

UNIVERSIDADE FEDERAL DO RIO GRANDE DO SUL
INSTITUTO DE INFORMÁTICA
CURSO DE PÓS-GRADUAÇÃO EM CIÊNCIAS DA COMPUTAÇÃO

**Análise Operatória de Ferramentas
Computacionais de Uso Individual e
Cooperativo**

por

PATRICIA ALEJANDRA BEHAR

Tese submetida à avaliação, como requisito parcial para
a obtenção do grau de Doutor em
Ciência da Computação



Prof. Dr. Antônio Carlos da Rocha Costa
Orientador

Porto Alegre, março de 1998.

CIP - CATALOGAÇÃO NA PUBLICAÇÃO

Behar, Patricia Alejandra

Análise lógico-operatória de ferramentas computacionais de uso individual e cooperativo / por Patricia Alejandra Behar.- Porto Alegre: CPGCC da UFRGS, 1997.

139 f. : il.

Tese (doutorado) - Universidade Federal do Rio Grande do Sul. Curso de Pós-Graduação em Ciência da Computação, Porto Alegre, BR - RS, 1997. Orientador: Costa, Antônio Carlos da Rocha.

1. Lógica Operatória. 2. Ferramentas Computacionais. 3. Ambientes Cooperativos. 4. Informática Educativa. I. Costa, Antônio Carlos da Rocha. II. Título

Informática - SBU
Educação - SBU
Informática: Educativa
Lógica Operatória

ENPq 1.32 04.00-2

UFRGS INSTITUTO DE INFORMÁTICA BIBLIOTECA		
REF. BIBLIOTECARIA:	37:681.32(043) B419AN	N.º REG: 34343
ORIGEM:	DATA:	PREÇO:
D	10/04/98	R\$ 30,00
FUNDO:	FORN.:	
II	II	

UNIVERSIDADE FEDERAL DO RIO GRANDE DO SUL

Reitora: Profa. Wrana Panizzi

Pró-Reitor de Pós-Graduação: Prof. José Carlos Ferraz Hennemann


Diretor do Instituto de Informática: Prof. Roberto Tom Price

Coordenador do CPGCC: Profa. Carla Maria Dal Sasso Freitas

Bibliotecária-Chefe do Instituto de Informática: Zita Prates de Oliveira

Bibliotecária responsável pela normalização de documentos: Ida Rossi

UNIVERSIDADE FEDERAL DO RIO GRANDE DO SUL
Sistema de Bibliotecas da UFRGS


37:681.32(043)
B419AN

INF
1998/158552-9
1998/04/16

“Descobri como é bom chegar quando se tem paciência, e para se chegar onde se quer que seja, aprendi que não é preciso dominar a força, mas a razão. É preciso, antes de mais nada, querer”.

Amir Klink

Para Papi, Mami, Dani y Gus...

Agradecimentos

Este percurso foi um longo caminho trilhado. Houve muitas curvas, desvios, muitas dúvidas e certezas, luzes que se viam no final do túnel e que de repente se apagavam. Mas, apesar disso, aprendi uma lição: de que estou somente começando, de que é preciso continuar, acreditar sempre no que se almeja, numa força interior,... Fazer das interrupções que a vida nos reserva, um caminho novo, fazer da queda um passo de dança, daquele medo... quem sabe uma escada? Do sonho de chegar lá...uma ponte, e da procura do ideal...um encontro. Muitas pessoas me ajudaram a trilhar este caminho e, se o Jorge Luiz Borges me permite alterar o seu poema Voltar a viver, que diz... *“Se pudesse voltar a viver a minha vida,... na próxima (agora palavras minhas), faria tudo de novo”*,...

Agradeço, em primeiro lugar, ao meu orientador, Prof. Dr. Antônio Carlos da Rocha Costa, com o qual eu pude contar durante todos estes anos e que foi muito mais do que um mestre, me ensinando a aprender, a ensinar e a crescer profissionalmente, por ter me ajudado a descobrir o caminho que tinha que seguir,...

À Prof. Dra. Léa da Cruz Fagundes pela sua energia positiva, sua garra constante em tudo o que significasse o desenvolvimento de algo inovador nesta área emergente, pelo exemplo de lutadora de um ideal, de uma educação melhor,...

À Prof. Dra Cleci Maraschin pela sua participação contínua em todos estes anos, suas contribuições significativas e imprescindíveis que surgiram através de discussões, seu interesse e apoio sem limites na construção do meu trabalho,...

Ao Instituto de Informática da UFRGS que me auxiliou com todo tipo de material e, principalmente com ajuda financeira para poder publicar e divulgar os meus trabalhos,...

Ao CNPq que me forneceu apoio financeiro durante o período da tese,...

Aos meus amigos que me deram todo o apoio afetivo, aquelas palavras de carinho, a “força amiga” que, sem a qual não produzimos, não vivemos,...

Ao Dani, meu irmão, por ter sido sempre meu “amigão”, meu parceiro de gargalhadas, de brincadeiras, de esportes, de praia, de viagens, pelos momentos de vida que passamos juntos e que nos fez sofrer, crescer e nos manter cada vez mais unidos, pelo sabor das vitórias que comemora sempre ao meu lado,...

Ao meu pai Moni por ter sido meu “orientador” em todos os sentidos da minha vida profissional, corrigindo meus projetos, por ter me passado o “gene” da pesquisa, o gostinho daquele certo/incerto próprio da investigação, por ter me aberto este caminho me dando todo seu exemplo acadêmico e, é claro, aquela fonte de amor de pai e a força para seguir em frente em todo este período,...

À minha mãe Rosita que, com sua ternura, simplicidade, paz, carinho e compreensão me manteve tranqüila, ou pelo menos tentava, com seu ouvido sempre pronto para ouvir alguma história que eu tinha para contar ou para desabafar, que me ergueu incessantemente diante das minhas tristezas, um exemplo de mulher, de amiga e companheira inseparável,...

Ao meu “maridão” Gus, sempre com seu sorriso estampado para me ver, ouvir, transmitindo toda essa doçura e alegria que me acalmou nos momentos difíceis, suas palavras de estímulo e confiança sempre presentes, seu amor ilimitado que transborda e que me fez terminar este período da minha vida feliz, em equilíbrio, pronta para enfrentar novos desafios...

Pela emoção de ter atingido mais uma meta de vida, uma etapa que, com certeza, me levará a encontrar uma nova estrada que me conduzirá,... quem sabe aonde?

Sumário

Lista de Figuras	10
Lista de Tabelas	13
Resumo	14
Abstract	15
1 Introdução	16
2 Plano de pesquisa	18
3 Nível individual	19
3.1 Teoria do sujeito individual - Função simbólica do sujeito	19
3.1.1 Conceitos básicos piagetianos	20
3.1.2 Lógica operatória piagetiana	21
3.1.3 Principais operações lógicas e infralógicas	24
3.2 “Reconstrução” do método piagetiano de análise lógico- operatória de experiências do sujeito individual	30
3.2.1 Procedimento de explicação das experiências	31
3.2.2 Aplicação do método em uma experiência piagetiana	32
3.3 Conceitos para o estudo operatório de ferramentas computacionais de uso individual	39
3.3.1 Caracterização do sujeito individual	40
3.3.2 Formas de interação do sujeito com ferramentas computacionais	42
3.3.3 Sujeito (usuário/programador) e o modelo mental do funcionamento do computador	43
3.4 Modelo geral de interação de um sujeito com uma ferramenta computacional de uso individual	44
4 Nível coletivo	47
4.1 Teoria do sujeito coletivo - Lógica operatória e cooperação	47
4.1.1 Construção coletiva de conhecimentos	49
4.2 Trabalho cooperativo apoiado por computador	50
4.2.1 Computação Cooperativa	51
4.3 Conceitos para o estudo operatório de ferramentas computacionais de uso cooperativo	55
4.3.1 Caracterização do sujeito coletivo	57
4.3.1.1 O sujeito coletivo e a cooperação	60
4.3.1.1.1 As operações lógicas nas trocas sociais do sujeito coletivo	61
4.3.1.1.2 A operatoriedade dos valores coletivos	62

4.3.2 Modelos gerais de interação de um sujeito coletivo com ferramentas computacionais de uso cooperativo	66
---	-----------

5 Análise operatória de ferramentas computacionais de uso individual	69
5.1 Editor gráfico Paintbrush	69
5.1.1 Descrição do ambiente	69
5.1.1.1 Operações do menu em relação ao sistema	70
5.1.1.1.1 Controle direto do sistema	70
5.1.1.1.2 Controle indireto do sistema	71
5.1.1.1.3 Manipulação direta sobre o desenho	72
5.1.1.1.4 Manipulação indireta sobre o desenho	72
5.1.2 Análise lógico-operatória do editor gráfico Painbrush	74
5.1.2.1 As operações infralógicas mais elementares e o contínuo	75
5.1.2.2 As operações infralógicas formadoras das relações topológicas	76
5.1.2.3 As operações infralógicas e o espaço euclidiano	78
5.2 Editores de texto	80
5.2.1 Análise lógico-operatória do editor de texto Word 6.0	81
5.2.2 Hipertextos	84
5.3 Ambientes de programação	85
5.3.1 Ambiente de programação Logo	85
5.3.2 Ambiente de programação Winlogo	87
5.3.2.1 Operações do processo interativo sujeito usuário e/ou programador x Ambiente Winlogo	89
5.3.2.1.1 Operações de controle do ambiente	89
5.3.2.1.2 Operações básicas do processo de interação	92
5.3.2.1.3 Operações de controle do processo de interação	93
6 Análise operatória de ferramentas computacionais de uso cooperativo	96
6.1 Microsoft Netmeeting 2.0 Beta 4	96
6.1.1 Descrição do ambiente	96
6.1.2 Operações lógicas/infralógicas do Microsoft Netmeeting 2.0 Beta 4	98
6.1.2.1 Operações lógicas/infralógicas do quadro de comunicações	100
6.1.2.2 Operações lógicas/infralógicas do bate-papo	102
6.2 Microsoft Chat 2.0 Beta 1	103
6.2.1 Descrição do ambiente	103
6.2.2 Operações lógicas/infralógicas do Microsoft Chat 2.0 Beta 1	106
6.3 Ambiente de desenvolvimento cooperativo de programação ENVY/400	112
6.3.1 Descrição do ambiente	112

6.3.2 Programação em equipe e controle de versões	116
6.3.3 Operações lógicas/infralógicas do ambiente de desenvolvimento cooperativo de programação ENVY/400	117
7 Proposta de uma base de desenvolvimento de um método de análise lógico-operatória do sujeito coletivo	124
8 Conclusões	132
Bibliografia	135

Lista de Figuras

FIGURA 1 - Plano de estudo no nível individual	18
FIGURA 2 - Plano de estudo no nível coletivo	19
FIGURA 3 - Reconstrução do método piagetiano de análise lógico-operatória de experiências	31
FIGURA 4 - Aplicação do método na experiência, relativa ao estágio pré-operatório .	36
FIGURA 5 - Aplicação do método na experiência, relativa ao estágio operatório concreto	37
FIGURA 6 - Aplicação do método na experiência, relativa ao estágio operatório formal	38
FIGURA 7 - Interação individual apoiada por computador	40
FIGURA 8 - Estruturas que compõem um sujeito individual	41
FIGURA 9 - Elementos que compõem a interação sujeito x ambiente computacional .	42
FIGURA 10 - Modelo geral de interação sujeito x ambiente computacional.....	46
FIGURA 11 - Formas de Cooperação em sistemas CSCW	53
FIGURA 12 - Natureza Geográfica de Sistemas de CSCW	54
FIGURA 13 - Interação interindividual apoiada por computador	56
FIGURA 14 - Definição recursiva de um sujeito coletivo	56
FIGURA 15 - Organização dos elementos envolvidos em uma interação interindividual	57
FIGURA 16 - Organização interna do sujeito coletivo	58
FIGURA 17 - Organização externa do sujeito coletivo	58
FIGURA 18 - Escala de valores em forma de relações assimétricas	63
FIGURA 19 - Escala de valores hierárquicos	63
FIGURA 20 - Modelo geral 1 de interação entre um sujeito coletivo x ambiente computacional cooperativo centralizado	67
FIGURA 21 - Modelo geral 2 de interação entre um sujeito coletivo x ambiente computacional cooperativo distribuído	68
FIGURA 22 - Modelo geral 3 de interação entre um sujeito coletivo x ambiente computacional cooperativo distribuído	69
FIGURA 23 - Interface do Editor Gráfico Paintbrush	72

FIGURA 24 - Modelo de interação sujeito x editor gráfico	74
FIGURA 25 - Interface do Editor de texto Word 6.0	81
FIGURA 26 - Modelo de interação sujeito x editor de texto	82
FIGURA 27 - Modelo de interação sujeito x ambiente de programação LOGO	86
FIGURA 28 - Interface do ambiente de programação Winlogo	88
FIGURA 29 - Modelo de interação sujeito x ambiente Winlogo	89
FIGURA 30 - Interface da ferramenta Microsoft Netmeeting 2.0 Beta 4	97
FIGURA 31 - Modelo geral de interação do sujeito coletivo com o Microsoft Netmeeting 2.0 Beta 4	98
FIGURA 32 - Modelo de interação dos sujeitos com a ferramenta quando ocorrem alterações nas representações dos mesmos	100
FIGURA 33 - Modelo de interação dos sujeitos com o Bate-papo	103
FIGURA 34 - Interface da ferramenta Microsoft Chat 2.0 Beta 1	104
FIGURA 35 - Modelo de interação entre um sujeito coletivo x ferramenta cooperativa Microsoft Chat 2.0	105
FIGURA 36 - Modelo de interação do sujeito coletivo com o ambiente ENVY/400 .	115
FIGURA 37- Biblioteca compartilhada ENVY	116
FIGURA 38 - Salvando e carregando desde o repositório compartilhado	117
FIGURA 39 - Edições e versões	117
FIGURA 40 - Organização externa do método de análise lógico-operatória no nível coletivo	125
FIGURA 41 - Base para o desenvolvimento de um método de análise lógico-operatória em relação aos valores de um sujeito coletivo	127
FIGURA 42 - Exemplo da aplicação do método de análise lógico-operatória em relação aos valores de um sujeito coletivo na ferramenta computacional cooperativa Microsoft Chat 2.0 Beta 1	128
FIGURA 43 - Base para o desenvolvimento de um método de análise lógico- operatória em relação aos objetos de um sujeito coletivo	129

- FIGURA 44 - Exemplo da aplicação do método de análise lógico-operatória em relação aos objetos de um sujeito coletivo na ferramenta computacional cooperativa Microsoft Netmeeting 2.0 Beta 4 **130**
- FIGURA 45 - Base para o desenvolvimento de um método de análise lógico-operatória em relação à linguagem de um sujeito coletivo **131**
- FIGURA 46 - Exemplo da aplicação do método de análise lógico-operatória em relação à linguagem de um sujeito coletivo no ambiente de desenvolvimento cooperativo de programação ENVY/400 **132**

Lista de Tabelas

TABELA 1 - Operações realizadas pelo sujeito em interação com o editor gráfico Painbrush	46
TABELA 2 - Operações realizadas pelo sujeito em interação com o editor de texto ..	84
TABELA 3 - Operações realizadas pelo sujeito em interação com o ambiente LOGO..	87

Resumo

Esta tese de doutoramento trata da integração da teoria piagetiana com a Ciência da Computação, mais especificamente, a análise de ferramentas computacionais do ponto de vista da lógica operatória. Para isso, foi preciso investigar, em primeiro lugar, a teoria do sujeito individual, no que se refere à função simbólica e reinterpretar estes conceitos no objeto. Neste caso, o objeto é a ferramenta computacional de uso individual. Portanto, somente a partir deste estudo foi possível construir o modelo geral de interação de um sujeito qualquer com uma ferramenta computacional, para depois analisá-la operatoriamente.

Em um segundo momento, introduziu-se o teoria do sujeito coletivo que, para caracterizá-lo, foi necessária a compreensão de alguns conceitos relevantes da teoria em questão envolvidos nas noções de interação interindividual e cooperação. A partir de então, foram construídos os modelos interativos mais simples de um sujeito coletivo com três tipos de ferramentas computacionais de uso cooperativo. Para isso, foram abordados alguns aspectos relativos à Computação Cooperativa ou CSCW - Computer Supported Cooperative Work. Da mesma forma que ocorre no nível individual, foram analisadas operatoriamente ferramentas computacionais de uso cooperativo.

PALAVRAS-CHAVE: Lógica Operatória, Ferramentas Computacionais, Ambientes Cooperativos, Informática Educativa.

**TITLE: "OPERATORY ANALYSIS OF COMPUTATIONAL TOOLS
FOR INDIVIDUAL AND COOPERATIVE USE"**

Abstract

This thesis integrates the piagetian theory with Computer Science, more specifically, the computational tools analysis, from an operatory-logical point of view. For that, it was necessary to investigate, in a first step, the individual subject theory, in relation to the symbolic function and to interpret this concept in the object. In that case, the "object" is a computational tool for individual use. After this study it was possible to construct the general interaction model of any subject with a computational tool and to analyse it in an operatory form.

In a second step, we introduce the collective subject theory. For that it was necessary to comprehend some relevant concepts of the theory in question involved in the notions of interindividual interaction and cooperation. After that, it was constructed the simplest interactive model of a collective subject in interaction with three types of computational tools for cooperative use. For that, some concepts related to CSCW or Computer Supported Cooperative Work were reported. In the same way that occurred in the individual level, the computational tools for cooperative use were analysed in an operatory form.

KEYWORDS: Logical-operatory, Computational tools, Cooperative environments, Computer Science in Education.

1 Introdução

Esta tese de doutoramento é o resultado de um trabalho de investigação que iniciou no segundo ano do curso, e que utiliza a teoria piagetiana para realizar uma análise operatória de ferramentas computacionais.

Na teoria piagetiana, o desenvolvimento do conhecimento de um sujeito envolve as idéias de construção e de interação social. Isso quer dizer que, através das interações que um sujeito estabelece com o meio, ele põe em ação seus sistemas de tratamento, de recepção e de significação, “transformando” os valores, objetos e/ou linguagem de conhecimento e transformando-se a si mesmo através da ação [PIA 73].

Podem, então, ser identificados dois tipos de interação:

- sujeito - objeto (interação individual) e, neste caso chamamos de objeto ao ambiente computacional;
- sujeito - sujeito - objeto (interação interindividual), onde existe uma situação de cooperação entre sujeitos para interação com o meio.

Por essa razão, este trabalho foi dividido em duas partes: em primeiro lugar foi realizado o estudo do sujeito individual em relação ao ambiente computacional e depois o estudo do sujeito coletivo em interação com as ferramentas computacionais cooperativas.

Para entender o papel do sujeito em relação ao computador ou, mais especificamente, a uma determinada ferramenta, os psicólogos piagetianos (observadores) acompanham e estudam a evolução cognitiva do sujeito, coletando os dados necessários até chegar no seu desenvolvimento máximo, com o objetivo de analisá-lo. Neste estudo, o sujeito é interpretado em termos da lógica-operatória piagetiana. Mas, para compreender de forma completa essa interação sujeito-computador, também é preciso entender a ferramenta que está envolvida na mesma. Quem faz esse estudo é a Ciência da Computação e pensamos que, para tornar essa descrição compatível com a do sujeito, isto é, descrita nos mesmos termos, o computador também deve ser reinterpretado nos termos da lógica-operatória.

Para concretizar esse objetivo, o primeiro passo foi realizar uma leitura em algumas das obras de Jean Piaget, em relação ao sujeito individual, do ponto de vista operatório. A partir do estudo de “Introducción a la lógica operatória” [CAS 82] e “Da lógica da criança à lógica do adolescente” [PIA 76], percebeu-se a necessidade de uma definição e delimitação mais clara dos diferentes aspectos utilizados na análise operatória apresentada nas respectivas obras. Por essa razão, reconstruiu-se o método de análise de experiências que estava implícito na análise que Piaget realizou, facilitando a compreensão do método através da sistematização dos aspectos que direcionavam o mesmo.

A reconstrução realizada encontra-se descrita em [BEH 95a] e faz uma apreensão dos elementos mais importantes envolvidos na análise das experiências, o significado da sua forma e do seu conteúdo e, principalmente, procura tornar mais visível o processo de desenvolvimento das operações lógicas do sujeito individual, facilitando a leitura das obras piagetianas.

Por outro lado, iniciou-se o estudo caracterizando em termos da lógica operatória o computador. Os resultados deste estudo foram apresentados em trabalhos que descrevem uma forma de análise operatória de algumas ferramentas computacionais como, por exemplo, do aplicativo Paintbrush do Windows 3.1 [BEH 96d], do ambiente de

programação WinLogo [BEH 96e], do editor de textos Word 6.0 e da linguagem de programação LOGO [BEH 96b], entre outras.

Dessa forma, foi possível caracterizar em termos lógicos e infralógicos, tanto o sujeito quanto as ferramentas computacionais.

Utilizando como base esses estudos realizados em relação ao sujeito individual, partimos para o estudo das interação interindividuais, isto é, vários sujeitos interagindo com um ambiente computacional. Portanto, foram extraídos os conceitos necessários da teoria piagetiana à nível interindividual para caracterizar um instrumento computacional coletivo de interação e realizar a análise operatória do mesmo.

Essa caracterização da ferramenta de uso coletivo, poderá servir como base para o desenvolvimento de um método de análise operatória de ferramentas de uso coletivo e aplicação do mesmo numa determinada experiência, que envolva a interação de um grupo de sujeitos com uma ferramenta computacional cooperativa. Ou seja, este seria a extensão do método que foi reconstruído no nível individual [BEH 95a].

Para concretizar esta linha de raciocínio, o texto foi dividido da seguinte forma: no próximo capítulo é descrito o plano de pesquisa proposto, mostrando o caminho que foi seguido ao longo desta abordagem.

O capítulo 3 descreve a abordagem conceitual relacionada ao uso individual de ferramentas computacionais, como a teoria do sujeito individual do ponto de vista piagetiano, as operações lógicas e infralógicas, os processos interativos e a construção do modelo de interação de um sujeito com a ferramenta computacional de uso individual.

No capítulo 4 é realizado o mesmo estudo, mas em relação às ferramentas de uso coletivo, enfatizando o estudo das interações interindividuais. A partir disso, é caracterizado o sujeito coletivo, os ambientes computacionais cooperativos, os processos de interação relacionados a eles e construídos os modelos gerais de interação entre um sujeito coletivo e as ferramentas computacionais de uso cooperativo.

O capítulo 5 apresenta a análise lógico-operatória das ferramentas computacionais de uso individual. No capítulo 6, esse mesmo estudo é realizado no nível cooperativo, de acordo com as formas de interação do sujeito com a ferramenta computacional.

No capítulo 7 são descritos alguns conceitos necessários para a construção de uma base de desenvolvimento de um método de análise lógico-operatória do sujeito coletivo.

O capítulo 8 apresenta as conclusões das idéias elaboradas ao longo do trabalho, mostrando o caminho percorrido para a concretização das hipóteses levantadas no início da pesquisa, as contribuições gerais e específicas da mesma e as possibilidades de trabalhos futuros.

2 Plano de pesquisa

O que nos interessa neste trabalho é aprofundar o estudo dos elementos que fazem parte dos modelos de interação entre sujeitos e ferramentas computacionais, partindo do estudo do sujeito individual para atingir o sujeito coletivo. Para isso, foi necessário extrair as características mais importantes e as operações envolvidas nos mesmos a fim de estabelecer as relações entre os elementos.

A partir daí, foi possível caracterizar e analisar operatorialmente as ferramentas computacionais cooperativas. Através deste primeiro esboço, é possível realizar nos próximos trabalhos uma experiência coletiva para validar as idéias que foram descritas e, se possível, utilizar estes estudos como base para a construção de um método de análise lógico-operatória no nível coletivo.

A seguir é mostrado o caminho de investigação que foi seguido.

NÍVEL INDIVIDUAL

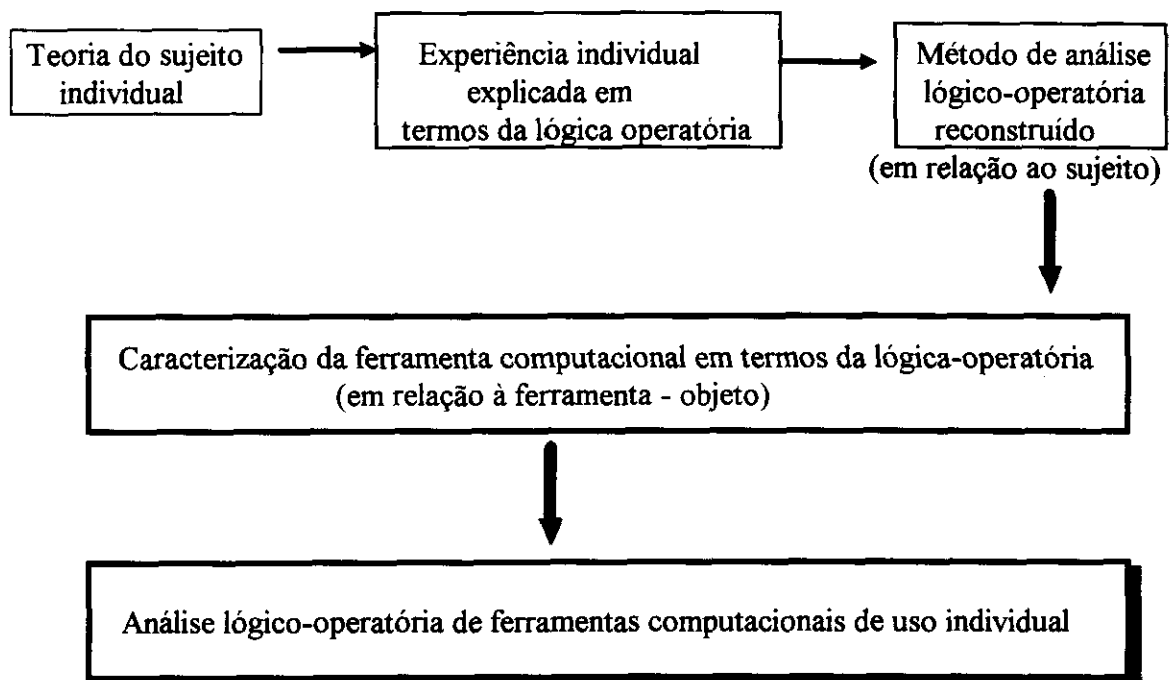


FIGURA 1 - Plano de estudo no nível individual

NÍVEL COLETIVO

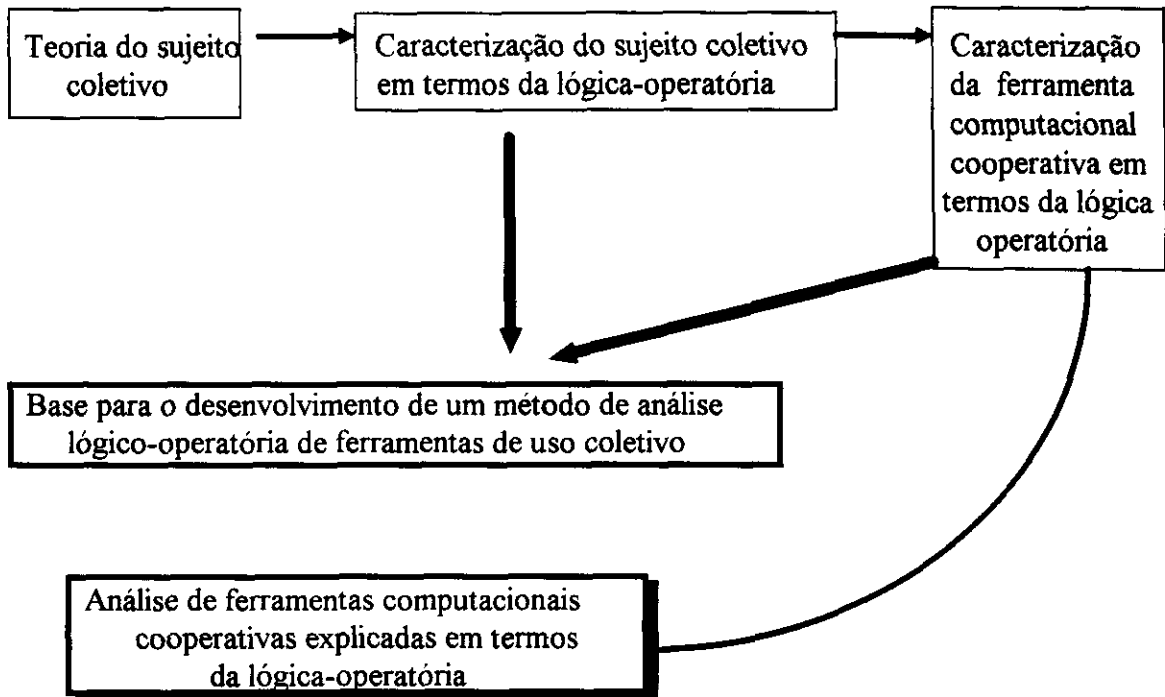


FIGURA 2 - Plano de estudo no nível coletivo

3 Nível individual

3.1 Teoria do sujeito individual - Função simbólica do sujeito

O objetivo final deste estudo é analisar, desde o ponto de vista operatório, uma determinada ferramenta computacional utilizada por um grupo de sujeitos para representar os seus conhecimentos. Mas para isso, é preciso entender, em primeiro lugar, qual a função que o sujeito individual utiliza para se expressar através do computador. Ele usa a sua *capacidade representativa* para expressar, através do computador, suas atividades representativas e operativas.

Por que estudar a atividade representativa do sujeito?

Porque tudo aquilo que o sujeito expressa seja no computador ou fora dele, é o resultado de um processo interno de construção de um sistema de significações. Por exemplo, cada objeto representado corresponde, progressivamente, a uma imagem mental que permite ao indivíduo evocar esse objeto na sua ausência. Para isso, ele precisa ter um meio de se manifestar, seja este através da linguagem oral e/ou escrita, da imitação diferida, da imagem gráfica ou do jogo simbólico.

Através da *função simbólica* é que se pode representar objetos ou acontecimentos não percebidos atualmente nas evocações por meio de símbolos ou signos diferenciados.

Essa é uma das razões de apresentar neste capítulo, alguns *conceitos básicos piagetianos* utilizados ao longo deste estudo, *o desenvolvimento da função simbólica do sujeito e o papel da atividade representativa*.

3.1.1 Conceitos básicos piagetianos

Existem inúmeras teorias que explicam a origem do conhecimento, seu desenvolvimento e, conseqüentemente, sua forma de representação. Neste estudo optou-se por seguir a concepção interacionista de desenvolvimento, que é a teoria Piagetiana.

Segundo a teoria, o conhecimento não parte nem do sujeito (S) nem do objeto (O), mas dialéticamente da interação indissociável de ambos, pois nenhum dos dois estão prontos ($S \Leftrightarrow O$). São totalidades que se diferenciam um pela ação do outro, representando um movimento dialético de troca, de mudanças recíprocas, na qual a noção de *desenvolvimento do conhecimento* se amplia e passa a se constituir das idéias de construção e interação com o meio social.

Desse ponto de vista, o conhecimento é construído não pela exposição do sujeito ao meio, mas pela ação, pela atividade do sujeito sobre o objeto a conhecer de modo a decompô-lo e a recompô-lo para aprendê-lo [PIA 76].

Logo, o conhecimento não só resulta da relação direta do sujeito com os objetos, mas de sua forma de interpretação e representação. Ele aprende, por exemplo, a desenhar, a medida que desenha, ou seja, da sua interação com o desenho, o que lhe propicia construir hipóteses acerca da natureza e função deste sistema.

Qual é o significado de atividade representativa?

Segundo Piaget, a representação pode ser entendida de duas formas:

1. *“ela se confunde com o pensamento, isto é, como toda inteligência que já não se apoia simplesmente nas percepções e nos movimentos (inteligência sensório-motora), mas antes num sistema de conceitos ou de esquemas mentais”* (no sentido lato);
2. *“ela se reduz à imagem mental ou à lembrança-mental, isto é, à evocação simbólica das realidades ausentes”* (no sentido restrito) [PIA 71].

Portanto quem diz representação, diz reunião (**ação**) de um “significante” (símbolo) que permite a evocação de um “significado” (conceito) fornecido pelo pensamento. A evolução da inteligência representativa inicia na atividade sensório-motora, passando pela atividade representativa egocêntrica (pensamento pré-conceitual e intuitivo) e terminando na atividade representativa de ordem operatória.

Pensar consiste em interligar significações; dessa forma a imagem é um significante e o conceito um significado. Representar é, portanto, *“reconstruir no plano mental o que estava estruturado no plano das ações”* [PIA 71] e, também, construir sobre essas reconstruções.

Segundo a teoria piagetiana [PIA 71], *“a função simbólica (ou semiótica) é concebida enquanto mecanismo comum aos diferentes sistemas de representações e enquanto mecanismo individual cuja existência prévia é necessária para tornar possível as interações de pensamento e, por conseguinte, a constituição ou a aquisição de significados coletivos”*.

As diversas formas de pensamento representativo interdependem e interagem umas com as outras, evoluindo todas em função de um equilíbrio progressivo.

Segundo Piaget [PIA 71], para poder representar “alguma coisa”, o sujeito utiliza a função simbólica, que ele prefere denominar de *função semiótica*, porque ela engloba os signos, arbitrários e sociais e os símbolos, analógicos, individuais e sociais. Portanto, é

esta função que permite ao sujeito representar objetos ou situações que podem estar fora de seu campo visual, através de imagens mentais, do desenho, da linguagem escrita ou, como pode ser constatado na ciência da computação, através de programas.

Como é o aparecimento desta função no sujeito?

A inteligência representativa do sujeito inicia quando existe uma diferenciação entre o significante e o significado. Ou seja, no estágio sensório-motor a criança ainda não possui uma representação, ou imagem mental de um determinado objeto, nem muito menos de suas próprias ações. A passagem da inteligência sensório-motora à inteligência representativa se opera através das diferentes condutas representativas: a imitação diferida, o jogo simbólico, o desenho ou imagem gráfica, a imagem mental e a evocação verbal das ações passadas através da linguagem. O que constitui, então, a função semiótica e, o que faz ultrapassar a atividade sensório-motora, é a capacidade do sujeito representar um objeto que não é visível, ou seja, ausente, através de símbolos ou signos. Isso implica em poder diferenciar e coordenar os significantes e os significados ao mesmo tempo. Portanto, a representação nasce da união de “significantes” que permitem evocar os objetos ausentes com um jogo de significação que os une aos elementos presentes. Surge esta função nova, que é a função simbólica.

Pode-se observar que, durante o *período sensório-motor*, o significante e o significado permanecem indiferenciados (atividade sensório-motora). Quando estes começam a se diferenciar, a criança tem acesso à linguagem e ao pensamento. Ela elabora imagens que lhe permitem compreender o mundo do seu ponto de vista, isto é, ela adquire a linguagem e vai formando um sistema de imagens na sua cabeça. Esse sistema é um sistema de signos particulares referentes a uma realidade particular e é por essa razão que a palavra ainda não tem o valor de um conceito; ela ainda não consegue generalizar. Mas a criança já começa a relacionar uma imagem ao seu correspondente simbólico, ou seja, mesmo que seja uma realidade fechada nela mesma, a criança já constrói o seu sistema de relações. Esses “conceitos particulares” são denominados por Piaget de *pré-conceitos* (atividade egocêntrica). Esta é a fase do egocentrismo e, com ela, o artificialismo, o animismo, o realismo intelectual, etc. O significante é para a criança um símbolo e, o significado um pré-conceito. Portanto, sua representação se encontra num estado imagístico e simbólico.

Num segundo momento, quando a criança se encontra na fase do *pensamento intuitivo*, esta une o pensamento pré-conceitual e simbólico ao operatório.

Quando o sujeito atinge a reversibilidade lógica, há um ponto de ruptura, e a imagem começa a ser subordinada pelas *operações reversíveis* (atividade operatória). Ou seja, através do sistema de operações que o sujeito constrói, ele diferencia o significante do significado e, finalmente, organiza as representações a nível espaço-temporal e causal.

O próximo passo do presente estudo é descrever alguns conceitos da lógica operatória piagetiana para compreender a análise das operações lógicas e infralógicas que são realizadas pelo sujeito em interação com as ferramentas computacionais e introduzir o papel da cooperação neste processo.

3.1.2 Lógica operatória piagetiana

Segundo Piaget [PIA 73], o pensamento do sujeito é formado com participação indispensável do grupo social a que este pertence, pois graças à linguagem e às coações de cada geração sobre as seguintes, o sujeito é, a todo momento, tributário do conjunto das aquisições anteriores, assim transmitidas pela via “externa” da educação. As noções

de pensamento e as regras lógicas, ultrapassam os limites da atividade individual e supõem a colaboração, cooperação entre indivíduos. As regras lógicas consistem em leis normativas necessárias às trocas de pensamento e, impostas por uma necessidade social, em oposição à anarquia das representações espontâneas do sujeito.

Jean Piaget estudou a evolução do pensamento lógico desde a criança até o adulto, com o objetivo de determinar o modo de sua construção [CAS 82] [PIA 76] [PIA 72]. Sua meta foi além da simples descrição das ações observadas nas crianças. Ele buscava uma explicação estrutural das mesmas. Piaget converteu a resposta desta questão, a um de seus princípios mais importantes da teoria: as ações das crianças (e também dos adultos) não se apresentam em forma caótica, desordenada e sem conexão, mas evidenciam formas de organização distintas em vários períodos de desenvolvimento. Estas formas de organização das ações são pensadas por ele como sendo estruturas de conjunto que, ao organizá-las, lhes dá significado, integrando-as em um todo coordenado e estruturado. Surge, então, a tarefa de especificar qual estrutura de conjunto que possibilita a obtenção cognitiva característica de cada etapa de desenvolvimento da inteligência. E, dessa forma, compreender o que uma criança pode ou não fazer em um determinado estágio e o que ela poderá construir no próximo.

Piaget preocupou-se em descrever como se dá no sujeito, a construção das estruturas de conjunto características dos períodos operatórios do pensamento da criança e, para isso, selecionou a linguagem da lógica e da matemática moderna. Estas estruturas de conjunto são apresentadas através da linguagem lógico-matemática, constituindo o objeto de estudo da lógica operatória.

Segundo Piaget [PIA 72], a lógica operatória descreve, então, as estruturas de conjunto da lógica natural dos sujeitos, através do aparelho teórico da lógica formal e da matemática. Esta lógica aparece como uma construção intermediária entre a lógica natural dos indivíduos e a lógica formal dos lógicos. Em resumo, a lógica operatória constitui os modelos formais que representam esta lógica natural, tais como o agrupamento de classes e de relações e o grupo das transformações proposicionais.

A razão de chamar lógica operatória à lógica que explica tais estruturas de conjunto da lógica natural, é que estas constituem sistemas de operações.

Mas, o que é **operação** para Piaget?

Segundo Piaget, é denominada de operação "*à ação interiorizada ou interiorizável, reversível e coordenada em uma estrutura total...*", ou ainda, "*é uma ação suscetível de voltar ao seu ponto de partida e de fazer composição com outras segundo esse duplo modo direto e inverso*" [PIA 71]. Assim, um conjunto de operações forma um sistema de operações.

Nesta abordagem, este conceito de operação em relação ao sujeito é reinterpretado para o objeto. Portanto, chamamos de operação na ferramenta computacional à aquela realizada pelo sujeito mas transferida para a máquina. Assim, estas ações realizadas na ferramenta também podem formar um sistema de operações. Logo, no presente estudo serão analisadas algumas operações lógicas e infralógicas tanto do sujeito quanto das ferramentas computacionais.

O que são as **operações lógicas**?

Segundo Piaget [CAS 81], a operação lógica "*é aquela que trata sobre objetos individuais considerados como invariantes e se limita a reuni-los ou a relacioná-los independentemente de suas vizinhanças e das distâncias espaço-temporais que os separam*". Este tipo de operação agrupa ou reúne os objetos em classes segundo propriedades em comum, dando lugar à inclusão de classes e às operações que existem

entre elas, como a adição, subtração, etc. Ela também ordena os objetos segundo diferenças, dando lugar às seriações aditivas e multiplicativas.

Neste caso, o objeto é tomado independentemente das relações espaço-temporais entre este e outros objetos e, além disso, é considerado como uma unidade, sem levar em conta as partes espaciais que o constituem. Por exemplo, se uma criança agrupa casinhas e outros objetos de acordo com as propriedades em comum, quer dizer que estas se constituem em classes. Ela não se atém ao fato das casinhas estarem espacialmente mais próximas umas das outras. Somar um conjunto de canetas a um outro, resulta em um conjunto maior, no qual não são levados em conta nem a localização espacial dos mesmos e nem as distâncias entre elas.

No caso de uma exemplificação utilizando um editor gráfico, podemos destacar as operações lógicas que são encontradas no desenvolvimento de um texto, no que se refere à relação das partes de um texto com seu produto final, a seqüência que deve ser seguida em um texto para organizar um todo coerente, a correspondência que deve existir entre as partes do mesmo, entre outras.

O que são as **operações infralógicas**?

Em termos gerais, a **operação infralógica** consiste em *“compor o objeto através de seus próprios elementos, conseguindo assim, não classes e nem relações independentes do espaço, mas sim, objetos totais de distintos tipos”*.

Trata-se, por exemplo, de reunir as partes de um objeto em um todo ou de colocá-los em uma ordem de sucessão determinada [PIA 93]. Assim, por exemplo, em um quebra-cabeças, para formar uma figura a partir dos elementos, é necessário considerar as relações espaciais dos fragmentos entre si, as vizinhanças, a fim de determinar qual o fragmento que deve ir ao lado do outro para formar a figura total. É evidente, então, que a figura é concebida como um todo contínuo em relação às suas partes. Em vez de reunir ou separar os objetos segundo as suas semelhanças ou diferenças, como acontece nas operações lógicas, nas operações infralógicas se reúnem ou se separam as partes que compõem um objeto, segundo a posição espacial que estas ocupem. Estas partes são espacialmente dependentes do objeto total. Como ocorre no quebra-cabeças quando uma criança constrói uma casinha (objeto total) com um triângulo (parte do objeto) e um quadrado (parte do objeto) embaixo, ela faz depender espacialmente ambas as partes, no sentido que o quadrado deve ir sempre embaixo do triângulo, devendo estar ambos em uma relação de proporcionalidade, ou seja, tem que levar em conta o fator da escala na construção dos objetos.

O termo **infralógico** significa que estes tipos de operações são formadoras da noção do objeto. Estas não se apoiam nos encaixes de classes mas sim, nas partes de um mesmo objeto total, substituindo a noção de semelhança pela de vizinhança, a de diferença pela de ordem ou colocação e a de número pela de medida.

Segundo nossa interpretação, quando um sujeito, por exemplo, desenvolve uma atividade em um editor gráfico, ele está trabalhando sobre o espaço representativo, compondo uma figura total através de suas partes. Isto significa que ele estará operando, preponderantemente, em um nível infralógico, compondo, a partir dos objetos parciais, objetos totais. Isso não quer dizer que o sujeito não esteja fazendo operações lógicas. Com certeza também estará.

Do ponto de vista formal, não existe nenhuma superioridade do nível lógico em relação ao infralógico. Estes dois não se pressupõem mutuamente, eles tratam de um mesmo sistema de operações, isto é, o agrupamento aplicado a modos diferentes de operar com os objetos.

Na próxima seção são descritas as principais operações lógicas e infralógicas. Algumas são ilustradas com exemplos, onde foram identificadas estas operações, nas ferramentas computacionais.

3.1.3 Principais operações lógicas e infralógicas

As principais operações lógicas são as seguintes:

1. **Classificação:** as operações de classificação agrupam os objetos segundo suas equivalências. Efetuar uma classificação é agrupar objetos segundo seus critérios comuns. A mais simples delas, aparece como uma seqüência linear de encaixes. Por exemplo, no caso do editor de texto Word 6.0, esta operação pode ser identificada quando se ativa o comando *Classificar texto*; menu *Tabela* => classifica em parágrafos, campos, data,... ou na opção *Estilo Texto* => manuscrito, artigo, anúncio, carta,...

Esta operação pode ser implementada da seguinte forma em um programa desenvolvido no ambiente WinLogo:

```
> Para Entre? : A :B :C
> Devolve E ? :B :A Menor? :B :C
> Fim
```

É um procedimento que devolve Verdadeiro se o segundo valor for um número menor que os outros dois. É o tipo de representação da interação em que o sistema é programado para identificar a relação de Menor entre 3 elementos diferentes, ou seja, estará classificando-os.

2. **Seriação:** estas operações agrupam os objetos segundo suas diferenças ordenadas; elas se constituem ao mesmo tempo que as classificações, por volta dos sete anos. Em um editor de texto esta operação é identificada quando se deve seguir uma ordem (texto), padronizar partes do texto, definir números dos capítulos (organizados em série), seções, etc.

3. **Proporcionalidade:** proporção é a relação entre duas relações; é a comparação entre duas medidas, a igualdade entre duas razões. O esquema das proporções opera a transição entre os esquemas provenientes das redes (combinatória proposicional) e os que provêm das estruturas de grupo (INRC). No aplicativo Paintbrush, esta operação aparece no caso de ativar o menu *Modificar*, a opção *Reduzir/Ampliar* ou *Mais Zoom/Menos Zoom* em relação a uma figura.

4. **Probabilidades:** está diretamente ligada à combinatória e ao esquema das proporções. Assim, é preciso ser capaz de fazer uma combinatória que permita ter em conta todas as associações possíveis entre os elementos em jogo e um cálculo das proporções, para apreender as probabilidades.

Através das pesquisas, foi constatado [PIA 93] que as primeiras operações e relações infralógicas que a criança constrói e utiliza são as seguintes operações e relações elementares:

1. **Vizinhança:** é a relação mais elementar do espaço, se refere à “proximidade” dos elementos percebidos num mesmo campo; esta leva vantagem em relação aos outros fatores de organização (semelhança, simetria, etc). Por exemplo: diversas partes desenhadas em um editor gráfico Paintbrush são “vizinhas” uma das outras ao invés de dispersadas nos quatro cantos da tela.

2. **Separação:** intervém na medida em que os elementos desenhados são distinguidos uns dos outros; dissociar dois elementos ou fornecer um meio para distingui-los. Exemplo: ativar a área de *Recorte* do Painbrush marcando uma parte de uma figura na tela e deslocá-la com o mouse para o outro canto da tela, separando-a do resto do desenho.

3. **Ordem:** relação espacial que se estabelece entre elementos ao mesmo tempo vizinhos e separados. Também a relação de simetria que pode ser simbolizado pela dupla ordem ABC, CBA. Por exemplo, quando não existe esta relação há inversão das relações de esquerda e direita, acima e abaixo, frente e trás. Em uma ferramenta qualquer, é preciso seguir uma certa ordem na execução dos comandos porque, caso contrário, estes não funcionarão. A opção *Colar* ou *Copiar*, não funciona se antes não for ativado o *Recorte*, marcando uma parte de uma figura ou texto.

4. **Circunscrição (ou de envolvimento):** esta relação é acentuada quando se trata de figuras mais simples, mas dão lugar a constantes erros em figuras complexas. A B C; onde o elemento B é percebido como estando entre A e C (circunscrição a uma dimensão). Por exemplo, desenhar com a caixa de ferramentas do Paintbrush uma figura com os olhos fora do rosto, botões do lado da cabeça e do corpo.

5. **Continuidade e descontinuidade:** é uma simples justaposição dos elementos em oposição à sua ligação contínua. Continuidade e descontinuidade das linhas e das superfícies. Um conjunto de elementos (partes de uma figura) conectados uns nos outros formando uma paisagem (figura completa), por exemplo, desenhada no Painbrush.

Estas relações e operações infralógicas dizem respeito às noções topológicas mais intuitivas na representação do espaço e são consideradas elementares do ponto de vista da construção teórica do mesmo. Elas são desenvolvidas antes de qualquer organização projetiva e/ou euclidiana.

As operações mais elementares que constituem as relações topológicas são:

1. **Partição e adição primitiva:** Ex: $A + A' = B$; $B + B' = C \dots$ e $C - B' = B$; $B - A' = A$. Este tipo de estrutura é o equivalente, no domínio infralógico das vizinhanças, à estrutura dos encaixes das classes A, B, C ... no domínio da lógica. No caso de utilizar o editor de texto Word, esta operação é identificada quando se ativa o comando *Inserir* ou *Deletar* uma página em um documento.

2. **Ordem de colocação:** se refere à operação de ordem direta e inversa entre os elementos: A, B, C, ... $A \Rightarrow B \Rightarrow C \dots$ e, $C \Rightarrow B \Rightarrow A \dots$. A ordem constitui um deslocamento (movimentos necessários ao sujeito para recorrer a seqüência).

```

> Para Dia : Dia
> Faz "Semana [domingo segunda terça quarta quinta sexta sábado]
> Devolve PosLista :Dia :Semana
> Fim
? E Dia "terça?
3

```

O resultado do programa devolve a ordem de colocação do elemento terça, de acordo com a posição na lista.

3. Reciprocidade das vizinhanças: essa noção de "vizinhança geográfica", que é a reciprocidade inerente, não é muito bem compreendida pela criança, isto é, se A é vizinho de B, então B necessariamente é vizinho de A. Portanto, se o sujeito parte do elemento A, considerando A' como vizinho (escreve-se A_1 e A_1'), ele também pode partir de A como elemento vizinho (A_2'). Então:

$A_1 + A_1' = A_2 + A_2' (= B)$; da mesma forma que:

$B_1 + B_1' = B_2 + B_2' (= C)$ e assim por diante.

Esta operação não foi identificada em nenhuma ferramenta computacional analisada.

4. Relações simétricas de intervalos: a reciprocidade das separações se traduz em termos de relações por um sistema de relações simétricas, que são as relações de intervalos. Transpondo esta noção para o domínio das noções topológicas, pode-se dizer que os intervalos são marcados pela relação "entre". Isto é, B está entre A e C na ordem direta, da mesma forma que está situado entre C e A, na ordem inversa. Daí, surge um sistema de relações simétricas que sempre pode ser extraído de uma seriação. Portanto,

$A \Leftrightarrow B (= 0)$; $A \Leftrightarrow C (=B)$; $A \Leftrightarrow D (=B,C)$, quer dizer que, "entre" A e B não existe nenhum elemento, entre A e C, o elemento B e entre A e D, os elementos B e D e, assim por diante. Esta operação aparece na ordem de colocação dos comandos. Por exemplo, no Word, para recortar uma parte do texto é preciso *Marcar (A)*, *Recortar (B)* e *Colar (C)*. Portanto, entre a operação de Marcar e Recortar não existe nenhuma outra, mas entre Marcar (A) e Colar (C) é necessário Recortar (B), senão a operação de Colar não funciona.

