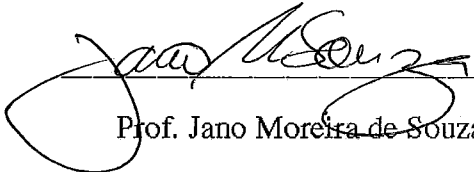


**FLECHA: UM EDITOR GRÁFICO COOPERATIVO PARA O MODELO DE
OBJETOS DO GEOTABA**

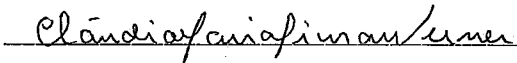
Claudia Susie Pina Camargo

TESE SUBMETIDA AO CORPO DOCENTE DA COORDENAÇÃO DOS PROGRAMAS DE PÓS-GRADUAÇÃO DE ENGENHARIA DA UNIVERSIDADE FEDERAL DO RIO DE JANEIRO COMO PARTE DOS REQUISITOS NECESSÁRIOS PARA A OBTENÇÃO DO GRAU DE MESTRE EM CIÊNCIAS EM ENGENHARIA DE SISTEMAS E COMPUTAÇÃO.

Aprovada por:


Prof. Jano Moreira de Souza, Ph.D.
(Presidente)


Prof. Pedro Manoel da Silveira, Ph.D.


Cláudia Maria Lima Werner, D.Sc.

RIO DE JANEIRO, RJ - BRASIL

DEZEMBRO DE 1992

CAMARGO, CLAUDIA SUSIE PINA

Flecha: Um Editor Gráfico Cooperativo para o Modelo de Objetos do
GEOTABA [Rio de Janeiro] 1992

XII, 81p 29,7 cm (COPPE/UFRJ, M.Sc., Engenharia de Sistemas e
Computação, 1992)

Tese - Universidade Federal do Rio de Janeiro, COPPE

1. Editor Cooperativo I. COPPE/UFRJ II. Título (série)

Ao Paulo e
a meus pais

AGRADECIMENTOS

Ao Paulo Fernando, pela sua dedicação, apoio e paciência de rever o texto.

Aos meus pais, por acreditarem em mim.

À tia Elodi, pela força nos momentos mais difíceis.

Ao professor Jano Moreira de Souza, pelo tema dessa tese e por me ajudar a tornar esse trabalho possível.

À Claudia Maria Lima Werner, pelas sugestões feitas durante a elaboração deste trabalho e, por dar a honra de participar da minha banca de tese.

Ao professor Pedro Manoel da Silveira, por dar a honra de participar da minha banca de tese.

Às secretárias, Cláudia e Ana Paula, pela paciência e pelo carinho.

Aos colegas e funcionários do Programa: Luís Carlos M. Monte, Arnaldo Belchior, Jugurta Lisboa Filho, Guilherme Travassos, Eliseu M. Chaves e Flavio Araujo de Mattos.

À CAPES pelo apoio financeiro.

Resumo da Tese apresentada à COPPE/UFRJ como parte dos requisitos necessários para obtenção do grau de Mestre em Ciências (M. Sc.)

**Flecha: Um Editor Gráfico Cooperativo
para o Modelo de Objetos do GEOTABA**

Claudia Susie Pina Camargo

Dezembro 1992

Orientador: Jano Moreira de Souza

Programa: Engenharia de Sistemas e Computação

O Flecha é uma das ferramentas cooperativas da Estação de Desenvolvimento de *Software* - TABA. Junto com essas ferramentas cooperativas, ele pretende fornecer um alto nível de comunicação entre os desenvolvedores e os responsáveis pelo desenvolvimento, nas atividades assíncronas e síncronas.

O Flecha é um editor gráfico para a representação gráfica do Modelo de Objetos (MOO) do GEOTABA - Sistema de Gerência de Objetos da Estação TABA. Ele auxilia a edição gráfica desta representação de objetos, dando um enfoque cooperativo. Ele permite a utilização de anotações, a troca de mensagens, facilidades para a interação entre colaboradores e para o compartilhamento de informação entre eles. Este editor, ainda, dá suporte à

colaboração simultânea, permitindo que diferentes pessoas trabalhem em diferentes partes do diagrama no mesmo instante, utilizando as versões mais atuais de cada parte, e tomando o conhecimento instantâneo das alterações realizadas pelos outros usuários.

