disso, a possibilidade de representá-lo através de dígrafos, utilizando, para tanto, o Sistema de Base de Dados usado pelo PSA. Tivemos que fazer, portanto, uma reformulação de [17], totalmente voltada para esses objetivos, procurando tornar tal documento mais acessível às pessoas envolvidas com o Sistema PSL/PSA, lançando mão, principalmente de exemplos.

A explicitação do modelo matemático da linguagem PSL plena, certamente contribuirá para o entendimento do Sistema, bem como para facilitar seu desenvolvimento futuro. Tal explicitação será importante, quando da implementação de testes que verificam a consistência, na descrição das Especificações de um Sistema de Informação, como pode ser evidenciado no trabalho de Germano [1], para um subconjunto da linguagem. Foi baseado na explicitação matemática deste subconjunto, que foi possível o desenvolvimento de estudos que conduziram à elaboração do teste de consistência para Descrições de Sistemas de Informação. Tal teste foi baseado na verificação da conexidade em um subdígrafo, selecionado com certos critérios, do dígrafo que constitui a descrição de um Sistema de Informação, e a partir dessa verificação, são inferidas conclusões quanto à consistência da descrição.

Através do melhor entendimento proporcionado pela explicitação do modelo matemático, torna-se mais fácil assimilar o Sistema, bem como aprender a utilizá-lo, modificá-lo para atender às necessidades específicas não previstas nas versões disponíveis, ou então, incorporar-lhe características que eventualmente não possua.

Ainda mais, se pensarmos na linguagem PSL como uma linguagem para a descrição de dígrafos categorizados com rótulos, veremos que sua sintaxe passa a ser quase que intuitiva. Com isso, as objeções que muitos fazem, quanto ao uso do PSL para a descrição de Sistemas de Informação, tais como a de aprendizado demorado e difícil e de utilização trabalhosa, não têm mais sentido.

Além da explicitação por nós feita, nos preocupamos em apresentar as idéias de Durchholz que estão voltadas ao estudo de Sistemas de Base de Dados, onde são enfocadas quatro das principais abordagens de modelos de Sistemas de Informação: por pontos, matricial, relacional e funcional. Mostramos o modelo matemático proposto por Lieberman, que é explicitamente matricial e, dessa forma, abordamos os quatro principais enfoques, propostos por Durchholz.

Da apresentação do trabalho de Durchholz, evidenciamos uma linha de pesquisa futura, que seria aquela voltada para a integração dos trabalhos de Teichroew, Lieberman e Bosak, tendo por base, o de Durchholz. Como já dissemos, seria interessante evidenciar cada um deles, dando os quatro enfoques propostos pelo último autor, utilizando a técnica colocada por ele à disposição, para se expressar relações como matrizes e funções.

Finalizando, gostaríamos ainda de acrescentar um tópico de pesquisa, que também achamos interessante ser desenvolvido, voltado já para um estudo da literatura de Teoria dos Grafos, classificando-a em dois grupos: um de teoremas similares e outros de algoritmos, de relevância para o desenvolvimento do Sistema e também indicando as possíveis áreas de aplicação e usos desses teoremas e algoritmos, no PSL/PSA.