5. Multiplicação biunívoca de elementos: um objeto fechado no interior de uma caixa não pode ser ligado a um elemento exterior a não ser atravessando uma das paredes, ou seja, uma superfície. Aqui nos referimos à superfície com uma dimensão 0, isto é, tem por fronteira o ponto, a um sistema a duas dimensões que tem por fronteira uma linha (dimensão 1) e a um sistema com três dimensões tendo como fronteira uma superfície (um sistema a duas dimensões). A elaboração dessas relações leva a um agrupamento multiplicativo tão logo o sujeito passa de uma dimensão a muitas. Essa operação se refere aos encaixes simultâneos de elementos segundo dois sistemas ao mesmo tempo.

Por exemplo: prepara-se uma tela ao sujeito onde ele precisa resolver um problema utilizando as operações do editor. São desenhadas várias figuras divididas em dois pedaços, tipo quebra-cabeças, dispostas em duas colunas. Além disso existem partes de figuras que não mantêm relação com nenhuma outra. Pede-se ao sujeito

ligar essas colunas relacionando as partes, mostrando quais os elementos que se ligam entre si, apagando as partes que não possuem correspondência com outras. Assim, o sujeito deverá montar essa rede de duas dimensões utilizando, por exemplo, o pincel da caixa de ferramentas (MDD). Com este, ele poderá fazer uma correspondência biunívoca, estabelecendo uma rede tal que: os elementos de uma seqüência (coluna) sejam colocados em correspondência com os elementos da outra seqüência, estabelecendo assim a conexão entre eles. As partes que não possuem nenhum tipo de relação poderão ser apagadas através dos comandos do menu *Editar, Recortar*. Dessa forma pode-se dizer que o sujeito estará fazendo uma **multiplicação biunívoca de elementos**.

6. Multiplicação biunívoca de relações: uma rede de 2 ou 3 dimensões pode ser construída em termos de relações. Este agrupamento multiplicativo de relações, aparece sob a forma de correspondências biunívocas que são estabelecidas pelo sujeito entre duas seqüências ordenadas. Isso significa que, os elementos A_1, B_1, C_1, \dots etc, de uma determinada seqüência são colocados em correspondência com os elementos A_1', B_1', C_1', \dots , de uma outra seqüência quando uma certa conexão é estabelecida entre A_1 e A_1' , B_1 e B_1' , ... Esta conexão muitas vezes não ultrapassa o trajeto seguido pelos movimentos do olhar, mas também pode ser desenhado o traço que une estas duas seqüências, ou seja, as linhas cruzariam as seqüências, segundo um sistema multiplicativo a duas dimensões.

Esta operação não foi identificada em nenhuma ferramenta analisada.

7 e 8. Multiplicação co-unívoca de elementos e de relações: as operações 5 e 6 se referem às correspondências biunívocas, ou seja, termo a termo. Esta, constrói correspondências um a diversos, isto é, de forma co-unívoca. Por exemplo: num mosaico qualquer, o elemento A_1 corresponderá a muitos elementos vizinhos $A_2 B_2 C_2, \dots$, essa seqüência $A_2 B_2 C_2, \dots$ corresponderá a um grande número de elementos vizinhos $A_3 B_3 C_3 D_3 F_3, \dots$ e, assim por diante, formando uma extensão progressiva de vizinhanças.

Essas relações infralógicas constroem sistemas únicos e contínuos que são os espaços totais.

Por outro lado, as operações infralógicas que constituem um sistema de coordenadas cartesiano se referem à localização espacial do objeto e os diversos tipos de deslocamentos que este pode fazer no espaço. As operações infralógicas mais elementares que constituem o espaço euclidiano, se apoiam num objeto relativo a um ponto de vista e nas mudanças de ponto de vista. Portanto, elas exprimem os caracteres do objeto relativamente à sua localização e seus deslocamentos.

As operações são as seguintes:

1. Adição e subtração de elementos: reunir e dissociar as partes de uma figura. Por exemplo: $A + A' = B$; $B + B' = C$, etc. Essa operação tem que assegurar a conservação do todo tanto a título de localização quanto da figura do objeto colocado. Por exemplo, no Paintbrush prepara-se uma experiência: as partes de um homem encontram-se desenhadas e espalhadas pela tela. Além disso, esses elementos não se encontram na sua posição correta, ou seja, a boca está em posição vertical, um braço está de cabeça para baixo e o outro de cabeça para cima, o nariz está virado e as cores de um dos olhos está invertidas. O homem (H -

objeto total) deverá ser composto dos vários elementos (objetos parciais): cabeça (C), nariz (N), boca (BC), olhos ($O = O_1 + O_2$), corpo (CO), braços ($B = B_1 + B_2$), pernas ($P = P_1 + P_2$), obedecendo a posição correta dos mesmos. O sujeito precisa relacionar as partes com as partes, tendo assim, a visão do todo. Para isso, necessita trabalhar com o sistema de coordenadas, isto é, com as noções de rotação e translação dos elementos (deslocamentos), da vertical e horizontal, entre outras.

Assim, cada vez que o sujeito adiciona ou exclui um elemento da figura total estará utilizando a operação infralógica de **adição e subtração de elementos**, assegurando sempre a conservação do todo, tanto a título de localização quanto da figura do objeto colocado.

2. Colocações e deslocamentos: colocar em ordem linear os objetos, mudança de ordem e deslocamentos dos objetos. Por exemplo, tem-se $A \rightarrow B \rightarrow C \rightarrow \text{etc}$, estes objetos são ditos “colocados” uns em relação aos outros. No sistema das operações constitutivas do espaço euclidiano, a operação inversa não é somente inverter a seqüência e sim, é a mudança de ordem ou de colocação, isto significa, um deslocamento, que pode tanto inverter a seqüência inteira ou um elemento em relação aos outros. Assim, $A \rightarrow C \rightarrow B$ por inversão de $B \rightarrow C$. Logo, pode-se observar que existe uma diferença entre a ordem dos elementos e a das colocações. A noção de deslocamento resume-se às noções qualitativas, ou seja, que relacionam as partes com o todo. A mesma experiência anterior é preparada na tela de edição e diversos elementos se encontram separados uns dos outros nos quatro cantos da tela. Pede-se ao sujeito para colocá-los numa certa ordem, portanto, estes elementos são ditos colocados uns em relação aos outros. No espaço euclidiano, há uma diferença entre a ordem dos elementos e a ordem das colocações. Essa operação é denominada de **colocações e deslocamentos** e ela pode ser realizada somente arrastando as partes da figura com o mouse ou, se o sujeito preferir, através do menu *Editar*, através das operações de *Copiar* e *Colar*.

3. Reciprocidade das referências: leva-se em conta diversas figuras vizinhas, constituídas pelas formas dos objetos ou de suas colocações, adicionadas segundo as operações do tipo 1, partindo de A_1 como referência. Logo, $A_1 + A_1' = B_1$ e $B_1 + B_1' = C$. Assim, sempre é possível chegar à mesma reunião C partindo de A_1' ou de B_1' como referência e, chamando-a A_2 . Portanto, $A_2 + A_2' + B_2' = C$. Esta operação leva à reciprocidade dos sistemas de coordenadas, sendo o elemento A_1 ou A_2 considerado como origem de cada um dos sistemas de coordenadas;

4. Ajustes dos intervalos ou distâncias: o intervalo entre dois pontos ordenados ao longo de uma reta é uma distância. A conservação das distâncias é assegurada pelo fato desses pontos e reta fazerem parte das “localizações” imóveis mesmo podendo ser percorridos por um objeto em movimento. A relação constituída pelo deslocamento deste último entre X e Y sendo uma relação assimétrica, a distância constitui a relação simétrica de intervalos correspondente. É simétrica porque de X a Y há uma mesma distância do que de Y a X, isto é, pode ser simbolizado por: $X \Leftrightarrow Y = Y \Leftrightarrow X$;

5. Multiplicação biunívoca dos elementos: uma seqüência linear de elementos $A_1 + A_1' = B_1$; $B_1 + B_1' = C_1$, etc... multiplicada por outra $A_2 + A_2' + B_2'$ etc... constitui

uma superfície. Estas duas seqüências, multiplicada por uma terceira $A_3 + A_3' + B_3'$, etc, produzem um volume.

6. Multiplicação biunívoca das relações de colocações e deslocamentos: esta operação diz respeito às mesmas traduzidas em termos de relações assimétricas (ordem de colocação e deslocamento), sendo que elas engendram um sistema de coordenadas. Esse sistema é uma rede de colocações ordenadas em função dos pontos de referência ou objetos considerados como imóveis, sendo tais colocações ordenadas segundo duas ou três dimensões simultaneamente: enquanto uma das seqüências ordenadas constitui um dos eixos do sistema e, a segunda, ordena de acordo com outra dimensão, isto é, constitui um segundo eixo.

7. Multiplicação co-unívoca dos elementos: ao contrário das correspondências biunívocas próprias aos dois sistemas precedentes, a multiplicação por correspondência co-unívoca engendra a noção do triângulo, a duas dimensões, ou do tetraedro a três.

8. Multiplicação co-unívoca das relações: trata-se de um intervalo simétrico crescente produzido por duas relações assimétricas de valor progressivo. Este sistema operatório é quem dá lugar à avaliação qualitativa dos ângulos antes da introdução de uma medida.

> Para RECOMEÇAR

> Avança 5

> Se CoordY > 90 [FixaRumo 180 -
Rumo]

> Se CoordY < 1 [FixaRumo 0 Stop]

> PULAR

> Fim

Este procedimento permite que a tartaruga pule. Para isso, tiveram que ser utilizadas as seguintes operações de controle da interação: **operação co-unívoca de relações**: Se CoordY > 90 [FixaRumo 180 - Rumo] e Se CoordY < 1 [FixaRumo 0 Stop]. PULAR é o procedimento de repetição do pulo da tartaruga (**operação de ordem cíclica**).

Depois de descritos alguns conceitos considerados importantes em relação à teoria individual, vamos para o segundo quadro apresentado na figura 2, que apresenta uma experiência individual em termos da lógica-operatória. O que se quer dizer com isso, é que foram descritas algumas experiências individuais (físicas e químicas) retiradas da obra de Piaget “Da lógica da criança à lógica do adolescente” [PIA 76], em termos da lógica-operatória utilizando um método de análise que é apresentado a seguir.

3.2 “Reconstrução” do método piagetiano de análise lógico-operatória de experiências do sujeito individual

O método de análise lógico-operatória reconstruído foi aplicado em três experiências [BEH 96f] que estão descritas no livro “Da lógica da criança à lógica do adolescente” [PIA 76]. Nessa reconstrução foram considerados alguns conceitos da obra de Piaget, “Recherches Sur L’abstraction Réfléchissante” [PIA 77].

No trabalho de reconstrução do método de análise da experiência, procura-se detalhar a formulação lógico-matemática das operações lógicas de que os sujeitos se valem na explicação dos fenômenos que procuram compreender, explicitar o seu significado em relação ao estágio em que são manifestadas e, a maneira como se constroem a partir das operações anteriores.

As palavras-chave utilizadas para analisar as experiências foram obtidas a partir dos seguintes critérios que envolvem tanto uma análise da participação dos sujeitos nas experiências, quanto a análise realizada por Piaget na explicação das mesmas. Assim, foram feitas:

- uma leitura minuciosa dos relatos das experiências, buscando analisar o discurso dos sujeitos através de conceitos da lógica formal e da matemática moderna - **Modelo Mental** do Sujeito -,
- uma representação das manifestações do sujeito contidas no relato de seu discurso, traduzidas em termos da lógica proposicional - **Simbolização Lógica** do Modelo Mental do sujeito - e,
- uma análise da terminologia e da formulação conceitual utilizada por Piaget, na explicação dos itens anteriores ¹.

O quadro do **modelo mental** é composto pelos seguintes itens: os **objetos** (os diferentes elementos que direcionam e dão suporte à experiência), suas **propriedades** (características particulares dos elementos), as **relações entre os objetos** (ligações entre elementos da experiência) e os **eventos** (acontecimentos que ocorrem independente da ação do sujeito). De acordo com a leitura que o sujeito faz do evento, ele é capaz de verificar ou não a ocorrência de determinadas transformações; essas transformações ocorrem à nível de **relações e propriedades**. Finalmente, apresenta-se a relação entre esses eventos, ou seja, o **mecanismo da experiência** (a causalidade) e a justificativa da mesma. Estas palavras-chave foram retiradas da ênfase que Piaget dá aos relatos de suas experiências, ressaltando a relação de sujeitos com objetos em determinadas situações.

Por outro lado, os termos utilizados para caracterizar a **simbolização lógica do modelo mental do sujeito** se apresentam na forma de: **proposições** (são as sentenças declarativas afirmativas e atômicas que expressam os componentes básicos do modelo mental); **proposições de eventos** que expressam a ocorrência de eventos; e, finalmente, as **operações sobre proposições** (que “ligam” as proposições, através dos conectivos lógicos: “e lógico”, negação, “ou lógico”, implicação, implicação recíproca, entre outros) manifestando o modelo mental que o sujeito elaborou a respeito do mecanismo da experiência. Estas palavras-chave foram retiradas da fundamentação teórica da lógica operatória.

Cabe salientar, por outro lado, que esse esquema de análise é admitidamente simplificador, e que a riqueza das formulações piagetianas nem sempre se encaixa com

¹ Estes aspectos variam de acordo com o estágio em que se encontra o sujeito da experiência.

facilidade neste quadro técnico. A vantagem que se vê, é a sua sistematicidade. Ele deve ser utilizado como guia genérico, não como uma métrica rígida.

Os aspectos anteriormente descritos estão ilustrados na figura 3.

3.2.1 Procedimento de explicação das experiências

O método reconstruído de análise de experiências requer, em primeiro lugar, a apresentação de uma resenha explicativa da experiência que está sendo analisada. Como são utilizados conceitos específicos de uma dada teoria (seja esta física, química, etc), estes têm que ser esclarecidos. Portanto, é feita uma descrição breve destes aspectos.

Logo depois, dá-se uma exposição sumária de cada estágio de desenvolvimento, apresentando uma breve idéia do comportamento e pré-requisitos mais importantes do pensamento do sujeito em relação à etapa analisada.

Estes itens são importantes para situar o leitor no contexto de abordagem da experiência. Além disso, apresenta-se uma resenha descrevendo quais as transformações ocorridas entre um estágio e outro. Estas transformações, como já foi enfatizado, se referem ao processo de abstração reflexionante.

Só então, se faz a descrição do modelo mental e da simbolização lógica, conforme esquematizado na figura 3.

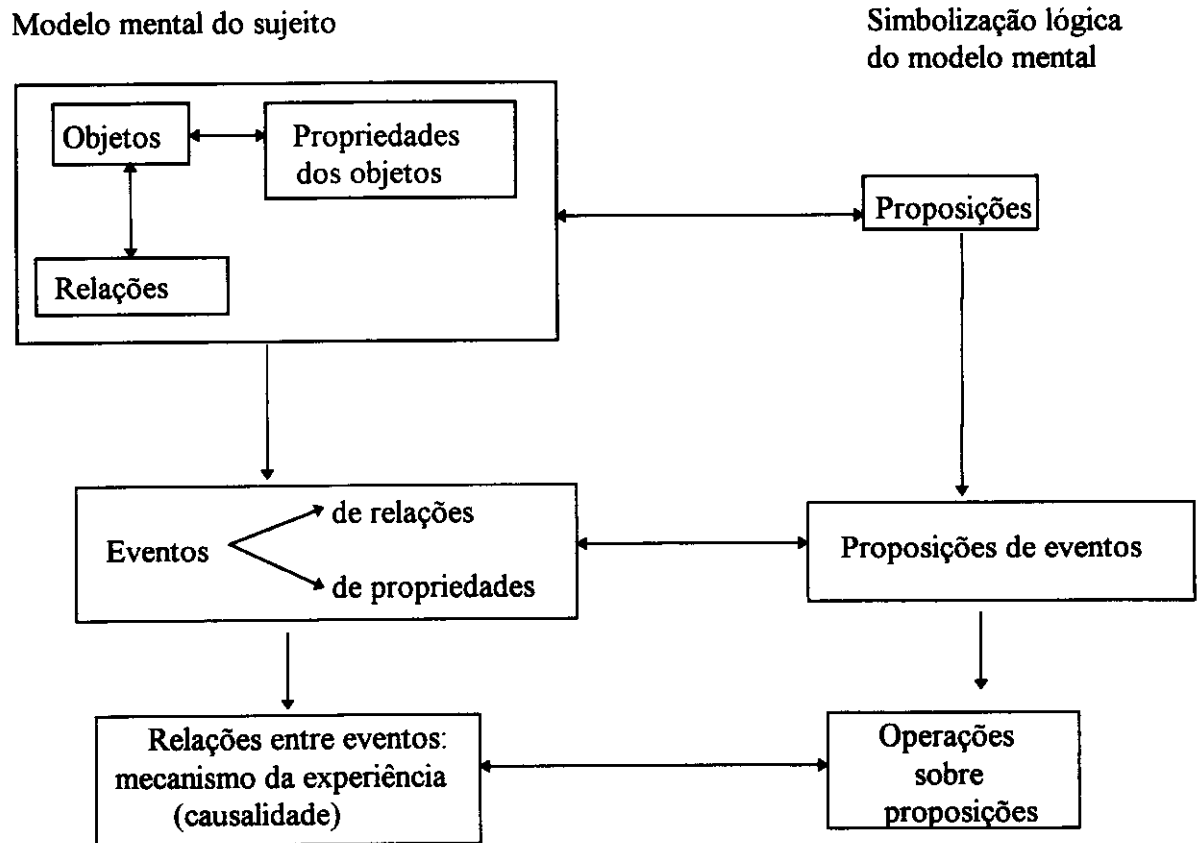


FIGURA 3 - Reconstrução do método piagetiano de análise lógico-operatória de experiências

Logo depois, se explica o que o sujeito alcançou à nível da teoria (física, química, entre outras) que apoia a experiência, apresentando, por último, as operações lógicas que o sujeito manifestou durante sua participação na experiência.

Através da reconstrução do método de análise da experiência, é possível descrever as diferentes experiências de forma bastante detalhada, obtendo uma visão clara das diferentes operações lógicas que se desenvolvem em cada estágio do pensamento do sujeito, desde o período pré-operatório até chegar no operatório formal. A aplicação do método em três experiências piagetianas encontra-se descrita em [BEH 96f]. Neste trabalho somente será apresentado o método aplicado a uma experiência, a título de ilustração.

3.2.2 Aplicação do método em uma experiência piagetiana

A seguir será apresentada a aplicação do método na experiência “*A igualdade dos ângulos de incidência e de reflexão e as operações de implicação recíproca*”, que foi retirada da obra de Piaget “*Da lógica da criança à lógica do adolescente*” [PIA 76].

Esta experiência trata da análise das explicações do sujeito a respeito da igualdade dos ângulos de incidência e de reflexão na experiência de rebatimento de um móvel contra uma parede, e o uso da operação de implicação recíproca na explicação desse fenômeno.

Segundo o método, o primeiro passo é dar uma **descrição de como se dá a experiência**, apontando os aspectos considerados mais importantes. Nesta experiência é utilizado um aparelho, que é uma espécie de jogo de bilhar, no qual as bolas são lançadas por um mecanismo de mola tubular que pode ser deslocado para várias direções em torno de um ponto fixo. A bola é lançada contra uma tela de projeção e é rebatida para o interior do aparelho. Coloca-se um alvo em diferentes e sucessivos pontos, e os sujeitos devem apenas procurar atingi-lo. Depois, dizem o que observaram. Ora, a igualdade entre os ângulos de incidência e de reflexão só é construída no nível IIIA (11-12 a 14 anos) e, muitas vezes, não é formulado antes do nível IIIB (14-15 anos), segundo Piaget [PIA 76]. O que se questiona é o porquê de uma noção tão conhecida desde 7-9 anos, como é o caso da igualdade de dois ângulos, só é utilizada tão tarde na indução de uma lei elementar. O segundo passo é **apresentar a teoria física que explica a experiência** em termos operatórios, definindo conceitos como os ângulos de incidência e de reflexão e estabelecendo a lei da igualdade dos mesmos como o princípio explicativo da experiência.

No primeiro estágio, ou seja, no nível pré-operatório, a atenção dos sujeitos está centralizada no triunfo ou no fracasso práticos (bola atingir o alvo), sem consideração do seu mecanismo, e com o desprezo freqüente da intervenção dos rebatimentos; a trajetória da bola não é concebida, a não ser no fim do estágio, como formada por segmentos retilíneos mas, ao contrário, como percorrendo uma espécie de curva; o sujeito age apenas para atingir o objetivo, e não se pergunta como é que o consegue; não toma consciência do caráter retilíneo dos segmentos de trajetórias, nem da existência dos rebatimentos; não percebe a presença de ângulos no ponto de rebatimento.

A partir de então, monta-se o quadro apresentado na figura 3, composto do **modelo mental** e da sua **simbolização lógica**. Os **objetos** do estágio pré-operacional são os seguintes: objetos empíricos: taco, bola (lançada), alvo, ponto de partida, ponto de chegada. As **propriedades dos objetos** são também empíricas: direção da bola, direção do taco. As **relações entre os objetos**: direção (bola lançada, alvo) (relação

empírica). Os **eventos de relações e de propriedades** são os seguintes: lançar a bola, ajustar taco, deslocar a bola desde o seu ponto de partida até o de chegada e as **proposições de eventos**, ou seja, a expressão dos eventos anteriores.

Logo depois, apresenta-se o **mecanismo da experiência**: a bola é lançada pelo taco e atinge um determinado ponto de chegada. O sujeito não se preocupa com os rebatimentos ocorridos, ele age apenas para atingir o objetivo e não se pergunta como é que o consegue. Isto ocorre, porque ainda não tomou consciência do caráter retilíneo dos segmentos de trajetórias, nem da existência dos rebatimentos. Por isso, não percebe a presença de ângulos no ponto de rebatimento.

A partir do modelo mental e da simbolização lógica do mesmo, descreve-se **o que o sujeito alcançou à nível da teoria física que apoia a experiência**. Portanto, no primeiro estágio, o sujeito só faz uma relação primitiva entre o ponto de partida e o ponto de chegada, ou seja, ele não consegue estabelecer nenhum tipo de relação explicativa.

Analisando todas as informações anteriormente apresentadas, chega-se a conclusão de que o sujeito possui operações correspondentes ao nível pré-operacional. Ou seja, seu pensamento não é ainda operatório. Ele desconhece a reversibilidade e a conservação de suas operações. Isto pode ser visto, através de sua ação em relação à experiência. O sujeito somente consegue observar o ponto de partida e o ponto de chegada (objetos), não se preocupando com os elementos (propriedades e relações entre objetos) envolvidos no processo de deslocamento da bola, desde o seu lançamento até o alvo.

No segundo estágio, ou seja, no operatório concreto, há uma mudança geral de comportamento do sujeito desse estágio em relação ao anterior. Isso porque um dos fatores que interfere na mudança de suas ações, se refere ao início do pensamento operatório. Esta mudança é percebida pela “nova” forma como o sujeito interage com os objetos da experiência.

Neste estágio os sujeitos isolam os elementos necessários para a descoberta da lei da igualdade dos ângulos de incidência e de reflexão; não chegam à construção da lei e nem a sua formulação verbal; não procuram a razão para as correspondências (de inclinação dos dois segmentos de trajetórias, antes e depois do rebatimento); consideram os segmentos em presença apenas do ponto de vista de suas direções; não têm idéias de dividir em dois ângulos iguais, o ângulo total formado por esses dois segmentos; não se limitam somente a agir, mas interiorizam as suas ações sob a forma de operações de deslocamento ou colocação. Os **objetos** são: o taco, a tela de projeção, bola (lançada), alvo, ponto de partida, ponto de chegada, ponto de rebatimento, trajetória da bola do ponto de partida ao ponto de chegada. As **propriedades** são: inclinação (maior ou menor) do taco e a direção da bola. A **relação entre os objetos** é a direção (taco, alvo), algumas noções de ângulo de projeção (não seu conceito), a inclinação (maior, menor) taco e a distância (bola lançada, ponto de rebatimento, inclinação do taco). Os **eventos de relações e propriedades** são os seguintes: o lançamento da bola vai depender da inclinação do taco, que implica em um ponto determinado de rebatimento na tela de projeção e, em consequência disso, vai atingir o alvo. As **proposições de eventos** são a verbalização dos eventos anteriores, na sua forma simbólica.

Em comparação ao estágio anterior, isto é, no pré-operacional, onde o sujeito somente estava preocupado com a ação inicial e final, neste, pode-se perceber o início da análise de alguns elementos, que intervêm no processo que ocorre entre essas duas ações. No que se refere ao modelo mental e a sua simbolização lógica, verifica-se uma ampliação de dados, demonstrando a construção de novas **operações lógicas** no sujeito.

Portanto, percebe-se uma mudança de comportamento do sujeito em relação às ações que desempenhava até então.

No estágio concreto, o **mecanismo da experiência** é concebido da seguinte forma: a bola é lançada pelo taco (depende da sua inclinação), atingindo um determinado ponto de rebatimento para depois, atingir o alvo. Esse mecanismo mostra que os sujeitos interiorizam suas ações sob a forma de operações de deslocamento ou colocação. Portanto, tomam consciência do fato do taco poder ser deslocado segundo certas inclinações, de que o trajeto da bola é composto por dois segmentos retilíneos e, de que estes dois segmentos formam entre si um ângulo. Este ângulo depende da inclinação do taco. As **proposições** são: inclinação do taco (mostra o objeto de incidência), o ponto de rebatimento, a linha de reflexão (saída do ângulo). As **operações sobre proposições** são a correspondência em relação aos três fatores anteriores, a seriação e a implicação. Logo depois, é descrito o que o sujeito atingiu à nível da teoria física que apoia a **experiência**.

Neste estágio os sujeitos conseguem isolar todos os elementos necessários para a descoberta da lei da igualdade dos ângulos de incidência e de reflexão mas, apesar disso, não chegam à construção desta lei, e nem a sua formulação verbal.

No terceiro e último estágio, isto é, no operatório formal, os sujeitos questionam o porquê de uma certa diferença de inclinação do taco corresponde, necessariamente, a uma diferença no trajeto; vêem necessidade de encontrar um fator que exprima a razão das relações constantes; descobrem a lei de igualdade dos ângulos, capaz de explicar as correspondências e verificam ou rejeitam hipóteses específicas. As hipóteses características desse nível, estão ainda muito perto de correspondências concretas. Os sujeitos procuram exprimir apenas o seu fator geral, que seja capaz de exprimir, além de relações constantes, a razão de tais relações; buscam uma razão para as inclinações observadas e, finalmente, descobrem a igualdade dos ângulos. Os **objetos** são: taco, tela de projeção, bola (lançada), alvo, ponto de rebatimento, ângulos. As **propriedades de objetos** são: força de lançamento da bola (suave, violento) e inclinação (ângulo) do taco. A **relação entre os objetos** (lógico-matemática): inclinação (maior, menor) do taco na tela de projeção e a distância (mais longe ou mais perto) da bola. Os **eventos de propriedades e de relações**: o sujeito faz a experiência verificando, em primeiro lugar, o ângulo reto, isto é, inclinação nula (x) e o ângulo de incidência e de reflexão são também iguais a zero. Depois ele passa para um ângulo de 45 graus e, finalmente testa qualquer tipo de ângulo (α), para concluir a lei de igualdade (o ângulo de incidência é igual ao de reflexão). As **proposições de eventos** se referem a seguinte questão: o aumento do ângulo de incidência implica no aumento do ângulo de reflexão: $p \Rightarrow q$. Portanto: $q \Rightarrow p$ e, vice-versa: $p \Leftrightarrow q$. Levando em consideração a hipótese de que a inclinação é nula (x): $x = 0$, então: $p = 0$ e $q = 0$ e, se $x = \alpha$ (qualquer inclinação) então $y = \alpha$ então, ou $(p \cdot q)$ ou $(-p \cdot -q)$ então $p = q$ (igualdade dos ângulos). O sujeito encontra correspondência concreta entre as inclinações (com a testagem do ângulo nulo, onde a bola volta exatamente ao seu ponto de partida, 45 graus e, finalmente, qualquer ângulo), buscando uma hipótese geral, capaz de explicar essas correspondências. Para o sujeito, o ângulo reto é composto de dois ângulos iguais (incidência e reflexão). Encontra um fator necessário capaz de explicar a razão entre as relações realizadas, isto é, a relação de igualdade entre os ângulos de incidência e reflexão, generalizando-a. Os sujeitos deste estágio se colocam imediata e simultaneamente do ponto de vista do conjunto dos possíveis e das ligações necessárias. Isso se explica porque possuem operações, ao mesmo tempo, combinatórias e suscetíveis de assegurar a necessidade dedutiva (**mecanismo da experiência**). As **proposições** são: o ângulo de incidência e, o ângulo

de reflexão. As **operações sobre proposições** são: a implicação recíproca (\Leftrightarrow) ou equivalência (\equiv). **O que o sujeito atingiu à nível da teoria física que apoia a experiência**: a descoberta da igualdade é o resultado da implicação recíproca, entre as inclinações correspondentes, e não o inverso. Esta implicação recíproca difere da simples correspondência concreta, na medida em que resulta de um cálculo dos possíveis e não apenas de verificações reais.

No operatório formal o sujeito é capaz de prever todas as relações que poderiam ser válidas e logo procura determinar por experimentação e análise, quais dessas relações possíveis têm validade real. Em lugar de limitar-se a organizar o que lhe chega através dos sentidos, como foi visto no estágio pré-operacional, o sujeito tem a capacidade potencial de imaginar o que poderia estar ali. Em relação ao estágio operatório concreto, os objetos, as suas propriedades e relações manipuladas pelo sujeito em seu raciocínio, já não são dados concretos, mas enunciados ou proposições que contém esses elementos. Por isso, o pensamento formal é sobretudo proposicional e combinatório.

O sujeito, neste último estágio, consegue tomar consciência do conjunto de todos os elementos que estão envolvidos na experiência e, em consequência, sua ação é diferenciada das ações percebidas anteriormente.

Utilizando a sistematização do método anteriormente apresentada, é possível demonstrar a evolução do pensamento do sujeito durante esta experiência nos estágios descritos, conforme as figuras 4, 5 e 6.

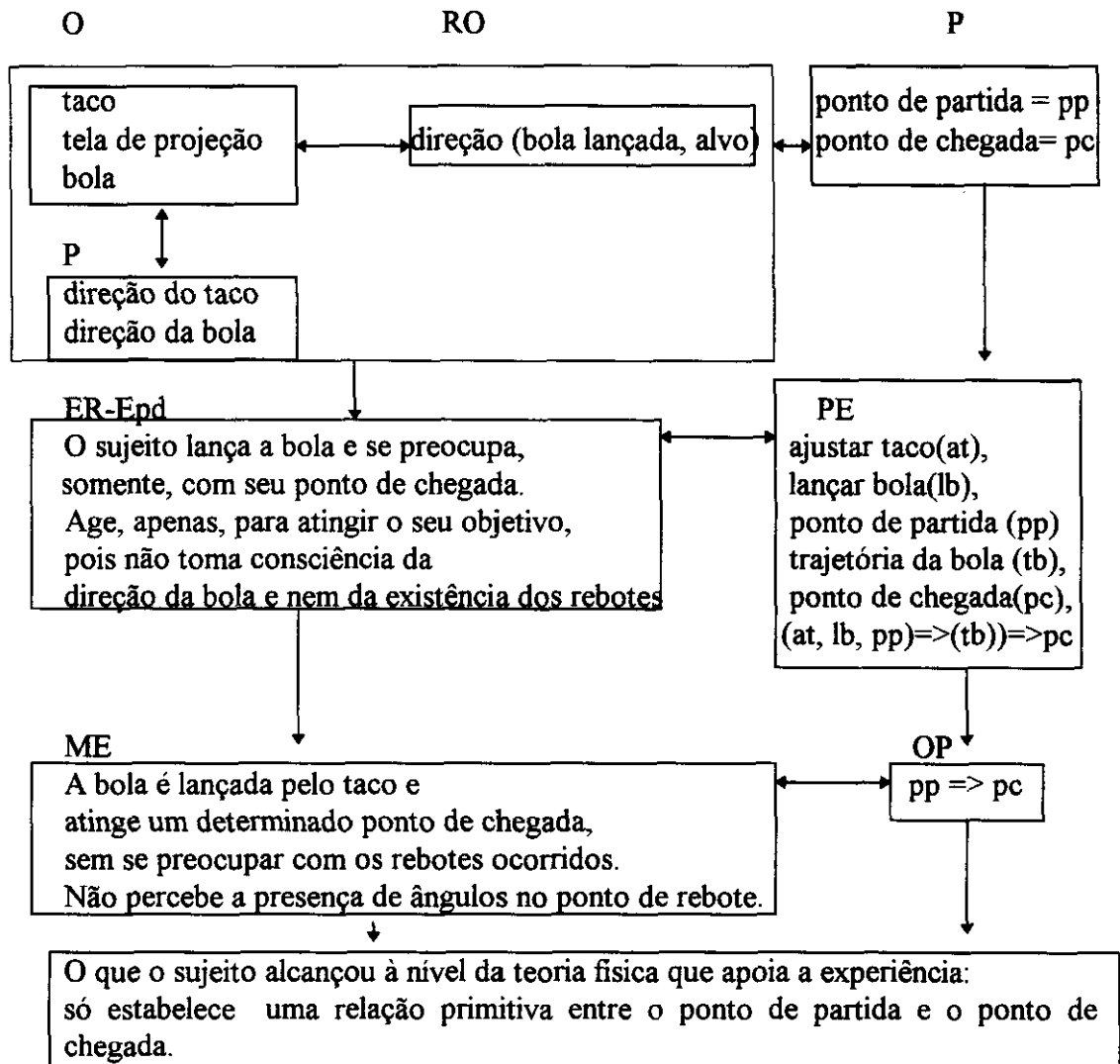
Estágio pré-operacional:

FIGURA 4 - Aplicação do método na experiência, relativa ao estágio pré-operatório

Estágio operatório concreto:

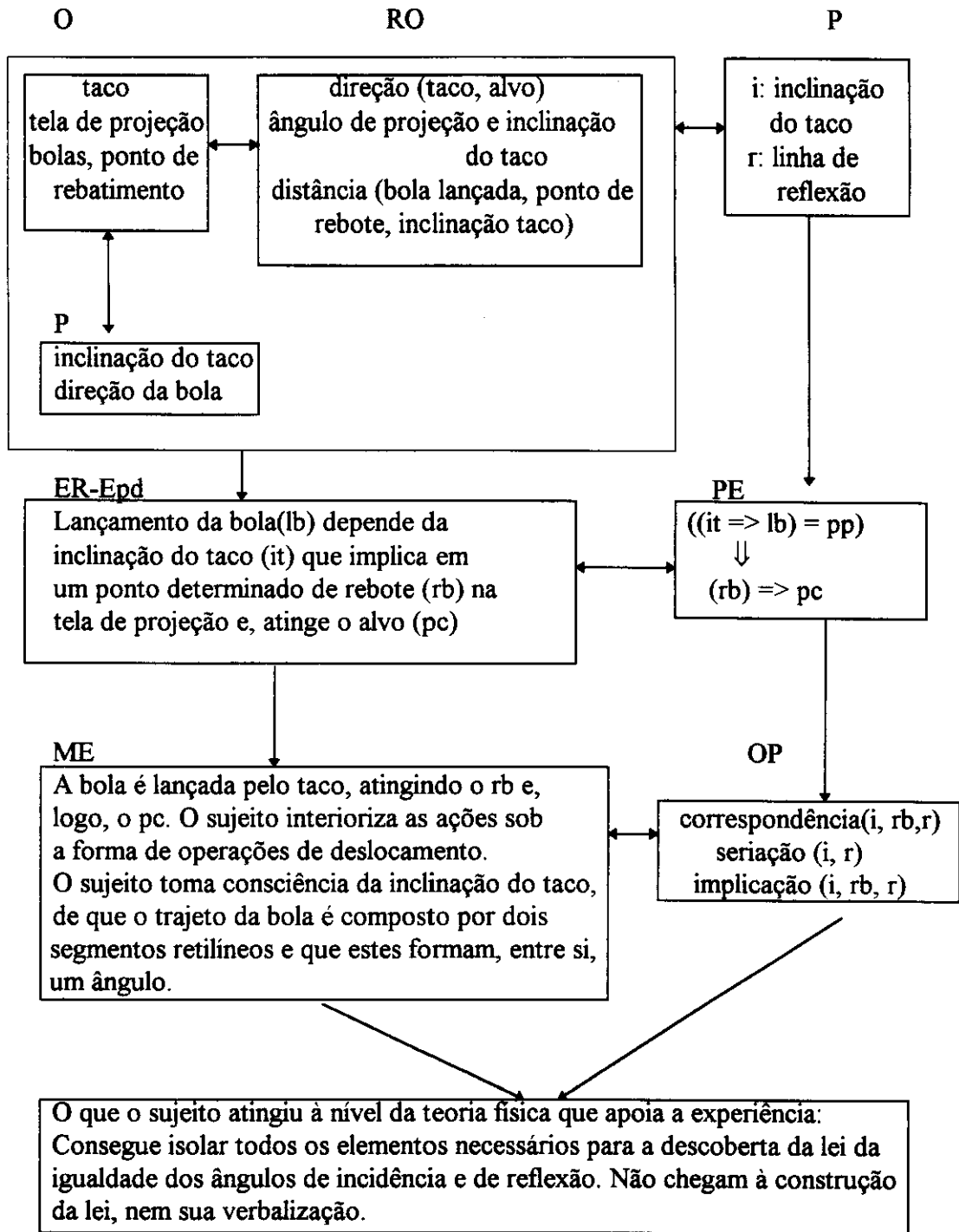


FIGURA 5- Aplicação do método na experiência, relativa ao estágio operatório concreto

Estágio operatório formal:

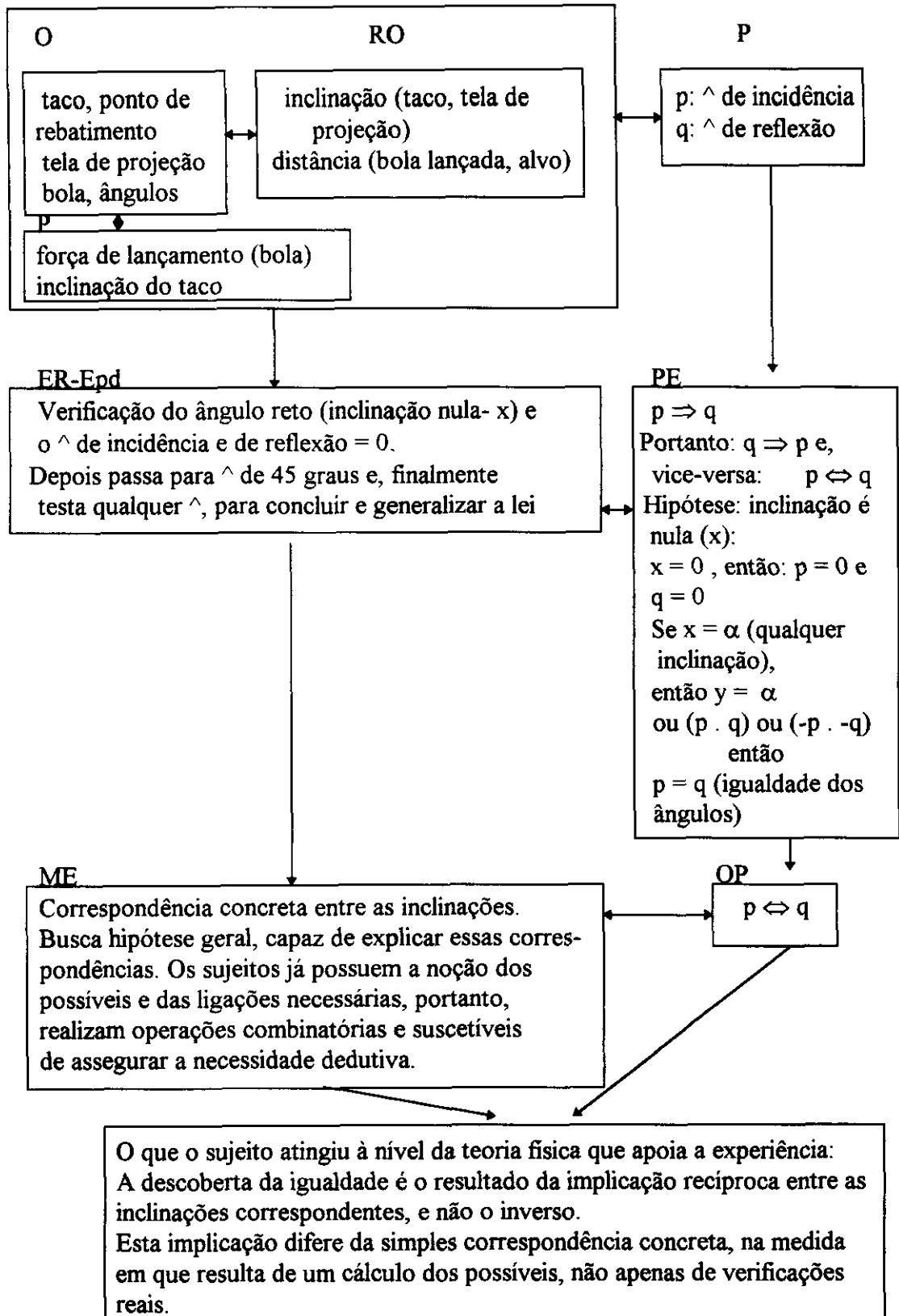


FIGURA 6 - Aplicação do método na experiência, relativa ao estágio operatório formal

Utilizando este estudo como base para realizar o estudo das operações lógicas desenvolvidas pelo sujeito em relação a uma experiência qualquer, transpomos estes

conceitos para fazer o estudo operatório em relação ao objeto, mais especificamente, à ferramenta computacional. O objetivo final é descrever o objeto em questão, em termos da lógica-operatória.

3.3 Conceitos para o estudo operatório de ferramentas computacionais de uso individual

No que se refere à análise operatória das ferramentas, este estudo se propõe a mostrar um dos caminhos encontrados quanto à forma de como elas podem ser vistas em relação às operações lógicas e infralógicas.

Para mostrar quais são os elementos envolvidos em uma interação entre um sujeito e uma ferramenta, é preciso se transportar para a *semiótica*, como descrito em [BEH 96b].

Segundo Rocha Costa [COS 83], é possível realizar uma análise *semiótica* dos processos e sistemas computacionais. A *semiótica* é o estudo dos sistemas de signos e, estes são elementos capazes de representar “algo” de diferentes formas. Quando um sujeito percebe o significante de um signo, atribui-lhe um significado e relaciona esse significado a um significante. E, quando este faz a transmissão desse significado a outro sujeito, ele estará fazendo uma *comunicação*. Portanto a linguagem é um sistema de signos utilizados em um processo de comunicação. Mas, para existir uma comunicação, deve haver, no mínimo, um emissor (um sujeito que vai transmitir) e um receptor (outro que vai receber), ou seja, estes podem ser chamados de “usuários da linguagem” [COS 83].

Como pode ser visto, então, existem três componentes fundamentais em um processo de comunicação deste tipo: uma *linguagem* (instrumento pelo qual os sujeitos ou, melhor dito, os usuários trocam informações sobre os objetos), um conjunto de *usuários* e um conjunto de *objetos* que se referem ao conteúdo da comunicação entre usuários.

Estes mesmos conceitos, podem ser levados para a Ciência da Computação, no que se refere ao tipo de “comunicação” que é mantida em um ambiente computacional. Portanto, fala-se dos elementos envolvidos na interação de um sujeito com um máquina. Neste caso, os usuários da linguagem são o *programador* (quem escreve o programa) e a *máquina* (que executa o programa). A linguagem para manter a comunicação é a *linguagem de programação* utilizada na escrita. E, o objeto que é transmitido entre os usuários que fazem parte do ambiente é a *computação* idealizada pelo programador e representada por ele no programa em forma de “*uma seqüência de operações realizadas sobre um conjunto de elementos denominados dados*” [COS 83].

Assim, a execução de uma computação por uma máquina, é mediada por relações semióticas que têm por centro a linguagem de programação.

É por essa razão que esta área de pesquisa envolve a função semiótica como uma das formas do sujeito fazer as suas representações através do ambiente computacional.

A interação deste estudo é analisada de acordo com os elementos apresentados na figura 7. Esta pode ser interpretada da seguinte forma, como descrito em [BEH 97b]: um sujeito qualquer (S) utiliza uma ferramenta computacional (FC) para manipular a sua representação (RC) que diz respeito a um determinado valor, objeto e/ou linguagem de estudo (V,O,L). A composição da ferramenta e da representação forma o computador (C).

Estes elementos deram origem aos elementos necessários para a construção do modelo de interação que é utilizado para a realização da análise operatória de ferramentas computacionais de uso individual e coletivo.

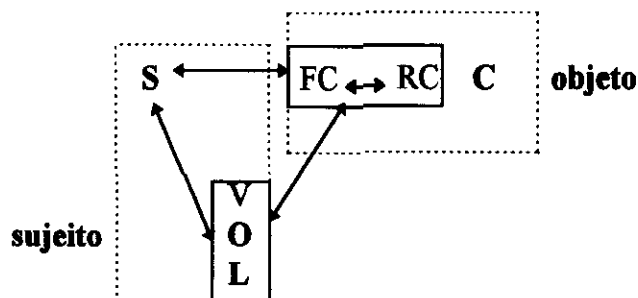


FIGURA 7 - Interação individual apoiada por computador

A seguir são apresentados outros conceitos necessários para o estudo operatório de ferramentas computacionais interativas.

Pode-se observar na figura acima que na interação sujeito x ambiente computacional, existem dois tipos de estruturas: a do **sujeito** e a do **objeto**. A seguir iremos caracterizar o modelo referente à estrutura do sujeito para depois construir o modelo do objeto e, por último juntar os dois (sujeito + objeto) num único modelo.

3.3.1 Caracterização do sujeito individual

Neste trabalho foi proposto um esquema simplificador em relação à caracterização do sujeito apresentado em [DOL 93], apoiado na intuição e experiência da autora [BEH 96b]. Mesmo faltando provas que justifiquem mais detalhadamente esta escolha, as noções descritas por Dolle [DOL 93] são analisadas de acordo com um novo enfoque, utilizando como base uma das obras de Piaget [PIA 73]. Dessa forma foi delimitado um campo de estudo factível, fazendo um corte de um setor do real psicológico em função de uma determinada idéia. Para isso, é necessário ter um referencial teórico que estabeleça um contexto adequado desenvolvendo uma nova via de pesquisa [BEH 97c].

Utilizando como base os estudos realizados em relação ao sujeito individual, vamos dizer que um sujeito é composto por estruturas afetivas, cognitivas e simbólicas. Isto é, levamos em conta o fator mental de um sujeito que é formado, segundo Piaget [PIA 73], por três aspectos indissociáveis: o estrutural (cognitivo), o energético (afetivo) e os sistemas de símbolos (simbólico), servindo de significantes a estas estruturas operatórias ou a estes valores individuais [RAM 96].

As estruturas afetivas dizem respeito aos valores do sujeito. Por exemplo, gostar ou não, interessa ou não, é importante isso ou aquilo, etc. As cognitivas se referem ao objeto em si, ou seja, são as responsáveis pelas operações realizadas em relação aos objetos, por exemplo, classificações, medições, seriações, soma, subtrações, etc. E, finalmente, as estruturas simbólicas, são as que dão significado representativo aos objetos, utilizando, para isso, os sinais, isto é, a linguagem. Inerentes a estas estruturas também temos a imagem mental do sujeito, que não será apresentada esquematicamente. Este modelo pode ser visto na figura 8.

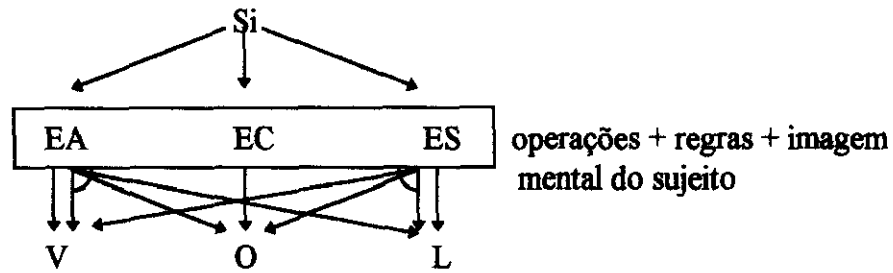


FIGURA 8 - Estruturas que compõem um sujeito individual

onde: Si: sujeito individual; EA: estrutura afetiva; EC: estrutura cognitiva; ES: estrutura simbólica; V: valores; O: objetos; L: linguagem.

Portanto, a partir da figura acima, podemos dizer que as estruturas afetivas se referem aos valores (EA (V)) e é de acordo com estes que o sujeito opera com objetos (EA (V,O)) e também a expressão de valores através de idéias (EA (V,L)). As estruturas cognitivas podem operar em relação aos objetos (EC (O)) independentemente dos valores e da linguagem. As estruturas simbólicas se referem à linguagem em si do sujeito (ES (L)), e a que relaciona esta com os objetos (ES (L,O)) e com os valores ((ES (L,V))).

Também é preciso esclarecer um novo conceito que, na nossa interpretação, não se refere somente ao fator social [PIA 73]. Estamos nos referindo à noção de *regra* que é definida de diversas formas segundo Abbagnano [ABB 87]. Aqui a utilizaremos, como sendo um *procedimento*, isto é, *uma seqüência de ações que devem ser feitas pelo sujeito relativa a fins específicos*. Usando termos computacionais, definimos a regra como “um programa, uma receita que mostra os diferentes meios com que se pode atingir um fim determinado”. Podemos dizer, então, que além das operações realizadas em relação aos valores, objetos e/ou linguagem, as regras estão também inseridas nas estruturas afetivas, cognitivas e simbólicas do sujeito individual. Portanto, estas últimas se referem à coordenação das ações em relação aos valores, objetos e à linguagem. Assim, a notação das estruturas mostradas acima ficaria da seguinte forma:

EA (opV, rV); EC (opO, rO); ES (opL, rL), onde:

opV: operações sobre os valores (V); opO: operações sobre os objetos (O); opL: operações sobre a linguagem (L); rV regras que coordenam (organizam) as operações sobre valores; rO: regras que coordenam as operações sobre os objetos; rL: regras que coordenam as operações sobre a linguagem.