Abstract of Thesis presented to COPPE/UFRJ as partial fulfillment of the requirements for the degree of Master of Science (M. Sc.).

**Flecha: A Cooperative Graphic Editor
for the Object Data Model of GEOTABA**

Claudia Susie Pina Camargo

December 1992

Thesis Supervisor: Jano Moreira de Souza

Department: Systems and Computer Engineering

Flecha is one of the cooperative tools of the TABA - Software Development Workstation. Together with these cooperative tools, it intends to provide a high level of communication and synergy to the developers and to the application development manager, in both, asynchronous or synchronous activities.

Flecha is a graphic editor for the graphic representation of the Object Data Model (MOO) of GEOTABA -Object Management System of the TABA Station. Flecha is a graphic editor for this objects' representation, emphasizing cooperativeness. It allows the use of annotations, message exchanging, facilities for the interaction among collaborators and information sharing. The editor also supports simultaneous collaboration, allowing different people to work in different parts of a diagram at the same time, using the most up to date

versions of each part, and acquiring instantaneous knowledge of changes introduced by other users.

ÍNDICE

I.	INTRODUÇÃO	1
II.	TRABALHO COOPERATIVO	4
II.1.	<i>Groupware</i>	6
II.1.1.	Correio Eletrônico	7
II.1.2.	Editores Cooperativos de Documentos	8
II.1.3.	Reuniões Face-a-face	8
II.1.4.	Sistemas de Conferências	10
II.1.5.	Sistemas de Calendário	11
II.2.	Obstáculos	12
II.2.1.	Benefícios mal distribuídos	13
II.2.2.	Falta de flexibilidade no tratamento de erros	13
II.2.3.	Falha na intuição do projetista	14
II.2.4.	Dificuldades na fase de avaliação	14
II.2.5.	Adaptação à organização	15
II.3.	A Importância da Comunicação, Colaboração e Coordenação	16
III.	EDITORES COOPERATIVOS E FERRAMENTAS DE SUPORTE À EDIÇÃO COOPERATIVA	19
III.1.	DistEdit	19
III.1.1.	Apresentação	19
III.1.2.	Características	20
III.1.3.	Funcionamento	21
III.1.4.	<i>Software</i> de Suporte	22
III.1.5.	Trabalhos Futuros	22
III.2.	Prep	23
III.2.1.	Apresentação	23

III.2.2.	Características	23
III.2.3.	Funcionamento	24
III.3.	InterNote	25
III.3.1.	Apresentação	25
III.3.2.	Características	25
III.3.3.	Funcionamento	27
III.3.4.	<i>Software</i> de Suporte	28
III.3.5.	Trabalhos Futuros	29
III.4.	Quilt	29
III.4.1.	Apresentação	29
III.4.2.	Características	30
III.4.3.	Funcionamento	31
III.4.4.	<i>Software</i> de Suporte	31
III.5.	Shared Books	32
III.5.1.	Apresentação	32
III.5.2.	Características	32
III.5.3.	Funcionamento	33
III.5.4.	<i>Software</i> de Suporte	34
III.5.5.	Trabalhos Futuros	34
III.6.	Video Draw	34
III.6.1.	Apresentação	34
III.6.2.	Características	35
III.6.3.	Funcionamento	36
III.7.	VMACS	36
III.7.1.	Apresentação	36
III.7.2.	Características	36
IV.	FLECHA: EDITOR GRÁFICO COOPERATIVO PARA O MODELO DE OBJETOS DO GEOTABA	38
IV.1.	Apresentação	38
IV.2.	Representação Gráfica para o Modelo de Objetos do GEOTABA	38
IV.2.1.	Modelo de Objetos	39
IV.2.2.	Diagrama de Objetos	41