BIBLIOGRAFIA

- 1 - GERMANO, F.S.R. - Modelos matemáticos da teoria dos sistemas de informação: tese apresentada ao ICMSC - USP, para a obtenção do título de doutor, São Carlos, ICMSC/USP, 1966, 2v.
- 2 - DURCHHOLZ, R. - Relation representation by tables and by functions. Information systems, 1:91-96, 1975.
- 3 - CHURCHMAN, C.W. - Introdução à teoria dos sistemas. 2^a ed. - Petrópolis, Vozes, 1972.
- 4 - DIAS, D.S. & GAZZANEO, G. - Projeto de sistemas de processamento de dados. Brasília, Inl., Rio de Janeiro, Capre, Livros Técnicos e Científicos, 1975.
- 5 - PRINCE, T.R. - Sistemas de informação: planejamento, gerência e controle, Rio de Janeiro, EDUSP/ Livros Técnicos e Científicos, 1975, V.1.
- 6 - LANGEFORS, B. - Information systems. Amsterdam, North - Holland Publishing Company, 1974. (Information processing 74).
- 7 - TEICHROEW, D. & BASTARACHE, M.J. - PSL user's manual. Michigan, University of Michigan, 1975. (ISDOS working paper, n° 98)
- 8 - GERMANO, F.S.R. e NICOLETTI M.C. - Descrição do pré-cálculo do custo industrial de produtos finais de uma fábrica de compressores, usando o sistema PSL/PSA, São Carlos, ICMSC/USP, 1976.
- 9 - TEICHROEW, D. & CARLSON, D.M. - A model of the system building process, Michigan, University of Michigan, 1972. (ISDOS working paper, n° 64).
- 10- CHAPIN, N. - Flowcharts, U.S.A., Auerbach Publishers, 1971

11 - International conference on software engineering, II. Proceedings:

ALFORD, M., TRW - A requirements engineering methodology for real-time processing requirements.

BELL, T.E., BIXTER, D.C., and DYER, M.E., TRW - An extendable approach to computer - aided.

DAVIS, C.G. and VICK, C.R., Ballistic Missile defense Advanced Technology Center. The software development system.

IRVINE, C.A. and Brackett, J.W., Softech Inc. Automated software engineering through structured data management.

ROSS, D.T. & SCHOMAN, K.E., Softech Inc. Structured analysis for requirements definition.

TEICHROEWN, D. & HERSHEY III, E.A. - PSL/PSA: A computer-aided technique for structured documentation and analysis of information processing systems.

12 - TEICHROEW, D. and HERSHEY III, E.A. - An introduction to PSL/PSA - Michigan, University of Michigan, 1974. (ISDOS working paper n° 86).

13 - CORTES, E.R. - PSL/PSA: Un método formal de especificación de requerimientos de un sistema de información administrativo: Memória para optar al título de Ingeniero civil industrial, Santiago, Chile, Universidade de Chile, 1976.

14 - BERGE, C. - Graphs and hypergraphs, Amsterdam, North - Holland, 1976.

15 - FURTADO, A.L. - Formal aspects of the relational model. Rio de Janeiro, PUC, 1966, (Monographs in Computer Science and Computer Applications - n° 6/76).

16 - BERZTISS, A.T. - Data structures: theory and practice, New York, Academic Press, 1976.

- 17 - HERSHEY III, E.A. & BASTARACHE, M.J. - The structure and contents of a PSA data base - Michigan, University of Michigan, 1974 (ISDOS working paper n° 87).
- 18 - MARTIN, J. - Computer data-base organization, New Jersey, Prentice-Hall, Inc, 1975.
- 19 - HERSHEY III, E.A. - A data base management systems for PSA based on DBTG. Michigan, University of Michigan, 1975. (ISDOS working paper n° 88).
- 20 - KNUTH, D.E. - The art of computer programming: fundamental algorithms, Massachusetts, Addison - Wesley, 1968.
- 21 - GERMANO, F.S.R. - Formulação matemática dos problemas de processamento da informação: dissertação apresentada ao ICMSC - USP, para obtenção do título de doutor, São Carlos, ICMSC/USP, 1973.
- 22 - LIEBERMAN, I.J. - A mathematical model for integrated business systems. National meeting of the Institute of Management Sciences, 2nd, New York, 1955.
- 23 - LANGERFORS, B. - Theoretical analysis of information systems. New York, Petrocelli Charter, Auerbach, 1973.
- 24 - CODASYL - An information algebra. CACM 5(4), APRIL, 1962
- 25 - HOMER, E.D. - A generalized model for analyzing management information systems. Management Science 8(4): 500-515, Jul. 1962.
- 26 - PRIDMORE, H.D. - The abstract information system concept and the problem of optimum design. Australian Computer Conference, III, 1966. Proceedings.
- 27 - CODD, E.F. - Further normalization of the data base relational model. Courant computer Science Symposia, VI Data Base Systems, New York, Prentice-Hall, 1971.