Cabe enfatizar ainda que, além destas três estruturas internas do sujeito, existe o objeto de estudo com o qual este interage (no nível individual). Portanto, como este objeto possui sua própria estrutura funcional, operacional, material, etc...(no caso de ser uma ferramenta computacional), ele também têm suas próprias operações para manipulá-lo e suas próprias regras de funcionamento. Ou seja, estamos nos referindo a um objeto que têm uma estrutura externa similar à estrutura interna do sujeito, mas neste caso, falamos do objeto que faz parte do processo interativo. Por exemplo, se um sujeito consegue trabalhar bem com um objeto qualquer, isso quer dizer que a sua estrutura (operações e regras) é compatível com a estrutura do próprio objeto de manipulação.

No caso de utilizar os valores individuais para se referir a um objeto EA (V,O), a notação ficaria da seguinte forma: EA (opVO, rVO). Ou seja, o sujeito coordena as operações sobre valores e objetos (OpV + OpO = OpVO), através de regras sobre

valores e objetos ($rV + rO = rVO$). Ex: Gostar ou não de um carro, isto é, diz respeito à valores sobre objetos, combinando os dois tipos de estruturas.

Portanto o modelo do sujeito individual $S(x)$ em interação com um determinado ambiente computacional pode ser visto da seguinte forma na figura 9:

valores, objetos e linguagem individual

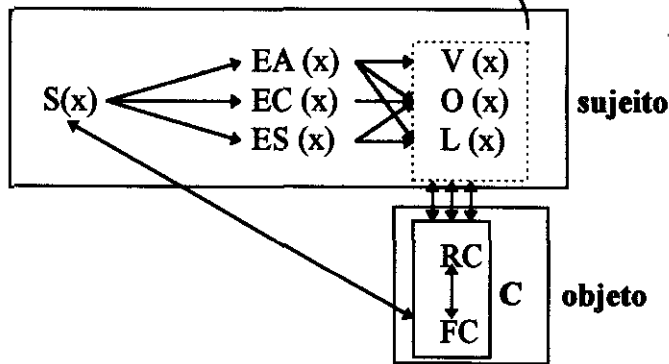


FIGURA 9 - Elementos que compõem a interação sujeito x ambiente computacional

Quando o sujeito tem que utilizar algum tipo de ferramenta computacional para representar qualquer coisa, ele é levado a pensar sobre o seu pensar para, então, poder transcrever ou manifestar as suas idéias. No momento em que este tem que expressar de forma escrita ou figurada o seu pensamento, ele pode refletir sobre o mesmo e, muitas vezes até reestruturá-lo, construir ou reconstruir sua imagem mental. Este processo pode levar o sujeito à construção de novos conhecimentos, mas tudo irá depender da maneira como o sujeito se relaciona com o ambiente. É por essa razão que utilizamos a seta bidirecional que liga os valores, objetos e linguagem ao computador. Como a imagem mental do sujeito é inerente às estruturas, através da interação com a ferramenta e/ou com a representação computacional, essa imagem pode ser alterada constantemente.

3.3.2 Formas de interação do sujeito com ferramentas computacionais

De acordo com o nosso ponto de vista, o sujeito pode interagir de três formas distintas com um ambiente computacional. Ou seja, não é possível analisar as operações realizadas pelo sujeito em interação com um programa, ou com a sua imagem gráfica/textual representada no computador, como uma coisa única. É preciso detalhar as diferentes formas de trabalho decorrentes deste tipo de relação. Neste estudo se utilizará o exemplo de um editor gráfico para explicar estas formas de interação. Por exemplo:

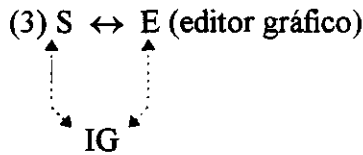
(1) S (sujeito) \leftrightarrow E (editor gráfico)

Esta relação refere-se às operações que o sujeito realiza no Editor Gráfico, referindo-se, exclusivamente, às formas de manipular e controlar o sistema, seja esta de forma direta ou indireta.

(2) $S \leftrightarrow$ IG (operações do sujeito realizadas diretamente sobre a imagem gráfica desenvolvida)

Esta relação envolve as operações que o sujeito faz sobre o seu desenho, independentemente das funções pré-definidas do editor. Ou seja, ele trabalha sobre a

imagem gráfica, utilizando as operações do editor que dependem da ação direta do usuário sobre este, sem intermediação dos comandos do programa.



Esta forma é utilizada pelo sujeito, interagindo com a imagem gráfica, mas por intermédio dos comandos do editor gráfico. Ou seja, o sujeito não age diretamente sobre o objeto de sua ação e sim, através das operações do editor.

Isto ilustra que não existe uma única forma de interação do sujeito com uma ferramenta computacional (ex: editor gráfico).

3.3.3 Sujeito (usuário/programador) e o modelo mental do funcionamento do computador

Os sujeitos, quando interagem com um ambiente, com outros indivíduos e com artefatos da tecnologia, formam um modelo mental de si próprios, das pessoas e das “coisas” com as quais estão interagindo. Estes modelos provêm poder de entendimento, capacitando as pessoas a prever e explicar aspectos de interação [ROC 93]. Ao mesmo tempo, estes modelos são modificados pela própria interação. Nesta abordagem é utilizado o termo “Modelo Mental do Funcionamento do Computador” (MMFC), à representação de objetos e suas respectivas operações, quando o sujeito interage com o computador. Por essa razão, é necessário esclarecer qual é o sujeito (S) ao qual estamos nos referindo ao longo deste estudo.

Se por um lado, o sujeito realiza operações espaciais sobre os “objetos reais” com os quais interage, é preciso entender qual a ação que este realiza quando se trata do computador, ou melhor dito, de um software em particular, ou seja, um “objeto” que possui a sua própria estrutura funcional e operacional.

O que é, então, uma “*operação*” (tanto do sujeito quanto da máquina)?

As operações se referem às transformações que ocorrem tanto a nível de processamento interno da máquina quanto aquelas que envolvem interações do sujeito com a ferramenta computacional. Portanto, como se trata de uma interação (S ↔ O (computador)), podem existir *operações realizadas pelo sujeito em relação à máquina e a ação que a máquina exerce sobre o sujeito*, e que podem modificar o estado desta e/ou do sujeito.

Chamamos de *S usuário* (Su), ao S que não interfere na programação do computador, isto é, ao indivíduo que interage com a máquina “externamente”. Assim, este S não precisa conhecer a estrutura material, operacional e funcional da máquina. Ele simplesmente opera e atua com o software, no caso do exemplo anterior, com o editor gráfico que lhe é oferecido, sem interferir na sua programação.

Por outro lado, existe o *S programador* (Sp). Este precisa se colocar nos dois pontos de vista, tanto do S usuário, quanto do objeto, ou seja, da estrutura da máquina. Ele deve possuir o conhecimento da lógica do computador para, então, poder programar e, ainda, tem que se colocar como S usuário para entender como o software interagiria com o mesmo. Assim, este S atua “internamente” à máquina.

Vamos supor agora um ambiente computacional heurístico, onde o sujeito é continuamente desequilibrado e levado a resolver os desafios propostos. Para isso, ele

terá que compor as heurísticas (método analítico para descobrir a verdade científica), através dos seus objetivos, das suas intenções, fins, percorrendo os determinados caminhos, buscando diferentes estratégias, a fim de solucionar o problema proposto. Esta ação está relacionada diretamente ao funcionamento do computador, ao software em execução. Nesse sentido, o computador funciona como uma máquina extremamente opaca, portanto não é visível, ou seja, não se pode abrir um computador e observar o que acontece quando um programa está sendo executado (no caso do S usuário). A forma então, que se tem para tentar explicar este processo, é através de representações, isto é, modelos, que de alguma forma espelhem o funcionamento, a sua lógica. Permitindo esta “visibilidade” do processo de execução, o usuário obtêm um melhor “feedback” de sua interação, o que poderá auxiliar no processo evolutivo de construção do seu modelo mental do funcionamento do computador.

Para construir então, esse modelo mental do funcionamento do computador, deve-se levar em conta o estágio de desenvolvimento cognitivo do indivíduo, as condutas resultantes da interação com o computador e, ainda, entender a forma como ele domina e opera a máquina, as leis, normas ou regras que segue. Portanto, cada sujeito possui um modelo mental (representação dos objetos + representação das operações) em relação à forma como funciona o computador, independentes de serem usuários e/ou programadores. E, é de acordo com este modelo que eles conseguem interagir com a máquina.

Portanto, neste trabalho é diferenciado o elemento sujeito da interação de acordo com o seu tipo, usuário ou programador.

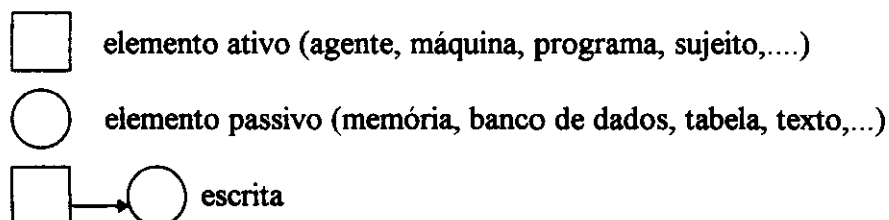
A seguir é descrito o modelo de interação do sujeito para o estudo operatório de ferramentas computacionais interativas, à nível individual e, posteriormente, à nível coletivo.

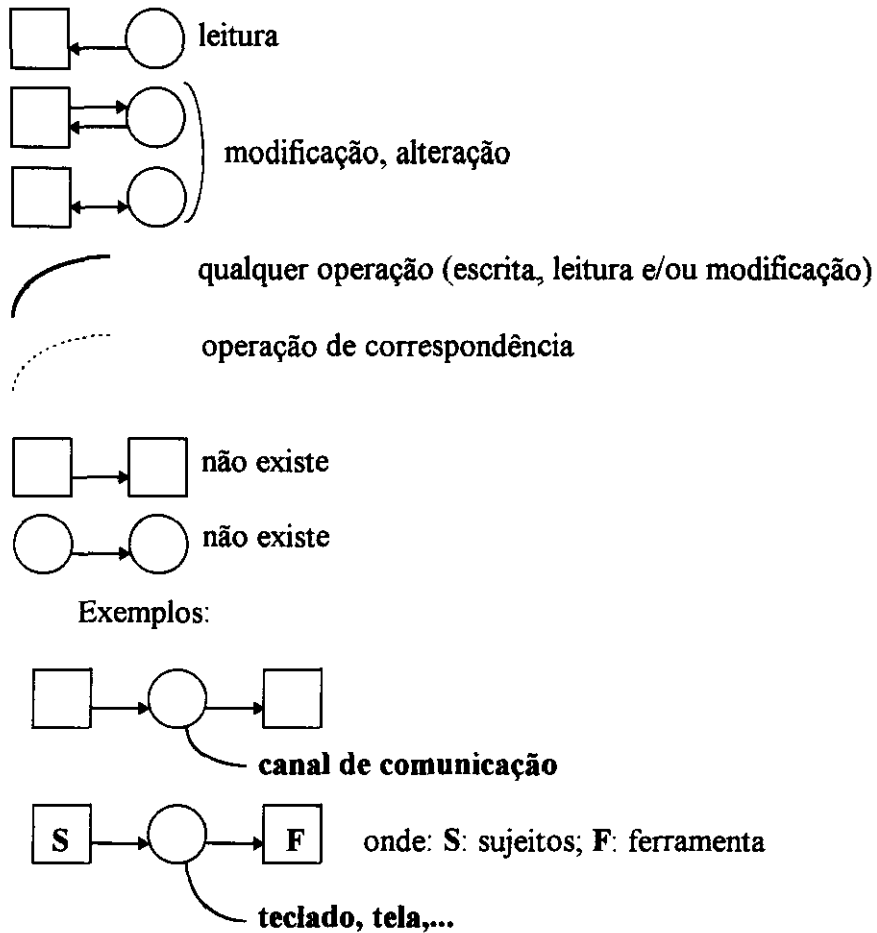
3.4 Modelo geral de interação de um sujeito com uma ferramenta computacional de uso individual

Levando em conta os elementos, as funções e operações mais importantes envolvidas em um processo de interação, foi criado um *modelo geral de interação* do sujeito com um sistema computacional, no qual será baseada a análise lógico-operatória das ferramentas computacionais, no nível individual.

O construtivismo piagetiano explica os processos de desenvolvimento e aprendizagem do sujeito como resultados da atividade do mesmo na interação com o ambiente. Por essa razão, neste modelo foi preciso levar em conta os diversos tipos de relações que podem ocorrer entre um sujeito (seja este usuário e/ou programador) e um sistema computacional.

Para isso, os elementos envolvidos no modelo foram padronizados da seguinte forma, de acordo com a notação gráfica usual para as redes de Petri [WEN 76] para facilitar a interpretação do mesmo:





O sujeito pode operar de forma direta ou indireta, em relação ao ambiente e à atividade que está sendo desenvolvida por ele. Isto quer dizer que existem, basicamente, quatro tipos de operações que podem ser executadas por ele em relação à ferramenta computacional:

1. **Controle Direto do Sistema (CDS)**: neste item são descritas as operações que dizem respeito à manipulação direta do sistema, isto é, os comandos que afetam o sistema através da ação do usuário;
 2. **Controle Indireto do Sistema (CIS)**: o ambiente é modificado pela ação do usuário, através de ações realizadas pelo sistema;
 3. **Manipulação Direta sobre a Representação (MDR)**: nesta forma de interação, o usuário opera diretamente sobre a representação, isto é, ele executa uma ação manipulando diretamente as ferramentas que afetam o objeto.
 4. **Manipulação Indireta sobre a Representação (MIR)**: esta se refere à execução de comandos do sistema, ativados pelo usuário, que alteram o "objeto".
- Estas operações são identificadas no modelo através da seguinte notação:

→ : operação direta; → : operação indireta

As operações realizadas pelo sujeito podem ser simples ou compostas, ou seja, uma ou mais de uma operação realizada em conjunto ou, ainda, uma seqüência de comandos.

A seguir serão exemplificados na tabela 1 os processos interativos do sujeito em relação à ferramenta ou a representação, utilizando o editor gráfico Painbrush. A análise detalhada deste editor encontra-se no capítulo 5.1.

TABELA 1- Operações realizadas pelo sujeito em interação com o editor gráfico Painbrush

	INTERAÇÃO COM A FERRAMENTA	INTERAÇÃO COM O DESENHO
CONTROLE DIRETO (CDS) ou MANIPULAÇÃO DIRETA (MDR)	O usuário interage através de operações diretas com o ambiente, isto é, executa ações sobre o mesmo, afetando o sistema. Ex: muda o formato das janelas do editor. Quem define o tamanho das janelas da interface é o sujeito e não o programa.	O sujeito interage diretamente, executando ações sobre o desenho, isto é, afetando não o ambiente, mas a imagem. Ex: o usuário modifica o desenho ativando o spray da caixa de ferramentas do editor ou, através da borracha, apaga parte do desenho, modificando-o
CONTROLE INDIRETO (CIS) ou MANIPULAÇÃO INDIRETA (MIR)	O usuário interage indiretamente com o sistema, ou seja, executa comandos pré-definidos que se referem somente ao ambiente. Ex: o sujeito ativa o comando SALVAR, do menu ARQUIVO para armazenar o seu desenho. Quem faz a operação de salvar é o editor.	O usuário executa comandos do ambiente que mudam a imagem, ou seja, além dele executar ações sobre o ambiente necessita das operações pré-definidas do mesmo para alterar o desenho. Ex: o sujeito ativa o comando ROTAÇÃO VERTICAL, do menu MODIFICAR para colocar uma figura na posição correta. Quem faz esta ação é o editor e não o usuário

A partir destas formas de interação com o ambiente computacional, é possível construir o modelo geral no nível individual, representado na figura 10:

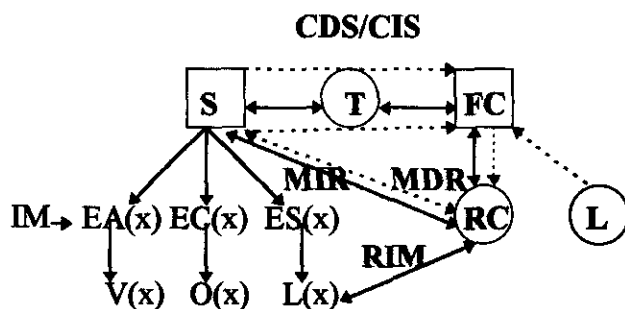


FIGURA 10 - Modelo geral de interação sujeito x ambiente computacional

onde: S: sujeito (Sujeito Usuário); FC: ferramenta computacional; L: linguagem utilizada para a representação computacional; RC: representação computacional (valor, objeto e/ou linguagem

representada em forma de imagem gráfica textual/figural); RIM: representação da imagem mental; IM: imagem mental; CDS: controle direto do sistema; CIS: controle indireto do sistema; MDR: manipulação direta da representação; MIR: manipulação indireta da representação; T: canal de comunicação: tela, teclado,...

Portanto, este modelo pode ser entendido da seguinte forma:

- Baseando-se na imagem mental (IM) que o sujeito (S) tem em relação a “algo” que deseja representar, sejam valores, objetos e/ou uma determinada linguagem, ele utiliza uma linguagem (L) para representação de sua IM. Esta linguagem, que pode ser a sua linguagem natural (de desenho ou de escrita) ou uma linguagem computacional, permite a representação deste “algo” no computador (RC). Para realizar essa representação, o sujeito utiliza a ferramenta computacional (FC) manipulada através do teclado, tela, etc.

Este é um modelo geral que foi construído como uma das formas encontradas para explicar o processo de interação de um sujeito e suas respectivas estruturas com uma ferramenta computacional qualquer, no plano individual. A seguir é introduzido o nível coletivo utilizando como base os conceitos abordados nas seções anteriores.

4 Nível Coletivo

4.1 Teoria do sujeito coletivo-Lógica operatória e cooperação

Os sujeitos que trabalham em um grupo de forma cooperativa, representando informações, são levados à refletir sobre o pensamento dos outros, respeitando-se, ajudando-se, trocando e aceitando idéias. O grupo deve ser aberto, flexível, constituído sobre a motivação e os interesses dos integrantes do mesmo, para que os sujeitos possam desenvolver dinamicamente, as suas estruturas cognitivas tanto individuais quanto coletivas. Portanto, o ambiente cooperativo possibilita a confrontação de visões diferentes entre os integrantes de um grupo.

A relação de um indivíduo com outros incide de forma decisiva sobre o processo de socialização, para a aquisição de aptidões e de habilidades, através do desenvolvimento das estruturas intelectuais. Esse é um dos principais fatores que levam ao progresso intelectual de um sujeito. Ou seja, a partir disso, pode ocorrer a reflexão, a construção ou “reconstrução” do próprio pensamento do sujeito.

Através das redes de computadores, o suporte à comunicação está sendo cada vez mais consolidado, resultando no desenvolvimento de aplicações que possibilitam o trabalho e a produção em conjunto.

A representação de conhecimentos de diversos participantes de uma equipe, utilizando uma determinada ferramenta computacional, permite que estes se comuniquem, trocando informações e que interajam com outras organizações, ou seja, outras equipes de trabalho, “crescendo” em conjunto.

Essa é uma das razões que nos levou à necessidade de distinguir o uso de ambientes computacionais individuais dos coletivos, analisando-os do ponto de vista da lógica operatória.

Para elaborar as operações descritas em 3.1.3, as construções e a descentração cognitivas são ligadas às construções e descentração afetiva e social. Assim, pode-se observar que as relações interindividuais têm uma importância muito significativa, e estas fazem com que as operações lógicas sejam, acima de tudo, *co-operações*.

As operações lógicas que se formam durante o processo de desenvolvimento cognitivo, têm como modelo as operações lógico-matemáticas que se organizam como estruturas mentais. Estas são denominadas, por Piaget, no estágio concreto, de *agrupamentos*, por terem, por um lado, uma similaridade à estrutura matemática de grupo e, por outro lado, ao reticulado.

As operações lógico-matemáticas desenvolvem-se durante os estádios concreto e formal. Neste último é que aparecem as operações proposicionais, o pensamento hipotético-dedutivo, e as transformações sobre essas operações, o denominado grupo INRC. Existe, portanto, uma estreita ligação dos níveis de desenvolvimento cognitivo do sujeito com as suas condutas que, neste caso, se referem somente às operações lógicas do indivíduo.

No nível da construção dos agrupamentos, no estágio concreto e depois, no nível formal, aparece o problema dos papéis do intercâmbio social e das estruturas individuais no desenvolvimento do pensamento. A lógica verdadeira, constituída nesses dois períodos, é acompanhada de duas espécies de caracteres sociais.

Por um lado, a medida que as intuições da criança se articulam e acabam por se agruparem operatoriamente, esta se torna cada vez mais apta para a cooperação, relação social distinta da influência, porque supõe uma reciprocidade entre indivíduos. Segundo Piaget, a cooperação significa a discussão dirigida objetivamente, a colaboração no trabalho, a troca de idéias, o controle mútuo, etc. A cooperação é, portanto, o ponto de partida de uma série de atitudes importantes para a constituição e desenvolvimento da lógica.

Por outro lado, a própria lógica é um conjunto de estados de consciência, de sentimentos intelectuais e de atitudes. Diz-se então, que a lógica comporta regras ou normas comuns, impostas e sancionadas por outros, é uma moral do pensamento. Portanto, a objetividade, a necessidade de verificação, o imperativo de conservar-se o sentido das palavras e das idéias são, tanto obrigações sociais como condição do pensamento operatorio.

É necessário portanto, esclarecer se o *agrupamento* é a causa ou o efeito desta cooperação. Esta estrutura constitui a coordenação das operações do sujeito. A cooperação é, então, "*uma coordenação de pontos de vista ou de ações, que emanam, de diferentes indivíduos*" [PIA 73].

O indivíduo, então, só consegue agrupar suas ações em um todo coerente, quando é capaz de permutar o seu pensamento e de cooperar com outros. Portanto, o agrupamento operatorio supõe uma vida social. Ele também é uma forma de equilíbrio tanto das ações interindividuais como das ações individuais, e encontra sua autonomia na vida social, pois os intercâmbios do pensamento obedecem uma lei de equilíbrio que, se constitui em um agrupamento operatorio. Assim, o significado da cooperação é *coordenar operações*.

Portanto, para elaborar as operações lógicas é preciso passar por um processo de descentralização na área afetiva e social. O que permite descentralizar-se é a permuta constante de idéias com os outros, assegurando a possibilidade de coordenar interiormente as relações provindas de pontos de vista distintos. A reversibilidade do pensamento encontra-se ligada a uma conservação coletiva. E, visto que um pensamento lógico é um pensamento social, é fácil observar que as leis de agrupamento constituem formas de equilíbrio geral. Estas refletem o equilíbrio das trocas interindividuais e as operações que o indivíduo socializado é capaz de raciocinar interiormente, em relação às novas idéias.

4.1.1 Construção coletiva de conhecimentos

Algumas questões que cercam os que procuram entender as implicações das relações sociais na cognição humana, são as seguintes:

- como a relação social interfere no desenvolvimento cognitivo;
- entender se o desenvolvimento das estruturas da inteligência decorrem de ações individuais, da natureza social ou das duas ao mesmo tempo;
- se o uso de ferramentas computacionais cooperativas que proporcionam a comunicação entre os sujeitos serve só para a troca de informações ou pode servir como suporte no desenvolvimento cognitivo individual e/ou coletivo, entre outras;
- durante a comunicação de dois sujeitos (interações interindividuais), o que pode e deve ser analisado que evidencia o desenvolvimento cognitivo do ponto de vista social da cooperação;

Para isso, tem que se compreender como se dá o processo de construção e de articulação do pensamento do sujeito quando este interage com outros e quais são as características que deve ter uma ferramenta computacional que auxilie este tipo de processo [PIA 73].

Mas quando e como será que um sujeito chega às interações interindividuais?

O ser humano (desde que nasce) está submerso em um meio social que atua sobre ele. Ou seja, do nascimento à vida adulta o sujeito é objeto de pressões sociais. Segundo o nível de desenvolvimento individual, os intercâmbios com o meio social são de natureza muito diversa, modificando a estrutura mental e individual.

Durante o período sensório-motor o bebê é objeto de múltiplas influências sociais, mas ainda não existe troca de pensamento, visto que a criança ignora o pensamento. Portanto, não existe nenhuma modificação profunda nas estruturas sensório-motoras através da vida social ambiente.

Já com a aquisição da linguagem (período simbólico e intuitivo) são introduzidas novas relações sociais que aparecem enriquecendo e transformando o pensamento do sujeito. Posteriormente, nos níveis pré-operacional, as estruturas próprias do pensamento nascente resistem a formação das relações sociais de cooperação, que seriam as únicas a acarretar a constituição de uma lógica.

Mas é nos níveis da construção de agrupamentos de operações concretas e sobretudo formais, que existe problema da troca social e das estruturas individuais na construção do pensamento. A medida que o pensamento intuitivo se articula e acaba por se agrupar operatoriamente, o sujeito vai se tornando cada vez mais apto à cooperação; relação social que pressupõe a reciprocidade entre os indivíduos que diferenciam os seus pontos de vista. Ou seja, só o pensamento “verdadeiro” pode ser socializado.

Portanto, cada relação social constitui uma totalidade de características novas, transformando a estrutura mental do sujeito. A interação entre dois sujeitos não é nem a soma dos seus conhecimentos e nem a realidade deles superpostas.

É através da interação interindividual que é possível ocorrer uma construção coletiva de conhecimentos (que dependerá da relação social estabelecida).

Uma das formas de ocorrer construção coletiva de conhecimentos é através da cooperação. O elemento motor da interação social é o mecanismo de *equilíbrio* [PIA 76] que pode solucionar os *conflitos sociocognitivos* (CSC) gerados pelas diferentes perspectivas dos sujeitos frente à solução de um problema. Tanto a *equilíbrio* quanto os *conflitos sociocognitivos* é que mobilizam e forçam as reestruturações intelectuais e, com isso, ocorre o progresso cognitivo dos sujeitos. É o resultado das contradições

internas entre os diversos esquemas dos sujeitos que se produz no decurso da interação social. Daí que vem a denominação CSC, que é uma das variáveis que repercutem sobre o desenvolvimento intelectual.

Para que ocorra um CSC os participantes de uma equipe devem ter certas ferramentas cognitivas à sua disposição, pois o sujeito só poderá tirar proveito de uma interação se ele já estiver apto a estabelecer uma diferença entre o seu ponto de vista e o do outro. No conflito sociocognitivo podem ser identificados os conceitos de **decentração e perturbação**.

A confrontação de perspectivas em uma interação social se constitui da perturbação e da decentração, pela possibilidade da retomada do ponto de vista de cada um dos sujeitos no processo das relações interindividuais. O CSC é um dos fatores que levam à construção coletiva de conhecimentos porque envolve estes dois conceitos e desencadeia o processo de equilíbrio.

A sociogênese (processo de construção das relações sociais) diz que: *“o conhecimento humano é essencialmente coletivo e a vida social constitui um dos fatores essenciais da formação e do crescimento dos conhecimentos dos sujeitos”* [PIA73].

O desenvolvimento intelectual de um sujeito é um processo que se origina das construções operatórias decorrentes de interações entre estes. Essas interações são o resultado do sujeito estar inserido em um sistema de relações com outros sujeitos, objetos de outra natureza. Através destas é que ele constrói seu mundo de significações.

Os fatores sociais de coordenação interindividual das ações permitem a construção de uma lógica: *“as regras da lógica não são impostas pelo grupo social, mas são o resultado da coordenação das ações interindividuais que ocorrem nas atividades em comum e na troca verbal”* [PIA 73].

4.2 Trabalho cooperativo apoiado por computador

O objetivo deste capítulo é caracterizar o trabalho cooperativo e particularizar a cooperação no processo de representações grupais. Ou seja, a importância deste estudo é investigar os diferentes ambientes computacionais que permitem e suportam a construção de um “saber coletivo”. Esta só é possível a partir das reciprocidades, da cooperação e, principalmente, através de uma interação cooperativa contínua entre os sujeitos que realizam uma ação em comum. Assim, o ambiente computacional deve proporcionar meios para a representação coletiva das idéias de uma equipe.

A representação coletiva tanto de idéias, conhecimentos ou informações é o resultado de uma série de ações realizadas por um grupo de sujeitos. Isto é, por trás desta representação existe um trabalho que é fruto da cooperação entre os sujeitos.

A cooperação é um sistema que coordena atividades ou forças de duas ou mais pessoas, devendo existir boa vontade em cooperar, colaborar, habilidade para se comunicar e disposição para discutir e acatar propostas advindas de cada participante do grupo. Esta deve ser eficaz para atingir seus objetivos a partir da organização coletiva e eficiente no sentido de satisfazer idéias individuais e, portanto representações individuais que se tornam coletivas.

O trabalho cooperativo contribui positivamente para as habilidades de convivência social e a auto-estima. Se este segue um enfoque heurístico, enfatiza a descoberta, privilegiando no sujeito a atividade criativa, a exploração e construção individual e coletiva de conhecimentos.

Nestes ambientes podem ocorrer situações de troca de informações e idéias, que é a base da busca de uma solução de um problema ou ainda, do “crescimento” (desenvolvimento cognitivo) de indivíduos e de um grupo. Por exemplo, um sujeito apresenta um texto através de um editor cooperativo. Ao receber este texto (ou mensagem), um outro usuário percebe a ação deste primeiro, isto é, toma consciência do outro, podendo colaborar ou não com sua opinião, apresentando os seus conhecimentos e compartilhando-os através da ferramenta computacional.

As interações interindividuais tratam de uma relação social que constitui uma totalidade de características novas, transformando a estrutura mental do sujeito. A interação entre dois ou mais sujeitos não é nem a soma das suas idéias, de seus conhecimentos, nem a realidade deles superpostas, mas é uma totalidade nova; um sistema de interação que modifica o sujeito na sua estrutura e, portanto, a estrutura do grupo como um todo.

É preciso, em primeiro lugar, introduzir os tipos de ferramentas computacionais que suportam a atividade cooperativa para, depois, definir os elementos necessários para a construção dos modelos mais gerais de interação destas com o sujeito coletivo.

4.2.1 Computação Cooperativa

O tipo de ambiente computacional que pode apoiar a representação coletiva, seja esta através de desenhos ou da escrita, é conhecido como CSCW (Computer Supported Cooperative Work) ou Computação Cooperativa [BAN 91] [ELL 93] [GAL 90] [GEE 91] [GRE 88] [MAL 93].

Neste item são apresentados alguns conceitos gerais em relação às ferramentas que suportam o trabalho em grupo, a fim de esclarecer alguns conceitos que são utilizados nos próximos capítulos.

As aplicações nesta área incluem:

- mecanismos para a comunicação entre os indivíduos do grupo, para que estes possam se ver, ouvir, enviar mensagens uns aos outros;
- mecanismos de compartilhamento de área de trabalho, permitindo que os indivíduos dividam os recursos do ambiente ao mesmo tempo ou em momentos diferentes e,
- mecanismos de compartilhamento de informações, onde várias pessoas trabalham sobre a mesma base de dados.

Estas aplicações que são desenvolvidas na Computação Cooperativa recebem o nome de *groupware* [BAN 91] [ELL 93].

São programas que enfatizam o ambiente multi-usuário, coordenando todos os que fazem parte do mesmo, controlando o fluxo de informações, evitando possíveis conflitos. O groupware sofre influências de, no mínimo, cinco áreas, entre as quais se encontram, os sistemas distribuídos, a inteligência artificial, comunicação, a interação homem-máquina e a teoria social, que é utilizada para planejar o sistema.

Os ambientes CSCW são divididos em quatro classes gerais, de acordo com o aspecto temporal e espacial [GAL 90] [GEE 91] [GRE 88] [MAL 93]:

1. Sistemas de Mensagem: o sujeito pode criar, enviar, receber e distribuir mensagens eletrônicas. Neste ambiente ocorrem interações remotas, não simultâneas entre os indivíduos. O exemplo típico é o “correio eletrônico” onde, no momento da criação da mensagem, existem facilidades de edição de

texto/formatação. Os usuários podem recuperar as mensagens, respondê-las, armazená-las e enviá-las a outros sujeitos e, ainda existem listas de distribuição que também podem ser armazenadas no sistema. Como pode ser visto este tipo de ferramenta permite que o usuário apresente as suas idéias a um grupo de pessoas. É chamada comunicação mediada pelo computador. O indivíduo pode utilizar o sistema para apresentar informações e, esta ação possibilita que estas pessoas encontrem outras com interesses em comum, criando ligações entre elas. O sistema de anotações também é outra forma de comunicar idéias ou opiniões sobre um determinado assunto.

2. Conferência por Computador: é outra forma de comunicação mediada por computador; uma evolução do correio eletrônico, com a diferença que a estrutura do sistema está imposta de acordo com a forma em que as mensagens estão agrupadas. Em geral, existe um assunto de discussão e os participantes apresentam suas contribuições. Há vários grupos (conferências), onde os membros da equipe tem seu tópico de interesse, no qual se baseiam para suas reuniões. O sistema armazena as informações lidas ou transmitidas por cada indivíduo envolvido na conferência. Estas são mantidas junto com as mensagens da conferência numa grande base de dados (central), podendo ser acessadas por qualquer membro da reunião. Atualmente existem inúmeros sistemas deste tipo, cada vez mais sofisticados, unindo capacidades em tempo-real, com modernos ambientes de janelas, conferência multimídia, entre outras tecnologias. Como pode ser visto, através da conferência por computador o sujeito pode mostrar as suas idéias, registrá-las e compartilhá-las com a equipe, desde que seja dentro do interesse e do objetivo do grupo. O acesso a uma conferência é normalmente controlado por um ou mais usuários com direitos especiais. Existem sistemas deste tipo baseados em ambientes gráficos, que permitem a apresentação e interação dos conhecimentos entre usuários através do uso de mídias, como voz, imagem, gráficos e desenhos.

3. Sala de Reuniões ou também denominadas de Sistemas de Tomada de Decisão: estes sistemas visam aumentar a produtividade no processo de tomada de decisão. Além da sala de conferência, estes são compostos por um projetor de uma grande tela, uma rede de computadores ou, simplesmente, um computador, com um número de terminais individuais e um terminal que faz o controle global do ambiente.

4. Sistemas de Autoração Grupal e Conversação Estruturada: este tipo de sistema ajuda nas negociações e discussões de um trabalho cooperativo. Um exemplo disso, é a autoria cooperativa de documentos (de texto ou com recursos multimídia), onde a apresentação final é um documento, fruto do consenso dos membros de uma equipe. Editores cooperativos, como os sistemas CoDraft [KIR 93] e GROVE [ELL 91], são sistemas de co-autoria, onde o produto final resulta das contribuições individuais e de negociações e argumentações fruto da apresentação de conhecimentos entre os sujeitos. O tipo de cooperação é, em geral, síncrona e a localização geográfica vai de face-a-face a localmente distribuída.

Os sistemas CSCW podem ser divididos em três classes, como mostra a figura 11 [ELL 93] [MAL 93].

◆ **Sistemas Puramente Síncronos:** existem quatro classes de programas que suportam o tipo síncrono:

1. sistemas baseados em estações (de trabalho) para atividades cooperativas, escrever, desenhar, esquematizar, esboçar, etc;
2. ferramentas de programação e software para implementação de groupware síncrono distribuído através das estações de trabalho, por exemplo, via compartilhamento de telas e janelas;
3. software para criação de reuniões eletrônicas e de decisão assim como os sistemas de suporte à decisão de grupo e, sistemas e software para criação de redes audiovisuais controladas computacionalmente.

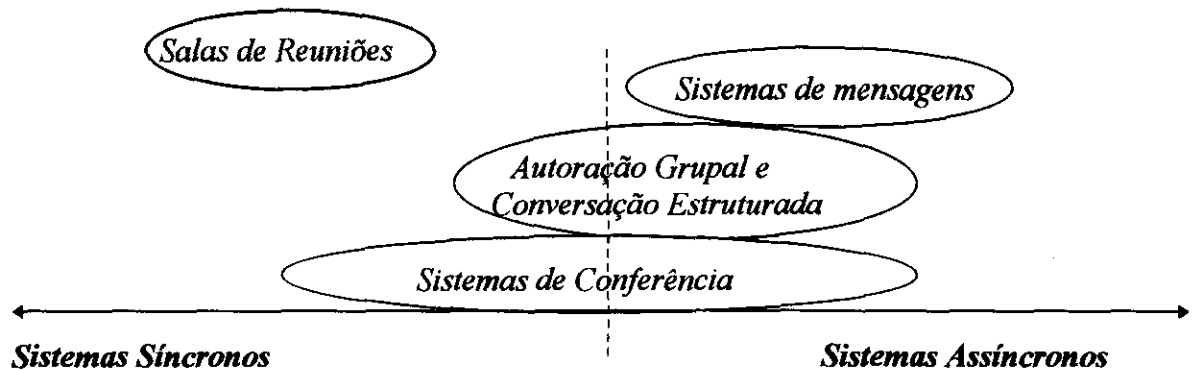


FIGURA 11 - Formas de Cooperação em sistemas CSCW

◆ **Sistemas Puramente Assíncronos:** esta classe pode ser dividida em três categorias:

1. e-mail e sistemas de conferências computadorizadas;
2. sistemas que suportam mensagens estruturadas, agentes e gerenciamento de fluxogramas e,
3. hipertexto cooperativo e sistemas de memória organizacional.

◆ **Sistemas Mistos:** exemplos de sistemas mistos são as conferências em tempo real, sistemas de autoração grupal e conversação estruturada, que trabalham de forma assíncrona em grande parte de sua utilização, mas também estão equipados com mecanismos para trabalho síncrono e funções de cooperação.

Os sistemas CSCW são considerados, do ponto de vista geográfico, remotos ou co-localizados. Esta classificação é mostrada na figura 12 [ELL 93] [GRE 88].

Enfim, para apresentar informações e trocar idéias nem sempre é necessário o suporte de um sistema sofisticado, basta estar disponível um quadro-negro, ou branco ou virtual para se rabiscar as idéias e comunicá-las. Os chamados blocos de rascunho virtuais e compartilhados são também espaços para mostrar a criatividade: Clearboard, GroupDraw, LiveBoard, TeamPaint, entre outros [MAL 93].

Os editores tradicionais cooperativos de texto e de desenhos são os instrumentos mais populares para apresentar e registrar textualmente o conhecimento de cada indivíduo integrado a um grupo. A edição cooperativa de textos é atendida por editores de texto convencionais ou por sistemas de hipertextos.

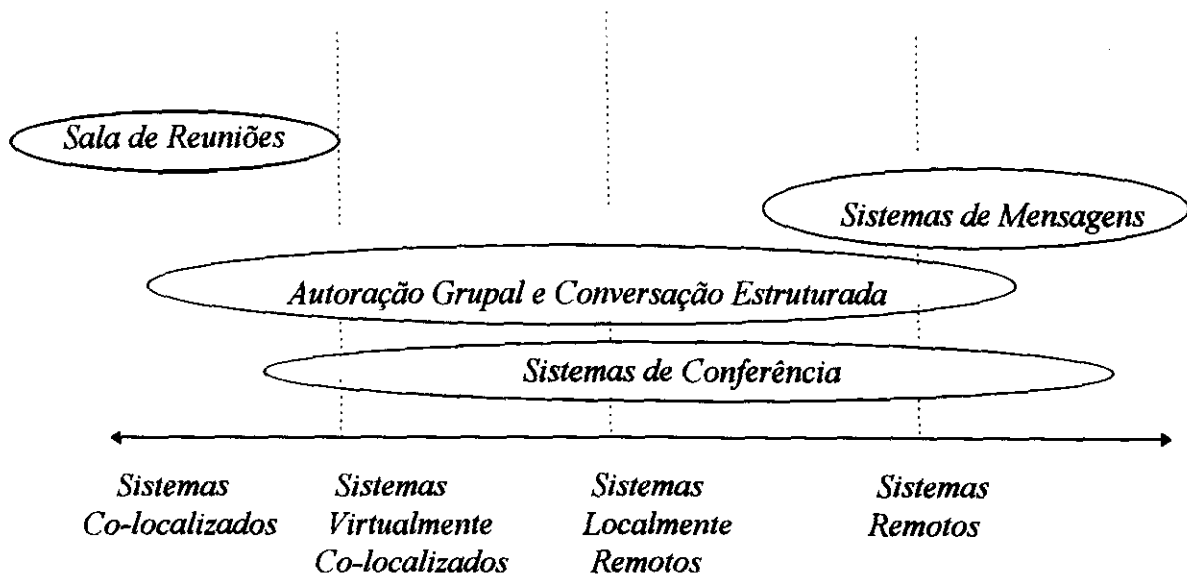


FIGURA 12 - Natureza Geográfica de Sistemas de CSCW

Os CSCW podem ser classificados de acordo com duas formas de controle [BAN 91]:

1. Controle Explícito: possuem técnicas de interação e cooperação de grupo e,
2. Controle Implícito: não possuem técnicas de representação ou coordenação do grupo de interação.

O tipo de cooperação destes sistemas é ditado pelo estilo de interação do ambiente. A classificação simples da representação e controle de cooperação em ambientes CSCW produz cinco classes de sistemas:

- Sistemas dos atos da fala [FLO 88] ou sistemas baseados em “conversação”: permitem uma abordagem lingüística para o trabalho cooperativo suportado por computador baseado, basicamente, na teoria dos atos da fala que considera a linguagem como uma série de ações. A cooperação é representada e controlada dentro deste tipo de sistema, utilizando algumas estruturas de redes detalhando os padrões de troca de mensagens. Um exemplo deste sistemas é COORDINATOR [WIN 87].
- Sistemas de procedimento de escritório: descreve tarefas executadas no escritório ou oficina em termos da combinação de efeitos de um número pequeno de sub-tarefas ou procedimentos. As pesquisas concentraram-se no desenvolvimento de linguagens que permitem a especificação de procedimentos de escritório e a descrição de suas interações. Esta classe de sistema se caracteriza pelo uso de linguagens procedurais para descrever e controlar a cooperação através da definição de papéis e atividades.
- Sistemas de Mensagem ativo semi-formal: semi-formal ou sistema de mensagem ativo provem o suporte de mecanismos para manipulação de mensagens automáticas, incluindo os conceitos de papéis (funções de cada agente) e de agentes autônomos.

- **Sistemas de conferência:** provem mecanismos de controle básicos mínimos e fixos dentro das aplicações. Em sistemas tradicionais, os conferencistas e moderadores controlam a entrada e saída de informações. Nos sistemas de conferência em tempo real, um mecanismo de controle de fluxo é que dita quem terá acesso para compartilhar o espaço da conferência, a qualquer momento.
- **Sistemas de controle livre ou Reunião de Grupo-Igual:** neste tipo de sistema não existe nenhum mecanismo de controle; confia-se na reunião dos participantes para formular os seus próprios protocolos de encontro. Todos os usuários têm o mesmo status e podem mudar e usar o sistema livremente. Estes sistemas não conservam a sua natureza e nem o grupo de trabalho envolvido no ambiente.

Os primeiros três tipos de sistemas possuem controle explícito e permitem a representação e editoração de informações de controle. Por outro lado, a conferência e os sistemas de controle livre têm controle implícito, ou seja, não contêm nenhuma representação de controle.

Com base nos conceitos descritos anteriormente, é possível entender as diferentes variáveis que influenciam na construção dos modelos gerais de interação de sujeitos os ambientes de computação cooperativa.

4.3 Conceitos para o estudo operatório de ferramentas computacionais de uso cooperativo

Da mesma forma que foi apresentado no nível individual, o que se pretende desenvolver nesta seção é a construção de modelos que auxiliem na análise lógico-operatória de ferramentas computacionais cooperativas. Ou seja, o caminho que se pretende seguir, baseando-se nos conceitos apresentados no capítulo anterior, visa responder as seguintes questões:

- quais são os elementos que são inseridos nesta interação, visto que se trata de uma interação interindividual, que leva em conta os aspectos afetivos, cognitivos e simbólicos?
- existe um sujeito coletivo usuário e/ou programador? quais são suas características e como estes interagem com uma ferramenta computacional cooperativa?
- quais são as diferentes formas e modelos de interação de um sujeito coletivo com uma ferramenta computacional cooperativa?
- como uma ferramenta deste tipo pode ser vista em relação a suas operações lógicas e/ou infralógicas?

Portanto, o primeiro passo para responder as questões levantadas é utilizar os conhecimentos descritos até agora, para construir o modelo de interação do sujeito coletivo com os ambientes computacionais cooperativos e, a partir destes, analisar as ferramentas de interação do ponto de vista da lógica operatória.

Utilizando o mesmo raciocínio da interação individual para o plano interindividual, o modelo interativo S-(V,O,L)-C apresentado na figura 7 pode ser estendido para o sujeito coletivo $(S_1 + S_n)$ -(V,O,L)-C, como mostrado na figura 13.

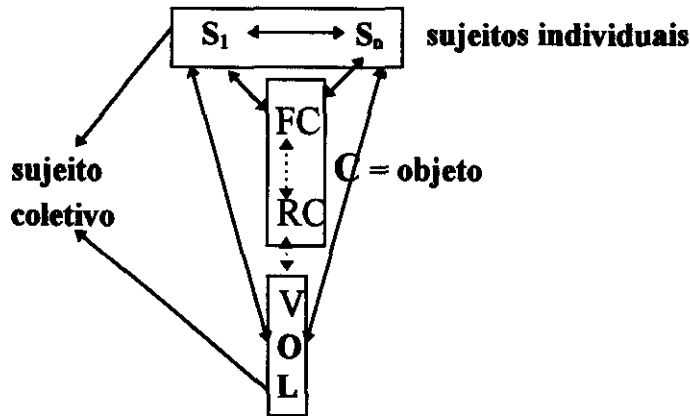


FIGURA 13 - Interação interindividual apoiada por computador

Pode ser observado que os mesmos elementos encontrados na interação individual se repetem na interação interindividual, com a diferença que, neste caso, é preciso levar em conta o aspecto social, a troca de idéias, os conflitos que podem ser gerados na comunicação, o compartilhamento de informações, entre outros, o que leva à constituição de uma nova entidade, que é o sujeito coletivo.

Da mesma forma que ocorre no plano individual, existe aqui também sujeitos usuários e/ou programadores coletivos mas, neste caso eles trabalham em equipe, isto é, em conjunto.

Antes de caracterizar o sujeito coletivo que foi delimitado neste estudo, é preciso defini-lo de forma abrangente.

Na nossa interpretação, um sujeito coletivo nada mais é do que um conjunto de sujeitos que fazem parte de um todo com características próprias construídas ou reconstruídas através do trabalho em grupo, e que se refletem no mesmo. Isso quer dizer que em um sujeito coletivo podemos encontrar vários sujeitos coletivos, isto é, outros grupos e, esses grupos que, nada mais são do que outros sujeitos coletivos, são formados por outras equipes de sujeitos, e assim por diante. No último nível, estariam os sujeitos individuais. Esquemáticamente esta idéia pode ser vista na figura 14.

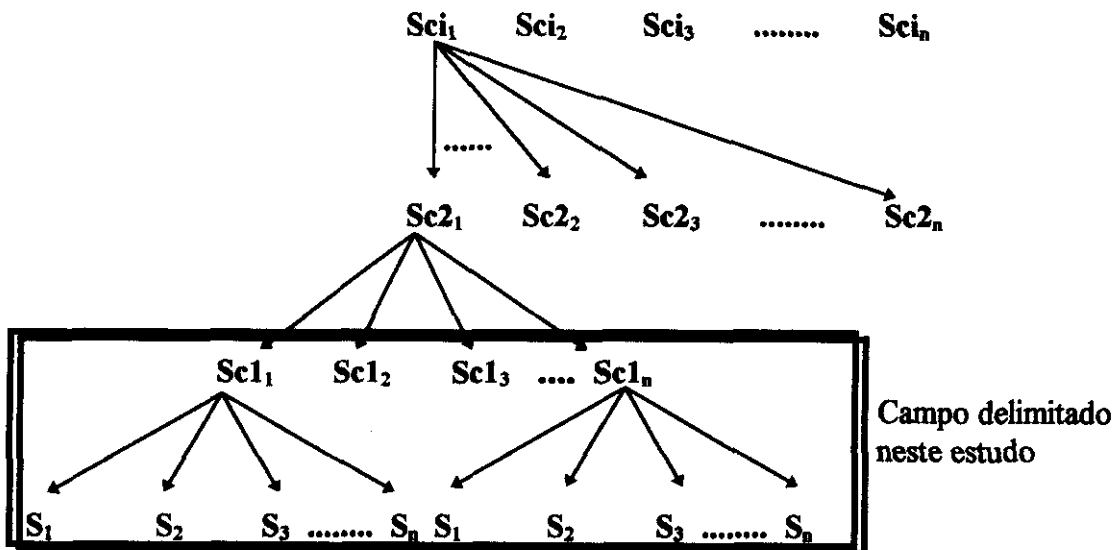


FIGURA 14 - Definição recursiva de um sujeito coletivo

onde: i, n : números naturais finitos; $S_1, S_2, S_3, \dots, S_n$: sujeitos individuais 1,2,3....até n ; $Sc1_1, Sc1_2, Sc1_3, \dots, Sc1_n$: sujeitos coletivos do nível 1; $Sc2_1, Sc2_2, Sc2_3, \dots, Sc2_n$: sujeitos coletivos do nível 2; $Sci_1, Sci_2, Sci_3, \dots, Sci_n$: sujeitos coletivos do nível i .

Logo, um sujeito coletivo sempre poderá ser formado por outros sujeitos coletivos, ou seja, na definição deste está envolvido o conceito de *recursividade*. Neste estudo somente será levado em conta os dois primeiros níveis da árvore, como mostrado na figura acima.

4.3.1 Caracterização do sujeito coletivo

Levando em conta agora a interação interindividual, são inseridos neste contexto os fatos sociais do sujeito, que Piaget [PIA 73] define como sendo as regras, os valores e os sinais coletivos. A existência de regras coletivas é o elemento que coordena as ações coletivas, isto é, referente ao caráter interindividual das interações. Os valores coletivos diferem dos valores atribuídos à simples relação entre sujeito e objeto, implicando em um elemento de troca interindividual. Os significantes próprios às interações coletivas, são os sinais convencionais, isto é, se opõem aos índices puros ou símbolos individuais.