IV.3.	Requisitos	45
IV.3.1.	Editor para a representação gráfica do MOO	45
IV.3.2.	Multi-usuário cooperativo	45
IV.3.3.	Diferentes tipos de autorizações	45
IV.3.4.	Comentários e Anotações	47
IV.3.5.	Atualização imediata de objetos	
	bloqueados em outros contextos	47
IV.3.6.	Integração de contextos	48
IV.3.7.	Histórico de Contextos e Diagramas	49
IV.3.8.	Comunicação com o PROTOGEO	49
V.	IMPLEMENTAÇÃO DO EDITOR	50
V.1.	Funcionamento	51
V.1.1.	Histórico	51
V.1.2.	Tela Principal	52
	V.1.2.1. Comandos	53
	V.1.2.2. Funções dos Ícones	53
	V.1.2.3. Funções do Cardápio	
	de Comandos	58
V.1.3.	Modo Leitor, Gerente	
	e Modelador	62
V.1.4.	Envio de Mensagens	64
V.1.5.	Atualização Imediata	64
V.1.6.	Divisão de Contextos	65
V.1.7.	Integração de Contextos	66
V.2.	Arquitetura do Flecha	67
V.3.	Esquema de Classes	68
	V.3.1. Classe de Aplicação	71
	V.3.2. Classe de Janela	71
	V.3.3. Classe de Diálogo	71
	V.3.4. Classe de Figura	72
	V.3.5. Classe de Registro	72
V.4.	Futuras Implementações	72
VI.	CONCLUSÕES	75
	REFERÊNCIAS BIBLIOGRÁFICAS	77

ÍNDICE DE FIGURAS

IV.1.	Diagrama de Objetos para a Companhia	41
IV.2.	Ícones e Caixas para Classes e Objetos	42
V.1.	Tela Principal	52
V.2.	Variável de Instância	55
V.3.	Relacionamento	56
V.4.	Arquitetura do Flecha	67
V.5.	Esquema de Classes do Flecha	69

I. INTRODUÇÃO

A modelagem de sistemas complexos é, essencialmente, uma tarefa que envolve um grupo de pessoas durante um certo período de tempo. Estas pessoas trabalham com um objetivo comum, porém com visões muito distintas a respeito do que está sendo criado.

Nesta aparente diversidade de interesses, é necessária muita harmonia e comunicação entre as partes envolvidas para atingir o objetivo final. Para situações deste tipo tem sido proposto o enfoque de Trabalho Cooperativo Suportado por Computador [Grei89].

O Trabalho Cooperativo Suportado por Computador se baseia na experiência e na colaboração de muitos especialistas, incluindo cientistas sociais e da computação, estudando como os grupos trabalham e como a tecnologia (especialmente, os computadores) pode ajudá-los a trabalhar [Elli91].

Aliando as idéias de trabalho cooperativo com o fornecimento de uma facilidade gráfica, foi elaborado o editor Flecha, uma ferramenta de software para o auxílio à confecção de diagramas orientados a objetos, mais especificamente, o Diagrama de Objetos, proposto por [Mont91], que é uma representação gráfica do Modelo de Objetos (MOO) [Mont90] do sistema de gerência de objetos GEOTABA. Sua função principal pode ser comparada à

função da representação gráfica do Diagrama de Entidades-Relacionamentos (DER) [Chen76]. Ambos representam modelagens semânticas de informação. O Diagrama de Objetos é diretamente mapeável no MOO, correspondendo, exatamente, a um esquema de classes e objetos implementado no GEOTABA.

O GEOTABA [Matt89] é um sistema de gerência de objetos que vem sendo desenvolvido no Programa de Engenharia de Sistemas e Computação da COPPE/UFRJ. Este sistema gerencia informações de acordo com o MOO, contidas no ambiente de desenvolvimento de *software* TABA [Roch91].

Esta tese tem como objetivo pesquisar aspectos sobre as características e o funcionamento de editores cooperativos e ferramentas de suporte à edição cooperativa, com a finalidade de propor e desenvolver um protótipo de um editor gráfico cooperativo, num ambiente multi-usuário, dando ênfase ao compartilhamento de informações entre co-autores (co-modeladores) e à comunicação através de comentários e mensagens diretas por pessoas autorizadas. Essa interação é feita em tempo real, ou seja, diferentes partes desse diagrama podem estar sendo modeladas, enquanto mensagens ou comentários estão sendo feitos. Existe o caso de duas dessas partes possuírem objetos comuns. Dessa forma, o autor que entrar primeiro em sessão terá livre acesso a ele, enquanto os outros autores terão seu acesso a esta parte bloqueado, podendo, no entanto, acompanhar as alterações realizadas pelo primeiro.

O capítulo 1 apresentou a Introdução da tese, descrevendo a motivação, uma pequena descrição, o objetivo e a organização da mesma. O capítulo 2 é dedicado à análise do trabalho cooperativo, estudando suas características, falhas e sucessos, sendo incluídos exemplos. O capítulo 3 examina editores

cooperativos e ferramentas de suporte à edição cooperativa. O capítulo 4 apresenta o Flecha, editor gráfico cooperativo para o Modelo de Objetos do GEOTABA. O capítulo 5 aborda sua implementação e sugere novas características para futuras implementações. Finalmente, o capítulo 6 apresenta as conclusões.

II. TRABALHO COOPERATIVO

A sociedade adquire a maioria de suas características através da maneira na qual as pessoas interagem entre si. Mesmo depois da presença do computador ser, atualmente, comum em muitas casas e escritórios, a interação das pessoas, uma com as outras, é quase a mesma que a dez anos atrás. Assim, como as tecnologias dos computadores e outras formas de comunicação eletrônica continuam a crescer, as pessoas, também, continuarão a interagir de novas e diferentes maneiras.

Foi assim que, em meados de 1980, pesquisadores pertencentes a diferentes comunidades intelectuais, como Ciência da Computação, Psicologia, Inteligência Artificial, Sociologia, Sistemas de Automação de Escritório, etc., começaram a reconhecer seus interesses comuns sobre o papel do computador no trabalho feito por grupos de pessoas. Esse novo campo de estudo é chamado de Trabalho Cooperativo Suportado por Computador (TCSC).

Através de suas experiências iniciais com este tipo de tecnologia, [Grei87a] constatou que as aplicações em TCSC mostraram muitas de suas propriedades particulares como, por exemplo, o fornecimento de um suporte à interação usuário-usuário. Desta forma, contribuiu para o aparecimento de muitas perguntas a respeito de suas idéias sobre o que são os sistemas de computadores e como as pessoas se relacionam com ele.

"Como as pessoas podem planejar um trabalho mais lucrativo unidas a outras pessoas ?"

"Que tipo de software pode ser desenvolvido ?"

"Como o trabalho em grupo pode ser definido para atingir o potencial esperado pelas pessoas e pela tecnologia ?"

"Qual o tamanho dos grupos que utilizam computadores para a colaboração ?"

[Grei87a] conclui que questões como estas serão respondidas, apenas, através de pesquisas específicas nesta área.

O TCSC estuda o relacionamento entre os sistemas de computador e os sistemas sociais. Uma parte dessa pesquisa se dedica à construção de ferramentas que ajudam as pessoas a trabalharem juntas. A outra parte está interessada nos meios pelos quais as pessoas utilizam ferramentas baseadas em computador, para trabalharem juntas, e como estas ferramentas e o relacionamento dentro do trabalho afetam cada uma delas. Ou seja, uma parte desta pesquisa dá ênfase à construção de ferramentas como: correio eletrônico, sistemas de conferências, editores cooperativos, etc., enquanto a outra parte estuda as ferramentas com ênfase nos aspectos que influenciam a produtividade do trabalho em grupo [Olso87].

II.1. Groupware

Groupware é um termo freqüentemente usado como um sinônimo da tecnologia do TCSC.