Portanto, colocando estes conceitos em forma esquemática, a interação interindividual pode ser mostrada na figura 15. Nesta, levamos em conta dois sujeitos $S(x)$ e $S(y)$.

valores, objetos e linguagem individual

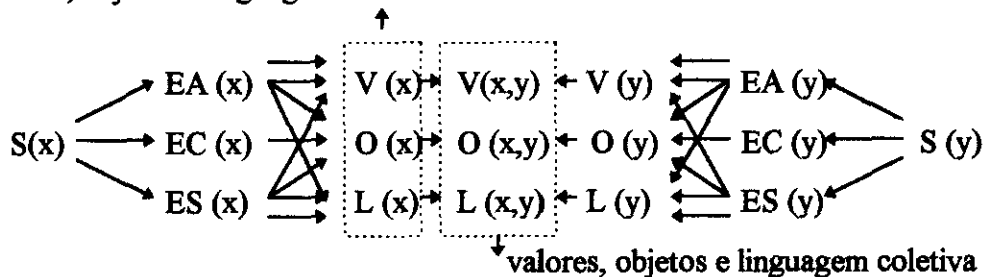


FIGURA 15 - Organização dos elementos envolvidos em uma interação interindividual

onde: $S(x)$: sujeito x ; $S(y)$: sujeito y ; $EA(x), EA(y)$: estrutura afetiva de $S(x)$, estrutura afetiva de $S(y)$; $EC(x), EC(y)$: estrutura cognitiva de $S(x)$, estrutura cognitiva de $S(y)$; $ES(x), ES(y)$: estrutura simbólica de $S(x)$, estrutura simbólica de $S(y)$; $V(x), V(y), V(x,y)$: valores de $S(x)$, valores de $S(y)$ e valores coletivos (x,y) ; $O(x), O(y), O(x,y)$: objetos de $S(x)$, objetos de $S(y)$ e objetos coletivos (x,y) ; $L(x), L(y), L(x,y)$: linguagem de $S(x)$, linguagem de $S(y)$ e linguagem coletiva (x,y) .

Exemplificando:

- Se $EA(x)(V) = EA(y)(V)$, os dois sujeitos operam e tem as mesmas regras em relação aos valores.
- Se $ES(x)(L) = ES(y)(L)$, utilizam as mesmas regras e operações para a linguagem.
- Se $EA(x)(O,V) = EA(y)(O,V)$, significa que os dois sujeitos tem os mesmos valores em relação aos objetos.
- Se $EA(x)(V,L) = EA(y)(V,L)$, significa que os dois sujeitos expressam os mesmos valores, emoções através de idéias.
- Se $ES(x)(O,L) = ES(y)(O,L)$, significa que os dois sujeitos (x e y) falam e dão nome aos objetos da mesma forma.
- Se $EC(x)(O) = EC(y)(O)$, significa que eles operam e tem as mesmas formas de coordenar estas ações em relação aos objetos.

- $V(x) = V(y)$ utilizam a mesma escala de valores.
- $V(x) \subset V(y)$ os valores do sujeito x são uma parte dos que têm o sujeito y .
- $V(x) \supset V(y)$, a escala de valores de x é mais rica do que a de y .
- $O(x) = O(y)$ os dois sujeitos tem o mesmo conhecimento em relação ao objeto.
- $O(x) \subset O(y)$ o conhecimento que o sujeito x tem do objeto está contido no conhecimento que y tem em relação ao mesmo.
- $O(x) \supset O(y)$, o conhecimento do objeto do sujeito y em relação ao sujeito x é muito mais pobre.
- $L(x) = L(y)$, quer dizer que a linguagem de x e y é a mesma.
- $L(x) \subset L(y)$ significa que a linguagem de x é só uma parte da linguagem de y .
- Se $L(x) \supset L(y)$, a linguagem de x é muito mais rica do que a de y .

O que seria, então, o **sujeito coletivo**? Neste caso, denominamos de sujeito coletivo “os sujeitos que se encontram no processo interativo, levando em conta suas estruturas afetiva, cognitiva e simbólica”.

Ou seja, os valores coletivos já não são mais de um sujeito ou de outro, mas se referem ao grupo. Da mesma forma a “representação” coletiva, ou os objetos de manipulação são objetos coletivos, isto é, uma nova totalidade que reflete as estruturas cognitivas, que operam sobre objetos individuais. A linguagem coletiva é uma composição de linguagens individuais que o grupo utiliza como meio de comunicação.

O sujeito coletivo é esquematizado na figura abaixo:

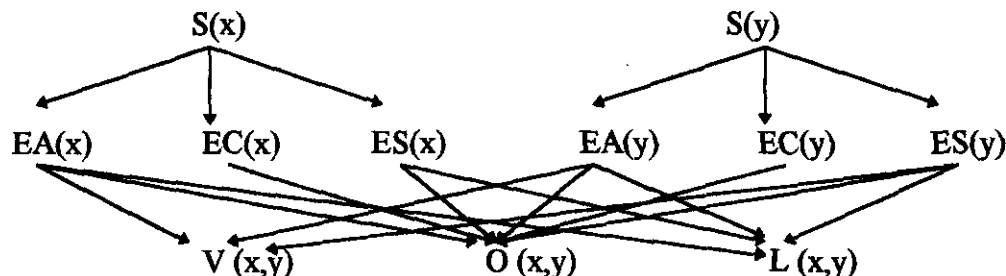


FIGURA 16 - Organização interna do sujeito coletivo

Neste caso, também podemos dizer que, além das estruturas afetivas, cognitivas e simbólicas de cada um dos sujeitos individuais envolvidos no processo interativo, também existem estas mesmas estruturas no plano coletivo, como pode ser ilustrado na figura 17.

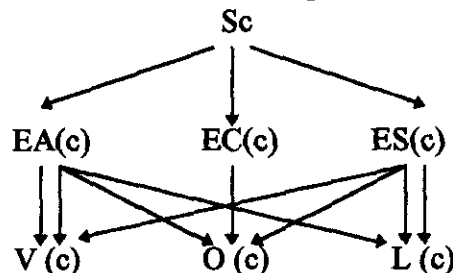


FIGURA 17 - Organização externa do sujeito coletivo

onde: Sc : sujeito coletivo; $EA(c)$: estrutura afetiva coletiva; $EC(c)$: estrutura cognitiva coletiva; $ES(c)$: estrutura simbólica coletiva; $V(c)$: valores coletivos; $O(c)$: objetos coletivos; $L(c)$: linguagem coletiva.

Da mesma forma que ocorre no nível individual, em que se encontram envolvidas nas estruturas, as operações e as regras em relação aos valores, objetos e linguagem, no nível coletivo, as estruturas coletivas funcionam de maneira semelhante. Ou seja, se um sujeito coletivo é composto pelos sujeitos $S(x)$ e $S(y)$ ($S_c = S(x) + S(y)$), isto quer dizer que:

$$S_c = EA(c) + EC(c) + ES(c)$$

onde $+$: é a notação para uma estrutura composta; não é simplesmente a soma, nem a justaposição.

Mas o que é $EA(c)$, $EC(c)$ e $ES(c)$?

$$EA(c) = EA(x) + EA(y)$$

$$EA(x) = (opV(x), rV(x)) \text{ e } EA(y) = (opV(y), rV(y)), \text{ portanto,}$$

$$EA(c) = (opV(x) + opV(y), rV(x) + rV(y))$$

Da mesma forma ocorre com $EC(c)$ e com $ES(c)$. Ou seja:

$$EC(c) = (opO(x) + opO(y), rO(x) + rO(y))$$

$$ES(c) = (opL(x) + opL(y), rL(x) + rL(y))$$

Como pode ser observado, as regras coletivas $r(c)$ sempre serão as regras de um sujeito ($r(x)$) compostas com as regras do segundo ($r(y)$) e, assim, também ocorre com as operações.

A nossa noção de regra vai ficar restrita a este contexto, não levando em conta as que dizem respeito aos fins genéricos relativos aos indivíduos como, por exemplo, o que e como fazer as coisas para ser feliz na vida e, as regras com fins genéricos mas relativas ao grupo, ou seja, referem-se às normas gerais de uma sociedade [ABB 87]. Logo, a única preocupação é entender como as regras resolvem o problema de coordenar as ações que envolvem valores, ações relativas aos objetos e ações que se referem à linguagem e a combinação dos mesmos.

Utilizando o mesmo raciocínio do sujeito individual para o sujeito coletivo, este último além de ter que combinar as estruturas de dois ou mais sujeitos para poder cooperar, ele também interage com um objeto. Isso significa que têm que ser coordenadas as estruturas de, no mínimo, dois sujeitos e, além disso, as operações e as regras das estruturas do mesmo têm que ser combinadas com a estrutura do objeto de interação. Isto é, para que um sujeito coletivo trabalhe de forma coerente com uma certa ferramenta computacional (é o objeto do nosso exemplo), as estruturas do sujeito coletivo têm que ser compatíveis com as operações e as regras que regem o ambiente computacional.

Portanto, o próximo passo é verificar se tanto os valores, quanto os objetos e os sinais (linguagem) podem ser operados logicamente e, definir quais são estas operações lógicas e/ou infralógicas.

Os fatos sociais se apresentam em forma de valores de troca. Estes consolidam socialmente os valores e os transformam, tornando-os dependentes, não somente da relação entre um sujeito e os objetos, mas do sistema total das relações entre dois ou mais sujeitos, por um lado, e os objetos, por outro.

Os valores de troca se referem a tudo o que está envolvido em uma troca, desde objetos utilizados pela ação prática, até as idéias e representações que ocasionam uma troca intelectual e até os valores afetivos interindividuais.

O sinal é um meio de expressão que serve para a transmissão das regras e dos valores. Ele é arbitrário e supõe uma convenção explícita e livre, como é o caso dos sinais matemáticos ou, ainda, obrigada como a linguagem corrente. O sistema de sinais é essencial à vida social; os sinais verbais, a escrita, os gestos, a mímica, enfim, todo tipo de manifestação da função simbólica do sujeito.

4.3.1.1 O sujeito coletivo e a cooperação

É preciso esclarecer neste estudo que um grupo que trabalha de forma coletiva não necessariamente está trabalhando de forma cooperativa. Mas neste contexto, como um dos objetivos principais é caracterizar uma ferramenta computacional cooperativa em termos da lógica operatória, levaremos em conta uma interação interindividual de cooperação com vistas a um fim específico. Portanto, o que interessa são as ações coletivas cooperativas. Logo, quando falamos de sujeito coletivo, estamos nos referindo *“ao grupo que realiza um conjunto de operações com um objetivo em comum, ou seja, um sujeito co-operativo”*.

Aqui analisaremos o sujeito coletivo, do ponto de vista de suas estruturas internas e, na seção 4.3.2, apresentamos o sujeito com seu objeto de interação que, no caso, são os ambientes computacionais cooperativos que possuem as suas próprias operações e regras de funcionamento.

Qual a relação entre a lógica e a vida social de um sujeito?

Ao lado das trocas de pensamento (comunicação verbal, transmissão oral de verdades anteriores) existem as trocas de ação (ajustamento recíproco de movimento e de trabalhos, transmissão de processos) as quais supõem uma cooperação efetiva. Ou seja, o que se quer dizer com isso, é que a cada interação intelectual corresponde uma estrutura intuitiva ou operatória determinada pela inteligência.

Portanto, por um lado existem escalas sucessivas de estruturação lógica e, por outro, cada uma destas escalas, é caracterizada por certo modo de cooperação ou de interação social.

A questão é: será que a estruturação lógica ou pré-lógica de um certo nível determina o modo de colaboração em jogo ou a estrutura das interações coletivas é que determinam a das operações intelectuais?

As interações são constituídas por ações e a cooperação, como já foi dito anteriormente, consiste num sistema de operações. Assim, as atividades do sujeito se exercendo sobre os objetos e as atividades dos sujeitos quando agem uns sobre os outros se reduzem, na realidade, a um só e mesmo sistema de conjunto, no qual o aspecto social e o lógico são inseparáveis tanto na forma quanto no conteúdo.

Segundo Piaget [PIA 73], se a lógica consiste em operações que precedem a ação, e se estas operações constituem por sua natureza sistemas de conjunto ou totalidades, então, pode-se dizer que, estes “agrupamentos” operatórios expressarão tanto os ajustamentos recíprocos e interindividuais de operações, quanto as operações interiores do pensamento de cada sujeito.

Se levamos em conta uma ação executada em conjunto por dois sujeitos, ou seja, colaborando um com o outro, é preciso que estes efetuem um ajustamento de suas ações. Portanto, em primeiro lugar, este ajustamento deverá ser feito através de uma série de operações qualitativas: correspondência das ações a elementos comuns, reciprocidade das ações simétricas, adição ou subtração das ações complementares, etc. Depois, se cada uma das ações dos colaboradores, sendo regulada por leis de composição reversível, constitui uma operação, o ajustamento destas ações de cooperação consiste

também em operações. Estas podem ser correspondências, reciprocidades ou simetrias, complementaridades, que são operações como as outras, assim como cada uma das operações executadas por cada sujeito. Estes podem escolher seu próprio sistema de referências, por exemplo, mas para existir cooperação será necessário coordenar num só estes sistemas de coordenadas. Isto é, será necessário coordenar as operações de cada sujeito num só sistema operatório, cujos atos de colaboração constituem as operações integrantes. Ou seja, deverão existir o tempo todo operações coletivas e formas de coordenar essas operações para ocorrer a cooperação (regras que devem ser seguidas).

Logo, a cooperação e as operações agrupadas são uma única e só realidade vista de duas formas diferentes. Portanto, não há resposta para a questão se é a constituição dos agrupamentos de operações concretas que permite a formação da cooperação ou vice-versa. O “agrupamento” é a forma comum de equilíbrio tanto das ações individuais quanto das interações interindividuais, porque não existem dois modos de equilibrar as ações e porque a ação sobre o outro é inseparável da ação sobre os objetos.

4.3.1.1.1 As operações lógicas nas trocas sociais do sujeito coletivo

Levando em conta a interação de dois sujeitos individuais, por exemplo, S_1 e S_2 , pode-se dizer que uma troca qualquer é uma ação de S_1 sobre S_2 . Isto é, existe um valor, simbolizado por $r(S_1)$, sacrificado por S_1 (tempo, trabalho, idéias, objetos) que alcança uma satisfação (positiva ou negativa) de S_2 , simbolizado por $s(S_2)$.

Assim, S_2 sacrifica os valores $r(S_2)$ agindo sobre S_1 que experimenta a satisfação $s(S_1)$. Estes são chamados de *valores reais*. Mas nem sempre a ação $r(S_1)$ de S_1 sobre S_2 é seguida de uma ação de volta $r(S_1)$. Aí entram duas espécies de *valores virtuais*: S_2 tendo experimentado a satisfação $s(S_2)$ contrai uma dívida $t(S_2)$ em favor de S_1 , enquanto esta mesma dívida constitui um crédito $v(S_1)$ para S_1 . Estes valores virtuais são de importância totalmente geral: os valores $t(S_1)$ ou $t(S_2)$ podem tomar a forma de gratidão e reconhecimento, que obrigam em diversos graus o sujeito. Por outro lado, os valores $v(S_1)$ ou $v(S_2)$ exprimem o sucesso, a autoridade, o crédito moral, adquirido graças às ações (r).

O equilíbrio da troca é determinado pelas condições de igualdade:

$$r(S_1) = s(S_2) = t(S_2) = v(S_1) = r(S_2) = s(S_1) = t(S_1) = v(S_2)$$

Mas este equilíbrio dificilmente é atingido; em geral, todas as desigualdades são possíveis como, por exemplo, $r(S_1) > s(S_2)$; $s(S_2) > t(S_2)$; $t(S_2) > v(S_1)$, etc se é desvalorizado ou superestimado os serviços prestados.

A gratidão, o reconhecimento, o sucesso, a autoridade moral podem ser consideradas composições lógicas de valores. Mas quando se trata, não da relação entre dois sujeitos mas entre um sujeito e uma coletividade B ou X, já não é mais uma composição aditiva, mas sim, trata-se de uma associação. Isto é, esta associação de interações (onde cada uma já está submetida a regulações e não a operações reversíveis), constitui um sistema de conjunto do tipo totalidades estatísticas, ou seja, o todo não é a soma algébrica das relações isoladas, mas simples composto provável.

O que nos interessa é definir se os sistemas de regras e valores podem ser reduzidos a proposições. E, o que é uma proposição? É o ato de comunicação, portanto, as proposições podem ser trocadas em forma de diálogo interior ou entre vários sujeitos distintos, pouco importa. O problema está em determinar em que consiste esta troca e quais são as operações envolvidas nela.

As trocas das proposições supõe um sistema abstrato de avaliações recíprocas, de definições e de normas. Esta troca também constitui um agrupamento de operações e são

as conservações obrigadas próprias a tal agrupamento que impõem a lógica das proposições suas regras fundamentais de agrupamento.

Está claro, portanto, que uma troca de idéias (onde estão inseridos os valores), isto é, de proposições obedece, do ponto de vista de sua forma exterior, aos esquemas de trocas em geral. Mas, no caso particular das proposições, os valores reais r e s e os valores virtuais t e v , resultantes das trocas entre dois sujeitos S_1 e S_2 , tomam a seguinte significação: $r(S_1)$ exprimirá o fato de que S_1 enuncia uma proposição, isto é, comunica um julgamento a S_2 ; $s(S_2)$ marcará em troca o acordo (ou desacordo) de S_2 , isto é, a validade atual que ele atribui à proposição de S_1 ; $t(S_2)$ traduzirá, por outro lado, a maneira pela qual S_2 conservará (ou não) seu acordo ou seu desacordo, isto é, esta validade atualmente reconhecida ou negada por ele. $v(S_1)$ enfim é, mas desta vez do ponto de vista de S_1 , a validade futura da proposição enunciada em $r(S_1)$ e reconhecida (ou negada) em $s(S_2)$. Tem-se então que, $r(S_1) \rightarrow s(S_2) \rightarrow t(S_2) \rightarrow v(S_1)$, etc.

Portanto, em uma troca de idéias entre sujeitos pode-se tirar duas conclusões: uma, é determinar as condições de equilíbrio desta troca (características do estado no qual os sujeitos estão de acordo ou satisfeitos) e, a outra, é que as condições de equilíbrio desta troca social implicam precisamente em um agrupamento de proposições. Ou seja, uma troca de proposições, enquanto conduta social, comporta por suas próprias leis de equilíbrio, uma lógica coincidindo com a lógica que usam os sujeitos para agrupar suas operações formais.

Para que ocorra o equilíbrio das trocas, é preciso satisfazer três condições:

1. S_1 e S_2 tem que ter uma escala comum de valores intelectuais, expressos através de símbolos comuns unívocos.
2. Igualdade geral dos valores em jogo (acordo sobre os valores reais)
3. Atualização possível em todo o tempo dos valores virtuais, isto é, ter a possibilidade de retornar sem cessar às validades reconhecidas anteriormente, ou seja, isto implica na reciprocidade.

Assim, o equilíbrio das trocas comporta um sistema de normas e, estas normas constituem agrupamentos que coincidem com os da lógica, apesar de não admitirem essa lógica em seu ponto de partida.

Ou seja, a troca das proposições constitui uma lógica pois acarreta o agrupamento de proposições trocadas. Assim, existirá um agrupamento próprio para cada sujeito, em função de suas trocas com o outro, e um agrupamento geral devido às correspondências, às reciprocidades ou às complementaridades de seus agrupamentos solidários.

Como pode ser visto, a troca entre sujeitos constitui uma lógica, que converge com a lógica das proposições individuais.

Em resumo, as funções do sujeito individual e as funções do coletivo chamam-se umas às outras para chegar ao equilíbrio lógico. Logo, o agrupamento é um sistema de substituições possíveis, seja no que se refere ao pensamento individual (operações da inteligência), seja de um sujeito com outro (cooperação social compreendida como um sistema de cooperações).

Estas duas formas constituem ao mesmo tempo uma lógica coletiva e individual, que caracteriza a forma de equilíbrio comum tanto às ações sociais quanto às individualizadas. É esse equilíbrio que axiomatiza a lógica formal.

4.3.1.1.2 A operatoriedade dos valores coletivos

Como já foi dito anteriormente, o fator social envolve um sistema de obrigações (regras), de trocas (valores) e de símbolos convencionais que servem de expressão às

regras e aos valores (sinais) [PIA 73]. O que nos interessa aqui são os valores sociais, ou seja, definir se estes podem ser agrupados operatoriamente. Para isso, é preciso definir o que é uma escala de valores.

Em primeiro lugar, pode-se afirmar que toda sociedade possui um número maior ou menor de *escala de valores*. Estes podem ser variáveis ou mais ou menos duráveis, heterogêneos uns aos outros ou suscetíveis de conduzir a uma espécie de duração média.

Assim, para cada sujeito, segundo o objetivo ao qual ele quer chegar e os meios que ele emprega para chegar lá, todos os objetos e todos os sujeitos que o interessam, assim como todas as ações, trabalhos e todos os “serviços” atual ou virtualmente prestados por eles, são suscetíveis de serem avaliados e comparados segundo algumas relações de valores, relações que constituem uma “escala” [PIA 73].

Do ponto de vista formal, pode-se representar uma escala de valores por um sistema de relações assimétrica, como pode ser visto na figura 18.

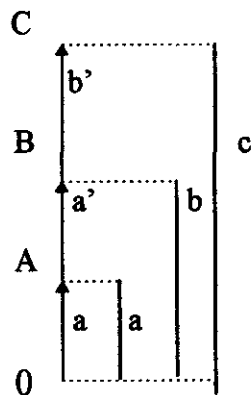


FIGURA 18 - Escala de valores em forma de relações assimétricas

Sejam A, B, C, ... etc, uma sucessão de termos de valor crescente. O $\uparrow aA$ = “A tem mais valor que O”; $A \uparrow a'B$ = “B tem mais valor que A”, e as duas operações seguintes:

1. Adição de valores: $\uparrow a + \uparrow a' = \uparrow b$ (ou $a + a' = b$); $b + b' = c$, etc.
2. Subtração de valores: $b - a' = a$; $c - b' = b$, etc.

Uma escala de valores também pode se apresentar em forma de árvore. Ou seja, existe um valor B1 (finalidade) à qual conduzem vários meios diferentes A1, A2 ou A3, como pode ser visto na figura 19.

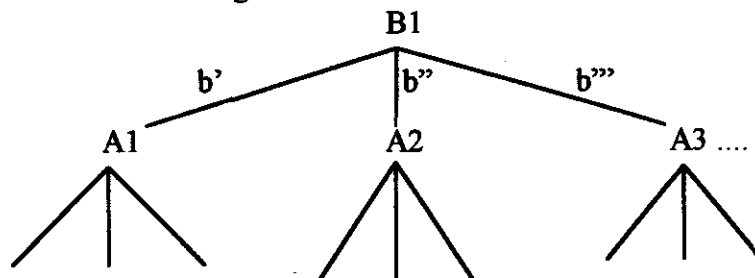


FIGURA 19 - Escala de valores hierárquicos

O mesmo sujeito pode conhecer simultaneamente várias escalas, segundo seus planos de atividade, estabelecendo hierarquias mais ou menos estáveis entre os fins de cada uma delas.

Esta abordagem está baseada unicamente na escala de valores do tipo apresentado na figura 18, por uma questão de simplicidade para analisar as operações oferecidas pelas

ferramentas computacionais em relação aos valores dos sujeitos envolvidos em uma interação através destas.

Toda ação ou reação de um sujeito para outro marca um crescimento de valores (satisfação), uma diminuição dos mesmos (prejuízo), ou uma diferença nula. Cada ação provocará uma ação de volta que pode ser de duas formas [PIA 73]:

1. ação material (valor atual; transferência de objetos em troca de serviço prestado) e,

2. ação virtual (aprovação ou censura, perseverar ou convite a cessar, promessa, valores virtuais).

A existência de valores se traduz numa valorização recíproca das ações ou “serviços”, sejam estes positivos ou negativos.

Se um sujeito x presta serviço a x' , podem ocorrer três possibilidades:

1. x' prestará serviço em troca de x ;
2. x' não prestará serviço mas se contenta em valorizar x ;
3. x' não prestará serviço nenhum a x , nem o valoriza.

Em cada um destes casos houve troca de valores, veremos, portanto, como exprimir esses valores em forma de esquemas logísticos.

Seja:

rx = ação de x sobre x'

sx' = satisfação de x' produzida pela ação rx

tx' = dívida de x' resultante da satisfação sx'

vx = valorização de x por x'

= : equivalência qualitativa

Em caso de equivalência simples, tem-se a seguinte equação lógica:

• $(rx = sx') + (sx' = tx') + (tx' = vx) = (vx = rx)$, por hipótese o sujeito x é valorizado por x' proporcionalmente ao serviço que lhe foi prestado

Essa equação é uma correspondência de série ou “similitude qualitativa”, baseada na multiplicação de regras assimétricas.

Além da equivalência geral, pode-se ter uma série de outras combinações, como, por exemplo:

• Se $rx > sx'$, tem-se:

$$(rx > sx') + (sx' = tx') + (tx' = vx) = (rx > vx)$$

Isto significa que x trabalha com prejuízo e que tem insucesso ou fracasso com sua ação social.

• Se $rx < sx'$, tem-se:

$$(rx < sx') + (sx' = tx') + (tx' = vx) = (rx < vx)$$

Neste caso, há benefício para x , cujo trabalho fácil é coroado de sucesso superior a seu esforço. Portanto, este é estimulado a perseverar no caminho que escolheu.

• Se $rx' > tx'$ enquanto $rx = sx'$ e $tx' = vx$

$$(rx = sx') + (sx' > tx') + (tx' = vx) = (rx > vx)$$

O sujeito x trabalha com prejuízo mas desta vez porque x' não quer reconhecer sua satisfação sx' .

• Se $sx' < tx'$ enquanto $rx = sx'$ e $tx' = vx$

$$(rx = sx') + (sx' < tx') + (tx' = vx) = (rx < vx)$$

Neste caso, há superavaliação de x por x' .

Da mesma forma que na lógica qualitativa geral existe a relação inversa de extensão (quantidade) e de compreensão (valor) dos conceitos, no plano dos valores também existe essa relação inversa entre o valor e a quantidade. Neste caso, quanto maior o

número de meios, menos são os valores que eles têm (emprego de um desvaloriza os outros) e menor quantidade de meios implica em ter mais valor.

Estendendo este raciocínio, como ficariam as trocas de valores não mais entre dois sujeitos individuais mas entre sociedades inteiras (sujeitos coletivos)?

Supondo que existam dois sujeitos (x, x') numa troca e que haja benefício recíproco, então, pode-se dizer que: $v_x > r_x$ e $v_{x'} > r_{x'}$. Assim, tem-se:

$$1. (r_x < s_{x'}) + (s_{x'} = t_{x'}) + (t_{x'} = v_x) = (r_x < v_x)$$

$$2. (r_{x'} < s_x) + (s_x = t_x) + (t_x = v_{x'}) = (r_{x'} < v_{x'})$$

Estas duas equações refletem uma relação de dupla valorização que constitui o que se pode chamar de “simpatia” entre x e x' . Ou seja, tudo o que um faz satisfaz o outro e vice-versa. Da mesma forma, “antipatia” vem a ser a desvalorização recíproca.

A valorização recíproca existe quando a escala de valores entre os sujeitos é a mesma, isto é, se diz quando estes estão de acordo, tem os mesmos gostos, etc...

Esta idéia não é só aplicada a uma troca interindividual, ela também é uma condição de existência para toda uma coletividade, ou seja, esta mesma idéia pode ser utilizada para dois sujeitos coletivos.

Toda escala de valores corresponde a uma coletividade de co-valorização constituída pelo conjunto de sujeitos co-permutadores segundo esta escala. Chama-se de classe de co-valorizantes a todo conjunto de sujeitos que troca seus valores segundo uma escala em comum. Isto é, por exemplo, a classe A é formada dos sujeitos x, x' e x'' , que admitem a mesma escala de valores. Assim, levamos em conta não mais a troca somente entre dois sujeitos mas, neste caso, entre todos os que fazem parte da classe A. Logo, a soma algébrica das trocas é $v_A = v_x + v_{x'} + v_{x''}$.

Portanto nesta coletividade pode existir benefício recíproco se esta for estável ($v_A > r_A$), desvalorização recíproca se a coletividade não for viável ($v_A < r_A$), ou ainda, pode existir um equilíbrio exato $v_A = r_A$.

Para que uma coletividade se conserve é preciso que apresente, no mínimo, uma escala comum de valores e que as trocas atinjam benefícios recíprocos ou o equilíbrio.

Uma escala de valores nada mais é do que a comparação ou a seriação das satisfações já obtidas ou a serem obtidas.

Mas, como assegurar a conservação dos valores?

Através de um sistema de operações designando, de forma duradoura, algumas relações e condições de equivalência aos valores presentes. As normas são estas operações de ordem formal e, os valores são os conteúdos destas formas. Por exemplo, um valor normativo será o valor resultante da aplicação de uma norma, enquanto esta é a regra ou obrigação propriamente dita.

Como pode ser visto, devem existir operações que coordenem os meios e os fins, as ações e as satisfações. Estas operações se referem à coordenação das ações em relação aos valores e elas constituem um sistema de correspondências ou reciprocidades, isto é, estamos nos referindo aos procedimentos necessários com fins específicos (regras).

4.3.2 Modelos gerais de interação de um sujeito coletivo com ferramentas computacionais de uso cooperativo

Baseando-se nos conceitos apresentados sobre a estrutura tanto do sujeito coletivo quanto dos ambientes cooperativos, é preciso construir os modelos mais gerais de interação dos sujeitos que trabalham de forma coletiva com uma ferramenta computacional cooperativa qualquer. A partir disso, é possível caracterizar operatoriamente as ferramentas computacionais cooperativas.

Existem várias formas através das quais os quatro tipos de operações descritas em 3.4 são executadas pelo sujeito coletivo em interação com um ambiente computacional cooperativo. Mas, neste caso, tudo irá depender das características das ferramentas em questão, como descrito em 4.2.1, se ela é centralizada ou distribuída, se permite tanto a representação individual quanto coletiva, ou somente a coletiva, se os sujeitos se comunicam de forma síncrona ou assíncrona, entre outros. Isto é, existem inúmeras variáveis que influenciam a realização prática do modelo interativo. Neste trabalho, apresentaremos o modelo de um sujeito coletivo com três tipos de processos interativos, produzindo, assim, três modelos simples de interação.

Em primeiro lugar, é mostrado o modelo mais simples, que é uma ferramenta centralizada, utilizada, de forma cooperativa por vários usuários (sujeito coletivo), na qual eles fazem uma representação coletiva de suas imagens mentais através da linguagem suportada pela ferramenta. Este modelo é mostrado na figura 20.

- A estrutura afetiva, cognitiva e simbólica de cada sujeito (S) pode executar operações sobre os valores, objetos e linguagem de acordo com as respectivas regras sobre a ferramenta computacional centralizada (FC). Isto é representado de forma cooperativa (RC) através da linguagem suportada pela ferramenta (LC). Assim, o sujeito (individual e/ou coletivo) pode executar as operações (lógicas e/ou infralógicas) sobre o ambiente (CDF/CIF). Também este pode manipular a representação computacional de forma direta e/ou indireta (MDR/MIR) através das operações lógicas e/ou infralógicas.

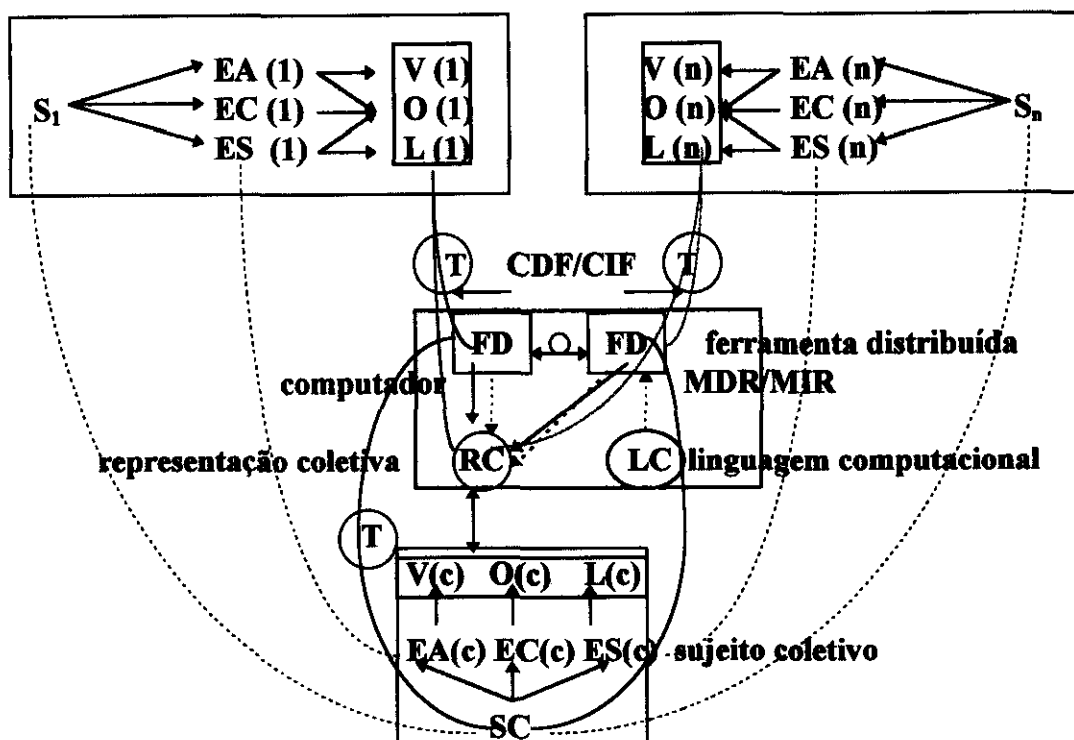


FIGURA 21 - Modelo geral 2 de interação entre um sujeito coletivo x ambiente computacional cooperativo distribuído

onde: S_1 : sujeito₁; S_n : sujeito_n; **FD**: ferramenta computacional distribuída; **T**: tela, teclado, etc...(canal de comunicação sujeito-ferramenta); **RC**: representação cooperativa; **LC**: linguagem computacional; **CDF/CIF**: controle direto/indireto sobre a ferramenta; **MDR/MIR**: manipulação direta/indireta sobre a representação coletiva.

O último modelo é mostrado na figura 22 e pode ser interpretado da seguinte forma:

- É uma ferramenta computacional distribuída (**FD**), onde cada sujeito (**S**) interage através de uma única linguagem suportada pelo ambiente (**LC**). Primeiro, cada sujeito opera de forma individual e, depois, cooperativamente (**RC**), refletindo as "representações individuais" (**RI**). As operações lógicas e/ou infralógicas podem ser executadas pela ferramenta (**CDF/CIF**). Aqui podem ser observadas tantas representações individuais quanto for o número de sujeitos individuais que fazem parte do ambiente. Tanto a **RI** quanto a **RC** podem ser manipuladas de forma direta e/ou indireta (**MDR/MIR**).

Como pode ser observado, foi necessário integrar os conceitos e definições apresentados até então, com os modelos de interação, a fim de utilizá-los como base para a análise lógico-operatória de ferramentas computacionais de uso cooperativo.

Estes são alguns dos modelos de interação que foram identificados e estudados. Neles será baseada a análise das diferentes ferramentas de manipulação da representação coletiva apresentada no capítulo 6.

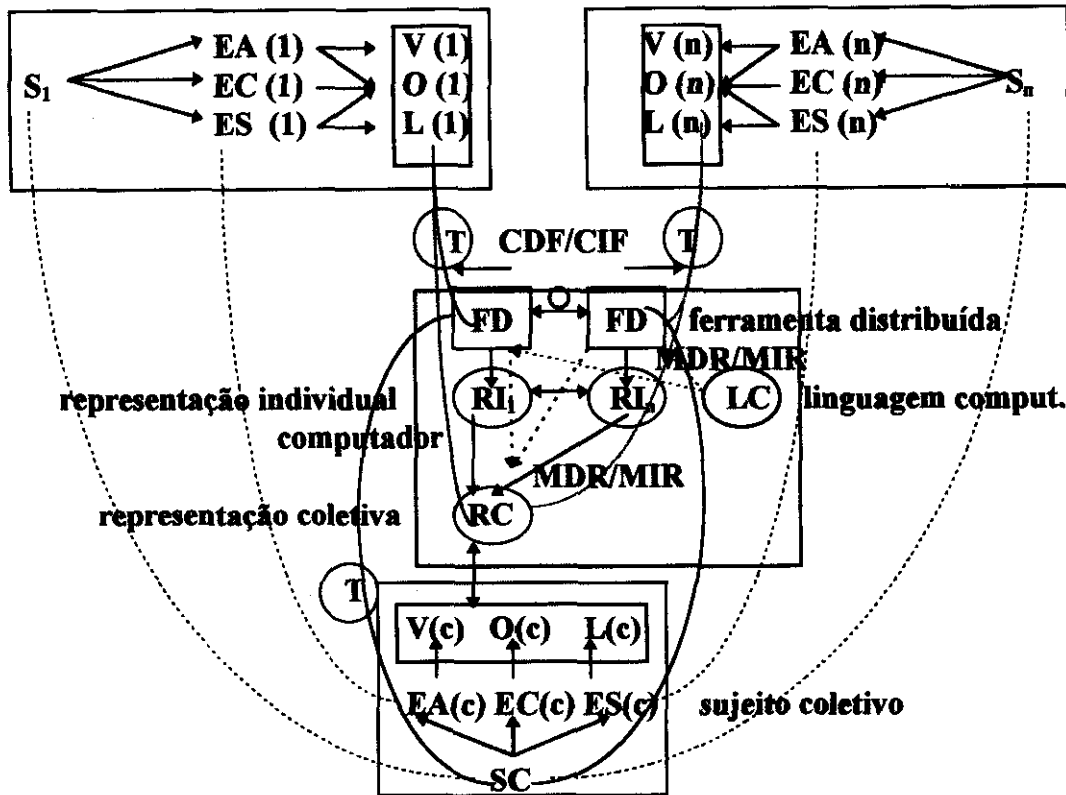


FIGURA 22 - Modelo geral 3 de interação entre um sujeito coletivo x ambiente computacional cooperativo distribuído

onde: S_1 : sujeito₁; S_n : sujeito_n; FD: ferramenta computacional distribuída; T: tela, teclado, etc...(canal de comunicação sujeito-ferramenta); RI_1 : representação individual do sujeito₁; RI_n : representação individual do sujeito_n; RC: representação coletiva; LC: linguagem computacional; CDF/CDI: controle direto/indireto da ferramenta; MDR/MIR: manipulação direta/indireta da representação individual e/ou coletiva.

O próximo passo é aplicar estes modelos para realizar a análise operatória das ferramentas computacionais cooperativas.

5 Análise operatória de ferramentas computacionais de uso individual

5.1 Editor Gráfico Paintbrush

5.1.1 Descrição do ambiente

O tipo de ambiente que propicia a representação do sujeito através de figuras, são os editores gráficos.

Segundo a nossa interpretação, quando um sujeito desenvolve uma atividade em um editor gráfico, ele está trabalhando sobre o espaço representativo, compondo uma

figura total através de suas partes, relacionando estas partes com outras e com um todo (desenho completo).

O editor gráfico é visto como uma ferramenta computacional, onde o sujeito tem disponível uma série de caixas de ferramentas e comandos pré-definidos para desenvolver a sua atividade. A imagem gráfica é vista como um objeto total que o indivíduo cria a partir de suas partes, de seus elementos, na tela do desenho.

Neste estudo interessa, basicamente, a forma como o sujeito representa as suas idéias graficamente, através do editor gráfico. Assim, os editores gráficos podem ser vistos como espaços gráficos onde será desenvolvida uma imagem gráfica. Estes constituem um “modo diferente” de representação espacial e, o espaço gráfico é uma das formas de espaço representativo.

A seguir é apresentado o aplicativo Paintbrush do Windows 3.1, que é um ambiente de **editoração de figuras**.

Como se trata de um editor gráfico, daremos ênfase às operações *infralógicas* suportadas pela própria ferramenta e pela interação desta com o sujeito.

O Paintbrush possibilita a criação de desenhos simples ou sofisticados, utilizando os botões do mouse e as instruções e funções pré-definidas. O Paintbrush contém:

- 1) uma caixa de ferramentas, que contém as ferramentas utilizadas para a criação do desenho. Estas permitem que o indivíduo desenhe, preencha, pinte, apague, digite texto e selecione itens dentro da área de desenho;
- 2) uma área de desenho, onde são criados os desenhos;
- 3) uma caixa de espessura da linha, que indica as espessuras de linha disponíveis a serem usadas e,
- 4) uma palheta de cores que contém as cores e padrões disponíveis para as ferramentas de desenho e para o fundo desta mesma área.
- 5) menus do aplicativo.

A interface do editor gráfico é mostrada na figura 23.

De acordo com o ambiente de interação no qual o sujeito deseja trabalhar (com o sistema ou com a imagem gráfica), são enquadradas as diferentes operações que podem ser executadas pelo programa, seguindo os níveis de interação acima definidos.

Somente nesta seção serão descritas todas as operações diretas e/ou indiretas tanto do controle do sistema quanto de manipulação de representações. Nas próximas análises das ferramentas serão abordadas estas operações de forma menos detalhada.

A tabela 1 referente a estas operações encontra-se na seção 3.4.

5.1.1.1 Operações do menu em relação ao ambiente de interação

5.1.1.1.1 Controle direto do sistema

1) **MODIFICAR O FORMATO DAS JANELAS DA INTERFACE:** o sujeito modifica o formato das janelas do ambiente, utilizando o mouse. Portanto, ele estará executando uma ação direta no sistema, afetando-o.

2) **VISUALIZAR:**

- **POSIÇÃO DO CURSOR:** mostra a posição em que se encontra o cursor. Essa escolha dependerá da ação do usuário.

3) **OPÇÕES:**

- **ATRIBUTOS DA FIGURA:** largura/altura - unidades: cm, pol, cores: preto/branco/colorido. O sujeito é quem escolhe o tamanho da figura que será desenvolvida.

- **FORMAS DO PINCEL:** definição da forma do pincel, ex: •, -, /, ■, \. O usuário altera o ambiente, modificando o formato do pincel, de acordo com a sua escolha.

- **EDITAR CORES:** o usuário cria suas cores personalizadas. Todas as cores e padrões do Painbrush são feitos variando a quantidade de cada um das cores primárias (vermelho, verde, azul) presentes na cor.

4) VISUALIZAR:

- **FIGURA:** apresentação de toda a atividade desenvolvida na área de desenho, desaparecendo a janela de interface do aplicativo.

5.1.1.1.2 Controle indireto do sistema

1) ARQUIVO

- **ABRIR:** abrir um novo arquivo: é preciso ir no menu *arquivo*, selecionar o comando de *abrir* e digitar (escolher) o nome arquivo que o usuário deseja trabalhar.

- **SALVAR:** salvar um arquivo corrente. O usuário executa o comando de *salvar* o desenho mas se trata de uma operação indireta pois quem guarda o arquivo é o próprio sistema, e não o usuário.

- **SALVAR COMO:** definir o tipo de arquivo (ex: PCX, BMP, PAL, etc). O sujeito faz a escolha do tipo de arquivo que se deseja salvar, mas é o ambiente quem executa este comando, afetando o sistema.

- **CONFIGURAR PÁGINA:** o sujeito define as medidas da página, tipo: cabeçalho, rodapé, margens da página e o sistema é quem faz a configuração da página de acordo com a escolha do usuário.

- **IMPRIMIR:** o usuário seleciona as diversas opções de impressão, ficando com o ambiente a tarefa de imprimir o arquivo.

- **CONFIGURAR IMPRESSÃO:** o sujeito é quem dá a ordem de alterar a impressora e as opções de impressão, mas o ambiente é quem executa este comando.

- **SAIR:** o usuário ordena o sistema a finalizar o aplicativo.

2) OPÇÕES:

- **CARREGAR CORES:** o ambiente carrega a paleta definida pelo usuário com extensão .PAL.

- **SALVAR CORES:** o sistema salva os arquivos tipo .PAL.

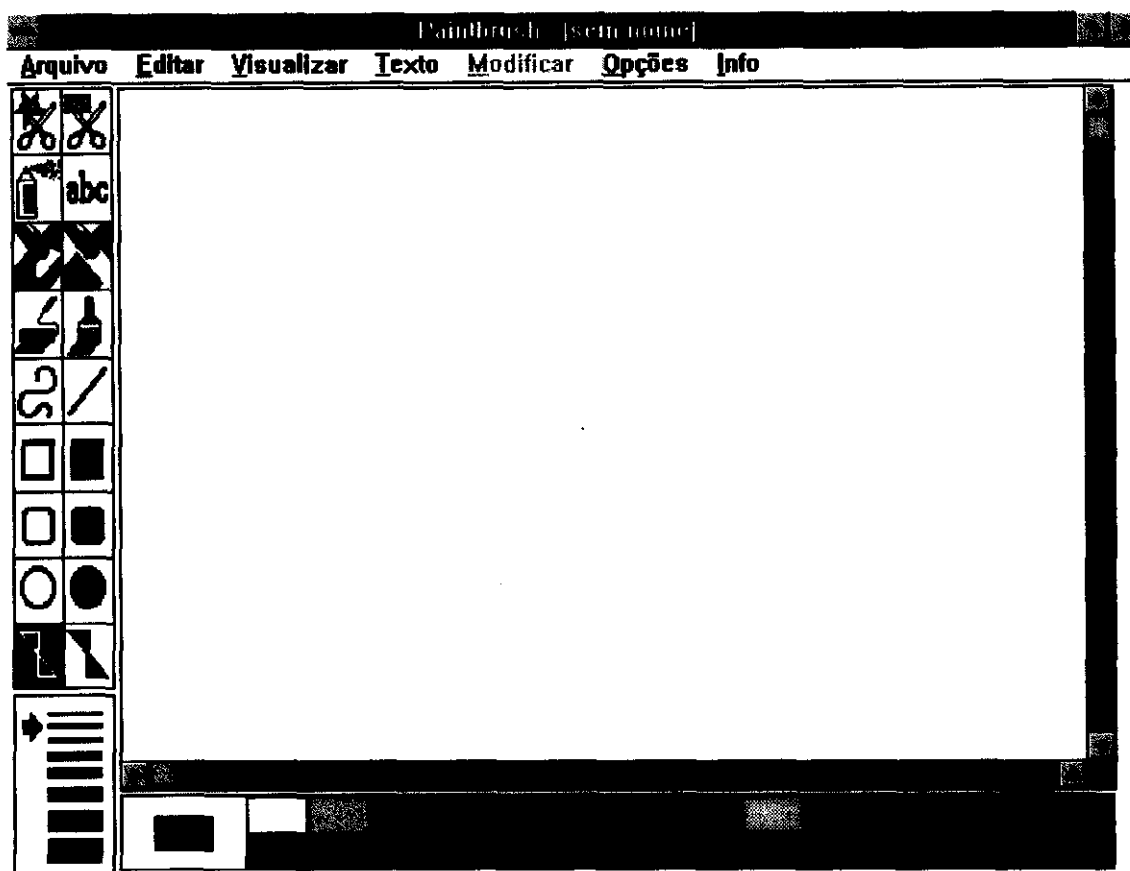


FIGURA 23 - Interface do Editor Gráfico Paintbrush

5.1.1.1.3 Manipulação direta sobre o desenho

Para a manipulação direta sobre o desenho são utilizadas basicamente:

- a caixa de ferramentas do editor para a criação do desenho
- a área de desenho
- a caixa de espessura da linha e,
- a paleta de cores.

Trabalhando com a caixa de ferramentas do Paintbrush, o usuário pode usar:

- 1) a área de corte (tesoura) regular ou irregular para marcar uma parte do desenho ou todo ele e, a partir daí fazer um simples deslocamento da área selecionada pressionando o botão do mouse;
- 2) utilizar o spray para desenvolver um desenho;
- 3) usar texto nos desenhos;
- 4) as borrachas para apagar o desenho;
- 5) o rolo de pintura;
- 6) o pincel e,
- 7) as demais ferramentas.

Selecionando uma ferramenta, o usuário executa uma ação sobre a área de desenho, alterando a imagem.

5.1.1.1.4 Manipulação indireta sobre o desenho

A seguir são apresentados os menus que correspondem à manipulação indireta sobre o desenho realizada pelo usuário em interação com o editor gráfico.

Trabalhando com a caixa de ferramentas do Paintbrush, o usuário pode utilizar a área de corte (tesoura) regular ou irregular para marcar uma parte do desenho ou todo ele. A partir daí ele pode:

- fazer um simples deslocamento da área de corte na direção do mouse (MDD) ou
- ativar o menu EDITAR, VISUALIZAR ou MODIFICAR (MID).

1) EDITAR

- RECORTAR: depois que foi selecionada a área de corte, o usuário ativa o comando de recorte para apagar a área marcada.
- COPIAR: copia a área marcada. Quem copia é o sistema.
- COLAR: cola a área marcada
- COPIAR PARA: salva área de corte como um arquivo
- COLAR DE: recupera uma área de corte anteriormente gravada em um arquivo e a coloca na área de desenho
- VOLTAR: desfaz a última operação realizada; apaga a parte do desenho criada desde a seleção da ferramenta que está sendo utilizada no momento. Também pode ser usada a tecla BACKSPACE.

Este comando ou tecla podem ser utilizados até que seja executada uma das seguintes ações:

- seleção de uma ferramenta;
- utilizar a barra de rolagem;
- alternar para um outro aplicativo;
- redimensionar a janela

2) VISUALIZAR:

- MAIS ZOOM: amplia uma parte do desenho (definido pela área de corte) possibilitando que sejam feitas alterações ponto a ponto (pixel)
- MENOS ZOOM: depois de serem feitas as alterações a partir do comando anterior, este traz o desenho para o seu formato normal

3) MODIFICAR

O Paintbrush possui diversas funções que permitem, através deste menu:

- ROTAÇÃO VERTICAL: rotação vertical da área de corte definida, de cima para baixo.
- ROTAÇÃO HORIZONTAL: rotação horizontal da área de corte definida, de um lado para outro.
- INVERTER CORES: inverte as cores da área de corte definida. Por exemplo, em um desenho preto e branco, a cor preta se transforma em branca e a branca se transforma em preta.
- REDUZIR + AMPLIAR: após definir a área de corte pode-se utilizar este comando para alterar as dimensões da área de corte, ou seja, reduzindo ou ampliando a mesma.
- INCLINAR: inclina uma área de corte definida no ângulo que for determinado
- LIMPAR: área original é apagada após redução ou ampliação da área de corte

4) OPÇÕES:

- OMITIR FORMATO DA FIGURA: quando o Painbrush copia um desenho para a Área de Transferência, ele copia em vários formatos para assegurar que o outro

aplicativo seja capaz de exibir o desenho. Este comando serve para limitar o formato da figura que será copiada para a área de transferência.

Depois de definidos os diversos comandos do editor gráfico de acordo com os ambientes de interação, é preciso destacar exclusivamente as ações executadas pelo sujeito em relação à atividade proposta. Isso porque o interesse desta abordagem é analisar as operações realizadas pelo usuário no espaço gráfico e não no que se refere à manipulação e controle do sistema. Mas é preciso identificar essa diferença para situar o leitor no contexto e justificar os aspectos a serem analisados.

5.1.2 Análise lógico-operatória do editor gráfico Painbrush

A seção anterior apresentou as diferentes operações do editor gráfico em relação ao controle e/ou manipulação do sistema e da representação respectivamente. A partir daí, analisaremos as operações infralógicas da ferramenta computacional.

Através das hipóteses que são levantadas a respeito do comportamento do sujeito em relação às operações infralógicas, é possível construir o modelo completo de interação do sujeito com um editor gráfico, mostrado na figura 24.

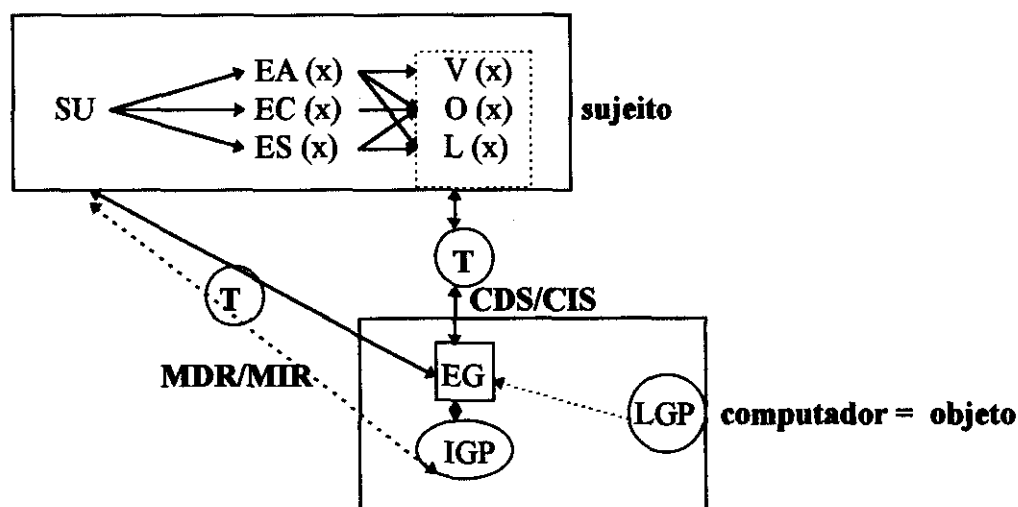


FIGURA 24 - Modelo de interação sujeito x editor gráfico

onde: SU: sujeito usuário; EG: editor gráfico; T: tela, teclado, etc...(canal de comunicação sujeito-ferramenta); IGP: imagem gráfico-plástica: representação individual; CDS: controle direto do sistema; CIS: controle indireto do sistema; MDR: manipulação direta sobre a representação (desenho) e, MIR: manipulação indireta sobre a representação.

A imagem gráfica em forma de figuras (IGP) é a representação simbólica da imagem mental (RIM) do sujeito. Ou seja, é o modelo de "algo" que o sujeito representa no computador, expressando-o em forma de "desenho" (no caso do editor gráfico). Para representar a IGP no computador, o usuário utiliza a sua própria linguagem gráfico-plástica (LGP).

Segundo Piaget [PIA 71], não é a representação mental do objeto que é desenhada, mas este evidencia o tipo de estruturação simbólica do sujeito num dado momento. A representação mental do objeto, ao ser traduzida para a linguagem gráfica do desenho, já não é a mesma, isto é, somente é um modelo. Neste caso, nos referimos ao modelo

computacional que o sujeito constrói em um sistema de representação que não é nem o desenho e nem a escrita, mas um tipo de sistema de representação computacional.

Como foi descrito, para que o sujeito interaja com um ambiente computadorizado ele precisa organizar esta nova realidade, entender o funcionamento da máquina, do software a ser utilizado. Para isso tem que ter um modelo mental do funcionamento do mesmo, construir os seus próprios conceitos em relação ao programa (ou ao computador) para poder operá-lo, manipulá-lo e, para isso, utiliza as estruturas lógicas e infralógicas do seu pensamento.

Para analisar as operações infralógicas que o sujeito realiza em interação com o editor gráfico, a leitura do modelo (apresentado na figura 10) foi feita em cima de dois tipos de situações:

1. na primeira, o sujeito produz uma imagem gráfica que lhe é proposta no editor gráfico e, as operações infralógicas são analisadas através da seqüência de comandos executada por ele e,

2. na segunda situação, é preparada a experiência no editor e proposto um desafio ao sujeito. As ações realizadas por ele para resolver o problema, são analisadas do ponto de vista da infralógica.

5.1.2.1 As operações infralógicas mais elementares e o contínuo

As operações infralógicas utilizadas pelo sujeito para o desenvolvimento de um desenho no editor gráfico, são apresentadas em forma de situações hipotéticas para servirem como exemplificações e, a partir daí introduzir as relações topológicas que constituem a representação espacial característico desse nível.

Na manipulação direta sobre um desenho, o usuário pode utilizar a caixa de ferramentas do Paintbrush para desenvolvê-lo. Neste exemplo se utilizam situações do tipo 1:

- Se pede ao usuário que desenhe uma casa ao lado de uma árvore. A primeira ação que ele faz é ativar alguma das opções da caixa de ferramentas. O quadrado pode ser utilizado para fazer a parte de baixo da casa e a linha pode ser usada para fazer o triângulo colocado em cima do quadrado como telhado da casa. Estes dois elementos, para constituírem um objeto total, devem obedecer a uma das operações mais elementares do espaço que é a operação de **ordem**. Por exemplo, quando não existe esta relação há inversão das relações de esquerda e direita, acima e abaixo, frente e trás. Além disso, se o sujeito quer que a árvore seja vizinha da casa, terá que desenhar estes objetos parciais colocados um ao lado do outro e, para isso, necessita ter bem clara a **operação de vizinhança**. Por exemplo: estes elementos desenhados têm que ser vizinhos um do outro, ao invés de dispersados nos dois cantos da área de edição.
- No caso da utilização do Paintbrush, para um usuário criar um desenho, ele precisa executar as seguintes ações computacionais parciais, que não precisam estar em uma ordem determinada, mas todas estas operações devem ser executadas: 1) seleção de uma cor de fundo; 2) seleção de uma cor para o primeiro plano; 3) seleção de uma largura para o desenho; 4) criação do desenho; 5) edição do desenho.
- O procedimento para digitação de um texto em uma imagem gráfica envolve as seguintes etapas, que também não necessitam estar numa ordem determinada:

1) seleção da cor de primeiro plano que será utilizada para o texto; 2) seleção da ferramenta “Texto” a partir da Caixa de Ferramentas; 3) escolha da fonte que será utilizada; 4) escolha do tamanho em pontos da fonte; 5) digitação e posicionamento do texto; 6) edição do texto.

Estes dois últimos tipos de situações são uma seqüência de operações que devem ser coordenadas pelo sujeito e, elas podem ser consideradas como geradoras de objetos parciais. Assim, estas são constituídas de operações infralógicas já que compõem um objeto total; no primeiro caso, a criação da imagem gráfica e no segundo, a digitação de um texto na área de desenho.

- Na situação do tipo 2, prepara-se uma experiência no editor. Esta se refere à montagem da figura de uma pessoa a partir de suas partes que se encontram espalhadas pela tela de edição. Ou seja, o rosto, os olhos, nariz, boca, corpo, braços e pernas de uma pessoa são colocados nos quatro cantos da tela. Pede-se ao sujeito para montar a figura através dos comandos de manipulação sobre o desenho.

O sujeito, então, precisa ativar a caixa de ferramentas e iniciar a sua atividade. Para isso, deve existir a **relação de envolvimento** que é acentuada quando se trata de figuras mais simples. O usuário então, utiliza os comandos de manipulação direta sobre o desenho para montar a figura proposta. Marca através da área de recorte a parte que deseja deslocar e, através do mouse arrasta a mesma até a posição desejada. O sujeito também pode utilizar os comandos de manipulação indireta sobre o desenho, copiando e colando os objetos parciais da figura até montar o desenho que é o objeto total.

Além disso deve existir uma **relação de continuidade**, que é uma ligação contínua dos elementos envolvidos no desenho (linhas, superfícies, etc), em oposição à justaposição dos mesmos (**relação de descontinuidade**).

- Outro exemplo: um sujeito desenvolveu um desenho composto de várias figuras, e se pede a ele que separe todas elas através dos comandos do editor. Se este utilizar a área de corte para marcar uma figura e logo deslocar cada uma delas, ele estará utilizando a **operação de separação** que intervém na medida em que os elementos desenhados são distinguidos uns dos outros, ou seja, o sujeito deve fornecer um meio para distingui-los.

5.1.2.2 As operações infralógicas formadoras das relações topológicas

Estas operações do editor gráfico são analisadas no nível da manipulação direta e indireta sobre o desenho. Como se trata das relações topológicas, é dada ênfase à quantificação intensiva, isto é, relacionar a parte (elemento parcial) com o todo (objeto total), mas não a parte com a parte. Por essa razão o sujeito limita-se a trabalhar com encaixes, ordens de sucessão e correspondências, ou seja, noções estas que não supõem nenhuma quantificação extensiva (relação de parte com parte).

Por exemplo:

- O sujeito fez uma figura qualquer e se pede a ele para separar somente uma determinada área do mesmo. Para isso ele pode selecionar a área de corte para marcar a parte que deseja separar do resto do desenho e arrastar a mesma com o mouse. Assim, o sujeito estará trabalhando diretamente sobre o desenho (MDD). Dependendo do nível de conhecimento em que se encontra o sujeito, ele poderá entrar no menu EDITAR, ativar o comando RECORTAR para apagar a área

marcada e COLAR em outra parte da tela. Assim ele estará utilizando comandos pré-definidos do editor (MID).

No momento em que o usuário desloca uma parte (objeto parcial: B') da sua figura (objeto total: C) fica com uma nova figura que é parte (B) da figura total. Assim, $B + B' = C$. Portanto, ele estará utilizando a **operação infralógica de partição e adição primitiva**.

Este tipo de operação é o equivalente, no domínio infralógico das vizinhanças, ao agrupamento dos encaixes das classes A, B, C ... no domínio da lógica.

Da mesma forma que ele pode deslocar a parte (B') que foi separada, ele também pode fazer a função inversa, associando-o novamente.

- O comando VOLTAR (MID), do menu EDITAR, se ativado logo depois que for deslocado o objeto parcial do objeto total também é uma forma de desfazer aquilo que foi feito, isto é, associa novamente B com B' formando C. Ou seja, neste tipo de operação computacional vê-se uma operação reversível de associação e dissociação dos elementos.

- Prepara-se uma experiência no editor: desenha-se uma mesa redonda no meio da tela e 5 pessoas colocadas na parte superior da tela de edição dispostas em uma fileira. Pede-se ao sujeito que coloque as pessoas em volta da mesa, utilizando as ferramentas do Paintbrush. Ele pode deslocar cada homenzinho utilizando a área de recorte e arrastando com o mouse até a posição desejada, agrupando todas as pessoas em volta da mesa (MDD).

- Uma outra forma de realizar esta operação seria propor ao sujeito desenhar varias pessoas em volta de uma mesa. Desta forma, ele teria que utilizar a caixa de ferramentas (círculo ou elipse) para desenhar a mesa, e depois montar uma pessoa com as ferramentas do editor. Depois de desenhada a primeira figura, ele poderá duplicar a mesma, sem necessidade de fazê-la novamente. Simplesmente utilizando os comandos de MID do editor. Assim, marcando a pessoa, o sujeito poderá acessar o menu de EDIÇÃO, COPIAR e COLAR essa figura, para depois colocá-la em volta da mesa. Essa é uma outra forma de analisar a operação infralógica de **ordem cíclica**, utilizando as operações computacionais do editor gráfico. Cabe destacar que não foi analisada a operação de duplicação do objeto parcial, através dos comandos pré-definidos.

- Da mesma forma o sujeito pode trabalhar com uma **seqüência linear**. Neste caso, sugere-se que ele desenhe várias figuras (vestido (v), casaco (c), bolsa (b), saia (s), calça (a)) e se pede que as alinhe numa seqüência. Isto é, que as coloque numa certa ordem, no varal (vcbsa). Depois pede-se ao sujeito que faça a operação contrária, ou seja, faça a inversão dessa seqüência (asbcv). Assim, através do deslocamento das figuras, utilizando a área de corte e arrastando os objetos do desenho com o mouse (MDD), o usuário estará utilizando as relações de vizinhança e separação. Ou seja, a coordenação crescente das ações que consistem em deslocar (transportar mentalmente) e recolocar os objetos, resultam na representação da ordem, seja esta linear ou cíclica.

- Prepara-se uma tela ao sujeito onde ele precisa resolver um problema utilizando as operações do editor. São desenhadas várias figuras divididas em dois pedaços, tipo quebra-cabeças, dispostas em duas colunas. Além disso existem partes de figuras que não mantêm relação com nenhuma outra. Pede-se ao sujeito ligar essas colunas relacionando as partes, mostrando quais os elementos que se ligam entre si, apagando as partes que não possuem correspondência com outras. Assim, o sujeito deverá montar uma rede de duas dimensões utilizando, por exemplo, o

pincel da caixa de ferramentas (MDD). Com este, ele poderá fazer uma correspondência biunívoca, estabelecendo uma rede tal que: os elementos A, B e C,... de uma seqüência (coluna) sejam colocados em correspondência com os elementos A₁' , B₁' , C₁' ,... da outra seqüência, estabelecendo uma certa conexão entre A₁ e A₁' , entre A₁ e A₂' e assim por diante. As partes que não possuem nenhum tipo de relação poderão ser apagadas através dos comandos do menu EDITAR, RECORTAR (MID). Dessa forma pode-se dizer que o sujeito estará fazendo uma **multiplicação biunívoca de relações**.

5.1.2.3 As operações infralógicas e o espaço euclidiano

Aqui são apresentadas e analisadas algumas operações infralógicas do sujeito em interação com o editor gráfico, desde o ponto de vista da representação euclidiana.

As coordenadas do espaço euclidiano consistem em relações de ordem aplicadas à três dimensões ao mesmo tempo: esquerda x direita, acima x abaixo, frente x trás, formando uma rede. Por exemplo:

- As partes de um homem encontram-se desenhadas e espalhadas pela tela. Além disso, esses elementos não se encontram na sua posição correta, ou seja, a boca está em posição vertical, um braço está voltado para baixo e o outro voltado para cima, o nariz está virado e as cores de um dos olhos está invertidas. O homem (H - objeto total) deverá ser composto dos vários elementos (objetos parciais): cabeça (C), nariz (N), boca (BC), olhos (O = O₁ + O₂), corpo (CO), braços (B = B₁ + B₂), pernas (P = P₁ + P₂), obedecendo a posição correta dos mesmos. O sujeito precisa relacionar as partes com as partes, tendo assim, a visão do todo. Para isso, necessita trabalhar com o sistema de coordenadas, isto é, com as noções de rotação e translação dos elementos (deslocamentos), da vertical e horizontal, entre outras. Utilizando os comandos de MDD e de MID, ele será capaz de:
- ativar a caixa de ferramentas para marcar a área que deseja rotacionar, por exemplo, a boca. Acessa o menu MODIFICAR, ROTAÇÃO HORIZONTAL, e coloca a boca na posição correta, isto é, ela estava em posição vertical e o sujeito, através dos comandos de MID, a rotaciona e a desloca até o lugar onde será montado o objeto total. Logo depois utiliza o mesmo menu, para fazer as rotações dos braços, a fim de posicioná-los adequadamente e os deslocamentos dos mesmos até o espaço onde se encontram as outras partes do desenho. O nariz é virado através dos comandos ROTAÇÃO VERTICAL do menu MODIFICAR (MID) e juntado à figura total. Um dos olhos é marcado através da área de corte, ativado o menu MODIFICAR, e a operação de INVERTER CORES. Assim, o sujeito monta o desenho completo (H), adicionando todos os elementos do mesmo:

$$H = C + rv(N) + rh(BC) + ((O_1 + ic(O_2)) + CO + ((rv(B_1) + rh(B_2)) + (P_1 + P_2)),$$

onde:

rv: rotação vertical;

rh: rotação horizontal e,

ic: inverte cores.

Assim, o sujeito utiliza a operação infralógica de **adição de elementos**, assegurando sempre a conservação do todo, tanto a título de localização quanto de figura do objeto colocado.

- Um outro ponto importante de verificar é a questão da proporção entre as partes do objeto. Se, por exemplo, a boca está do tamanho do braço, o sujeito deverá diminuí-la, utilizando os comandos REDUZIR + AMPLIAR do menu MODIFICAR (MID), colocando essa parte em **proporção com as outras**, ou seja, o sujeito tem que utilizar a relação de proporcionalidade.
- Uma outra operação que pode ser executada utilizando o mesmo exemplo, é caso se proponha ao sujeito dissociar as partes da figura do Paintbrush. Para isso, o sujeito pode utilizar as áreas de corte e os comandos de manipulação indireta sobre o desenho, RECORTAR, COLAR e COPIAR, ou ainda, a separar os elementos através do mouse. Portanto, o objeto é interpretado em função de sua forma ou de sua localização, como uma “figura do espaço”, dentro de um sistema de localizações. Estas localizações são as relações entre esses elementos. A **operação de subtração de elementos** consiste em dissociar as partes da figura considerada. No caso, $H = C + N + BC + O + CO + B + P$, e, se separar o braço B1 ou B2, por exemplo, fica: $B - B1 = B2$ ou $B - B2 = B1$ ou ainda, a cabeça do resto do resto do corpo: $H - C = N + BC + O + CO + B + P$. Pode-se fazer também a operação inversa que é a de **adição de elementos**.
- Quando essa mesma experiência é preparada na tela de edição, os diversos elementos se encontram separados uns dos outros nos quatro cantos da tela. Pede-se ao sujeito para colocá-los numa certa ordem, portanto, estes elementos são ditos colocados uns em relação aos outros, por exemplo:
 $C \rightarrow (\text{que contém } N \rightarrow BC \rightarrow O \text{ (relação de envolvimento)}) \rightarrow CO \rightarrow B \rightarrow P$
 $H = C + N + BC + O + CO + B + P$
 Neste caso, a operação inversa não é somente inverter a seqüência $P \rightarrow B \rightarrow CO \rightarrow (O \rightarrow BC \rightarrow N) \rightarrow C$, mas sim, fazer um “deslocamento” que pode inverter essa seqüência inteira ou deslocar somente um elemento qualquer para outra posição:
 $H = C + N + O + BC + CO + B + P$
 No espaço euclidiano, há uma diferença entre a ordem dos elementos e a ordem das colocações. Essa operação é denominada de **colocações e deslocamentos** e ela pode ser realizada somente através dos comandos de MDD ou, se o sujeito preferir, através do menu EDITAR, através das operações de COPIAR E COLAR.
- Os comandos MAIS ZOOM (MZ) e MENOS ZOOM (mZ), do menu VISUALIZAR (MID), são operações que tem que obedecer uma certa ordem de colocação, isto é, sempre que ativar qualquer um deles, é preciso ativar o outro para voltar à posição inicial (PI), ou vice-versa. Ou seja, $MZ \Leftrightarrow mZ$ ou $mZ \Leftrightarrow MZ$; $MZ + mZ = PI$ assim como $mZ + MZ = PI$. Daí as ordens direta e inversa entre os elementos e trabalhar com estes comandos é sinônimo de trabalhar com um sistema de referências.
- Um outro experimento interessante de analisar as operações infralógicas que constituem o espaço euclidiano, é desenhar na tela uma montanha e várias figuras espalhadas na tela de edição como: casa, árvore, edifício, carro. Pede-se ao sujeito que as coloque em cima da montanha. Se elas estiverem mal posicionadas, podem ser utilizados os comandos do editor para reposicioná-las. Algumas das figuras estão inclinadas e outras não. Se o sujeito já se encontra no final do estágio de desenvolvimento da horizontal e da vertical, ele utilizará a operação de INCLINAR, do menu MODIFICAR, colocando as figuras em posição vertical (retas). A partir de então, utilizando o comando de deslocamento, ela poderá

posicionar as diversas figuras na montanha. Se ela ainda não atingiu a noção de horizontal e vertical, deixará as figuras inclinadas, acompanhando a linha da montanha.

- Outro experimento para analisar as operações infralógicas é, além do sujeito ter que posicionar as figuras (inclinando-as ou não) através de operações computacionais, apresentar-lhe algumas delas numa escala maior ou menor. Dessa forma o sujeito terá que REDUZIR + AMPLIAR as figuras, no menu MODIFICAR, para que estas formem um todo coerente, ou seja, ela deverá relacionar as partes do desenho, colocando os elementos do mesmo em um tamanho que seja comum a todos eles.

Esses tipos de operações infralógicas realizadas nestas últimas duas situações, dizem respeito às relações **distância-proporção -posição** .

Neste estudo, não serão abordadas as operações infralógicas que constituem o espaço projetivo, por uma limitação do editor gráfico.

5.2 Editores de texto

A “escrita computacional” é uma nova concepção de representação da linguagem, já que o sujeito utiliza os recursos das ferramentas computacionais para editoração de textos. Assim, o sujeito deixa de usar o método “convencional” para representar o seu conhecimento de forma textual, para se apropriar de uma “nova forma de escrita”, onde não existe mais necessariamente aquela seqüencialidade na forma de manifestar as suas idéias.

Os editores de texto são programas que permitem que o sujeito registre as suas idéias no computador, codificando-as de forma escrita.

Com o uso de editores, o sujeito pode copiar, recortar, unir partes de textos, “re-organizando” constantemente a sua forma de pensar, a medida que vai desenvolvendo a sua atividade.

Apesar de sua apresentação final ser em forma linear, o seu desenvolvimento passa por diversas etapas que não são necessariamente escritas seqüencialmente.

Essa é uma das razões que mostram porque que a “escrita computacional” é baseada em um “novo paradigma”, isto é, no que se refere a sua forma de desenvolvimento [BEH 96b].

Também cabe destacar que a maneira como é construído o sistema de representação de escrita no computador toma, muitas vezes, a forma de um hipertexto. Ou seja, o sujeito pode escrever partes do documento sem nenhuma ligação, que depois se dá conta que estas tem uma certa relação e, assim, junta ou separa estas partes no final da apresentação. Logo, isso não deixa de ser uma “forma de navegar” no texto que está sendo desenvolvido, apesar do produto final ser em forma linear.

Neste item é descrito o Word 6.0 de **editoração textual**, do Windows 3.1, mostrando uma das formas de realizar a análise operatória do mesmo.

Também é analisado esse “novo paradigma” de escrita computacional através dos **sistemas de hipertextos**.

5.2.1 Análise lógico-operatória do editor de texto Word 6.0

A análise das operações diretas e/ou indiretas tanto de controle do sistema quanto de manipulação da representação foi realizada de forma completa no capítulo anterior, para situar o leitor no contexto descrito. Como na análise operatória somente interessam as operações lógicas e/ou infralógicas da ferramenta através da ação do sujeito sobre esta ou sobre a sua atividade, neste item não serão tratados os comandos de controle sobre o ambiente.

Utilizando o modelo geral apresentado na figura 10 (cap. 3.4), a seguir é descrito o modelo específico de um sujeito em interação com um editor de texto, analisado do ponto de vista operatório. Este é mostrado na figura 25.

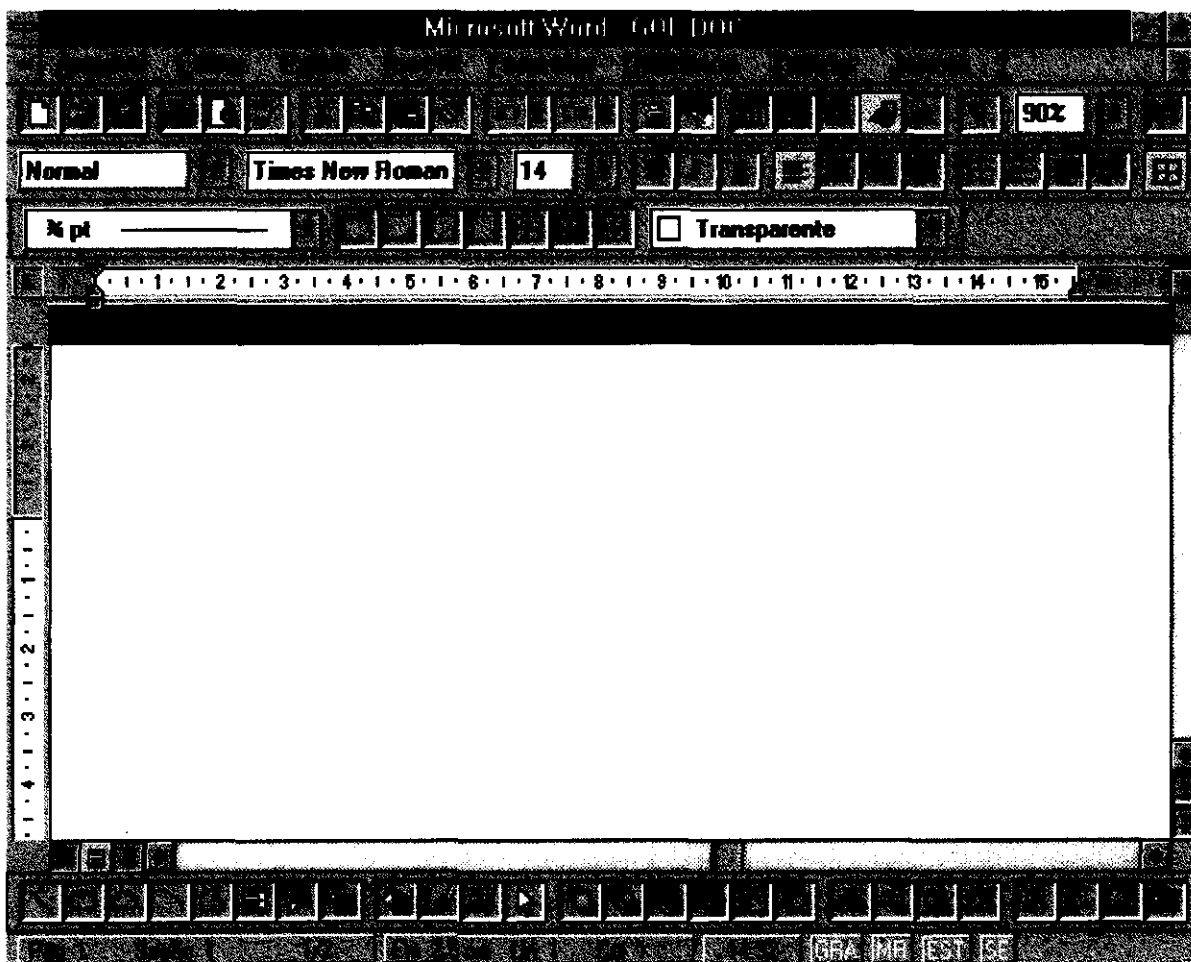


FIGURA 25 - Interface do Editor de texto Word 6.0

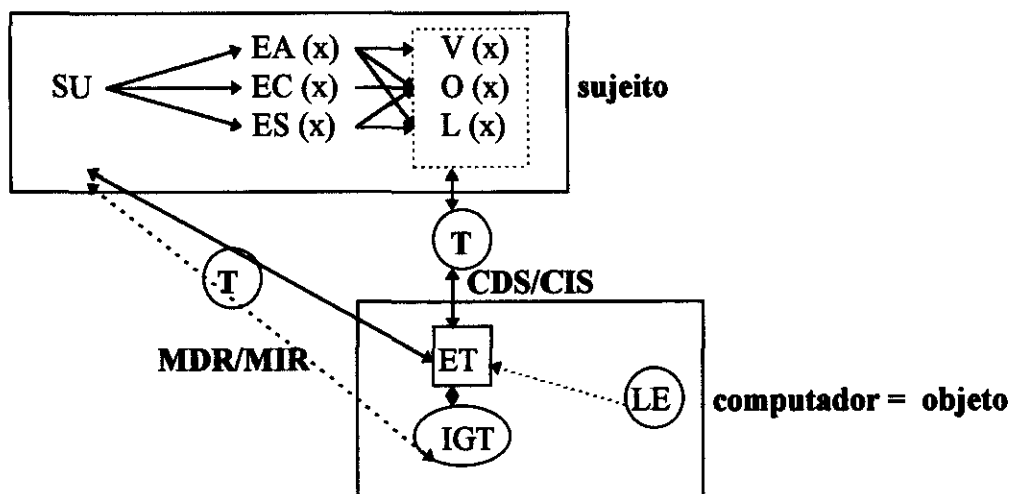


FIGURA 26 - Modelo de interação sujeito x editor de texto

onde: SU: sujeito usuário; ET: editor de texto; T: tela, teclado, etc...(canal de comunicação sujeito-ferramenta); IGT: imagem gráfica textual: representação individual; LE: linguagem escrita; CDS: controle direto do sistema; CIS: controle indireto do sistema; MDR: manipulação direta sobre a representação (texto) e, MIR: manipulação indireta sobre o texto.

Seguindo o mesmo modelo especificado na figura 9, apresentamos a interação de um sujeito com uma imagem gráfica representada em forma textual, através de um editor de texto. Logo, o sujeito possui a “imagem mental” de algo que deseja representar e, utiliza a sua própria linguagem natural escrita para representá-la, no computador, em forma de imagem gráfica textual (IGT).

O quadro resultante é mostrado na figura 26.

No que se refere à “escrita computacional” em um editor de texto, a análise operatória pode ser feita à nível lógico e/ou infralógico.

Em relação às operações lógicas, o documento escrito pode ser **classificado** através da ativação do comando CLASSIFICAR TEXTO, do menu TABELA (operação lógica de classificação), em parágrafos, campos, e tipos como texto, data e número (MIR).

Também o texto pode ser classificado pelo seu ESTILO, no menu FORMATAR, como manuscrito, artigo, formato tese, anuncio, carta, entre outros (MIR).

Outra operação lógica é a **seriação** no que se refere à estrutura do texto desenvolvido pelo sujeito (MDR). Além de seguir uma certa ordem, devem ser padronizadas todas as partes do texto, através da formatação dos capítulos, seções, tabelas, etc, do mesmo. Esses comandos podem ser ativados através do menu FORMATAR (MIR).

Para verificação da sintática, existe no editor o comando de VERIFICAÇÃO DE ORTOGRAFIA, do menu UTILITÁRIOS. Este pode ser utilizado pelo sujeito como uma operação lógica, referente à **sintaxe intratexto**.

Na estrutura do texto também deve existir uma **correspondência** em relação ao índice ou sumário e o número das páginas, e seu respectivo capítulo, seção, tabela, entre outros, que compõem o documento (parte com parte) (MDR).

Quanto às operações infralógicas, estas também se referem à estrutura do texto, mas no que diz respeito à relação de fazer corresponder as diversas partes com o todo (texto completo). Isto é, deverá existir uma clareza na organização do texto e uma

apresentação coerente e consistente. Isso dependerá da relação parte com o todo e parte com parte (**quantificação intensiva e extensiva**).

No que se refere à MDR, o usuário opera sobre o texto desenvolvido, executando operações diretas sobre ele. Por exemplo:

- **ARQUIVO: NOVO:** define o modelo do arquivo, para iniciar uma atividade nele;
- **FORMATAR: FONTE:** definir o tipo de fonte de letra e espaçamento entre caracteres; **PARÁGRAFO:** definir recuos, espaçamento e o fluxo do texto; **TABULAÇÃO:** escolher a marca de tabulação e o deslocamento (em cm) referente à tabulação; **BORDAS E SOMBREAMENTO:** definir tipos de bordas e sombreamentos; **COLUNAS:** especificar quantidade de colunas, largura e espaçamento; **MAIÚSCULAS E MINÚSCULAS:** define toda a sentença, ou a área marcada em minúsculas ou maiúsculas; **CAPITULAR:** definir fonte, tamanho da letra e distância do texto; **MARCADORES E NUMERAÇÃO:** define os tipos de marcadores, numeração ou múltiplos níveis; **GALERIA DE ESTILOS:** mostra toda a galeria de estilos, podendo inserir outras; **ESTILO:** define estilo do documento; **OBJETO DE DESENHO:** define padrões de desenho.

A manipulação indireta sobre o texto (representação) - **MIR** - refere-se à execução dos comandos do sistema, ativados pelo usuário, alterando a imagem textual que se encontra na tela.

Podemos especificar os seguintes comandos realizados pelo editor:

- **ARQUIVO: CONFIGURAR PÁGINA:** define as margens, tamanho do papel, origem do papel, layout de como vai sair o texto a ser desenvolvido; **SALVAR, SALVAR COMO, SALVAR TUDO; LOCALIZAR ARQUIVO; RESUMO INFORMATIVO:** quando ativado pelo sujeito, o sistema mostra um resumo do assunto do arquivo em questão; **MODELOS:** mostra o modelo do documento, o usuário pode anexar outros modelos, ou redefinir o atual, fazendo uma busca na biblioteca de modelos; **VISUALIZAR IMPRESSÃO:** mostra como o documento vai ser impresso;
- **EDITAR: VOLTA** a última ação executada pelo usuário através do sistema; **RECORTAR; COPIAR; COLAR; COLAR ESPECIAL; LIMPA; SELECIONA TUDO; LOCALIZA; SUBSTITUI; IR PARA.**
- **INSERIR: QUEBRA:** de página ou de coluna; **NÚMEROS DE PÁGINA; ANOTAÇÃO; DATA E HORA; CAMPO; SÍMBOLO; CAMPO DE FORMULÁRIO; NOTAS:** de rodapé e de fim; **LEGENDA; REFERÊNCIA CRUZADA; ÍNDICES; ARQUIVO; MOLDURA; FIGURA; OBJETO; BANCO DE DADOS**
- **UTILITÁRIOS: VERIFICAR ORTOGRAFIA; DICIONÁRIO DE SINÔNIMOS; HIFENIZAÇÃO; IDIOMA; CONTAR PALAVRAS; AUTOCORREÇÃO,** entre outras;
- **TABELA:** operações referentes ao desenvolvimento de uma tabela; **CLASSIFICAÇÃO TEXTO;** entre outras.

TABELA 2 - Operações realizadas pelo sujeito em interação com o editor de texto

	INTERAÇÃO COM O AMBIENTE	INTERAÇÃO COM O TEXTO
MANIPULAÇÃO DIRETA	o usuário interage através de operações diretas com o ambiente, isto é, executa ações sobre o mesmo, afetando o sistema. Ex: muda o formato das janelas do editor. Quem define o tamanho das janelas da interface é o sujeito e não o programa.	o sujeito interage diretamente sobre o texto, isto é, afetando não o ambiente, mas a imagem gráfica. Ex: o usuário altera o texto original, marcando uma parte do texto e apagando a mesma, ou, ainda, inserindo uma parte nova.
MANIPULAÇÃO INDIRETA	o usuário interage indiretamente com o sistema, ou seja, executa comandos pré-definidos que se referem somente ao ambiente. Ex: o sujeito ativa o comando SALVAR, do menu ARQUIVO para armazenar o seu texto. Quem faz a operação de salvar é o próprio editor.	o usuário executa comandos do ambiente que mudam a imagem, ou seja, além dele executar ações sobre o ambiente necessita das operações pré-definidas do mesmo para alterar o texto. Ex: o sujeito marca uma área determinada do texto, ativa o comando RECORTAR E COLAR, do menu EDITAR para copiá-la em outra parte do texto. Quem faz esta ação é o editor e não o usuário.

Portanto, foi apresentada uma das classificações encontradas para as operações lógicas e/ou infralógicas do editor de texto. Para isso, foi construído e analisado um modelo que reflete esta classificação.

5.2.2 Hipertextos

Os hipertextos surgiram com o propósito de permitir aos sujeitos trabalhar em sistemas onde é oferecido um novo mecanismo de extração e recuperação de dados através de diversos caminhos [WOO 90].

Um hipertexto é uma ferramenta que permite ao sujeito relacionar suas idéias, através de enlaces que auxiliam na “lembança”, facilitando a formação de conceitos. Como este tipo de sistema oferece estruturas não lineares para o desenvolvimento do texto, os conceitos do sujeito podem ser relacionados por associações, fazendo com que o mesmo construa o seu sistema de representação através da escrita computacional, baseando-se nas suas necessidades, interesses e motivações.

Os hipertextos permitem que o sujeito adicione novas informações através de anotações, combinando o uso de diferentes meios e formas de representação. Assim, este tem a liberdade de “navegar” no ambiente, organizando suas idéias, efetuando um processo de ação e construção do seu pensamento lógico.

Utilizando como exemplo, o protótipo de hipertextos construído por Maenza [MAE 94], denominado SHIPOI (Sistema de Hipertexto Para Organização de Idéias), analisaremos algumas operações lógicas e/ou infralógicas encontradas nesta “nova concepção” de escrita.

Neste protótipo o sujeito elabora um caderno de idéias, podendo ou não acessar a base de dados do sistema para extrair algum tipo de informação. A partir dela, o usuário pode fazer enlaces que indiquem o relacionamento entre diversos conceitos. Naturalmente, o sujeito pegará as ligações que ele acredite serem as mais significativas.

Para construir a sua apresentação, isto é, seu “hipertexto”, o sujeito deve utilizar entre outras, as **operações lógicas de classificação e seriação**. Portanto, ele classifica e seleciona os conceitos que formam o seu texto, efetuando uma seriação ao criar os enlaces que os relacionam.

Assim, este trabalha com uma estrutura do tipo rede, podendo fazer uso das propriedades de **inversão**, quando ele volta ao conceito (nó) anterior, através de recursos oferecidos por este tipo de ambiente e, ainda, **associatividade**, podendo utilizar mais de um caminho para chegar num mesmo nó.

Este é um breve relato de um protótipo de um hipertexto, no qual o sujeito pode formular idéias, captando relações entre elas, organizando-as e operando com classes.

Nesse sentido, vê-se que um texto escrito com papel e lápis, passa por um processo de evolução quando é transferido para a escrita computacional. Assim, o desenvolvimento de um texto, utilizando um processador de textos, permite uma “outra estrutura operatória” de pensamento e, mais ainda, tratando-se de um hipertexto, onde o número de operações lógicas e infralógicas usadas pelo sujeito pode aumentar consideravelmente, através da construção de estruturas integradas, dependentes e dinâmicas.

Dessa forma pode-se observar como os textos convencionais escritos de forma linear, ficam sobrepostos e interligados, mudando a “concepção do sistema de escrita”, através dos sistemas de hipertextos.

5.3 Ambientes de programação

5.3.1 Ambiente de programação LOGO

A linguagem de programação possibilita ao sujeito representar e manipular seus próprios esquemas de procedimentos, ou seja, suas ações em seqüências.

Portanto, um dos principais aspectos a serem enfatizados no que se refere à análise operatória de um ambiente de programação, é a própria estrutura de um programa. Ou seja, um programa possui uma determinada **forma** e um certo **conteúdo**, específico dele. Portanto, essa análise detalhada sempre dependerá da linguagem utilizada, da estrutura em que foi programado o algoritmo e, principalmente, do seu conteúdo.

O modelo de interação de um sujeito com o ambiente de programação LOGO é mostrado na figura 27:

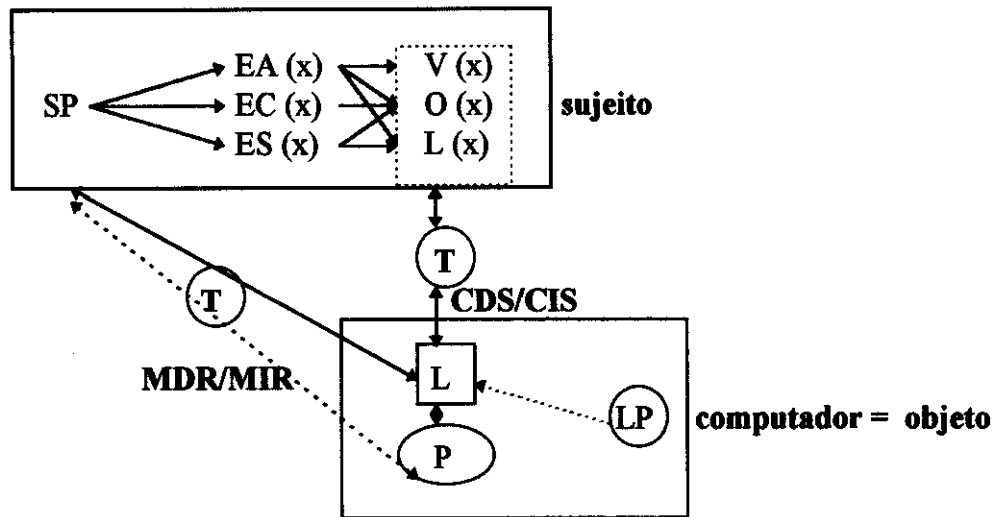


FIGURA 27 - Modelo de interação sujeito x ambiente de programação LOGO

onde: SP: sujeito programador; L: ambiente LOGO; T: tela, teclado, etc... (canal de comunicação sujeito-ferramenta); P: programa: representação individual; LP: linguagem de programação LOGO; CDS: controle direto do sistema; CIS: controle indireto do sistema; MDR: manipulação direta sobre a representação (programa) e, MIR: manipulação indireta sobre o programa.

O próprio ato de programar significa organizar as operações lógicas e infralógicas do pensamento do sujeito. Ele sempre irá compor módulos (partes) de um programa (todo) e, conseqüentemente, estará utilizando as **operações infralógicas**, compondo um objeto total, a partir de seus elementos. E assim sucessivamente, porque este programa pode ser, em outro momento, o módulo de um segundo, e estes dois fazerem parte de um terceiro. Logo, o sujeito sempre estará operando com objetos parciais.

Dependendo da forma como o sujeito irá estruturar o conteúdo do programa, ele poderá realizar transformações como a de **inversão**, **reciprocidade**, **correlação** ou, ainda, de **negação**.

A função semiótica do sujeito evoca-se no momento em que este simboliza, através de uma linguagem de programação, as seqüências representadas parcialmente, em um todo simultâneo (programa).

Quando um sujeito simplesmente preenche a tela com letras ou até com riscos quaisquer, pode ser comparado com o mesmo fato que acontece quando este traça linhas utilizando papel e lápis, não atribuindo nenhum significado a seus grafismos, isto é, não comporta um simbolismo e simplesmente consiste em repetir, pelo simples prazer. São as chamadas **garatujas**. Aqui não é encontrada nenhuma operação lógica e nem infralógica.

Já num segundo momento, o sujeito pode iniciar o processo de produção de um determinado desenho. Para isso, ele tem que trabalhar com o sistema de coordenadas e, conseqüentemente, com as operações infralógicas de **colocações** e **deslocamentos**.

Quando o sujeito compreende a noção de programa, ele explora uma regra de **organização** da escrita do programa, isto é, segue uma determinada seqüência, ou seja, as operações devem obedecer uma certa **ordem de seriação**.

TABELA 3 - Operações realizadas pelo sujeito em interação com o ambiente LOGO

	INTERAÇÃO COM O PROGRAMA
MANIPULAÇÃO DIRETA	o sujeito interage diretamente com o programa, inserindo, alterando ou deletando comandos do mesmo, isto é, afetando não o sistema, mas somente o programa. Ex: o usuário modifica o programa que desenha um quadrado: pf 10 gd 90 pf 10 gd 90 pf 10 gd 90 pf 10 gd 90, inserindo mais instruções, como, por exemplo: pf 10 gd 45 pf 10 gd 45
MANIPULAÇÃO INDIRETA	o usuário ativa os comandos do ambiente para executar um programa, ou seja, além dele definir o programa necessita das operações pré-definidas do próprio sistema para mandar interpretá-lo e executá-lo.

5.3.2 Ambiente de programação Winlogo

Nesta análise considera-se o ambiente Winlogo desde o ponto de vista das operações realizadas por um sujeito programador qualquer em interação com o mesmo, assim como as próprias operações que podem ser executadas pelo sistema.

O WinLogo é um ambiente integrado de programação e aprendizagem em linguagem LOGO [WIL 92]. Ele é composto por uma interface gráfica para comunicação com o usuário, possui menus descendentes, janelas para os diferentes ambientes de trabalho, e o mouse para facilitar a utilização de todos estes recursos. Além disso este software tem várias opções em termos de cor e resolução. Existem ferramentas que facilitam a criação de micromundos e aplicações. A interface do sistema pode ser visto na figura 28.

Este ambiente trabalha exclusivamente com janelas. As mais importantes são as de: Gráficos, Textos e de Trabalho. Existem também outras janelas inicialmente ocultas que podem ser ativadas pelo sujeito como, por exemplo, a área de Formas, do Editor, do Verificador, de Variáveis e de Ajuda.

Um programa (totalidade) é construído através das operações infralógicas, ou seja, através de objetos parciais (símbolos que constituem as instruções ou os comandos). Este objeto, além de ser fruto de uma representação interna do sujeito programador (das suas imagens, relações, operações, composições, coordenações), também é manipulável, ou seja, pode ser guardado, recuperado e modificado por ele. Portanto, se trata de um objeto externo, objetivo, com permanência independente do sujeito e é dinâmico.

Na nossa interpretação, o sujeito programador, à medida que vai utilizando um sistema, vai realizando “operações computacionais” e, conseqüentemente, desenvolvendo estruturas que exigem o uso de códigos e símbolos próprios do ambiente. Através desta interação entre a ferramenta, que é o ambiente, e o sujeito, podem ser identificados dois tipos de “operações computacionais”: as que são feitas sobre o próprio ambiente (OLWL- operações lógicas do Winlogo e OIWL - operações infralógicas do Winlogo) e

aquelas realizadas pelo sujeito na interação com o sistema através do uso dos comandos da linguagem, que se dividem em comandos básicos de interação (*OLCB e OICB - operações lógicas e infralógicas de comandos básicos*) e os comandos de controle da interação (*OLCI e OICI*), como será visto no modelo da figura 29.

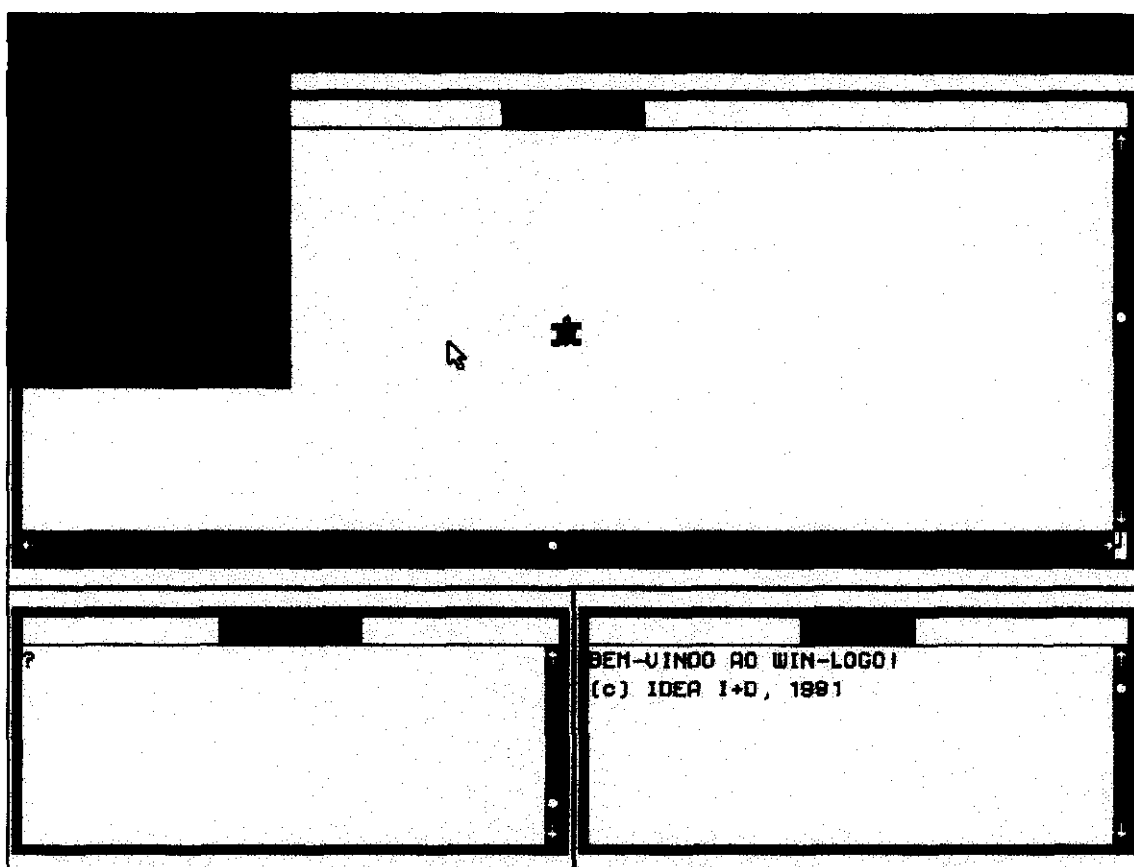


FIGURA 28 - Interface do ambiente de programação Winlogo

Levando em conta unicamente a interação do sujeito programador com o sistema através do uso da linguagem, enfatizaremos dentro desse tipo de relação, a combinação de comandos básicos da linguagem feita pelo programador. Esta trata dos comandos de controle da interação (organização da interação), ou seja, é através deles que o programador realiza as composições de comandos básicos (ações) de interação. Por essa razão, faremos essa distinção entre as operações lógicas e infralógicas de controle da interação (comandos básicos combinados através dos comandos de controle) e as operações lógicas e infralógicas em relação aos comandos básicos da linguagem. Esta idéia nos serviu como embasamento para analisar operatorialmente os comandos tanto do ambiente, do sujeito e do processo de interação entre os dois.