[Elli91] define *groupware* da seguinte forma:

"São sistemas baseados em computador que suportam grupos de pessoas envolvidas numa mesma tarefa, ou objetivo, e que fornecem uma *interface* para um ambiente compartilhado."

Dada a variedade destes sistemas, os fatores de tempo e espaço podem ser aplicados de forma a representar quatro grandes categorias. Na primeira categoria, representando o mesmo tempo e o mesmo lugar, estão as reuniões face-a-face; na segunda, representando o mesmo tempo em lugares diferentes, os sistemas de conferências; na terceira, tempos diferentes e mesmo lugar, o editor gráfico assíncrono; e, finalmente, na quarta, tempos e lugares diferentes, os sistemas de correios eletrônicos.

O contraste de um sistema de *groupware* para outro depende do enfoque dado às pessoas e do seu relacionamento dentro do grupo sob a influência do computador, dentro do trabalho de cada uma delas.

Uma outra maneira simples de entender *groupware* é definindo-o como uma classe de produtos, como uma caixa que contém ferramentas para diversas tarefas [Oppe88]. Como as ferramentas dentro da caixa, *groupware* corresponde

a diferentes trabalhos ou missões dentro do trabalho em grupo. As sub-seções que se seguem apresentam alguns exemplos dessas ferramentas.

II.1.1. Correio Eletrônico

O correio eletrônico é um meio de comunicação projetado para a transmissão de mensagens de pessoa a pessoa. É adaptado para algumas situações em grupo, mas não é voltado, realmente, para as necessidades do trabalho em grupo.

A principal característica do correio eletrônico, segundo [Feld87], é o baixo custo investido para descobrir pessoas com interesses comuns. Eles são assíncronos e permitem enviar mensagens para várias pessoas sem nenhum esforço a mais do que levaria para enviar para apenas uma pessoa.

A natureza assíncrona de sua comunicação torna desnecessário às pessoas se comunicarem simultaneamente. Neste caso, o custo e as restrições de horários são muito menores se comparados com a comunicação síncrona, que ocorre nas interações face-a-face ou por telefone.

Uma outra característica importante é a sua habilidade de definir grupos de pessoas e enviar mensagens, simultaneamente, para qualquer pessoa dentro do grupo. As pessoas identificam tópicos de interesses. Faz-se uma lista de nomes e endereços dos interessados e informações afins são compartilhadas.

É claro que a mensagem, também, poderá ser enviada na sua forma manuscrita. A diferença, no entanto, é que a mensagem manuscrita deve ser

copiada, endereçada e despachada para o destinatário, ou seja, um processo mais dispendioso.

II.1.2. Editores Cooperativos de Documentos

Os editores cooperativos de documentos são ferramentas colaborativas que permitem que um mesmo documento seja editado por um grupo de pessoas num ambiente distribuído. Alguns desses editores, como o ForComment [Oppe88], permitem a participação de revisores, que podem adicionar comentários sem alterar a versão original. Existem, também, os editores que permitem grupos de pessoas alterarem, no mesmo instante, o mesmo documento, que é o caso do Shared Books [Lewi88]. No capítulo 3 mais detalhes sobre esses editores são apresentados.

II.1.3. Reuniões Face-a-face

As reuniões têm um papel muito importante para a maioria das atividades humanas. As reuniões exploram idéias, resolvem desacordos e utilizam bastante o trabalho em grupo até alcançar seus objetivos.

Organizar e obter uma reunião eficiente pode ser uma tarefa muito difícil e que gasta muito tempo. Mesmo depois de uma extensa preparação, ainda não se pode ter garantia de seu sucesso ou que, realmente, seus objetivos serão atingidos.

Numa reunião convencional é comum a utilização do quadro-negro. Este instrumento permite o uso de figuras e textos que facilitam a exposição de

idéias e o andamento da reunião. No entanto, o espaço de utilização do quadro é limitado; alguns itens têm que desaparecer para liberar espaço para novas informações. É inconveniente quando os itens são rearrumados, necessitando de serem apagados e redesenhados manualmente. A escrita a mão no quadro-negro pode ser ilegível e o armazenamento de informações, também, é dificultado, pois de uma reunião para outra o quadro pode ser apagado.