Assim, o modelo construído foi baseado na tríade sujeito (S) - computador (C) - valor (V), objeto (O) e linguagem (L), isto é, a interação do sujeito com um ambiente computadorizado pode ser lida da seguinte forma:

- um sujeito (S) representa seus valores, objetos e/ou linguagem em forma de programas (P), interagindo com o ambiente WinLogo e utilizando a linguagem Logo de programação. Só que estes valores, objetos e/ou linguagem são

representados através da imagem mental do sujeito em relação ao processo interativo do sujeito usuário (Su) com o programa que está sendo desenvolvido. As operações são classificadas em infralógicas e lógicas no que se refere ao controle do ambiente (OIWL/OLWL), aos comandos básicos (ações) da interação (OICB/OLCB) e à organização de comandos básicos através dos comandos de controle da interação (OICI/OLCI).

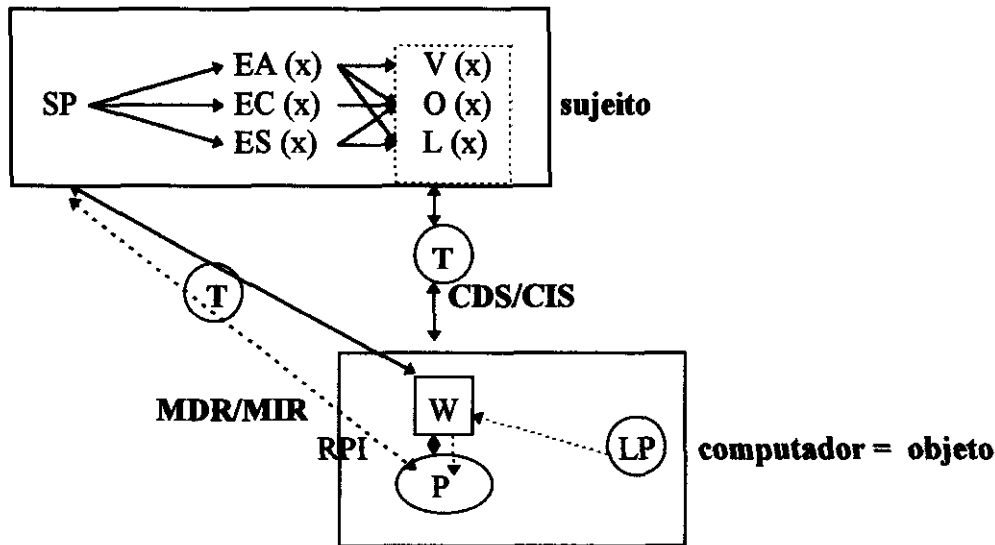


FIGURA 29 - Modelo de interação sujeito x ambiente Winlogo

onde: SP: sujeito programador; W: ambiente WINLOGO; T: tela, teclado, etc...(canal de comunicação sujeito-ferramenta); P: programa: representação individual; LP: linguagem de programação WINLOGO; CDS: controle direto do sistema; CIS: controle indireto do sistema; MDR: manipulação direta sobre a representação (programa) e, MIR: manipulação indireta sobre o programa; RPI: representação do processo de interação do sujeito usuário com o programa.

5.3.2.1 Operações do processo interativo sujeito usuário e/ou programador x Ambiente Winlogo

Nesta seção são descritas as operações que representam o processo de interação do ambiente com o sujeito usuário e/ou programador. O sujeito programador utiliza comandos pré-definidos da linguagem combinados com operações de controle da interação, a fim de organizar e coordenar essas operações básicas. Estas podem se referir à consulta, competição, a interrupção da execução de um procedimento, a sincronização, etc, entre o ambiente e o sujeito usuário.

Neste estudo não são apresentados todos os comandos de manipulação do ambiente e da linguagem de forma exaustiva; somente são analisados operatoricamente aqueles que foram considerados mais importantes.

A seguir são descritas algumas operações lógicas e infralógicas de controle do ambiente e da linguagem.

5.3.2.1.1 Operações de controle do ambiente

Neste item são descritas algumas janelas e menus do ambiente em que foram analisadas as operações consideradas mais importantes.

1) Áreas:

♦ **Trabalho:** é o canal de comunicação entre o usuário e o ambiente, isto é, o lugar onde o sujeito escreve as ordens que deseja que o sistema realize.

Esta janela permite corrigir um texto que foi escrito, fazendo as modificações necessárias e executando novamente o que foi alterado, para que o Winlogo o reinterprete. Pode-se observar então, que esta operação se refere à **operação de reversibilidade**, isto é, volta-se atrás e se refaz o que foi escrito.

A opção **interpreta** é uma poderosa ferramenta no momento de depurar procedimentos. Pode-se selecionar um bloco de instruções e pedir ao Winlogo que o interprete. Pode-se também selecionar em um conjunto de primitivas, somente aquelas que se deseja executar (**operação lógica de seleção e classificação**).

♦ **Textos:** é através desta janela que o ambiente se comunica com o sujeito. Ele manda todas as mensagens quando se define um procedimento ou se detém a execução e ainda, as mensagens de erro. Também aparecem nesta janela os resultados das operações de escrita. É possível ativar a janela de erros, isto é, as mensagens deixam de aparecer na janela de textos e passam a aparecer no meio da tela, de forma que não pode ser ignorada (**operação de correspondência**).

♦ **Gráficos:** é o mundo das tartarugas. Nela aparecem representados todos os traços e estilos, os cenários e a escrita gráfica. Este mundo é uma superfície plana e limitada, ou seja, se trabalha com um espaço euclidiano. Existem diversas formas de representação gráfica. Estas formas se referem ao efeito que causará sobre os gráficos um aumento ou diminuição do tamanho da janela. Portanto, pode-se mudar a **forma de representação**. Os modos de representação são o natural, proporcional e o modo à escala que se referem à **operação lógica de proporcionalidade/escala** do ambiente.

♦ **Editor:** esta é a área de tratamento de texto, onde podem ser editados os procedimentos do sujeito ou até escrever uma história. Winlogo incorpora nesta janela todas as facilidades de um processador de texto para que, o trabalho do sujeito seja mais rápido e eficiente. Aqui podemos mencionar as operações lógicas que podem ser efetuadas na produção de um texto, como a **seriação**, no que se refere à seqüencialidade que deve seguir um texto, a **correspondência** e a **continuidade** entre partes de um texto, a **operação de verificação sintática intratexto**, entre outros. Através desta janela pode-se trabalhar com blocos. Bloco é um conjunto de caracteres que pode conter desde um só caracter até várias páginas. É preciso identificar o bloco, através da sua marcação e, a partir disso, pode-se copiar, apagar, mover ou enviar uma cópia do mesmo para a janela de Trabalho, ou até mesmo desmarcar o bloco que foi marcado (**operação lógica de seleção e operação infralógica de colocação e de deslocamento**). Nesta janela, o Winlogo permite substituir palavras ou frases, através da opção Procura/Substitui do menu Localizar. Nesta é apresentada uma caixa de diálogo, onde é possível solicitar a procura ou a substituição de texto (**operação de colaboração sujeito/programa**).

♦ **Verificador:** proporciona uma ferramenta de controle sobre a execução do interpretador do Winlogo. Através desta janela pode-se seguir passo a passo o caminho que segue o interpretador ao executar as instruções de um procedimento. Nesta janela existe a opção Animação para observar o caminho lógico que seguem os procedimentos do sujeito, com as respectivas chamadas a subprocedimentos e os valores dos parâmetros de cada um, em cada momento da execução. Para deter a

execução num determinado ponto, pode-se acrescentar uma marca de interrupção (**operação de interrupção da execução**).

- ♦ **Formas:** permite, através do seu editor, definir novas formas para a Tartaruga, quer criando-as novamente ou modificando as que já existem. Winlogo possui um conjunto de 128 formas pré-definidas. Ao carregar o sistema, a Tartaruga adota uma forma de acordo com a sua orientação. Existe um comando que permite, através da lista que se tem como parâmetro, fazer a associação forma-orientação, que é FixaForma (**operação infralógica de colocação e deslocamento**).

- ♦ **Variáveis:** esta permite, através de sua janela, observar como muda o conteúdo das suas variáveis, à medida que se executam os procedimentos do sujeito. Esta área também pode utilizar-se como colaborador do verificador, ou seja, é uma **operação de correspondência entre ponto de controle** (no texto do programa) e **valores das variáveis**.

2) Menus:

- ♦ **Menus Áreas e de Edição** aparecem todas as opções anteriormente descritas.

- ♦ **Menu Formas:** existem as opções de desloca acima, desloca abaixo, desloca direita e esquerda, simetria horizontal e vertical, simetria diagonal um, simetria diagonal dois, roda direita, roda esquerda, inverte, apaga, copia, e desfaz (**operação infralógica de colocação e deslocamento**).

- ♦ **Menu Gráficos:** permite consultar e modificar todos os atributos do mundo gráfico. Todas as opções conduzem a caixas de diálogo.

- **Mundo:** abre a caixa de diálogo, a partir da qual é possível mudar os limites do mundo gráfico, da área representada na janela e a cor do fundo da janela (**operação de escala/seleção**).

- **Tartarugas:** abre uma caixa de diálogo na qual se obtém informação sobre o estado de todas as tartarugas (**operação de correspondência entre a “imagem mental” do sujeito e os estados das tartarugas**).

- ♦ **Menu Janelas:** permite mover, fechar, modificar, imprimir e apagar a janela ativa, assim como carregar o arquivo dentro dela ou guardar o seu conteúdo.

- **Move:** desloca janela com os cursores. ESC anula o movimento solicitado (**operações infralógica de deslocamento**).

- **Modifica:** muda o tamanho da janela ativa (**operação lógica de proporcionalidade/escala**).

- **Maximiza:** aumenta a janela para o seu tamanho máximo (**operação lógica de proporcionalidade**).

- **Tamanho/Posição:** pode-se ver e modificar o tamanho e posição da janela ativa (**operação lógica de escala**).

- ♦ **Menu Localizar:** permite procurar e substituir palavras ou frases por outras (**operação lógica de seleção/operação de cooperação sujeito/ambiente**).

- ♦ **Menu Sistemas:** tudo o que se refere ao conteúdo do espaço de trabalho (procedimentos e variáveis) e à correção dos erros.

- **Árvore de Procedimentos:** abre uma janela de informação que mostra, em forma de esquema, a estrutura de procedimentos, a partir do procedimento indicado na janela (**operação de classificação do ambiente**).

- **Desativa Janela de erros:** as mensagens de erro não aparecem na janela de Textos (**operação lógica de inversão da ação Ativa Janela de erros**).

- ♦ **Menu Textos:** define todos os atributos da área de textos.

- ♦ **Menu Utilitários:** se refere às ações complementares de configuração, da tela e da impressora. Também é possível sair para o Sistema Operacional, sem abandonar o ambiente.

5.3.2.1.2 Operações básicas do processo de interação

Aqui são descritos alguns comandos pré-definidos da linguagem, ou seja, as ações controladas pelo sujeito usuário e/ou programador em relação ao ambiente Winlogo.

- ♦ **Consulta:**

- > o sistema devolve o valor absoluto de um número: *valor absoluto*;
- > o sistema devolve o número inteiro mais próximo do número indicado em num: *Arredonda num*;
- > devolve um carácter lido do teclado ou do arquivo aberto para a leitura: *LêCar*. A mesma coisa vale para *LêCars*; lê quantidade de caracteres indicada em num, *Lêpalavra* lê uma palavra e ainda, para *LêLista*, onde o sistema devolve a lista lida a partir do teclado, na janela de Textos.
- > faz a leitura a partir do teclado e devolve a posição, rumo e cor atual das tartarugas ativas: *LêListaGráficos e LêPalavraGráficos*.

- ♦ **Classificação:**

- > agrupa os procedimentos e as variáveis indicadas em [objetos] com o nome grupo: *Agrupa "grupo[objetos] e AgrupaTudo "grupo*;
- > apresenta na janela Textos, o objeto indicado em obj mostrando qual a sua categoria. Se obj for uma palavra ou número mostra-o sem nenhum tipo de identificador: *Apresenta obj*;
- > mostra o índice de todos os ficheiros e diretórios que se encontram na unidade e diretório especificados em caminho: *Diretório "caminho*.

- ♦ **Escala/Proporção:**

- > ativa o modo de ajustamento do cenário ao tamanho da janela: *AjustaCenário*;
- > ajusta a escala de representação dos eixos X e Y ao passar do modo escalado ao natural: *AjustaProporção*;
- > o sistema fixa a escala de ampliação ou redução dos gráficos: *FixaAmpliação num*.
- > define a nova área representada na janela segundo as coordenadas indicada em coord: *FixaÁrea [coord]*.

- ♦ **Colocações e deslocamentos:**

- > o sistema devolve o valor do eixo X em que se encontra a tartaruga. Se existirem várias tartarugas o sistema devolve uma lista com o valor de todas elas, por ordem numérica: *CoorX*. Da mesma forma ocorre com o comando *CoorY*.
- > desloca o conteúdo da janela de Textos o número de linhas indicado em num: *DeslocaTexto num*.
- > o sistema devolve o rumo que teria a tartaruga no caso de estar orientada para o ponto cujas coordenadas se indicam em coord: *Direcção [coord]*.
- > faz rodar as tartarugas ativas para a sua direita os graus indicados em ang: *Direita ang*.

♦ **Negação:**

> desativa a tecla ESC: *DesativaEsc*.

♦ **Adição e Subtração de elementos:**

> o sistema elimina a primeira ocorrência do objeto obj dentro da lista de objetos: *Elimina obj [objetos]*.

> o sistema insere o elemento obj na posição especificada na lista de objetos: *Insere posição obj [objectos]*;

♦ **Interrupção do procedimento:**

> permite que o Winlogo interrompa a execução de um procedimento quando se carrega a tecla ESC: *ESC*.

> pára a execução durante um tempo equivalente às centésimas de segundo indicadas em num: *Espera num*.

5.3.2.1.3 Operações de controle do processo de interação

Estas operações referem-se à coordenação e organização dos comandos básicos descritos no item anterior. Ou seja, são as todas as ações que o sujeito programador decide em relação ao que o sujeito usuário deve realizar, a fim de controlar o processo de interação entre este e o programa.

- | | |
|------------------------|---------------------|
| ♦ >Para SEMPRE :VEZES | >Escreve :VEZES + 1 |
| >DesativaEsc | >Fim |
| >Se :VEZES = 50 [stop] | ?SEMPRE 1 |

Primeiro, desativa a tecla ESC. Não será possível interromper a procedimento SEMPRE carregando na tecla ESC. A operação de interrupção no meio da execução de um procedimento é feita através da tecla ESC. Como ela foi desativada neste procedimento, este só parará ao cumprir-se a condição de parada (ser = 50). Neste programa pode-se observar a **operação lógica de seriação**, porque os comandos devem obedecer uma certa ordem e ainda, a operação de controle alternativo da interação que se refere à **multiplicação de duas relações**, à condição de parada e a de soma de elementos.

- | | |
|-------------------------------|-------------------------------------|
| ♦ >Para MULT | > Executa [(Escreve :N1 \$* :N2 “ = |
| > Faz “N1 Aleatório 20 | : RESULTADO)] |
| > Faz “N2 Aleatório 20 | > Espera 65 |
| > Faz “RESULTADO Produto : N1 | > Fim |
| : N2 | |

Este procedimento efetua multiplicações aleatórias. Neste caso, o programador tem que pensar na interação que é feita com o usuário, programando o comando

- ◆ >Para ENTRE? :A :B :C
 - >Devolve E Menor? :B :A Menor? :B :C
 - >Fim

Este procedimento devolve VERDADEIRO se o segundo valor for um número menor que os outros dois. Ou seja, é um tipo de representação da interação em que o sistema **opera à nível de seriação** de um elemento em relação aos outros, segundo as condições pré-definidas (Menor?).

- ◆ >Para DIA :DIA
 - >Faz "SEMANA [DOMINGO SEGUNDA TERÇA QUARTA QUINTA SEXTA SÁBADO]
 - >Devolve PosLista :DIA :SEMANA
 - >Fim

? E DIA "TERÇA?

3

Este procedimento devolve o dia da semana em número, ou seja, refere-se a comandos de controle de interação que operam a nível de **correspondência**.

- >Para RODAR

<ul style="list-style-type: none"> >Avança 2 Espera 3 >Se Tecla? [RODA LêCar] >RODAR >Fim 	<ul style="list-style-type: none"> >Para RODA :DIR >Se :DIR = "D [Direita 90] >Se :DIR = "E [Esquerda 90] >Fim
---	--

A tartaruga move-se continuamente na mesma direção até que sejam pressionadas as teclas D ou E. Esta é a representação de um processo de interação, que usa a recursividade, ou seja, uma **operação de ordem cíclica**, com uso de **interrupções programadas** pelo sujeito.

O objetivo desta análise é enfatizar a importância da leitura das operações lógicas e infralógicas de controle do ambiente, dos comandos básicos e de controle da interação.

6 Análise operatória de ferramentas computacionais de uso coletivo

6.1 Microsoft Netmeeting 2.0 Beta 4

6.1.1 Descrição do ambiente

A ferramenta Microsoft Netmeeting para Windows 95, versão 2.0 Beta 4, maio 1997 [MIC 97b], é um ambiente que integra vários aplicativos de teleconferência. Trata-se de um groupware que contém um conjunto de aplicativos e componentes de rede que permite comunicações de áudio, vídeo e dados com interações síncronas (tempo real) e distribuídas (diferentes lugares) pela Internet. Como pode ser observado, este ambiente proporciona à pessoas uma nova forma de falar, trabalhar e compartilhar em torno da Internet. A sua interface é mostrada na figura 30.

Algumas características do Microsoft NetMeeting:

1) *Comunicação*: pode ser feita através da fala (uso de áudio), da escrita através da troca de mensagens (bate-papo, chat ou "talk"), vídeo, quadro de comunicações (desenho e/ou escrita) ou, ainda, através da linguagem que é permitida pelo aplicativo que está sendo compartilhado na conferência.

2) *Conflitos*: não existe um módulo que soluciona conflitos entre os membros da equipe de trabalho. Tanto no bate-papo, quadro de comunicações e compartilhando qualquer aplicativo, o primeiro que clicar no mouse vai poder escrever, falar e desenhar.

3) *Coordenação das ações*: regras que têm que ser seguidas pelo grupo terão que ser estabelecidas pelo próprio grupo que está em conferência, com o objetivo de trabalhar de forma harmoniosa, definindo assim os objetivos do trabalho, as atividades de cada um ou do grupo como um todo, divisão das tarefas (se necessário), prazos, etc.

4) *Recursos de sistema de autoria cooperativa*: sujeitos da conferência podem compor um documento ou um desenho em conjunto utilizando, para isso, várias páginas (editor cooperativo).

5) *Janela de trabalho privativa*: existe a opção de, mesmo trabalhando em um sistema de teleconferência, configurar o mesmo para que o usuário trabalhe de forma individual, sem nenhum tipo de compartilhamento dos seus dados.

6) *Janela de trabalho compartilhado*: o sujeito compartilha aplicativos e todas as pessoas que fazem parte da conferência podem ver e usar o aplicativo.

7) *Controlar/Colaborar*: dois tipos de compartilhamento entre usuários. O controlador é o que está controlando naquele momento o sistema, ou seja, é para ele que é feito o pedido de recebimento de chamada e, é este que aceita ou não. O colaborador é o outro membro que faz parte da conferência mas este somente tem o direito de escrever/desenhar/falar, trocar mensagens através dos aplicativos disponíveis. Quem abre uma conferência é o coordenador ou controlador da mesma e tem direito de fechá-la

quando quiser. Se este sair da conferência todos os colaboradores que estão na mesma serão automaticamente desligados.

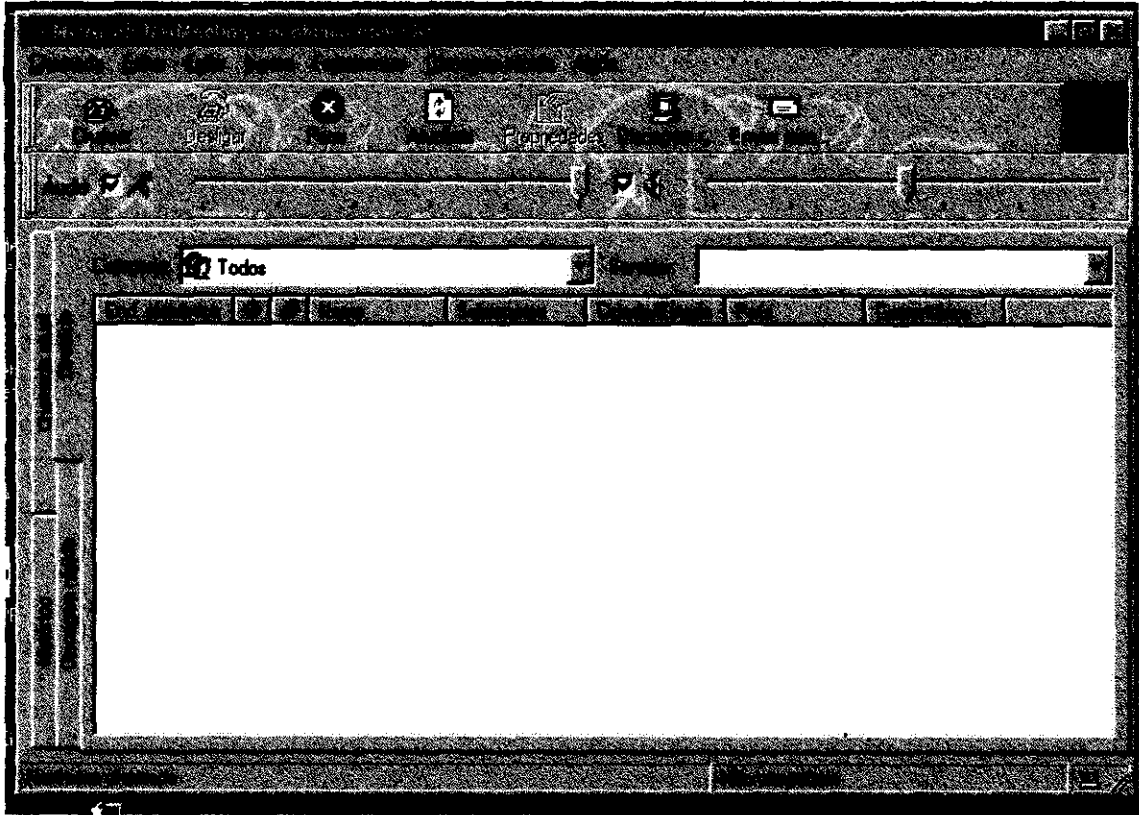


FIGURA 30 - Interface da ferramenta Microsoft Netmeeting 2.0 Beta 4

O modelo geral de interação do sujeito individual/coletivo com a ferramenta Microsoft Netmeeting 2.0 Beta 4, pode ser visto no figura 31. Cabe enfatizar que como se trata de um sistema de teleconferência, na opção de compartilhamento pode ser utilizado qualquer software que rode em Windows 95. Portanto, o modelo específico de cada aplicativo que será compartilhado dependerá do mesmo. Logo, somente será detalhado mais adiante o quadro de comunicações e o bate-papo.

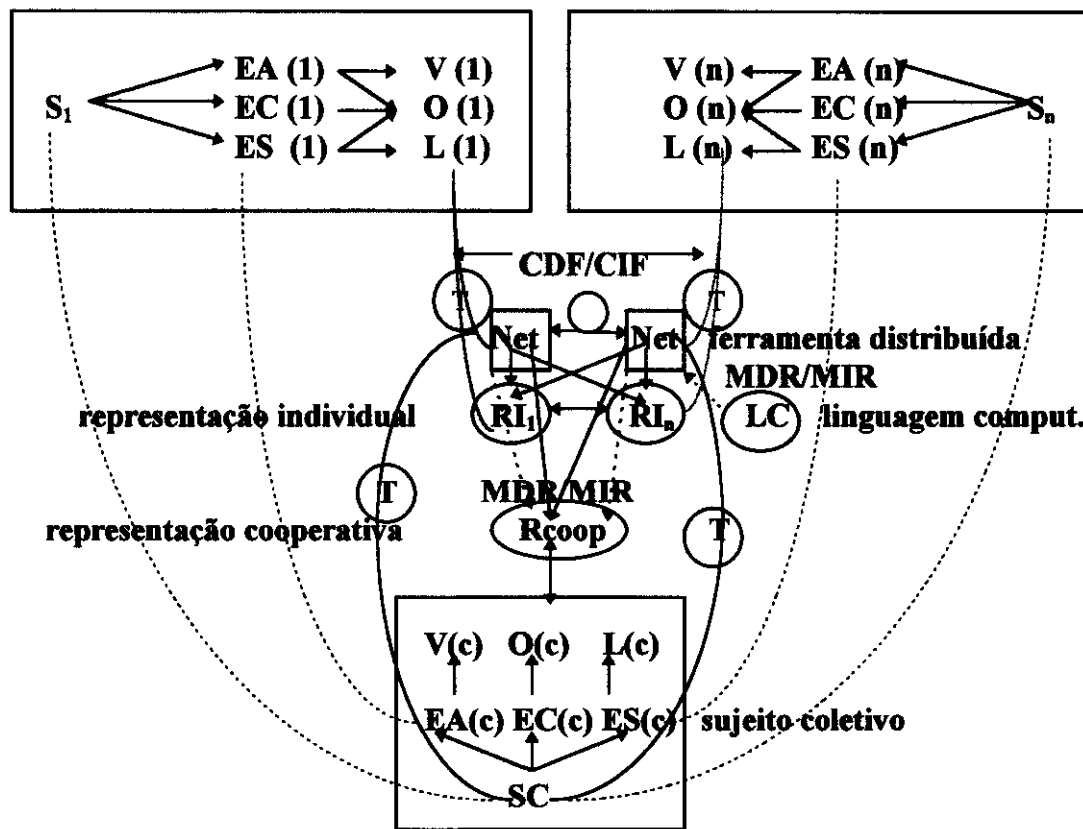


FIGURA 31 - Modelo geral de interação do sujeito coletivo com o Microsoft Netmeeting 2.0 Beta 4

onde: S_1 : sujeito₁; S_n : sujeito_n; Net: ferramenta computacional distribuída Netmeeting; T: tela, teclado, etc...(canal de comunicação sujeito-ferramenta); RI_1 : representação individual do sujeito₁; RI_n : representação individual do sujeito_n; RCoop: representação cooperativa; LC: linguagem computacional; CDF/CDI: controle direto/indireto da ferramenta e MDR/MIR: manipulação direta/indireta da representação individual e/ou coletiva (arquivos).

6.1.2 Operações lógicas/infralógicas do Microsoft Netmeeting 2.0 Beta 4

Nesta ferramenta as operações coletivas são identificadas no momento em que são ativadas a opção de compartilhamento do aplicativo, uso do quadro de comunicações ou do bate-papo.

Em qualquer uma destas opções, os sujeitos trabalham de forma coletiva seja, por exemplo, criando um slide ou um texto de forma cooperativa (no caso de usar o PowerPoint ou o Word do Windows95), seja realizando um desenho em conjunto com ou sem texto (quadro de comunicações) ou trocando mensagens sincronicamente.

A coordenação das ações se dá, geralmente, pela seqüência em que o usuário clica no mouse, isto é, o primeiro que clicar, tem direito a usar a ferramenta, se um outro usuário clicar logo depois, este segundo é quem estará utilizando o sistema e, assim por diante. Ou seja, no que se refere à coordenação das ações, esta ferramenta não possui muitos recursos para solução de conflitos entre usuários. O valor poder se dá pela rapidez de quem clicar o mouse primeiro.

1) Operação lógica de negação:

1.1) Chamada; Não incomodar: não receber nenhuma chamada. A janela de trabalho é exclusiva do sujeito. Até não desativar esta opção o sujeito não poderá receber chamadas. Esta operação é do tipo **controle indireto sobre a ferramenta**, porque é uma operação que não manipula representações mas diz respeito à forma como vai ser controlado o ambiente de trabalho. Neste caso, o usuário ativa a opção que não quer ser incomodado, isto é, quer trabalhar o tempo que for necessário na sua janela privativa. Quem executa o comando é a ferramenta, por essa razão trata-se de uma operação do tipo indireta (V,O).

1.2) Ferramentas; Transferência de arquivo; Cancelar recepção: mesmo que o outro sujeito envie o arquivo, o usuário nega esta operação, através da opção de cancelamento (V,O).

1.3) Chamada; Desligar: o usuário desliga uma chamada.

2) Operação lógica de interrupção:

2.1) Chamada; Interromper execução da chamada: interrompe a discagem feita. Quem ativa esta operação é o usuário, portanto, pode-se dizer que se trata de uma operação de **controle direto sobre a ferramenta**

3) Operação lógica inversa:

3.1) Ferramentas; Transferência de arquivo; Cancelar envio: desfazer a ação anterior de enviar arquivo.

4) Operação lógica de correspondência:

4.1) No menu Editor, pode-se recortar, copiar ou colar o texto. Portanto, são operações de **manipulação indireta da representação** que, neste caso, é a janela de texto. Ex: o sujeito marca uma área determinada do texto, ativa o comando **RECORTAR** e **COLAR**, do menu **EDITOR** para copiá-la em outra parte do texto. Quem faz esta ação é a ferramenta e não o usuário e, trata-se de uma correspondência entre as partes do texto para que o mesmo seja coerente (operação do tipo (O)).

5) Operações coletivas (relação controlador/colaborador):

5.1) Chamada; Receber conferência: para receber uma conferência do NetMeeting. Quando o usuário receber uma conferência, não precisa chamar ninguém. As outras pessoas podem ingressar e sair, como fariam em uma sala de bate-papo de um serviço qualquer. O usuário pode configurar a conferência de modo que os chamadores sejam aceitos automaticamente ou de modo que possa filtrar os chamadores. A conferência permanecerá ativa até que o usuário se desligue. Como pode-se observar, trata-se de uma ação do tipo individual que reflete no coletivo porque o usuário estará ativando uma opção em que todo o tempo que ele está utilizando a ferramenta poderão entrar e sair os participantes da conferência. Portanto, é um canal aberto para que se iniciem operações compartilhadas do sujeito coletivo.

5.2) Ferramentas; Compartilhar Aplicativo: se o usuário compartilhar, por exemplo, uma janela do Windows Explorer, como Meu computador ou uma pasta do seu computador, estará compartilhando todas as janelas do Windows Explorer que estiverem abertas. Além disso, uma vez que o usuário tenha compartilhado uma janela do Windows

Explorer, todo o aplicativo que ele iniciar enquanto estiver na conferência, será também automaticamente compartilhado com os participantes da conferência. Se este não quiser que outras pessoas da conferência assumam o controle do aplicativo que ele compartilhou, deve ativar no menu **Ferramentas**, a opção **Trabalhar sozinho** (janela individual de trabalho).

5.3) Ferramentas; Iniciar colaboração: para trabalhar com um aplicativo compartilhado por outra pessoa. O usuário pode assumir o controle do aplicativo clicando duas vezes na janela do aplicativo (*valor poder*). Este sujeito será definido como **colaborador** e, quem abriu o aplicativo será o **controlador**.

5.4) Ferramentas; Transferência de arquivo; Enviar arquivo: um sujeito pode enviar um arquivo para os participantes que desejar da sua conferência. Para enviar um arquivo para uma única pessoa, o usuário deverá clicar no ícone para aquela pessoa com o botão direito do mouse, e em seguida clicar em Enviar arquivo. Este também pode arrastar o arquivo pela lista de pessoas da conferência e pode especificar a pasta em que os arquivos estão armazenados quando as pessoas os enviarem para ele. Esta operação é do tipo **manipulação indireta da representação** (arquivo) porque quem ativa a opção de envio é o usuário, mas a ação é do sistema, isto é, é este quem envia o arquivo aos outros participantes. O modelo de interação neste caso é mostrado, simplificado, na figura 32.

Pode-se observar que um sujeito, no momento em que envia um arquivo para outro sujeito, ele estará alterando a representação dos outros participantes.

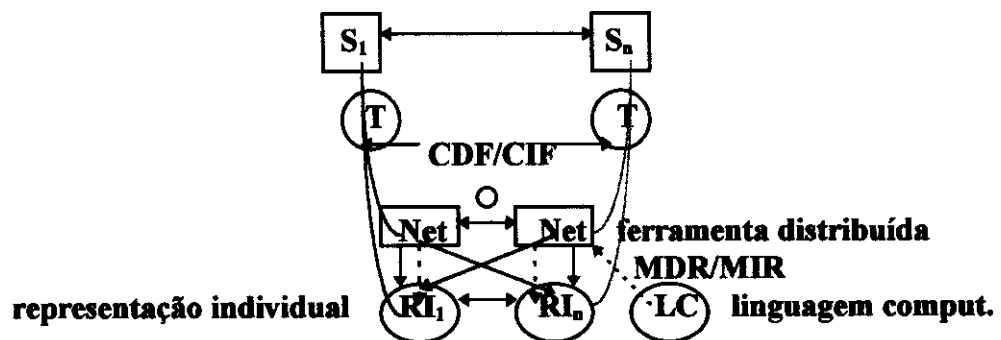


FIGURA 32 - Modelo de interação dos sujeitos com a ferramenta quando ocorrem alterações nas representações dos mesmos

6.1.2.1 Operações lógicas/infralógicas do quadro de comunicações

É um tipo de *whiteboard* com recursos para desenho e escrita cooperativa, isto é, é uma ferramenta computacional cooperativa tanto de desenhos quanto de textos, ou ainda, das duas coisas misturadas. Possui vários recursos para serem utilizados de forma cooperativa.

1) Operações coletivas:

1.1) Arquivo; Novo: quem abre um arquivo novo é o chamado **controlador**. Isto é, o usuário que abre o arquivo tem **poder** (valor) sobre o mesmo, operação do tipo (V,O). Apesar de compartilhá-lo com todos os participantes da conferência que, neste caso, serão usuários **colaboradores**, no momento em que o controlador decide se desligar da

mesma, todos são automaticamente desconectados. É uma operação de **manipulação indireta sobre a representação** (arquivo).

1.2) Qualquer participante pode abrir um trabalho previamente elaborado e guardado na memória do sistema. No momento de abrir o arquivo, o participante terá que operar em cima de uma operação do tipo (L,O), porque o objeto (arquivo), já possui uma linguagem (nome) relacionado a ele. É uma operação de **manipulação indireta sobre a representação**.

1.3) A ação de salvar um trabalho que já tem um nome relacionado a ele (L,O), pode ser feita por qualquer participante da conferência (MIR).

1.4) No momento em que um participante tem que salvar um trabalho realizado pelo grupo, ele precisa definir o nome do arquivo, tipo e diretório, portanto estará operando de forma (L,O) e **manipulando indiretamente a representação** (arquivo).

1.5) A operação de **Sair** (sai do quadro de comunicações), é uma ação individual e se reflete sobre o coletivo, já que se o participante que estiver saindo é o controlador, automaticamente todos os outros usuários serão desconectados automaticamente. Se for um colaborador, somente este será desligado da conferência.

1.6) Operação infralógica coletiva de partição e adição primitiva:

1.6.1) A opção do menu **Editar; Excluir** é uma **operação infralógica coletiva de partição primitiva** porque se trata de uma ação realizada na conferência por qualquer um dos participantes e somente funciona quando é selecionada uma parte ou toda a representação e este deseja excluí-la. Portanto, esta operação é realizada em cima da **representação coletiva** e irá alterá-la, excluindo parte da representação. É por essa razão que do total coletivo (toda a representação) irá se excluir parte dela. É uma operação de **manipulação coletiva indireta sobre a representação**.

1.6.2) No mesmo menu, a opção de **Recortar**, na qual se faz um recorte de uma área previamente selecionada, também se refere a uma **operação coletiva de partição primitiva**, já que será tirada uma parte de toda a representação (operação de **manipulação indireta sobre a representação**).

1.6.3) Já a opção **Colar**, cuja ação é colar na representação corrente uma área selecionada e previamente recortada, diz respeito a adição de uma parte no todo, portanto, também se trata de uma **operação de manipulação indireta da representação**.

1.6.4) As operações **Excluir página** (deleta uma página do trabalho), **Inserir página antes** (insere uma nova página antes da corrente), **Inserir página depois** (insere uma nova página depois da corrente), tratam de uma parte da representação que tem que ser adicionada ou particionada. Portanto, está se falando de **operações infralógicas de manipulação da representação coletiva**.

1.6.5) No menu **Ferramentas**, a opção **Área selecionada**, o usuário seleciona uma janela para colar no quadro de comunicações. Pode-se observar que esta operação é do

tipo **manipulação direta sobre a representação**, já que o participante estará trabalhando com a barra de ferramentas.

1.7) Operação lógica coletiva inversa:

1.7.1) Editar; Restaurar: função inversa (undo) do excluir (**operação de manipulação indireta sobre a representação**).

1.8) Operação lógica coletiva de proporção:

1.8.1) No menu **Exibir**, a opção **Zoom**, é uma operação de zoom com dimensões definidas pelo sistema, que é ativado por um participante mas que reflete no janela de todos os outros que fazem parte da conferência. Trata-se, portanto, de uma **operação de manipulação indireta sobre a representação**.

As seguintes operações são de **manipulação direta sobre a representação**, já que existe uma barra de ferramentas onde o usuário pode trabalhar de forma “direta” com a representação coletiva.

1.9) No menu **Ferramentas**, as opções **Seletor**: seleciona área de trabalho; **Caneta**: ativa pincel para fazer desenhos; **Realçador**: faz o realce; **Texto**: ativa a escrita, entre outras opções.

1.10) Operação coletiva de coordenação das ações:

1.10.1) No menu **Ferramentas**, a opção **Indicador remoto**, ativa e desativa uma mão. Esta é uma maneira de indicar o que deseja que o outro usuário veja ou, pode ser uma forma de aviso ao outro participante.

1.10.2) A opção **Bloquear conteúdo**, é uma forma de controlar a ferramenta sem intervenção de outro participante. Quem ativou o bloqueio deve desativá-lo para deixar o outro sujeito desenhar ou escrever, ou seja, trata-se de uma operação de coordenação das ações. O outro usuário é informado do bloqueio porque aparecerá na sua tela o cursor em forma de cadeado mostrando que ele não pode trabalhar até este não voltar a sua forma normal. Como neste sistema quem clica no mouse tem direito a usar a ferramenta, esta foi uma forma que se encontrou de um sujeito poder representar o que ele quiser na tela sem que o outro participante interfira, até ele liberar a sua representação para o resto dos usuários.

2) Operação lógica individual de classificação:

2.1) No menu **Editar**, a opção **Classificar páginas** funciona somente na janela privativa, mostrando todas as páginas desenvolvidas no quadro de comunicações. Esta operação não reflete sobre o coletivo, trata-se de uma operação ligada somente ao sujeito individual.

6.1.2.2 Operações lógicas/infralógicas do bate-papo

O aplicativo **Bate-papo** é um canal de conversação de dois ou mais participantes. Para enviar uma mensagem na janela Bate-papo, os usuários deverão digitar a mensagem que desejam enviar e, em seguida, pressionar ENTER.

O modelo de interação dos sujeitos com o Bate-papo pode ser visto, simplificada, como na figura 33.

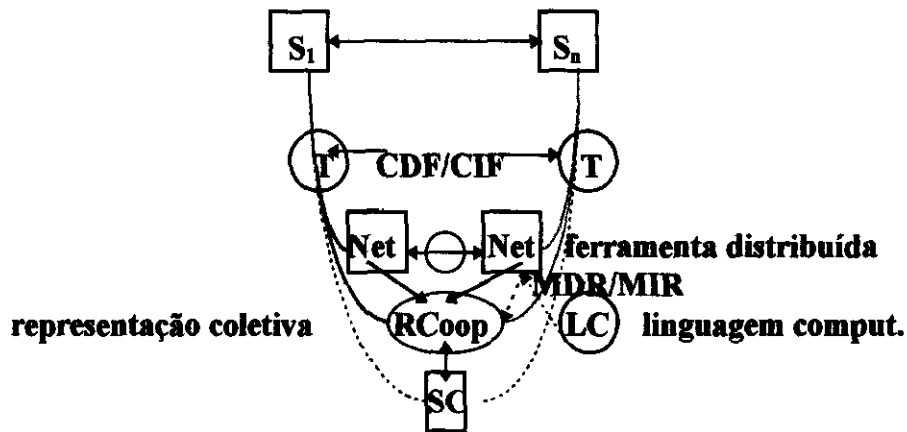


FIGURA 33 - Modelo de interação dos sujeitos com o Bate-papo

1) Operações coletivas:

1.1) No menu **Arquivo**; a opção **Abrir**, abre um arquivo com mensagens previamente armazenadas. Essa ação refletirá no coletivo, já que se um usuário abrir um determinado arquivo, todos os que estão fazendo parte da conferência, poderão ver esta ação na sua estação de trabalho. É uma operação coletiva de **manipulação indireta da representação** (arquivo que contém as mensagens previamente armazenadas).

1.2) A opção **Salvar como**, guarda mensagens definindo o nome do arquivo, tipo, etc. Essa operação pode ser efetuada por qualquer conferencista que deseja guardar em um diretório determinado o arquivo. Para isso este precisará relacionar um nome à representação (L,O).

1.3) **Sair do bate-papo**, significa desconectar-se da conversa. Isso irá refletir no coletivo, avisando os outros participantes que o usuário em questão está saindo da conferência.

1.4) Operação infralógica coletiva de adição e partição primitiva:

1.4.1) Tanto a opção **Editar**, para editar tem que marcar a mensagem desejada; **Recortar**, deletar a mensagem marcada; **Copiar**, copiar a mensagem marcada ou **Colar**, colar a mensagem marcada para onde está o cursor, estão trabalhando com partes da representação que são deletadas ou adicionadas. Portanto, estas operações refletem todas na representação coletiva (todo), modificando-a. São operações de manipulação indireta sobre a representação.

6.2 Microsoft Chat 2.0 Beta 1

6.2.1 Descrição do ambiente

O Microsoft Chat [MIC 97a] é um programa de bate-papo na Internet que dá a opção de conversar dentro de uma tira de quadrinhos. Como acontece com um programa de bate-papo padrão, o usuário pode entrar em uma sala de bate-papo em um servidor de Internet e manter conversas em tempo real com outras pessoas. Mas, com o Microsoft Chat, este também pode escolher um personagem de história em quadrinhos para representá-lo, e a sua conversa aparece como balões de fala dentro dos quadrinhos de

uma tira. Como um personagem de história em quadrinhos, o usuário pode expressar uma ampla gama de emoções, enviar "pensamentos", sussurrar para um destinatário selecionado e salvar a sessão de bate-papo para visualização off-line.

A janela do Microsoft Chat é composta das seguintes partes:

- um painel de visualização onde a conversa é exibida
- um painel de lista de participantes que lista todas as pessoas em uma sala de bate-papo
- um painel de composição (uma caixa de texto no final da janela) onde o usuário digita as suas mensagens
- botões que o usuário usa para enviar as suas mensagens

Se o usuário optar por conversar dentro de uma tira de quadrinhos, poderá ver também:

- um painel "estrelando" que exibe o título do bate-papo e lista os participantes
- um painel de autovisualização que mostra como é o seu personagem
- uma roda de emoções que permite que o usuário controle todas as emoções do seu personagem

A interface do sistema pode ser visto na figura 34.

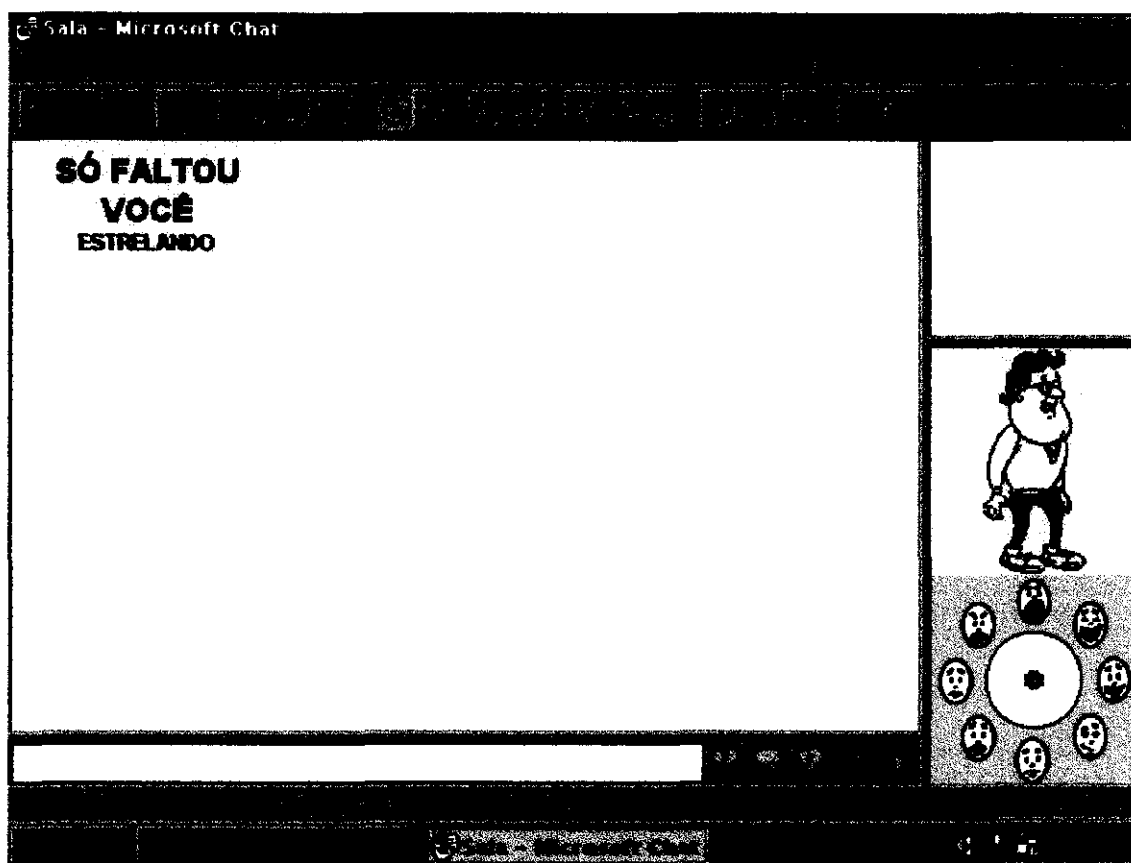


FIGURA 34 - Interface da ferramenta Microsoft Chat 2.0 Beta 1

representação do sujeito,; RCoop: representação cooperativa; LC: linguagem computacional; CDF/CIF: controle direto/indireto sobre a ferramenta; MDR/MIR: manipulação direta/indireta sobre a representação coletiva (conversa).

6.2.2 Operações coletivas lógicas/infralógicas da ferramenta Microsoft Chat 2.0 Beta 1

Nesta ferramenta todas as operações realizadas por qualquer usuário que participa da conversa são refletidas no coletivo porque toda ação é representada no tela, seja esta em forma de texto ou em forma de tira de quadrinhos, através da fantasia dos personagens.

Basicamente o que interessa aqui é analisar as operações sobre as emoções, sentimentos, isto é, sobre os valores trocados entre personagens (participantes do bate-papo). Mas, assim como nas outras ferramentas cooperativas analisadas, também existem operações lógicas e infralógicas sobre a representação, que é o produto de uma conversa entre usuários.

A análise desta ferramenta foi baseada nas seções 4.3.1.1.1 e 4.3.1.1.2, sobre a operatoriedade nas trocas sociais, mais especificamente, tratando das operações lógicas e/ou infralógicas de valores.

É sobre este contexto que introduzimos as operações da ferramenta Microsoft Chat 2.0 Beta 1 que permite, entre outras coisas, a troca social, troca de idéias (proposições) onde são inseridas emoções, e que a ação de um sujeito leva à reação de um outro. O que analisaremos é se a ferramenta possui operações que permitem essa troca social e como é realizada.

Neste ambiente a operação coletiva nada mais é do que a justaposição das operações individuais. Com isso, quer se dizer que o produto final contém ações que provocam reações de uns para outros, através das mensagens mandadas e que fazem com que o produto final não seja a soma de operações. Logo, nas operações individuais podem ser identificadas as reflexões dos outros, através das respostas dadas, ou da mudança de opinião em relação a algo, ou os conflitos gerados e a solução dos mesmos através da troca social que, neste sistema se dá através do diálogo, como forma de comunicação.

A seguir, serão analisadas algumas operações lógica/infralógicas individuais e/ou coletivas da ferramenta em relação à representação.

1) Operação infralógica individual de adição e partição primitiva: estas operações somente funcionam em modo texto

1.1) No menu **Editar**, as opções de **Recortar**, **Copiar** e **Colar** são operações onde, o sujeito que as está ativando, estará trabalhando com partes (partes da conversa), excluindo ou adicionando-as de um todo (conversa total). Trata-se, portanto, de operações de **manipulação indireta sobre a representação** (texto que representa a conversa: objeto). Todas as ações são executadas em cima da representação individual, portanto, sobre as operações do sujeito individual.

2) Operação lógica individual de correspondência:

2.1) No menu **Arquivo**; a opção **Nova Conexão**: conecta com os endereços favoritos, ou nome do servidor. Também vai para sala de bate-papo ou mostra todas as salas de bate-papo disponíveis. A caixa **Servidor**, contém a sala de bate-papo em que o usuário deseja entrar. Na caixa **Ir para sala de bate-papo**, contém o nome da sala de bate-papo

(L,O) em que o usuário deseja entrar. Esta operação é de **controle indireto da ferramenta**, porque diz respeito a uma conexão do mesmo (não uma representação) e quem faz a conexão, através da ação do sujeito é a própria ferramenta.

3) Operação lógica inversa:

3.1) No menu **Editar**, a opção **Desfazer**, desfaz a ação anterior, *undo* sobre a representação coletiva. É uma operação de **manipulação indireta sobre a representação**

3.2) No menu **Sala**, a opção **Desconectar**: desconectar-se da sala de bate-papo que está participando, é a inversa da conexão. Trata-se de uma operação de **controle indireto sobre a ferramenta**.

4) Operações individuais:

4.1) No menu **Exibir**, na opção **Tira de quadrinhos**: o modo de conversação é definida em tira de quadrinhos, e o usuário deverá se fantasiar com algum personagem. Na opção **Texto simples**, o bate-papo somente aparece em modo simples de texto. Se o usuário optar um ou outro modo de conversa (quadrinhos ou texto), será alterada a forma de representação individual do mesmo e não a representação de todos os participantes.

5) Operações sobre representação de valores:

5.1) Na caixa de texto, o usuário digita o que deseja dizer (linguagem) e a forma como deseja que esta mensagem seja representada (representação da linguagem):

- *dizer*: aparece dentro de um balão de fala;
- *pensar*: aparece dentro de um balão do pensamento;
- *sussurrar*: ação que reflete na representação dos personagens que o usuário selecionou na lista de participantes;
- *ação*: aparece dentro de uma caixa no alto do painel mais recente da tira de quadrinhos.
- *tocar som*: toca um som

Pode ser visto que, de acordo com a forma como é enviada a mensagem, o usuário estará expressando uma emoção através de uma idéia (V,L) e, dependendo desta, este estará esperando uma resposta dos outros participantes, isto é, uma reação.

Nesta opção pode-se falar somente com o usuário que se deseja, ou seja, não é necessário mandar a mensagem para todos os participantes. Basta que o usuário selecione alguém na lista de participantes antes de clicar no botão dizer ou de pressionar enter, e a ferramenta Chat faz parecer que o usuário está falando diretamente com essa pessoa. Todas as operações são de **manipulação “direta” sobre a representação (idéia)**, porque o usuário estará trabalhando com a barra de ferramentas como se fosse uma ação direta deste para o outro participante.

5.2) No menu **Exibir**, na opção **Macros**: o usuário digita uma mensagem que deseja que a macro envie. Também pode-se usar macros para enviar a mesma mensagem repetidas vezes sem ter que digitá-la a cada vez, por exemplo: *De onde você é?* Esta operação é de **manipulação indireta sobre a representação (mensagem a ser enviada)**, já que quem manda é a ferramenta, o usuário somente ativa a opção de macros.

5.3) Nas Opções, através da Automação cria-se saudações e macros a serem executadas de forma automática. Para enviar saudações automáticas a novos participantes da sala, na área Saudação automática, o usuário deve selecionar como deseja enviar a saudação. Para enviar a saudação apenas à pessoa que está entrando na sala, deve clicar em Sussurrado. Para enviar a saudação a todas as pessoas na sala sempre que alguém novo entrar, deve clicar em Dito. Na caixa Saudação automática, o usuário tem que digitar a mensagem que deseja enviar. Este pode usar o texto especial %name! e %room para personalizar a sua saudação. O Microsoft Chat, sempre que encontra %name!, o substitui pelo codinome da pessoa e, sempre que encontra %room, o substitui pelo nome da sala. Portanto, da mesma forma que a operação anterior, trata-se de **manipulação indireta sobre a representação** e, além disso, é uma operação de **classificação**.

5.4) No mesmo menu, em Opções, as Informações pessoais mostram os dados pessoais do usuário. Na caixa Codinome, o usuário deve digitar um nome pelo qual gostaria de ser conhecido. Na caixa Breve autodescrição, o usuário tem que digitar alguma coisa que gostaria que as pessoas soubessem a seu respeito. Essas informações aparecem quando outras pessoas solicitam o seu perfil. Para que outras pessoas possam enviar mensagens de correio eletrônico ou visitar a home page da World Wide Web do usuário. Portanto, trata-se de uma **operação de controle indireto sobre a ferramenta**, porque o usuário estará se apresentando aos outros participantes através dos dados armazenados na ferramenta. Ou seja, esses dados estarão implícitos no seu sistema.

5.5) Ainda nas Opções, o Modo de quadrinhos, onde se estabelece as fontes, padrões e o número de painéis que o Microsoft usa horizontalmente para exibir a tira de quadrinhos (de 1 a 4), também faz parte das operações de controle indireto da ferramenta.

5.6) Na Configuração, onde se faz a configuração da janela de trabalho, do tipo, avisar que tem que salvar antes de sair, permitir sussurros, tocar sons, receber convites de bate-papo, etc., o usuário estará realizando operações de controle indireto sobre a ferramenta.

5.7) Operação de partição e adição primitiva:

5.7.1) No Personagem, o usuário deve selecionar um dos doze personagens pré-definidos para que este se fantasie no modo de quadrinhos. O personagem aparecerá na área de Visualização. Para ver o personagem com diferentes emoções, tem que arrastar o ponto preto no centro da roda de emoções para diferentes posições. Para alterar a emoção de um personagem tem que se arrastar o ponto preto no centro da roda de emoções pela roda. Quanto mais para perto da borda da roda o usuário move o ponto, mais forte é a emoção. Este pode escolher oito emoções diferentes para o seu personagem: grito, raiva, felicidade, riso, tédio, susto, tristeza, timidez. O usuário também pode "congelar" a emoção do seu personagem clicando o botão direito do mouse no seu personagem no painel de autovisualização e depois em Congelar. A expressão não se altera até que o usuário clique novamente em Congelar. Neste caso, a operação é de manipulação indireta sobre a representação (objeto: personagem + idéia + emoção + gesto). Assim pode-se ver que aparece a operação infralógica de partição e adição primitiva que, nada mais é do que a adição e subtração de valores mais o gesto, ou seja, operações do tipo (V,O) de manipulação indireta sobre a representação.

Essa operação pode ser vista na seção 4.3.1.1.2, sobre a operatoriedade dos valores coletivos [PIA 73]:

1. Adição de valores: $\uparrow a + \uparrow a' = \uparrow b$ (ou $a + a' = b$); $b + b' = c$, etc.
2. Subtração de valores: $b - a' = a$; $c - b' = b$, etc.

Essa operação é identificada quando o usuário vai incrementando o seu objeto, com mais emoções, mais idéias e assim, por diante. Ou seja, ele estará construindo o seu todo (personagem completo: idéias + emoções + gestos) através de suas partes. Assim, toda ação de um participante para outro marca um crescimento de valores (satisfação), uma diminuição dos mesmos (prejuízo), ou uma diferença nula. Cada ação provocará uma ação de volta, isto é, uma reação do outro sujeito.

A existência de valores se traduz numa valorização recíproca das ações ou “serviços”, sejam estes positivos ou negativos. Então as emoções dos personagens da ferramenta, estarão limitados em realizar as seguintes ações através do *grito*, *raiva*, *felicidade*, *riso*, *tédio*, *susto*, *tristeza*, *timidez*.

Os gestos também fazem parte do personagem completo e, dependendo da ação do mesmo acarretará reações na troca social. Por exemplo: quando o usuário começa uma frase com a palavra:

Eu	Aponta para si mesmo
Você	Aponta para a outra pessoa
Olá ou oi	Acena
Tchau	Acena
Bem-vindo	Acena
Alô	Acena

Quando a frase do usuário contém as palavras:

Você é	Aponta para a outra pessoa
Você vai	Aponta para a outra pessoa
Você foi	Aponta para a outra pessoa
Você não é	Aponta para a outra pessoa
Não foi você	Aponta para a outra pessoa
Sou	Aponta para si mesmo
Eu vou	Aponta para si mesmo
Vou	Aponta para si mesmo
Estou	Aponta para si mesmo

Quando o usuário digita a palavra ou os símbolos abaixo, a personagem demonstra a seguinte emoção:

Tudo em maiúsculas	Grito
!!!	Grito
RDR (rolar de rir)	Riso
GAR (gargalhar)	Riso
:) ou :-)	Felicidade
:(ou :-(Tristeza
;-)	Timidez

5.7.2) Também um usuário pode enviar uma expressão de personagem de história em quadrinhos sem falar. Na roda de emoções, deverá arrastar o ponto preto para uma emoção, clicar o botão direito do mouse na emoção e depois em Enviar expressão. A

expressão aparece no painel mais recente da tira de quadrinhos. Neste caso, somente está enviando um gesto com uma emoção sem estes estarem necessariamente ligados à uma idéia.

5.7.3) No menu **Favoritos**, a opção **Adicionar aos favoritos**, é uma operação para adicionar uma sala de bate-papo à lista Favoritos, ou seja, a lista é o todo e será adicionado mais um elemento, representada por uma sala. Trata-se, portanto de uma **operação infralógica de adição primitiva e de manipulação indireta sobre a representação**.

5.7.4) Para retirar alguém de uma sala temporariamente, na lista de participantes, o usuário deve selecionar a pessoa que deseja retirar (remover temporariamente). No menu **Participante**, tem que clicar em **Coordenador** e depois em **Retirar**. Na caixa **Por que você está retirando** (codinome), o usuário deve digitar a razão pela qual está removendo a pessoa da sala de bate-papo. A razão que for digitada aparecerá para os outros participantes da sala. Retirar alguém não impede que essa pessoa entre novamente na sala de bate-papo. Também se pode desejar vetar alguém permanentemente. Apenas o coordenador de uma sala de bate-papo pode retirar alguém de uma sala. Para vetar alguém na sala de bate-papo permanentemente: na lista de participantes, o usuário deve selecionar a pessoa que deseja vetar. No menu **Participante**, tem que clicar em **Coordenador** e depois em **Vetar/Reintegrar**. Vetar alguém impede que a pessoa torne a entrar na sala até que o usuário a reintegre. Como este vai precisar fornecer essas informações se um dia quiser permitir que a pessoa volte à sala, pode ser conveniente anotar as informações exibidas na caixa **Pessoa a ser vetada ou reintegrada**. Apenas o coordenador de uma sala de bate-papo pode vetar alguém em uma sala (operação do tipo (V,O)). Estas operações são do tipo infralógica, porque está se trabalhando com partes (participantes) de um todo (sala de bate-papo com todos os integrantes) e é de **partição e adição primitiva** porque assim como podem ser retirados (temporariamente), os participantes também poderão vir a fazer parte da sala a qualquer momento. É uma entrada e saída contínua de elementos. Mas quando se *Veta* alguém está ocorrendo uma **operação de negação** deste participante à sala de bate-papo.

5.8) Operação de conexão (operação individual que reflete no coletivo):

5.8.1) Entrar na sala: entrar numa sala para conversar com outros usuários (faz parte do sujeito coletivo).

5.8.2) Conectar: conectar-se a uma determinada sala de bate-papo, definindo o servidor; mesma operação que *Nova Conexão*, do menu *Arquivo*. Trata de uma operação de controle indireto sobre a ferramenta, do tipo (L,O), onde L: é o nome do usuário que está se ligando a uma sala, que é o objeto (O).

5.9) Operação inversa (operação individual que reflete no coletivo):

5.9.1) Abandonar sala: abandonar a sala que estava conversando e fazendo parte do sujeito coletivo, ou seja, é a inversa da operação de entrar em uma sala (sujeito deixa de ser coletivo para se tornar individual).

5.9.2) No menu **Participante**, na opção **Ignorar**: ignora-se uma pessoa, assim o usuário não vê mais as mensagens daquele participante, mas os outros participantes da sala de bate-papo continuarão a vê-las.

5.10) Relação coordenador/colaboradores:

5.10.1) Criar sala: o usuário cria uma sala de bate-papo, dando um nome a ela e uma descrição do tópico do bate-papo da sua sala. O usuário se torna automaticamente o coordenador de qualquer sala que criar. Apenas o coordenador de uma sala pode alterar as suas propriedades. Trata-se de uma **operação de manipulação indireta sobre o objeto (forma de representação da sala)**, do tipo (V,O), onde V: coordenador e O: sala de bate-papo.

5.10.2) Quando um usuário é o coordenador da sua própria sala de bate-papo, ele pode: decidir o tópico da sala de bate-papo, controlar o acesso à sala de bate-papo com uma senha, limitar o número de participantes na sala de bate-papo, designar quem pode participar e quem pode ser apenas espectador, vetar participantes que atrapalhem ou ofendam a sala de bate-papo. Ou seja, está se trabalhando com o valor poder (V,O) em cima do objeto que é a sala de bate-papo, ou seja, uma **operação de controle indireto sobre a ferramenta**.

5.10.3) Para permitir que outras pessoas sejam coordenadores em sala de bate-papo de um usuário: na lista de participantes, o usuário deve selecionar a pessoa que deseja tornar um coordenador. No menu Participante, o este deve clicar em Coordenador. Apenas o coordenador de uma sala de bate-papo pode selecionar outras pessoas para que se tornem coordenadores. Operação do tipo (V,O).

5.11) Operação lógica de correspondência:

5.11.1) A opção **Lista de salas** serve para localizar uma sala de bate-papo específica. Na caixa Exibir o usuário deverá digitar palavras de pesquisa que correspondam ao nome da sala que o ele está procurando. Se este quiser procurar também nos tópicos da sala de bate-papo, deverá marcar a caixa de seleção e procurar também em tópicos de sala. Ou seja, trata de uma **operação lógica de correspondência** do nome da sala em relação a uma lista e o resultado da operação será quando o sistema encontra a sala de bate-papo desejada. É uma operação de **manipulação indireta sobre a representação**, do tipo (L,O), onde L: nome da sala que está se localizando e O: representação da sala de bate-papo.

5.11.2) No menu **Participantes**, na opção **Lista de usuários**, ocorre a mesma situação que na operação anterior. É para localizar outras pessoas, portanto, na área Pesquisar, deve ser definido o que o usuário deseja procurar. O usuário pode sussurrar ou enviar convites para as pessoas exibidas na caixa de diálogo Lista de usuários, mesmo se elas estiverem em uma sala de bate-papo diferente. Trata-se também de uma **operação lógica de correspondência**, de manipulação indireta sobre a representação do tipo (L,O) mas, neste caso, L: nome do usuário e O: Lista de usuários onde está sendo localizado o participante desejado.

5.11.3) No menu **Participante**, a opção **Netmeeting** é para fazer uma chamada do NetMeeting para um participante da sala de bate-papo. Tanto o usuário que fez a chamada quanto a pessoa para quem está fazendo a chamada devem ter o NetMeeting instalado, ou seja, deverá existir entre eles uma **relação de correspondência**.

5.11.4) Encontrando um usuário fora da sala de bate-papo: há várias maneiras do usuário poder se comunicar em particular com participantes de uma sala de bate-papo. Basta

selecionar uma das pessoas da lista de participantes e: (1) sussurrar na sala de bate-papo para que apenas a(s) pessoa(s) selecionada(s) possa(m) ver a sua mensagem particular, (2) fazer uma chamada do NetMeeting, se este e o outro participante tiverem o Microsoft NetMeeting instalado. Portanto deverá existir uma **correspondência** lógica entre os nomes que foram selecionadas e a lista de nomes das pessoas da sala. Os usuários podem compartilhar áudio, vídeo e programas fora do Microsoft Chat enquanto participam de uma sala de bate-papo. Também podem enviar uma mensagem de correio eletrônico, se o perfil de usuário do outro participante incluir um endereço de correio eletrônico.

5.12) Operação lógica de seriação:

5.12.1) No menu **Participante**, a opção **Convidar**, serve para convidar outras pessoas para uma sala de bate-papo. Para isso deve-se digitar os codinomes das pessoas que o usuário deseja convidar para a sala de bate-papo. Ou seja, é uma operação onde deve existir uma operação de seriação na colocação dos elementos separados por vírgulas (codinomes das pessoas). É uma operação de **manipulação indireta da representação**, do tipo (L,O).

5.13) Operação lógica de classificação:

5.13.1) No menu **Participantes**, as opções **Obter perfil** e **Obter identidade** para obter informações sobre outras pessoas é uma forma dos participantes estarem classificados segundo seu perfil e sua identidade. É uma operação do tipo (L,O), onde L: é a identificação do perfil ou da identidade e O: é a representação do usuário.

5.14) Operação de consulta (usuário/sistema):

5.14.1) No menu **Participante**, a opção **Tempo de resposta**: mostra o tempo de resposta do outro participante, ou seja, é uma operação de **controle indireto sobre o sistema**.

5.14.2) A opção **Hora local**, mostra ao usuário a hora local do bate-papo.

5.15) Relação co-unívoca (1 para muitos):

5.15.1) O usuário que participa do Chat pode enviar mensagens sussurradas usando a Caixa de sussurro. Isso permite que este controle várias conversas sussurradas ao mesmo tempo.

6.3 Ambiente de desenvolvimento cooperativo de programação ENVY/400

6.3.1 Descrição do ambiente

Este ambiente pode ser utilizado por equipes de desenvolvimento de programas compostas por um pequeno ou grande número de sujeitos, os quais são responsáveis pela realização de programas em conjunto [IBM 93].

O ambiente proporciona um extenso conjunto de ferramentas de programação em equipe, serviços de acesso à objetos AS/400 e, ainda, ferramentas de programação visual para aplicações de interfaces do tipo CUA 91.

O código das aplicações é escrito em Smalltalk, uma linguagem orientada a objetos que inclui uma coleção ampla de códigos fonte. Esta linguagem suporta classes definidas pelo próprio usuário que permitem a criação de objetos de trabalho e gerenciamento automático de memória.

A biblioteca compartilhada baseada em AS/400, que é um arquivo armazenado em folders compartilhados do tipo AS/400, mantém uma historia completa de todas as mudanças da aplicação que acontecem durante o seu ciclo de vida. As facilidades de exportação/importação seletivas suportam a transferência de versões de aplicações e componentes de aplicações entre organizações de desenvolvimento dispersados geograficamente.

O sistema ENVY/400 foi o primeiro ambiente de desenvolvimento orientado a objetos oferecido para programadores AS/400. Utilizando este ambiente, as equipes de desenvolvimento de programas podem criar cooperativamente aplicações combinando interfaces orientadas a objeto do tipo CUA 91 com objetos AS/400 - todos vindos de um ambiente de desenvolvimento individual.

ENVY/400 é um ambiente completo para prototipação, desenvolvimento, teste, manutenção e “empacotamento” de aplicações em Smalltalk. Este sistema aumenta a produtividade da equipe de programação através do suporte compartilhado para AS/400 baseado em um repositório de métodos fonte e compilados. Todas as características do produto ENVY estão disponíveis para a equipe, incluindo o controle de versões e gerenciamento de configuração.

O ambiente ENVY/400 suporta o desenvolvimento eficiente de interfaces, através do padrão de janelas CUA 91, utilizando ferramentas de programação visual. Isto é, usuários diferentes criam suas próprias interfaces utilizando o Gerador de Programas Visual (VPG).

As aplicações cooperativas cliente/servidor podem ser projetadas, implementadas e testadas usando ENVY/400. Os banco de dados AS/400, arquivos e outros serviços podem ser diretamente acessados da aplicação que está rodando na estação de trabalho (PWS).

O ambiente de desenvolvimento ENVY/400 gerencia as configurações das aplicações para aliviar o desenvolvimento multiplataforma e o empacotamento dos sistemas de produção final, isto é, o conjunto de aplicações.

Existem três tipos de configuração da aplicação:

1. uma configuração sem nenhum tipo de conexão com AS/400. Ou seja, trata-se do uso individual, onde a estação de trabalho realiza todas as funções, independentemente do hospedeiro (AS/400);
2. uma aplicação rodando em uma estação privada de trabalho e acessando os dados do hospedeiro através dos serviços disponíveis (compartilhados) e,
3. uma aplicação cooperativa completa com PWS e os componentes do hospedeiro, isto é, o AS/400 funciona como controlador.

Como pode ser observado, o ambiente pode ser utilizado de diversas formas, de acordo com a sua configuração.

Cada estação de trabalho tem a sua própria imagem em relação ao repositório, ou seja, à biblioteca compartilhada e, é a partir desta que são criados os componentes da aplicação.

A aplicação ENVY/400 provem uma interface consistente para disparar recursos distribuídos para o desenvolvimento de aplicações distribuídas através de estações de trabalho rodando em Windows ou OS/2 e AS/400 sobre uma conexão de rede. Os objetos são denominados de agentes e são providos dentro do ambiente Smalltalk

orientado a objetos, como um significado consistente para acesso à recursos através da rede e, localmente, na estação de trabalho. Os agentes utilizam os despachantes (intermediários) para realizar o trabalho e os agentes são introduzidos nos serviços através deles. A tríade agente-despachante-serviço caracteriza a aplicação.

Quem faz o processo de empacotamento da aplicação para a liberação final é o Empacotador/ENVY, isto é, este torna a aplicação pública da seguinte forma: pega o código fonte da aplicação, compila em código objeto e edita a união do resultado com as bibliotecas necessárias. O módulo resultante é a versão em tempo de execução da aplicação. As ferramentas que são usadas para produzir a aplicação, como os editores, projetos de telas e ambientes de depuração, são, naturalmente, bem separadas da aplicação desenvolvida em uma estação de trabalho. Logo, várias aplicações individuais são empacotadas em uma aplicação cooperativa.

O ENVY/400 é integrado com um Gerenciador ENVY de programação cooperativa e facilidades de controle de versões.

Os ambientes de engenharia de software, assim como o Gerenciador ENVY, são planejados para prover as facilidades requeridas para o desenvolvimento e manutenção de grandes sistemas de software. Isso requer facilidades para coordenar todos os aspectos referentes ao ciclo de vida do software. Portanto, este sistema provem os seguintes serviços:

- gerenciamento de componentes: suporta a identificação dos objetos que são introduzidos no sistema e sua relação com outros;
- gerenciamento da história: é responsável pela manutenção completa da história de todos os componentes de software desenvolvidos utilizando o ambiente. Isto, mantém separadamente a história completa de cada método, classe, aplicação e mapeamento de configuração criadas;
- gerenciamento de configuração: o sistema define o número de componentes de software e as relações entre eles. Cada componente pode ter um número de diferentes versões e edições. Uma função importante do ambiente de desenvolvimento de software é prover os seus usuários com configurações dessas edições de componentes;
- gerenciamento de mudanças: assegurar que as mudanças de componentes são executadas pelas pessoas autorizadas para isso. Em um ambiente multi-usuário, assim como o Gerenciador ENVY, o gerenciamento de mudanças é necessário para assegurar que o desenvolvimento concorrente dos componentes é manipulado de forma correta e efetiva.

Portanto, o sistema cooperativo ENVY/400, além de ser um ambiente cooperativo de desenvolvimento de programas, é um ambiente também de aplicação.

Em termos gerais, o modelo de interação dos sujeitos usuários/programadores pode ser visto da seguinte forma:

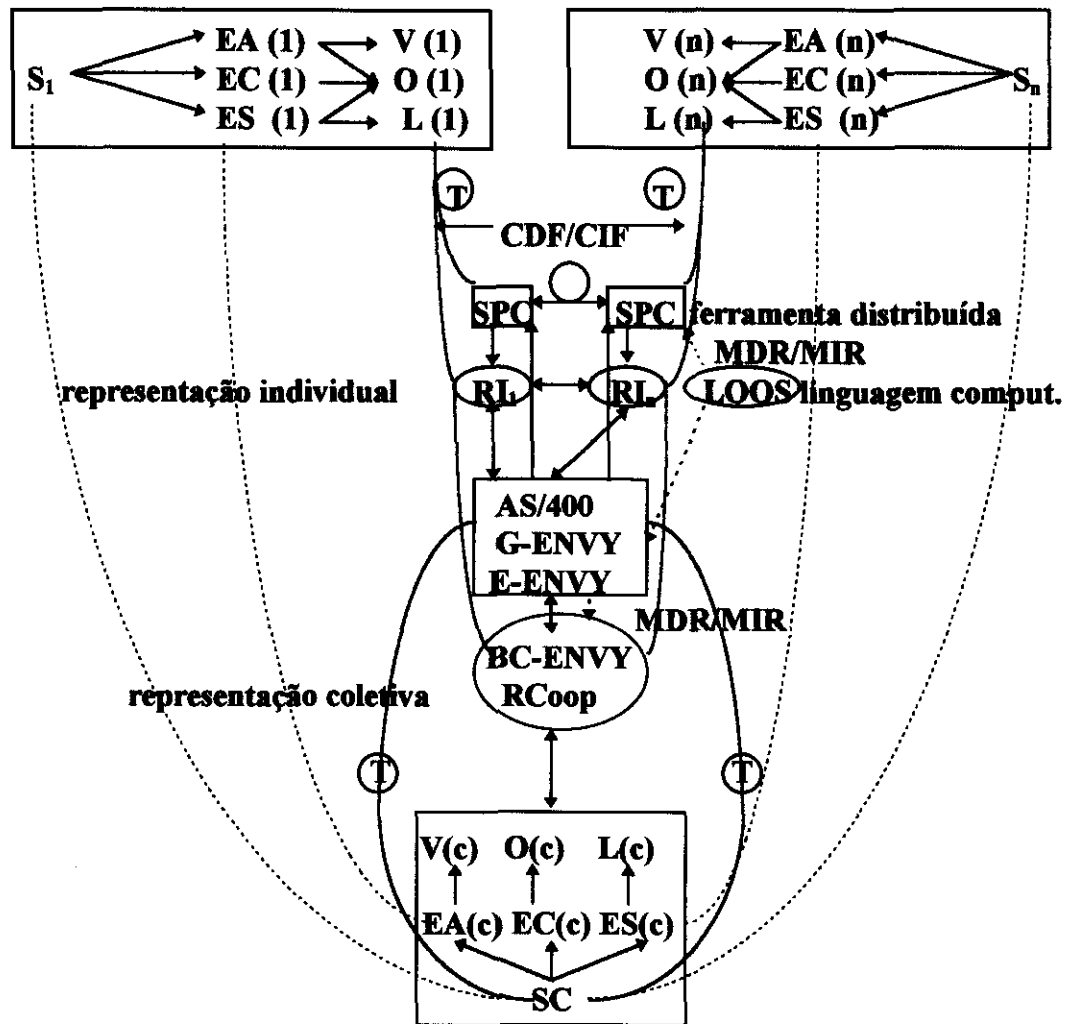


FIGURA 36 - Modelo de interação do sujeito coletivo com o ambiente ENVY/400

onde: S_1 : sujeito₁; S_n : sujeito_n; AS/400: serviços do hospedeiro AS/400; SPC: suporte PC; T: tela, teclado, etc...(canal de comunicação sujeito-ferramenta); RI_1 : representação do sujeito₁ (unidades de programas); RI_n : representação do sujeito_n; BC-ENVY: biblioteca compartilhada baseada em AS/400 do ambiente ENVY; G-ENVY: Gerenciador ENVY E-ENVY: Empacotador ENVY; LOOS: linguagem orientada a objetos Smalltalk; RCoop: representação cooperativa; MDR: manipulação direta sobre a representação; MIR: manipulação indireta sobre a representação; MDRCoop: manipulação direta sobre a representação cooperativa; MIRCoop: manipulação indireta sobre a representação cooperativa.

Logo, este modelo pode ser lido da seguinte forma:

- cada sujeito (S) tem a sua estação de trabalho Windows 3.1 ou OS/2 v2.0 e um suporte de PC. Através desta ferramenta, eles interagem com os serviços AS/400, que também tem um suporte de PC. Cada sujeito cria os seus componentes, que são unidades de programas, desenvolvendo a sua própria representação individual. Tanto esta quanto os componentes entram no módulo Gerenciador/Empacotador ENVY (G-ENVY, E-ENVY), empacotando todas as aplicações geradas nas estações de trabalho e liberando a representação cooperativa (RCoop), que é o produto final. Todos os sujeitos compartilham a biblioteca do ambiente ENVY/400 (BC-ENVY). Os componentes e as aplicações podem ser manipuladas pelos programadores de forma direta ou indireta.

6.3.2 Programação em equipe e controle de versões

Uma biblioteca desenvolvida coletivamente é uma base de dados do gerenciador ENVY em formato de arquivo PC e é armazenado em um folder compartilhado AS/400. Ele armazena versões e edições de todos os objetos ordenados desde aplicações, sub-aplicações, classes até métodos individuais.

A figura 38 mostra como se carrega e salva sub-aplicações para/desde a biblioteca compartilhada.

Uma aplicação ENVY/400 ou sub-aplicação é uma coleção de classes existentes, sejam estas novas ou extensivas, cuja forma é uma componente reutilizável. A biblioteca compartilhada mostrada na figura 37 é instalada em um folder compartilhado AS/400 DASD.

As ferramentas desenvolvidas colaborativamente providas com ENVY/400, contém browsers extensivos para o gerenciamento de aplicações, mudança de história, decisões de onde usar e onde chamar (mensagens enviadas, recebidas, referenciadas), capacidade para recuperação de informações "crashes" de forma rápida e carga e descarga de componentes desde as imagens dos desenvolvedores. Os desenvolvedores são distribuídos segundo as propriedades de classes e aplicações; somente o dono de uma aplicação pode liberar uma nova versão para o resto da equipe de desenvolvedores.

Dessa forma, são substituídos os estilos antigos registrados/enviados de versões gerenciadas em alta velocidade.

A figura 39 mostra como uma aplicação é aberta para modificações (novas edições), como as classes e os métodos são alterados e como a aplicação é versionada e liberada para o resto da equipe de desenvolvedores.

Qualquer componente ENVY-gerenciado pode ser aberto para alterações fazendo uma nova edição. As edições são armazenadas por data/tempo.

Quando uma edição é versionada, torna-se congelada e é dada uma identificação versão ID a ela. O browser dispara imediatamente quando encontra diferença entre duas versões ou edições. O gerenciador de projeto e desenvolvimento tem total visibilidade de todas as partes do ciclo de vida do projeto.

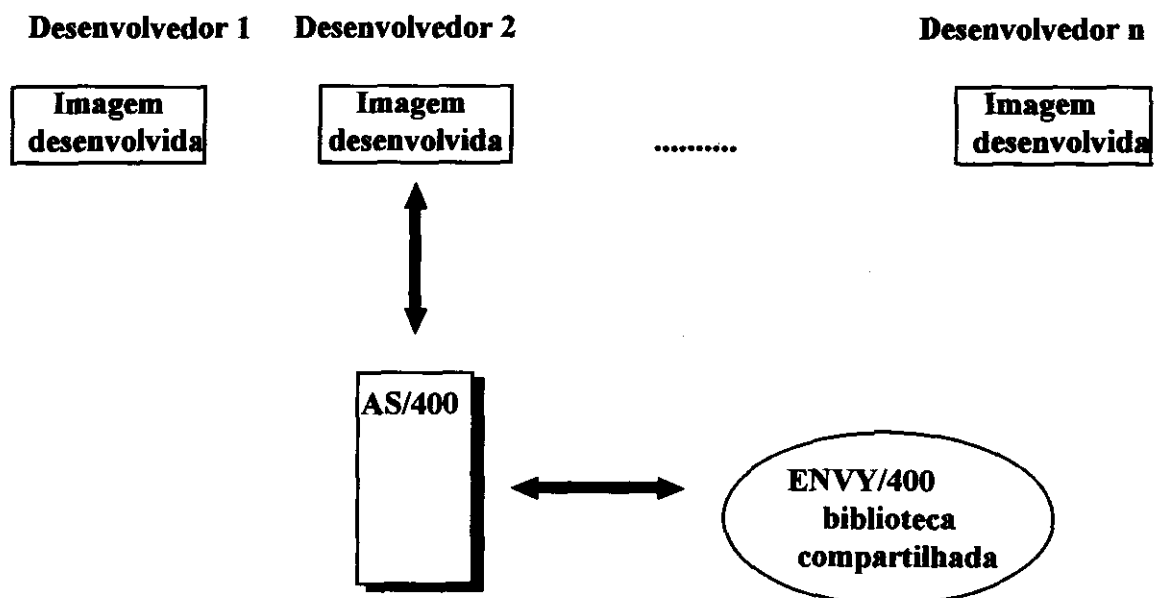


FIGURA 37- Biblioteca compartilhada ENVY

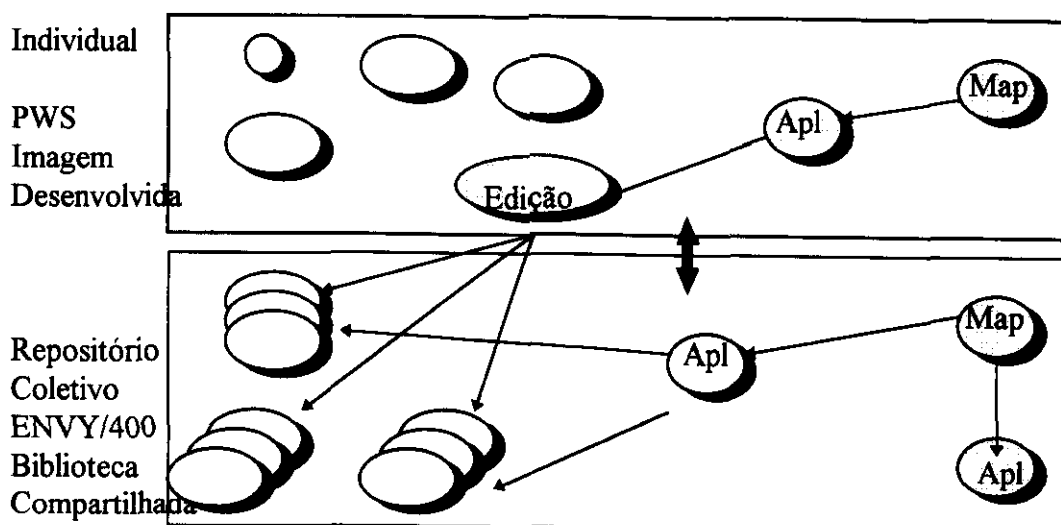


FIGURA 38 - Salvando e carregando desde o repositório compartilhado

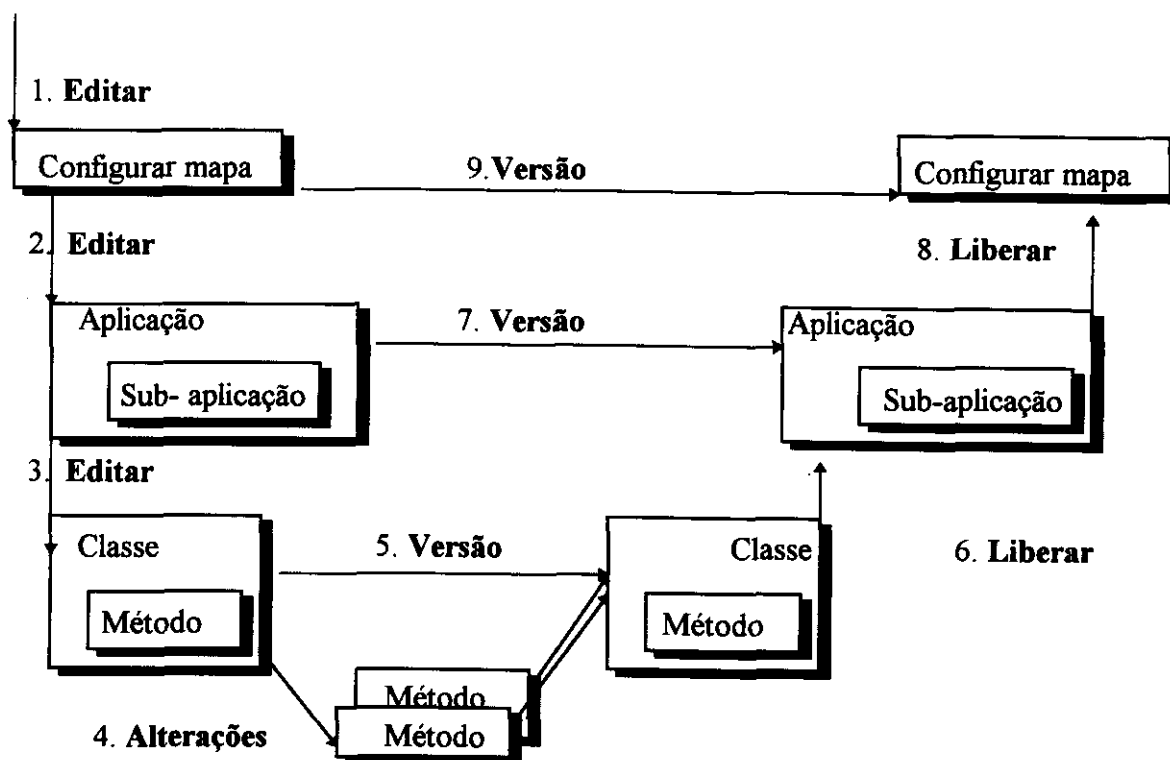


FIGURA 39 - Edições e versões

6.3.3 Operações lógicas/infralógicas do ambiente de desenvolvimento cooperativo de programação ENVY/400

Quando se fala em programação em equipe estamos nos referindo às co-operações, isto é, operações realizadas em conjunto, visando um objetivo em comum, o desenvolvimento de programas. Neste sistema a operação coletiva nada mais é do que a combinação de operações individuais. Assim, pode-se dizer que todas as operações realizadas pela equipe de desenvolvedores são do tipo individual mas todas as ações

refletem sobre o coletivo no momento em que os desenvolvedores tem que compartilhar e liberar (valor) tudo o que eles estão fazendo de forma imediata. Portanto, o produto final é cooperativo porque todos irão utilizar nos seus componentes de software elementos desenvolvidos por toda a equipe.

A operação coletiva diz respeito à criação de versões de uma representação coletiva. A criação de versões é o versionamento e, este, pode ser feito tanto pelo sujeito individual quanto pelo sujeito coletivo.

No caso do sujeito individual, este pode criar diferentes versões. Portanto pode ser visto da seguinte forma:

$$S_1 (v_0, v_1, v_2, v_3, \dots, v_n);$$

$$S_2 (v_0, v_1, v_2, v_3, \dots, v_n);$$

$$S_3 (v_0, v_1, v_2, v_3, \dots, v_n), \text{ onde:}$$

S_1 : sujeito₁; S_2 : sujeito₂; S_3 : sujeito₃; v_0 : versão 0; v_1 : versão 1; v_2 : versão 2; v_3 : versão 3; v_n : versão n

No caso do sujeito coletivo, o grupo de desenvolvedores estará criando diferentes versões, portanto:

$$SC: \begin{pmatrix} S_1 (v_0, v_1, v_2, v_3, \dots, v_n) \\ S_2 (v_0, v_1, v_2, v_3, \dots, v_n) \\ S_3 (v_0, v_1, v_2, v_3, \dots, v_n), \dots \end{pmatrix}$$

então:

Sujeito Coletivo: {[v₀,S₁], [v₀,S₂], [v₀,S₃], [v₁,S₁], [v₁,S₂], [v₁,S₃], [v₂,S₁],.....}, ou seja, o sujeito coletivo é a combinação dos diferentes sujeitos individuais e as versões criadas por eles. Este nada mais é do que a biblioteca compartilhada.

Sendo assim, as versões individuais são criadas a partir da biblioteca compartilhada e, portanto, chamam o sujeito coletivo. Ou seja, este é criado a partir do individual e vice-versa.

O sujeito coletivo também pode ser visto como a combinação das ações individuais da seguinte forma:

$$\begin{array}{c} \xrightarrow{S_2} \quad \xrightarrow{S_1} \\ \text{Sujeito Coletivo} \rightarrow (v_0, v_1, v_2, v_3, v_4, v_5, \dots, v_n), \text{ ou seja:} \\ \downarrow \quad \xrightarrow{S_3} \quad \xrightarrow{S_2} \\ S_1 \end{array}$$

O S₁ desenvolve a versão 0 (v₀), o S₂ carrega a versão 0 e cria a partir dela a versão 1 (v₁), o S₃ pega a versão 1 criada pelo S₂ e desenvolve a versão 2 (v₂), o S₁ pega a versão 2 e cria a versão 5 (v₅) e o S₂ cria a partir da versão 3 (v₃), a versão 4 (v₄). Com isso quer se mostrar que as operações individuais nada mais são do que operações realizadas em cima de operações, ou seja, *co-operações*. A ação cooperativa está presente continuamente no momento em que um sujeito individual utiliza uma versão que já se encontra na biblioteca compartilhada e, portanto, no sujeito coletivo, para criar uma nova.

Mas, para isso, tem que levar em conta que no sujeito coletivo está presente a estrutura afetiva que diz respeito aos valores. Para identificar o *valor poder* que um desenvolvedor tem em relação à representação do objeto (versão) criada por ele, a leitura pode ser feita da seguinte forma: (versão <valor>, sujeito). Se a versão vir acompanhada de um * no lugar de <valor>, significa que esta versão é pública e, se não aparecer nada, ainda é uma versão que somente pode ser gerenciada pelo sujeito que a criou; o sujeito é o gerente da versão. Exemplificando:

Sujeito Coletivo: {[v₀, S₁], [v₁*, S₁], [v₀*, S₂], [v₁, S₂],...}, isso quer dizer que, a versão 0 do sujeito 1 não pode ser utilizada por nenhum outro sujeito a não ser que

este a libere publicamente, mas a versão 1, do mesmo sujeito pode ser utilizada publicamente, ou seja, por qualquer usuário do sistema; a versão 0 criada pelo sujeito 2 é pública, mas a versão 1 somente pode ser utilizada por ele mesmo.

O objetivo deste estudo é identificar na ferramenta ENVY/400 algumas operações lógicas e/ou infralógicas tanto de controle da ferramenta quanto de manipulação da representação de forma direta e/ou indireta. Portanto, o seguinte passo é caracterizar estas operações identificando-as no ambiente em questão.

1) Operação elementar infralógica de adição e partição primitiva:

1.1) Falamos de uma operação infralógica porque um componente de software (todo) é formado de partes (mapas, aplicações, sub-aplicações, classes, métodos,...) que dependem umas das outras. Esta operação pode ser identificada no ambiente ENVY/400 na **criação de novos componentes**. Esta descreve as operações requeridas para a criação de novas instâncias de componentes de software armazenadas no gerenciador/ENVY. Por exemplo, na *configuração de mapas* é preciso ir para o menu **Create Configuration Map**, no prompt, digitar o nome do novo mapa e uma nova configuração de mapa será criada. A nova edição é gerenciada pela pessoa que a criou. Quando o mapa for criado, a edição inicial contém todas as aplicações que foram carregadas na imagem. Como pode ser observado, este caso de operação trata de um nome a ser dado à representação de um objeto que, neste caso, é o mapa a ser configurado. Portanto, é uma **operação individual de manipulação direta da representação** (MDR-mapa) do tipo (L,O). O próximo passo é usar os itens do menu **Add New Application...**, **Release Other...** e, **Delete Application** da lista de aplicações da janela para modificar o mapa editado a fim que contenha as aplicações desejadas. É possível adicionar aplicações no mapa que não são carregadas na imagem. Logo, é uma **operação elementar infralógica de adição primitiva** pois será adicionado ao mapa uma aplicação, portanto, o mapa (todo) já existe com suas aplicações (partes) e será colocada mais uma, para depois liberá-la ou até deletar uma aplicação. Portanto:

C: todo = mapa; B: partes = aplicações; B': aplicação adicionada, logo

$B + B' = C \Rightarrow$ *operação infralógica de adição primitiva.*

Se for utilizada a operação **Delete Application** para subtrair uma aplicação da lista:

$C - B' = B$ (novo mapa somente com o conjunto de aplicações contido em B) \Rightarrow *operação infralógica de partição primitiva.*

1.2) Para criar um novo usuário, cada biblioteca mantém uma lista de objetos usuários, representando os desenvolvedores que irão utilizar a biblioteca. Os atributos desses objetos usuários são: Nome único: o token usado pelo sistema para identificar o usuário. Os valores possíveis para o nome único serão identificadores do login da rede ou números empregados. Nome completo: a forma do nome de usuário será usado para descrever o usuário na interface de usuário do gerenciador/ENVY. Nome de rede: a identificação (ID) do login da rede para os usuários. Sempre que a imagem é conectada em uma biblioteca, a senha será requerida. Cada biblioteca tem um usuário conhecido como **Livrary Supervisor**. Somente este usuário irá criar ou modificar os usuários. Para adicionar um novo usuário para a biblioteca, ativar o menu ENVY e executar as

seguintes operações: selecionar **System => Create User...**, entrar o nome único do usuário, nome completo e nome de ID da rede, na caixa de diálogo.

Na criação de um novo usuário estará aparecendo a **operação infralógica de adição primitiva** já que haverá um incremento no número de elementos usuário. Além disso, o usuário tem que ser identificado através da linguagem (L,O) com três tipos de nomeações. Existirá ainda a relação de *valor poder* sobre os usuários; quem cria e modifica é conhecido como Library Supervisor (V,O). Esta operações são de **manipulação indireta sobre a representação**.

1.3) Descarregar um componente (classe, aplicação) de uma imagem é a remoção deste da imagem. Não tem qualquer efeito sobre a biblioteca. Selecionar o nome do mapa que se deseja descarregar, selecionar a versão e edição desejada, no menu, selecionar a opção **Unload Edition**. Cada edição ou versão de aplicação que foi liberada na edição de configuração de mapa será descarregada da imagem. Configurações de mapas são nomeadas coleções de aplicações. É possível que somente pedaços de configurações de mapa estejam presentes na imagem. Se este for o caso, somente esses pedaços carregados serão removidos.

Um componente de uma imagem é o todo (D) composto de mapas (M), classes (C), aplicações (A) (diferentes partes).

Assim, descarregar um mapa (M1), por exemplo, que é um conjunto de aplicações (A1, C1,...) significa removê-lo da imagem. Assim, pode-se observar a **operação de partição primitiva**, onde se $D: (M1: A1, C1, \dots) + (M2: A2, C2, \dots)$, e assim por diante.... então: a imagem ficará da seguinte forma: $D - (M1: A1, C1, \dots) = (M2: A2, C2, \dots)$, e assim por diante...mas sem remover o mapa da biblioteca compartilhada. Ou seja, somente é removido do sujeito individual e não do coletivo.

1.4) Outra operação é a de deleção de um componente. Esta operação tem efeitos sobre a biblioteca. Ex: se é deletado um método de uma classe, o método é removido da imagem e a classe editada/gravada na biblioteca é atualizada, então o método não é mais parte da classe. O mesmo acontece deletando classes de aplicações. Portanto, além do componente ser removido da imagem, a mesma **operação de partição primitiva** será identificada na biblioteca compartilhada, no momento da deleção do componente. Assim, o componente é removido tanto do sujeito individual quanto do sujeito coletivo.

2) Operação infralógica de ordem de colocação:

2.1) Para criar uma aplicação, devem ser feitos os seguintes passos: na primeira janela, selecionar no menu **Applications => Create**, no prompt, digitar o nome da nova aplicação e vai aparecer uma mensagem se já existir um nome igual ao que foi dado. Será aberta a opção de **PrerequisitesDialog** que permite que o usuário defina os pré-requisitos para uma nova aplicação. Depois da escolha, selecione o botão de **OK**. Assim uma nova aplicação será criada. Essa nova edição é gerenciada pela pessoa que a criou. Os pré-requisitos descrevem a situação onde um componente deve existir em uma definição do sistema antes de vir um outro. Por exemplo: aplicações definem pré-requisitos de aplicações, indicando que esses pré-requisitos são requeridos pela aplicação antes desta funcionar. Portanto, está se referindo à criação de uma aplicação que deverá seguir uma ordem para ser criada e poder funcionar. Neste caso, é uma **operação de manipulação indireta da representação** (aplicação), onde esta é criada através do nome que se dá a ela. Logo, está se falando de uma estrutura do tipo (L,O). Mas, desta vez, esta operação somente é executada se existir uma operação de ordem de colocação

da aplicação. Estamos frente a uma **operação infralógica de ordem de colocação** porque os pré-requisitos se referem a uma ordem que tem que ser seguida pelos elementos do sistema para que a aplicação possa funcionar.

Neste caso, também podem ser identificadas as operações infralógicas de **adição primitiva**, através da criação de uma aplicação que será parte (B') de um todo (c: mapa), e a de **ordem de colocação**.

Além disso, esta aplicação somente poderá ser gerenciada pela pessoa que a criou, portanto, percebe-se uma relação de **poder** sobre o objeto que é o valor sobre a aplicação (V,O) - manipulação direta sobre a representação - MDR. O valor poder é uma **operação coletiva** porque diz respeito a um grupo.

2.2) Na opção Importando e exportando, os componentes do Gerenciador/ENVY podem ser importados e exportados de diferentes bibliotecas como requerido pelas organizações de desenvolvimento individual. O processo de importação/exportação mantém o controle de versão e provêm outras importantes capacidades. Os benefícios são: execução improvisada comparada ao tradicional processo de colocar arquivo/retirar arquivo, mantimento de versões através das múltiplas bibliotecas, criação de novas bibliotecas com versões selecionadas e não versões intermediárias, criação de pequenos arquivos backup de componentes selecionadas, habilidade de trocas rápidas de componentes desenvolvidos entre bibliotecas dispersadas geograficamente. Um componente deve ser versionado antes de ser exportado.

Exportar e importar significa deslocar de um lugar a outro. Logo é uma **operação de ordem de colocação** se os elementos se mantêm sempre os mesmos e somente são mudados os lugares "físicos" dos mesmos. Mas, no caso da criação de novas bibliotecas, ou arquivos, a **operação infralógica é de adição primitiva** já que está adicionando novos elementos ao sistema. Nestes casos, trata-se de operações de **manipulação indireta sobre a representação** porque o usuário somente dá a ordem mas quem executa é a ferramenta.

3) Operação lógica de correspondência:

3.1) Para criar um método da ApplicationsBrowser é preciso selecionar a aplicação que se deseja adicionar o método, selecionar a classe que se deseja adicionar o método, no menu, selecionar a opção **New Method Template** e na janela de texto, editar o texto que corresponde ao ambiente do método desejado. Depois salvar (**Save**) o texto do menu. Selecionar uma classe (todo) onde será adicionado o método (parte), significa que além de ser uma **operação infralógica de adição primitiva**, deverá ser escrito um texto seguindo certas regras sintáticas da linguagem computacional. Portanto deverá existir uma **correspondência** e uma ordem de colocação dos elementos da linguagem. Trata-se de uma **operação individual de manipulação direta sobre o método (representação)** mas, no momento em que o usuário dá uma ordem para salvar o método quem salva é o sistema, portanto, é uma **operação indireta de manipulação sobre a representação**.

3.2) Esta operação de correspondência também pode ser identificada na **criação de uma nova edição**. Aqui descreve-se as operações usadas para a criação de novas edições de componentes existentes. Na *Configuração de mapas*, ativar o menu **ConfigurationMapsBrowser**, selecionar a configuração de mapas, depois selecionar a versão de configuração de mapas de onde será copiada a nova edição. Novas edições somente podem ser criadas de versões. Podem ser criadas só pelo gerente da versão

selecionada. Ativar e selecionar **Create a New Edition**. Uma nova edição conterà as mesmas versões de aplicações que a versão da qual foi criada

Nesta operação está presente a relação de valor sobre o objeto, porque quem pode criar uma nova edição de mapa somente pode ser o gerente que criou a versão da edição de onde será copiada. Portanto => **valor de poder (V)** sobre a nova edição do mapa (O)

A nova edição é copiada de uma versão que já existe mas terá um nome diferente. Assim, pode-se identificar que se a edição de onde foi copiada (B) é composta das partes $A1 + A2$, por exemplo, a nova edição B' terá as mesmas partes $A1' + A2'$, assim, $B = B'$. Portanto, está é uma **operação de correspondência**.