Muitas das funções que são mal feitas ou impossíveis de serem feitas com o quadro-negro são implementadas, facilmente, através dos computadores. Os sistemas de janelas e os desenhadores, por exemplo, fornecem flexibilidade para a organização de textos e figuras. Os sistemas de arquivos tornam possível recuperar informações geradas através de reuniões passadas; revisar antigos argumentos; mostrar o histórico desses argumentos; e resumir discussões. As *workstations* independentes permitem os participantes compartilharem visões, apontar objetos que estão sob discussão e trabalharem em diferentes aspectos de um problema, simultaneamente. Como resultado, seus participantes não se sentem apenas como um membro passivo de um grupo, mas também como colaboradores.

Foi para explorar essas idéias que a Xerox Parc desenvolveu a ferramenta Colab [Stef87a] [Stef87b]. Na Colab, os computadores suportam processos colaborativos em reuniões face-a-face. Esta ferramenta foi projetada para pequenos grupos de trabalho conectados por uma rede local. Seu projeto foi conduzido através das experiências com reuniões face-a-face convencionais, com o objetivo principal de tornar este tipo de reunião mais eficiente e de fornecer uma oportunidade de estudo sobre o efeito das ferramentas de computadores sobre as reuniões.

II.1.4. Sistemas de Conferências

Os sistemas de conferências evitam algumas limitações pertencentes aos tipos de conferências atuais e acrescentam novas características ao processo de discussão em grupo. Quando um executivo deixa o seu escritório, para participar de uma reunião, deixa também para trás importantes informações que chegam na sua ausência. O objetivo do sistema de conferência é fazer com que as pessoas participem das reuniões sem saírem de seus escritórios. Assim, as informações que aí chegarem não serão perdidas e os participantes estarão mais acessíveis aos outros.

A utilização de computadores nas conferências faz com que novos serviços sejam utilizados como, por exemplo, programas que tomam notas, ou que guardam o estado de uma reunião para ser utilizado numa outra discussão. E os benefícios tornam-se ainda maiores ao fazer reuniões com participantes que estão distantes geograficamente.

[Ahuj88] descreve o sistema de conferências Rapport. O Rapport oferece suporte a conferências interativas, em tempo real e distribuídas geograficamente. O sistema permite a participação de dois ou mais usuários. O Rapport utiliza computadores conectados numa rede, que compartilham dados e vozes, e cria um ambiente de suporte a vários tipos diferentes de reuniões, que inclui conversas ao telefone e discussões entre colegas. O sistema permite, ainda, a interação de usuários a longas distâncias e a participação em várias conferências ao mesmo tempo.

II.1.5. Sistemas de Calendário

Os calendários eletrônicos são sistemas utilizados por várias pessoas para a marcação de reuniões. Cada calendário possui um conjunto de tarefas definidas e uma especificação de quais tarefas serão acessadas pelos diferentes usuários. O sistema já fornece um conjunto de tarefas pré-definidas, que são responsáveis pelo controle de acesso aos calendários. O próprio dono do calendário pode modificá-lo, definindo novas regras que refletem suas preferências ou um uso particular do seu calendário.

Existem vários tipos de calendários que representam agendas para uma só pessoa, para uma sala de conferências ou para eventos abertos como seminários. As *interfaces* para todos esses calendários são semelhantes, mas diferem quanto ao seu objetivo. Por exemplo, calendários para uma sala de conferência utilizam um acesso público que permite que qualquer um confirme sua presença de forma a não haver conflitos, se este for o caso. Com os calendários pessoais existe a opção de aceitar marcações em conflito.

Os usuários reais podem ser: uma secretária, que pode confirmar ou cancelar compromissos; uma pessoa qualquer, desejando uma consulta ou um encontro com o executivo (neste caso, a marcação pode ser rejeitada ou confirmada, tanto pelo próprio executivo quanto pela secretária); o público, que pode, apenas, ver se os blocos de tempo estão ocupados, ou não; e os membros da equipe do executivo, que têm o acesso a detalhes dos compromissos.