3.3) Carregando um componente de edição: uma das capacidades mais importantes do Gerenciador/ENVY é a habilidades de carregar métodos, classes, aplicações e configurações de mapa desde a biblioteca. A abordagem do Gerenciador/ENVY é manter na biblioteca todos os códigos compilados que foram gerados pelos desenvolvedores. Cada operação de carga executada sobre uma imagem é uma operação atômica. Isso significa que ou toda a carga é feita com sucesso ou toda ela é abortada. Será descrito como as edições de componentes são carregadas da biblioteca para a imagem. Em todos os casos o processo é similar. O primeiro passo é identificar o componente a ser carregado. O segundo é identificar que versão ou edição de componente será carregado. Por exemplo, para *Configurar mapas* é preciso selecionar o nome do mapa que será carregado, selecionar a versão e edição desejada, do menu, ativar a opção **Load Edition**. Cada uma das edições ou versões de aplicação que foram liberadas na edição de configuração de mapa serão carregadas na imagem. Mapas configurados são nomeados e são versionadas as coleções de aplicações. Carregando um na imagem, carrega todas as edições ou versões de aplicações que são parte dela. Portanto, não existe uma configuração do objeto mapa que é carregado na imagem. Quando um mapa é carregado, o usuário trabalha diretamente com as aplicações que o contém.

Esta operação pode ser comparada a uma **operação do tipo lógica**, que trata de objetos individuais e não das suas partes. Isso porque, como foi dito anteriormente, cada operação de carga sobre uma imagem é do tipo atômica, ou seja, não pode ser dividida.

Trata-se de uma operação de **manipulação indireta sobre a representação** que, no caso, é a configuração de mapa, porque quem dá a ordem é o usuário mas quem executa é a ferramenta.

3.4) Conectando uma imagem à biblioteca: a imagem do desenvolvedor deve ser conectada na biblioteca do Gerenciador/ENVY. Uma imagem deve ser conectada a diferentes bibliotecas de tempos em tempos baseados nos requerimentos organizacionais. A cada tempo uma imagem é conectada somente a uma biblioteca. Esta biblioteca deve ter todas as aplicações que estão correntemente carregadas na imagem. Uma checagem completa consistente é executada quando há conexão com uma nova biblioteca. Para conectar uma imagem na biblioteca, acessar o menu ENVY e executar os seguintes passos: selecionar **System => Connect Image To...**, selecionar o caminho do diretório e digitar o nome do arquivo da biblioteca na caixa de diálogo.

A conexão da imagem à biblioteca significa que em todos os casos de conexão existirá uma **operação lógica de correspondência** do tipo biunívoca (1 para 1) ou co-unívoca (1 para muitos). Além disso é preciso identificar o nome do arquivo através de

uma operação do tipo (L,O). Estas operações são de **manipulação indireta sobre a representação**.

4) Operação com valores:

4.1) Para criar uma nova edição de uma sub-aplicação, devem ser executados os seguintes passos: selecionar a versão da sub-aplicação de onde será copiada a nova edição. Novas edições somente podem ser criadas de versões e selecionar do menu a opção **Create New Edition**. A nova edição conterá a mesma versão de classe que a versão de onde foi criada. É recomendado que quando for criada uma nova edição de sub-aplicação que seja liberada imediatamente para a edição de aplicação onde esta fará parte dela.

Esta nova edição de sub-aplicação tem que seguir uma *regra coletiva* do ambiente que é a **liberação imediata** (valor) para ser utilizada também por outros programadores.

4.2) **Versionando**: descrevem-se os passos requeridos para versionar componentes de software armazenados na biblioteca. Quando um componente é versionado, o sistema não permite nenhum tipo de alteração no componente.

Quando um componente é versionado por um determinado usuário, somente este tem o **poder (valor)** de alterá-lo porque este é o gerente do mesmo; quem o versionou (V,O). Portanto trata-se sempre de uma **operação de manipulação indireta** (porque quem versiona é o sistema com a ordem do usuário) **sobre o objeto** (mapas, aplicações, etc...).

5) Operações com a linguagem:

5.1) A edição de novos métodos é criada automaticamente sempre que uma alteração no texto do método é salvo pelo usuário. Antes de modificar um método, a aplicação que contém a classe definida ou a extensão devem ser uma edição. A classe pode ser aberta para alterações, mas para conveniência do desenvolvedor, uma edição será criada automaticamente se a versão é modificada. Para alterar o método desde a **ApplicationBrowser**: selecionar a aplicação que contém o método que se deseja modificar, selecionar a classe que contém o método que se deseja modificar, no janela texto, editar o método que corresponde ao ambiente desejado. Depois salvar (**Save**) o texto do menu.

Neste caso, como se trata de uma alteração no texto que define o método, está se falando da linguagem computacional. Assim uma **modificação na semântica**, significa, um novo método a ser salvo no ambiente. Portanto, a ordem dos elementos e os próprios elementos da linguagem pode ser alterados. Trata-se da **operação infralógica de ordem de colocação e de correspondência** (linguagem, método). A criação da nova edição do método, é uma **operação de manipulação indireta sobre a representação**, porque no momento que é modificada o texto, o ambiente salva o texto do menu.

6) Operação lógica inversa:

6.1) **Recuperando imagens "crashes"**: quando as alterações são perdidas, por não ter salvo elas ou porque ocorreu algum erro inesperado no sistema, todas as modificações podem ser recuperadas.

Em todos os casos de recuperação de aplicações, classes e métodos, trata-se da **operação lógica inversa**, pois estará compensando algo que foi perdido, revertendo a ação anterior.

7) Operações do sujeito coletivo:

7.1) Operações da biblioteca: uma biblioteca do Gerenciador/ENVY é uma propriedade compartilhada de banco de dados de objetos “gerenciados”. Uma equipe de desenvolvedores tem total flexibilidade em relação ao número de bibliotecas e seus conteúdos. Uma biblioteca Gerenciador/ENVY vazia pode ser criada a qualquer momento. Depois que foi criada com sucesso uma nova biblioteca, as configurações de mapas, aplicações e classes podem ser exportadas para uma nova biblioteca. Quando a nova biblioteca for criada, inicialmente não há usuários definidos. Para criar uma nova biblioteca, realizar os seguintes procedimentos: selecionar **System => Create New Library...**, selecionar o caminho do diretório e entrar o nome do arquivo da biblioteca da caixa de diálogo do sistema de arquivos que aparecerá. Não existem restrições quanto ao nome do arquivo e extensão do mesmo.

Não é identificada a relação de poder sobre o objeto porque inicialmente não há usuários definidos. Trata-se da **operação infralógica de adição primitiva** através da criação de novos elementos para serem adicionados à biblioteca. Também é preciso utilizar a linguagem para dar um nome ao arquivo, portanto, é uma **operação do tipo (L,O) de manipulação indireta sobre a representação.**

7 Proposta de uma base de desenvolvimento de um método de análise lógico-operatória do sujeito coletivo

Através do que foi descrito ao longo deste estudo, pode-se ter uma idéia dos elementos necessários para construir o método de análise lógico-operatória no nível coletivo.

Portanto, é preciso construir três quadros similares ao da figura 3 mas reformulando-o e estendendo-o através da inserção dos valores, da linguagem e dos objetos, junto com as operações e regras que coordenam as ações referentes a cada uma das estruturas afetiva, cognitiva e simbólica do sujeito coletivo.

Se pensamos que o sujeito coletivo pode ser tratado como uma só entidade, como mostra a figura 17, temos que levar em conta que este sujeito terá também os mesmos objetos de manipulação, relações entre eles, propriedades, a mesma escala de valores e a linguagem utilizada. O que se pretende investigar é como se comportam as operações lógicas destas três estruturas, assim como a coordenação das mesmas (regras).

Logo, podemos mostrar a organização externa do método de análise lógico-operatória de um sujeito coletivo na figura 40.

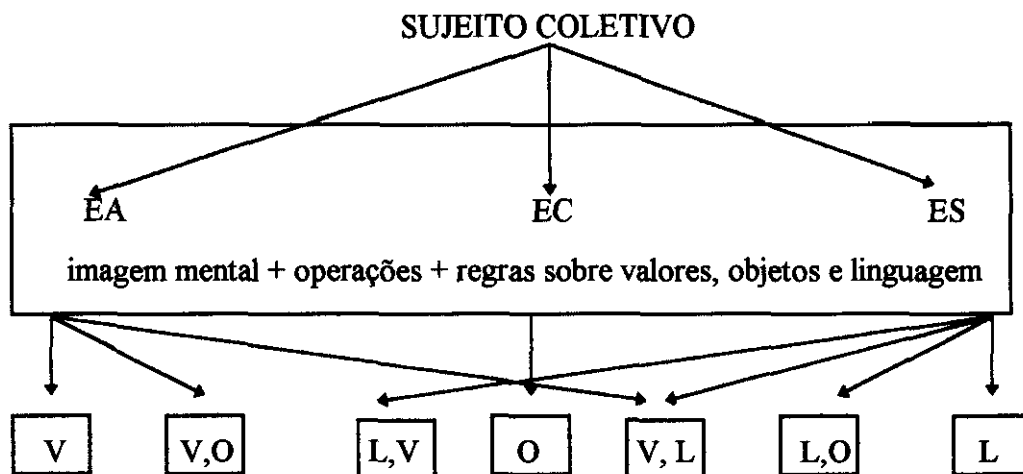


FIGURA 40 - Organização externa do método de análise lógico-operatória no nível coletivo

Cada quadrado é tratado separadamente e pode ser detalhado da seguinte forma:
Por exemplo, tratando-se dos valores V, é preciso definir:

- qual a escala de valores do sujeito coletivo;
- quais as relações estabelecidas entre esses valores (satisfação implica em uma dívida);
- propriedades dos valores (cognitivos: verdade/falsidade, sucesso/fracasso, reconhecimento/ gratidão, etc.; afetivos: satisfação positiva, negativa, nula, etc.) e,
- operações e a coordenação destas sobre os valores.

No caso de valores que envolvem objetos V, O :

- quais os objetos;
- quais os valores sobre esses objetos;
- as propriedades dos objetos e dos valores relacionados a eles;
- proposições de valores (atribuição de valor ao objeto) e,
- operações de valores sobre objetos e as suas respectivas regras.

Levando em conta os valores refletidos através de idéias V, L (\Rightarrow ; \Leftarrow) :

- qual é a linguagem (forma de comunicação);
- quais os valores sobre a linguagem;
- as propriedades da linguagem e dos valores relacionados a ela;
- proposições de valores (atribuição de valor à linguagem ou expressão de valores através da linguagem) e,
- operações de valores sobre a linguagem e da linguagem sobre os valores e as suas respectivas regras.

Da mesma forma ocorre com as estruturas cognitivas que se referem aos objetos

O :

- quais os objetos;
- suas propriedades;
- as relações estabelecidas entre os objetos;

- as proposições sobre objetos e,
- as operações e as formas de coordenação dessas ações sobre objetos.

Sobre a linguagem, podemos destacar os seguintes aspectos a serem analisados operatorialmente L :

- qual a linguagem;
- suas propriedades;
- as relações que o sujeito coletivo estabelece com a linguagem;
- as proposições sobre a linguagem e,
- as operações e regras da linguagem.

Por último temos a relação da estrutura simbólica com a cognitiva, isto é, como dar um nome a um determinado objeto L, O . Para isso, tem que definir:

- quais são os objetos e a linguagem;
- qual o significante do objeto (*propriedade*);
- as relações entre o valor simbólico e o objeto;
- as proposições da linguagem de objetos e,
- operações e regras sobre os mesmos.

No quadro final teremos as operações e as regras realizadas tanto com os valores, os valores de objetos, atribuição de valor à linguagem; os objetos em si, a linguagem de objetos, expressão de valores através da linguagem e a linguagem propriamente dita. Portanto, é sobre este resultado que é analisado operatorialmente o sujeito coletivo.

Existem inúmeras lacunas a serem investigadas sobre este assunto, que serão parte do objeto de estudo de possíveis trabalhos futuros. Aqui somente são apresentados os diferentes elementos e a relação entre eles que tem que levar em conta para a construção do método de análise lógico-operatória do sujeito coletivo.

Modelo mental do sujeito

Simbolização lógica
do modelo mental

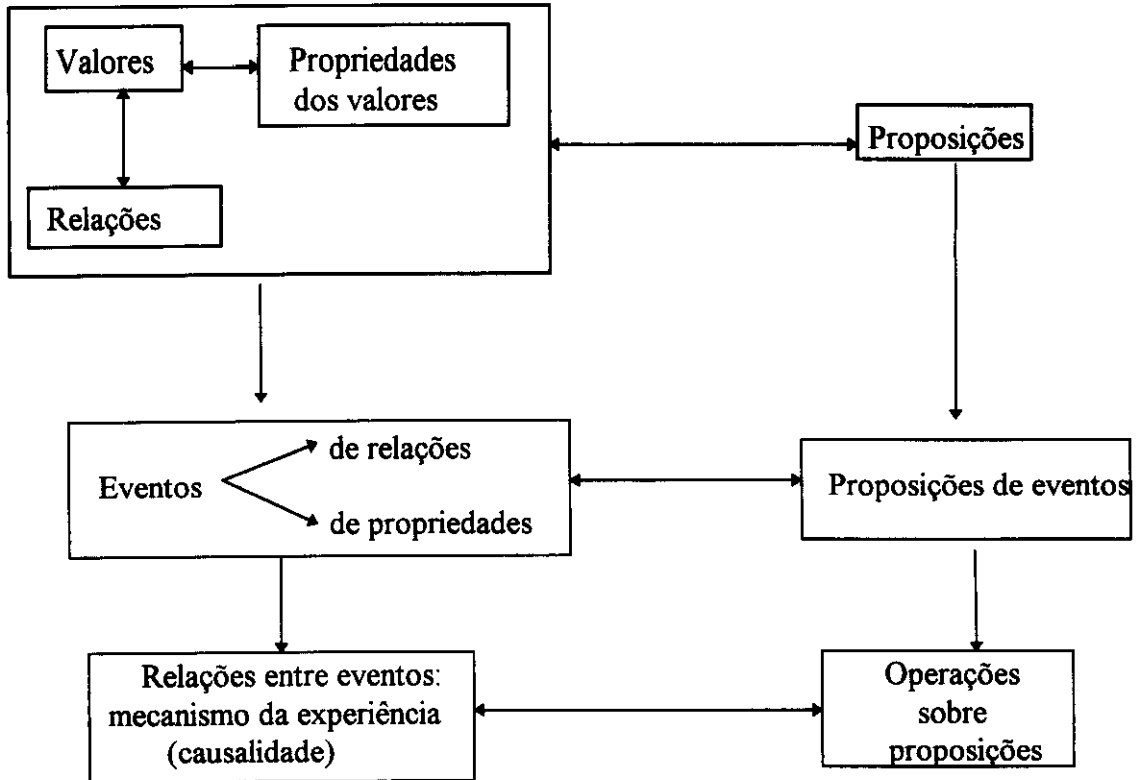


FIGURA 41 - Base para o desenvolvimento de um método de análise lógico-operatória em relação aos valores de um sujeito coletivo

Modelo mental do sujeito coletivo

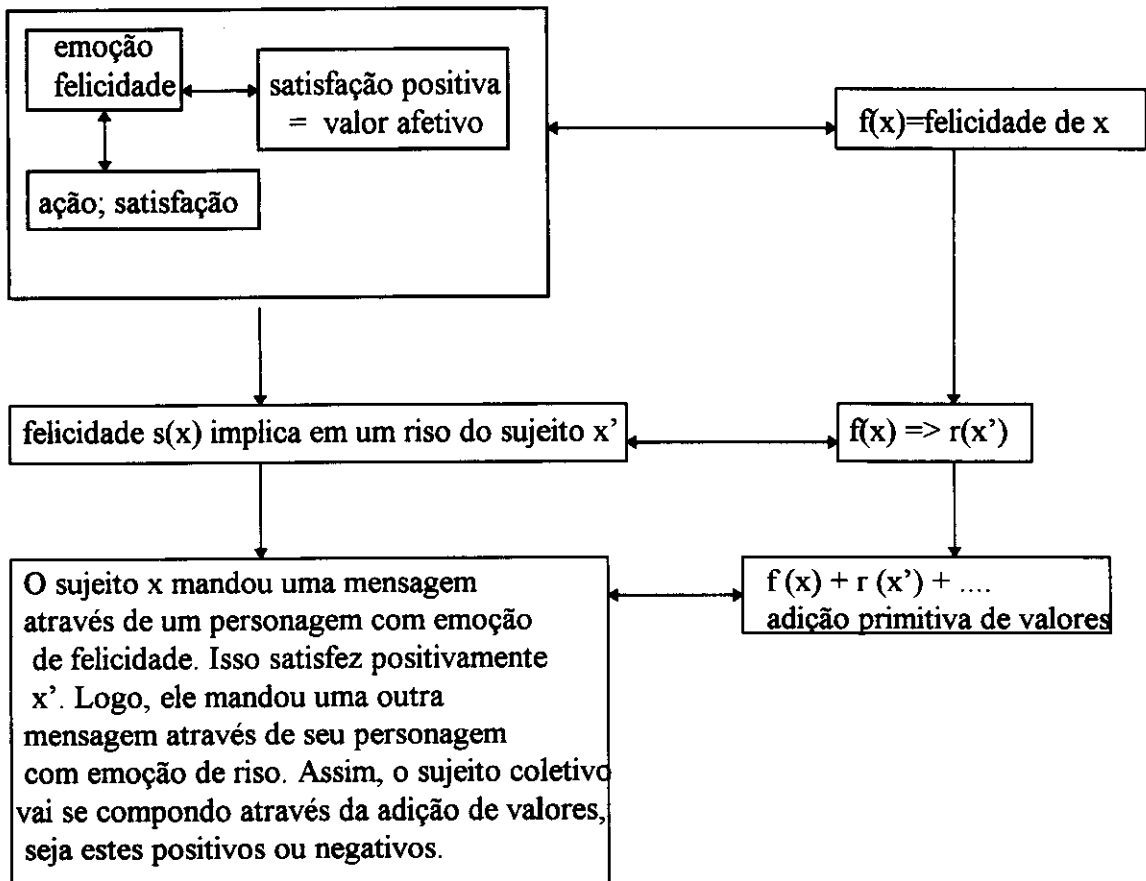
Simbolização lógica
do modelo mental

FIGURA 42 - Exemplo da aplicação do método de análise lógico-operatória em relação aos valores de um sujeito coletivo na ferramenta computacional cooperativa Microsoft Chat 2.0 Beta 1

Modelo mental do sujeito

Simbolização lógica
do modelo mental

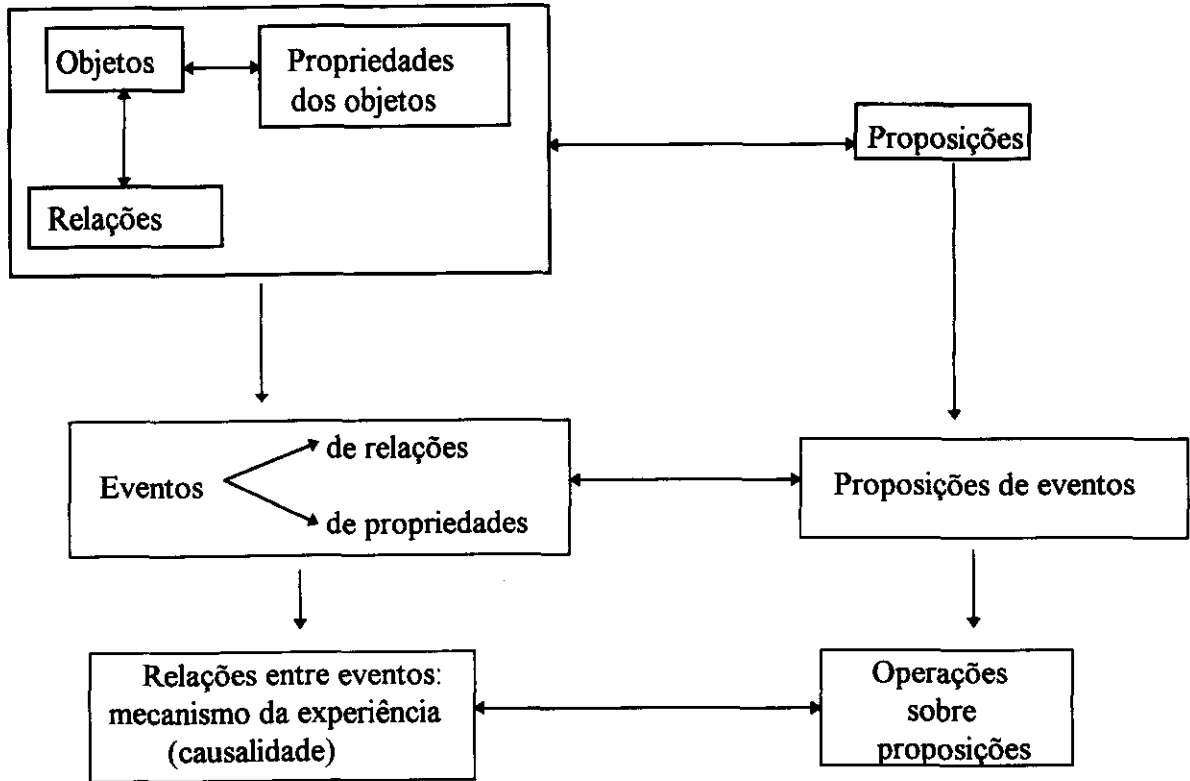


FIGURA 43 - Base para o desenvolvimento de um método de análise lógico-operatória em relação aos objetos de um sujeito coletivo

Modelo mental do sujeito

Simbolização lógica do modelo mental

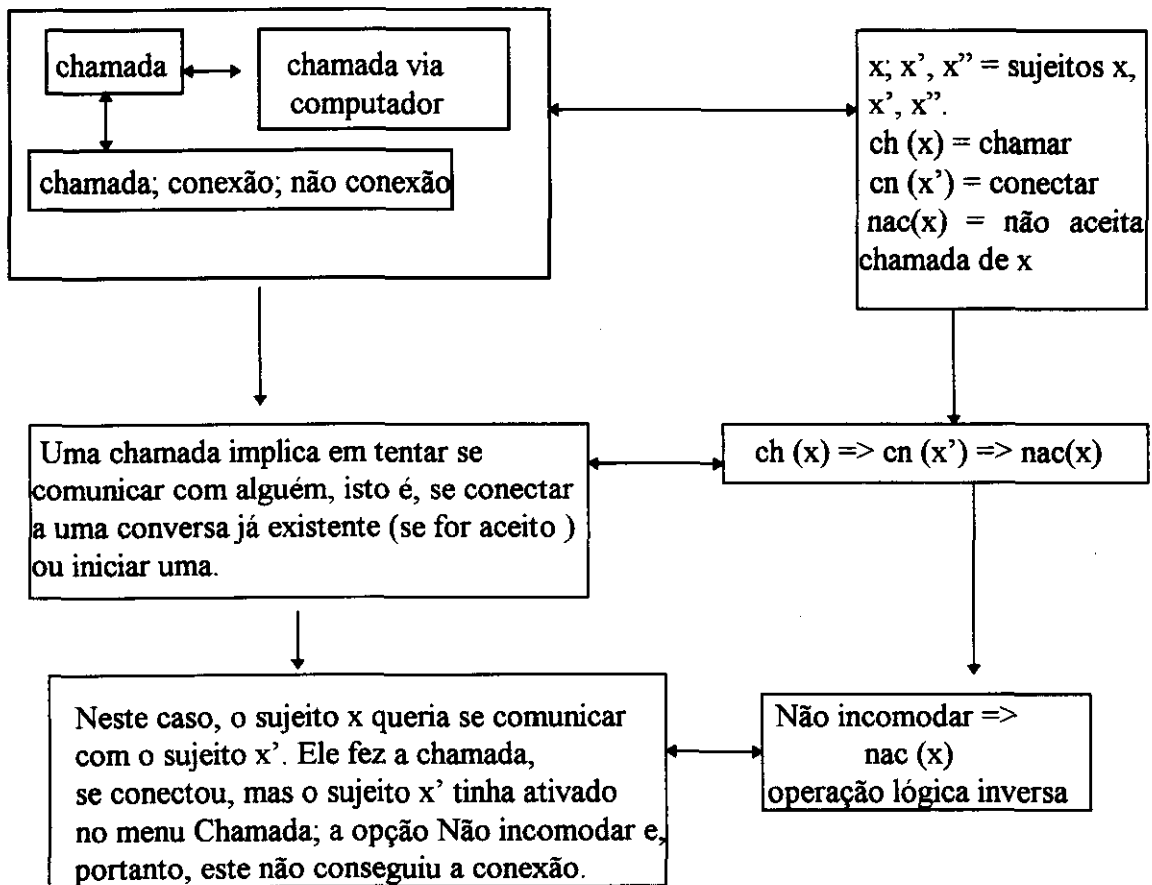


FIGURA 44 - Exemplo da aplicação do método de análise lógico-operatória em relação aos objetos de um sujeito coletivo na ferramenta computacional cooperativa Microsoft Netmeeting 2.0 Beta 4

Modelo mental do sujeito

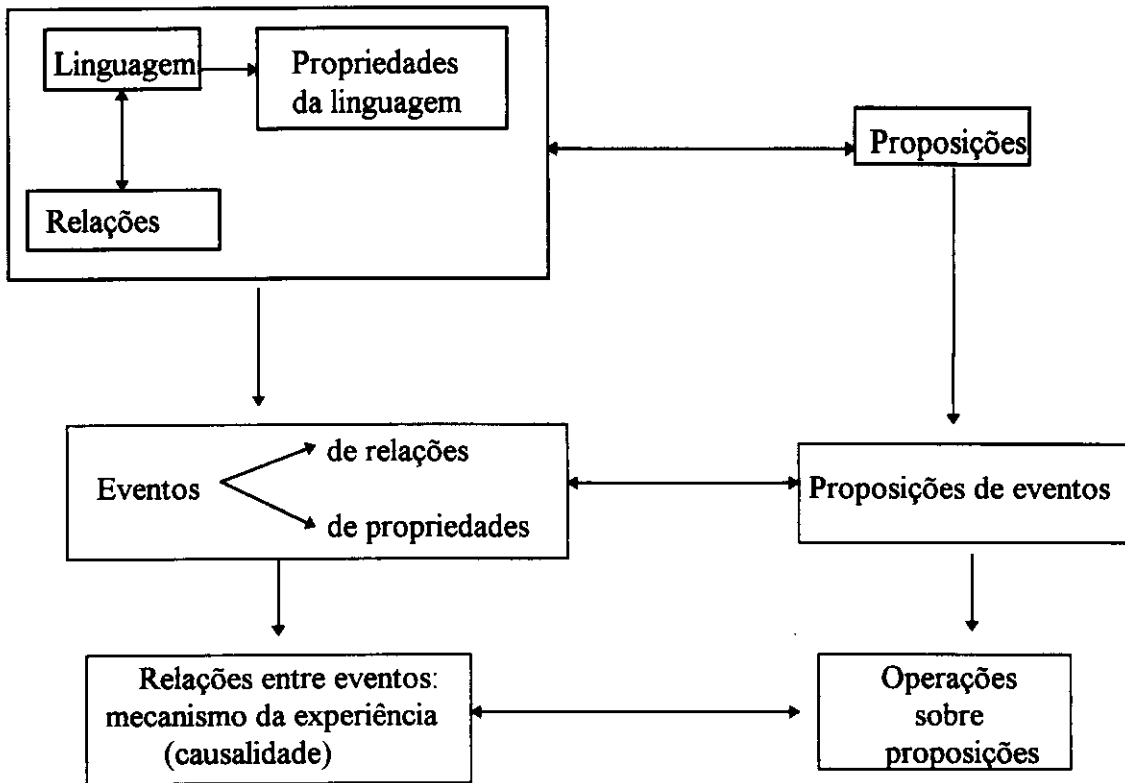
Simbolização lógica
do modelo mental

FIGURA 45 - Base para o desenvolvimento de um método de análise lógico-operatória em relação à linguagem de um sujeito coletivo

Modelo mental do sujeito

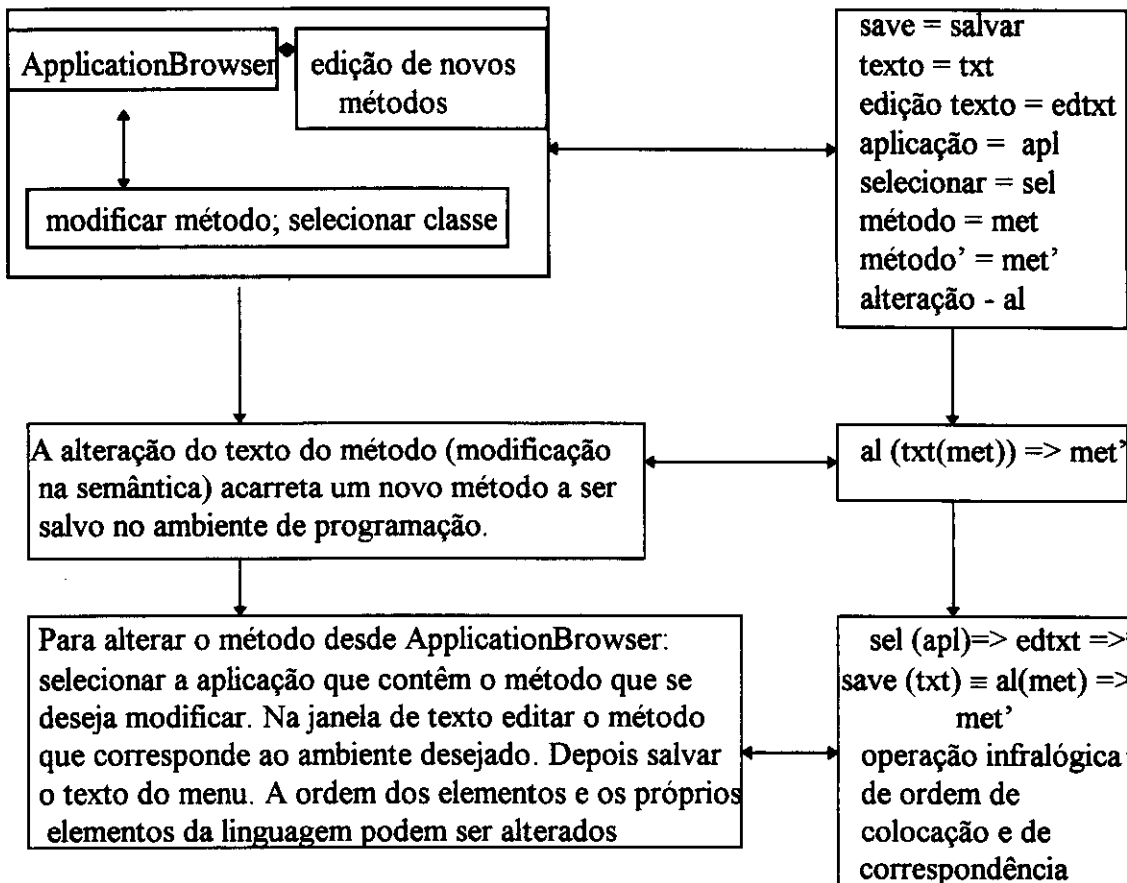
Simbolização lógica
do modelo mental

FIGURA 46 - Exemplo da aplicação do método de análise lógico-operatória em relação à linguagem de um sujeito coletivo no ambiente de desenvolvimento cooperativo de programação ENVY/400

8 Conclusões

O objetivo do trabalho foi delinear as várias contribuições que pode trazer a análise lógico-operatória no nível individual e cooperativo em relação às ferramentas computacionais. Para isso, foi preciso enfatizar os aspectos mais relevantes da mesma, que foram obtidos através da integração dos conhecimentos relacionados à teoria piagetiana, mais especificamente, da lógica operatória com a Ciência da Computação, focalizando a área de Computação Cooperativa ou CSCW- Computer Supported Cooperative Work.

Em primeiro lugar, focalizou-se o processo de construção das idéias que foram levantadas e desenvolvidas para a concretização desta pesquisa. Como analisar a interação entre um sujeito e um objeto? Por que não transportar conceitos em relação à construção do conhecimento humano para a máquina? Se o sujeito é interpretado de uma forma, por que não interpretar os ambientes computacionais utilizando os mesmos recursos? A partir de então, foi selecionada a Lógica Operatória, da teoria de Jean Piaget, para analisar ferramentas computacionais. Logo, tanto o sujeito quanto a ferramenta computacional foram interpretados em termos da lógica operatória. Assim, as estruturas do sujeito tinham que ser compatíveis com as estruturas do objeto

(ferramentas computacionais) para estes poderem interagir de forma harmoniosa. Portanto, a ocorrência de um possível bloqueio por parte do sujeito, por exemplo, significaria que estas estruturas não estão de acordo, ou seja, que existem operações e regras na ferramenta computacional que ainda não foram desenvolvidas no sujeito. Portanto, este não estaria apto ainda para interagir com a mesma.

Estas questões serviram para direcionar este estudo, mas houve confronto com uma proposição: como analisar os sujeitos que trabalham com ferramentas computacionais cooperativas? Que tipo de sujeito é este? Como interpretá-lo?

Por essa razão, o estudo foi realizado em primeiro lugar, em relação ao sujeito individual e sua interação com as ferramentas computacionais de uso individual. Em um segundo momento, foram amadurecidas as idéias desenvolvidas e foi possível realizar a mesma investigação mas em cima do sujeito coletivo em interação com as ferramentas computacionais de uso cooperativo.

Também foi destacado o processo de construção dos modelos de interação do “sujeito individual e coletivo”, em relação às ferramentas computacionais. Portanto, foi preciso investigar os elementos envolvidos em uma interação entre um sujeito e uma ferramenta para, a partir disso, construir o modelo geral de interação. A primeira hipótese levantada foi: por que não procurar subsídios na semiótica? Como as ferramentas computacionais podem servir para a representação de “algo”, mas em um espaço computacional, iniciaram os estudos transportando os conceitos da semiótica para este outro plano. Portanto, a partir desta área foram originados os elementos necessários para a construção do modelo de interação que foi utilizado para a realização da análise operatória de ferramentas computacionais de uso individual e coletivo. São eles: o sujeito, a ferramenta computacional, a representação computacional e os valores, objetos e a linguagem de estudo.

Cabe destacar que, como trabalhos futuros a serem complementados com este estudo, pode ser enfatizada a construção de um método de análise em relação à leitura das ferramentas cooperativas, em termos operatórios. Nesta pesquisa foi desenvolvida uma “reconstrução” do método piagetiano de análise lógico-operatória de experiências em relação ao sujeito individual. A necessidade de analisar as ferramentas computacionais de uso individual, nos levou a uma definição e delimitação mais clara dos diferentes aspectos utilizados na análise operatória apresentada nas obras piagetianas. Assim, construiu-se este método, com o fim de apreender os elementos mais importantes envolvidos na análise das experiências, acompanhando o processo de desenvolvimento das operações lógicas/infralógicas do sujeito individual. Por essa razão, reconstruiu-se um método de análise de experiências que estava implícito na análise que Piaget realizou, facilitando a compreensão do método através da sistematização dos aspectos que direcionavam o mesmo. Somente a partir deste processo é que foi possível caracterizar o “objeto” operatoriamente, da mesma forma que foi realizado com o sujeito. Logo, este trabalho apresentou algumas idéias na proposta realizada que devem ser amadurecidas para a construção do método de análise lógico-operatória em relação ao sujeito coletivo.

Em síntese, no nível individual, foram analisados o aplicativo Paintbrush do Windows 3.1, o editor de textos Word 6.0, a linguagem de programação LOGO e o ambiente de programação WinLogo. Dessa forma, foram interpretados em termos lógicos e infralógicos, tanto o sujeito quanto as ferramentas computacionais de interação em questão.

Utilizando como base os estudos realizados em relação ao sujeito individual e os conceitos da teoria piagetiana à nível interindividual, caracterizaram-se os instrumentos computacionais coletivos de interação. Portanto, foram analisados operatoriamente as

ferramentas Microsoft Netmeeting 2.0 Beta 4, Microsoft Chat 2.0 Beta 1 e o ambiente de desenvolvimento cooperativo de programação ENVY/400.

Como contribuições específicas desta tese de doutorado, podem ser destacadas as seguintes:

- a aplicação, no modelo de interação, do sujeito usuário e do sujeito programador;
- os elementos envolvidos no modelo de interação foram padronizados de acordo com a notação gráfica usual para as redes de Petri para facilitar a interpretação do mesmo;
- a classificação das operações de controle direto/indireto do sistema e manipulação direta/indireta sobre a representação;
- aplicação da classificação anterior em termos das operações lógicas e infralógicas;
- a construção do modelo do sujeito individual e coletivo, em termos de estruturas afetiva, cognitiva e simbólica, as quais são compostas pelos valores, objetos e a linguagem, respectivamente, integrado aos elementos que fazem parte das ferramentas computacionais de uso individual e coletivo, compondo um só modelo, o qual foi analisado de forma operatória.

Portanto, foi desenvolvido um instrumento que poderá servir, visto sob a ótica do sujeito, como uma contribuição para os educadores e pesquisadores em geral, já que estes poderão utilizá-lo como guia para auxiliar no desenvolvimento lógico-operatório de sujeitos envolvidos em experiências através do uso de ferramentas computacionais de uso individual e cooperativo.

Por outro lado, isto é, visto do ponto de vista do objeto, este instrumento poderá ser utilizado para avaliar softwares e groupwares, de acordo com as operações lógicas e infralógicas. Além disso, o projeto e desenvolvimento de ferramentas computacionais poderá ser baseado na lógica operatória piagetiana.

Bibliografia

- [ABB 87] ABBAGNANO, Nicola. **Dicionário de filosofia**. 2. ed. São Paulo: Mestre Jou, 1987. 980 p.
- [BAN 91] BANNON, Liam. CSCW: Four Characters in Search of a Context. In: BOWERS, J. M.; BENFORD, S. D. **Studies in Computer Supported Cooperative Work**. [S.l.]: Elsevier Science Publishers B. V, 1991.
- [BAR 77] BARDIN, L. **Análise do Conteúdo**. São Paulo: Livraria Martins Fontes, 1977.
- [BEH 95a] BEHAR, Patricia A.; CERON, Maria T.; COSTA, Antonio C. R Uma Reconstrução do Método Piagetiano de Análise Lógico-Operatória de Experiências. In: CONGRESSO INTERNACIONAL LOGO, 7., 1995, Porto Alegre, RS. **Anais...** Porto Alegre: Instituto de Informática da UFRGS, 1995.
- [BEH 95b] BEHAR, Patricia A.; COSTA, Antônio Carlos da Rocha. Estudo de tese em Computação Cooperativa para a Construção Coletiva de Conhecimentos. In: SIMPÓSIO BRASILEIRO DE INFORMÁTICA NA EDUCAÇÃO, 6., 1995, Florianópolis, SC. **Anais...** Florianópolis: SBC, 1995.
- [BEH 96a] BEHAR, Patricia A.; COSTA, A.C.R. Logical-Operatory Analysis of the Knowledge Construction Process in a Cooperative Computational Environment. In: INTERNATIONAL CONFERENCE ON INFORMATION SYSTEMS ANALYSIS AND SHYNTESIS, ISAS, 1996, Orlando, Florida. **Proceedings...** Orlando: International Institute of Informatics and Systemics, 1996.
- [BEH 96b] BEHAR, Patricia Alejandra. **Análise Operatória de Ferramentas Computacionais de Manipulação de Representações**: exame de qualificação. Porto Alegre: CPGCC da UFRGS, 1996. 101 f.
- [BEH 96c] BEHAR, Patricia A.; COSTA, A.C.R. Computação cooperativa no processo de construção coletiva de conhecimentos. In: CONGRESO IBEROAMERICANO DE INFORMÁTICA NA EDUCAÇÃO, 3., 1996, Barranquilla, Colombia. **Anais...** Colombia: RIBIE, 1996.
- [BEH 96d] BEHAR, Patricia A; COSTA, A.C.R. Sobre las operaciones infralógicas de sujetos en interacción con un editor gráfico. In: CONGRESO INTERNACIONAL DE INFORMÁTICA EDUCATIVA, 1996, Buenos Aires, Argentina. **Memórias...** Buenos Aires: Funda Austral, 1996.
- [BEH 96e] BEHAR, Patricia A.; COSTA, A.C.R. Análise operatória do processo de interação Sujeito programador X Ambiente Winlogo. In: SIMPÓSIO BRASILEIRO DE INFORMÁTICA NA EDUCAÇÃO, 7., 1996, Belo Horizonte, Minas Gerais. **Anais...** Belo Horizonte: UFMG, 1996.

- [BEH 96f] BEHAR, Patricia A.; CERON, M.T; COSTA, A.C.R. **Análise lógico-operatória de experiências: aplicação de uma reconstrução do método piagetiano.** Porto Alegre: CPGCC da UFRGS, 1996.
- [BEH 97a] BEHAR, Patricia A.; COSTA, A.C.R. Interaction models for the operatory analysis of cooperative computational environments. In: **INFORMATION TECHNOLOGY FOR COMPETITIVENESS EXPERIENCES AND DEMANDS FOR EDUCATION AND TRAINING**, 1997, Florianópolis, Santa Catarina. **Proceedings...** Florianópolis: CTAI, 1997.
- [BEH 97b] BEHAR, Patricia A.; COSTA, A.C.R. Base models for organizing the logical operatory analysis of cooperative computational environments. In: **WORLD CONFERENCE ON EDUCATIONAL MULTIMEDIA AND HYPERMEDIA AND WORLD CONFERENCES ON EDUCATIONAL TELECOMMUNICATIONS, ED-MEDIA & ED-TELECOM 97**, 1997, Calgary, Canadá. **Proceedings...** Calgary: AACE, 1997.
- [BEH 97c] BEHAR, Patricia A.; COSTA, A.C.R. Concepts for the logical-operatory modeling of cooperative computational environments. In: **THE VIRTUAL CAMPUS: TRENDS FOR HIGHER EDUCATION AND TRAINING, IFIP 3.3 AND 3.6 JOINT WORKING CONFERENCE**, 1997, Madrid, Espanha. **Proceedings...** Madrid: Universidad Nacional de Educación a Distancia, 1997.
- [BEH 97d] BEHAR, Patricia A.; COSTA, A.C.R. Caracterização operatória do processo interativo de um sujeito coletivo com ferramentas computacionais cooperativas. In: **SIMPÓSIO BRASILEIRO DE INFORMÁTICA NA EDUCAÇÃO**, 8., 1997, São José dos Campos, São Paulo. **Anais...** São José dos Campos: ITA, 1997.
- [BEH 97e] BEHAR, Patricia A.; COSTA, A.C.R. Uso da lógica operatória como forma de análise de ferramentas computacionais cooperativas. In: **WORKSHOP SISTEMAS DE TUTORIA INTELIGENTE APLICADOS À EDUCAÇÃO**, 1997, São José dos Campos, São Paulo. **Anais...** São José dos Campos: ITA, 1997.
- [BEH 97f] BEHAR, Patricia A. Nuevas tecnologías de la informática y de la comunicación y sus aplicaciones en Psicopedagogía y en la Educación Especial. In: **CONGRESO INFORMÁTICO SAN JUAN**, 1997, San Juan, Argentina. **Memórias...** San Juan: Facultad de Ciencias Exactas de la Universidad Nacional de San Juan, 1997.
- [CAS 82] CASTORINA, José Antonio; PALAU, Gladys Dora. **Introducción a la lógica operatória de Piaget-Alcances y significado para la psicología genética.** Buenos Aires: Ediciones Paidós, 1982.
- [COS 83] COSTA, A.C.R. **Um referencial teórico para o estudo de máquinas orientadas a linguagens de alto nível.** Porto Alegre: Instituto de Informática, UFRGS, 1983.

- [COT 95] COSTA, A.R.F. **Interações Interindividuais em rede telemática**. Porto Alegre: Pós-Graduação em Psicologia do Desenvolvimento da UFRGS, 1995. Dissertação de mestrado.
- [DIE 96] DIETRICH, E. **Projeto de um sistema de suporte à autoria cooperativa de hiperdocumentos**. Porto Alegre: CPGCC da UFRGS, 1996. Dissertação de mestrado.
- [DOL 93] DOLLE, J.M. **Para além de Freud e Piaget. Referenciais para novas perspectivas em psicologia**. Rio de Janeiro: Vozes, 1993
- [DUR 98] DURKHEIM, E. **Représentations individuelles et représentations collectives**. Gênova: Revista Revue de Métaphysique et de Morale, 1898.
- [ELL 93] ELLIS, C.A. et al. Groupware: some issues and experiences. In: BAECKER, Ronald. **Groupware and Computer-Supported Cooperative Work**, USA: Morgan Kaufmann Publishers, 1993.
- [FLO 88] FLORES, F et al. Computer systems and the design of organizational interaction. **ACM Transactions on Office Information Systems**, New York, v. 6, n. 2, 1988.
- [FRI 95] FRITZKE, Udo Júnior. **Projeto e Implementação de um suporte para aplicações cooperativas do tipo editor distribuído**. Florianópolis: Programa de Pós-Graduação em Engenharia Elétrica da Universidade Federal de Santa Catarina, 1995. Dissertação de mestrado.
- [GAL 90] GALEGHER et al. **Intellectual Teamwork: Social and Technological Foundation of Cooperative Work**. USA: Lawrence Erlbaum, 1990.
- [GEE 91] GREENBERG, Saul. Personalizable groupware: acomodating individual roles and group differences. In: EUROPEAN CONFERENCE OF COMPUTER SUPPORTED COOPERATIVE WORK, 1991, Amsterdam, Holanda. **Proceedings...** Amsterdam: Kluwer Press, 1991.
- [GRE 88] GREIF, Irene. **Computer-Supported Cooperative Work: a book of readings**. USA: Morgan Kaufmann Publishers, 1988.
- [IBM 93] IBM. **ENVY/400 Developer Guide & ENVY/400 User's Guide**, v. 2, release 2, modification 0. USA: IBM, 1993.
- [KIR 93] KIRSCH, T. et al. Communication Support for Cooperative Work. **Computer Communications**, Surrey, England v. 16, n. 9, Sept. 1993.
- [MAL 93] MALONE, Thomas; CROWSON, Kevin. What is Coordination Theory and How it help Design Cooperative Work Systems? In: BAECKER, Ronald. **Groupware and Computer-Supported Cooperative Work**. USA: Morgan Kaufmann Publishers, 1993.
- [MIC 97a] MICROSOFT. **Chat 2.0 Beta 1**. Disponível em <http://www.microsoft.com/chat> (jun. 1997).
- [MIC 97b] MICROSOFT, Netmeeting. **Netmeeting 2.0 Beta 4**. Disponível em <http://www.microsoft.com/netmeeting> (maio 1997).

- [PIA 58] PIAGET, J. **Psicologia da Inteligência**. Rio de Janeiro: Fundo de Cultura, 1958.
- [PIA 71] PIAGET, J. **A formação do símbolo na criança**. Rio de Janeiro: Zahar, 1971.
- [PIA 72] PIAGET, J. **Ensaio da lógica operatória**. São Paulo: Ed. da USP, 1972.
- [PIA 73] PIAGET, J. **Estudos Sociológicos**. Rio de Janeiro: Forense, 1973.
- [PIA 76a] PIAGET, J. **A equilibrção das estruturas cognitivas**. Rio de Janeiro: Zahar, 1976.
- [PIA 76b] PIAGET, J.; INHELDER, B. **Da lógica da criança à lógica do adolescente**. São Paulo: Pioneira, 1976.
- [PIA 82] PIAGET, J. **O nascimento da inteligência na criança**. Rio de Janeiro: Zahar, 1982.
- [PIA 83] PIAGET, J. **Psicologia da Inteligência**. Rio de Janeiro: Zahar, 1983.
- [PIA 86] PIAGET, J. **O possível e o necessário**. Porto Alegre: Artes Médicas, 1986.
- [PIA 93] PIAGET, J.; INHELDER, B. **A representação do espaço na criança**. Porto Alegre: Artes Médicas, 1993.
- [RAM 96] RAMOS, Edla M. F. **Análise ergonômica do sistema hiperNet buscando o aprendizado da cooperação e da autonomia**. Florianópolis, SC: Programa de Pós-Graduação em Engenharia de Produção da Universidade Federal de Santa Catarina, 1996. Tese de doutorado.
- [ROC 93] ROCHA, H. V. de. Representações Computacionais auxiliares ao entendimento de conceitos de programação. In: VALENTE, José Armando. **Computadores e Conhecimento: repensando a educação**. Campinas: Gráfica Central da UNICAMP, 1993.
- [TOR 95] TONAGLIA, J.C. **Multi-Editor Cooperativo para Aprendizagem**. Rio de Janeiro: COOPE da UFRJ, 1995. Dissertação de mestrado.
- [WEN 76] WENDT, Siegfried. Models and structures for microprogramming. In: SYMPOSIUM ON MICROARCHITECTURE, 2., 1976, Netherlands, Holland. **Proceedings...** Netherlands: North-Holland Publishing Company, 1976.
- [WIL 92] WINLOGO. **Ambiente integrado de programação e aprendizagem em linguagem LOGO**, versão 1.3 individual, Coimbra, Portugal. Coimbra: CNOTINFOR, 1992.
- [WIN 86] WINOGRAD, T.; FLORES, F. **Understanding computers and cognition: a new foundation for designing**. Norwood, NJ: Ablex, 1986.
- [WIN 87] WINOGRAD, T. A language/action perspective on the design of cooperative work. In: HUMAN COMPUTER INTERACTION, 1987, Amsterdam, Holland. **Proceedings...** North-Holland: Elsevier Science Publishers B.V., 1987.

[WOO 90] WOODHEAD, N. **Hypertext & Hipermedia:** Theory and Aplications.
England: Addison Welsley Publishing Company, 1990.



CURSO DE PÓS-GRADUAÇÃO EM CIÊNCIA DA COMPUTAÇÃO

Análise Operatória de Ferramentas Computacionais de Uso Individual e Cooperativo

por

Patrícia Alejandra Behar

Tese apresentada aos Senhores:

Prof. Dra. Edla M. F. Ramos (UFSC)

Prof. Dr. Paulo Sergio Marchelli (USM)

Prof. Dra. Liane Margarida Rockenbach Tarouco

Prof. Dra. Cleci Maraschin (Psicologia/UFRGS)

Vista e permitida a impressão.

Porto Alegre, 18/03/13.

Prof. Dr. Antônio Carlos da Rocha Costa,
Orientador.