Feito em Smalltalk e operando em estações SUN, o Visual Scheduler [Bear90] é um sistema de calendário pessoal e um sistema de marcação de

reuniões. Ele requer um sistema que conecte todos os participantes, onde um usuário pode editar seu próprio calendário e ver os calendários dos outros. As marcações de reuniões são feitas de acordo com a composição dos calendários de todos os outros participantes.

II.2. Obstáculos

O desenvolvimento de sistemas de *groupware* nem sempre são um sucesso. No entanto, esta falha não está ligada diretamente ao *software* ou às próprias firmas que os utilizam. Sua causa principal é atribuída às mudanças feitas na interação entre as pessoas no seu ambiente de trabalho. Os obstáculos encontrados, segundo [Grud88], estão na falta de conhecimento sobre o aspecto de interação de grupo.

Apesar da grande dificuldade encontrada nessa área, os obstáculos podem ser superados sem afetar os principais objetivos do *groupware*. Para isso, devem ser feitos: grandes investimentos em recursos adequados para solucionar os problemas encontrados; desenvolvimento de pesquisas e metodologias apropriadas; os focos onde os problemas acontecem devem ser encontrados e estudados, bem como os focos onde as aplicações atuam com sucesso; e, finalmente, a preparação adequada dos usuários para a introdução de novas aplicações [Grud89].

A seguir, são apresentadas as possíveis falhas que levam ao insucesso dos sistemas de *groupware*.

II.2.1. Benefícios mal distribuídos

Os sistemas de *groupware* suportam pessoas que trabalham em funções diferentes e que pertencem a hierarquias diferentes. No entanto, estes sistemas oferecem mais benefícios a uns do que a outros, ou seja, aqueles que têm um trabalho adicional na manipulação da ferramenta, mas que não são diretamente beneficiados. Esta aplicação possui grandes possibilidades de falhar. É o caso, por exemplo, dos calendários eletrônicos. Eles possuem a função de comunicação entre executivos que possuem secretárias, que os mantêm atualizados. O calendário eletrônico pode ser usado, simultaneamente, pela secretária, na marcação de horários; pelo próprio executivo, na revisão; ou por outros elementos do grupo, no seu planejamento. [Grud89] ressalta que, para uma perfeita utilização, o calendário deve ser mantido sempre atualizado, o que é uma tarefa da secretária.

Enquanto o executivo vê apenas vantagens no calendário eletrônico, outros, responsáveis pela atualização dos mesmos, não pensam da mesma forma. Portanto, a solução seria o aumento do benefício coletivo.

II.2.2. Falta de flexibilidade no tratamento de erros

As aplicações podem falhar se não permitirem exceções no tratamento de erros, aceitando características de improvisação nas atividades em grupo.

Os sistemas foram desenvolvidos para dar suporte às atividades em grupo ou aos procedimentos supostos como capazes de acontecer. Portanto, a

flexibilidade no tratamento dos erros não previstos facilita sua aceitação e conseqüente sucesso.

II.2.3. Falha na intuição do projetista

O processo de projeto algumas vezes fracassa por causa de falhas nas intuições para aplicações multi-usuárias. O gerente é o responsável pelas tomadas de decisões em uma empresa e, na posição de aceitar os recursos necessários para os projetos de desenvolvimento da aplicação, confia na sua intuição. A experiência do gerente, considerando uma aplicação de *groupware*, é baseada, geralmente, nas aplicações mono-usuárias. Neste tipo de aplicação, um gerente com uma boa intuição pode perceber, rapidamente, a experiência do usuário com aquela ferramenta. Mas uma aplicação de *groupware* envolve a participação simultânea de pessoas de diferentes níveis de conhecimento, preferências e trabalhos, todas as quais têm que participar de uma forma ou de outra no sucesso da aplicação. A intuição do gerente possui grandes chances de falhar ao analisar todas as situações e todas as reações dos usuários.

II.2.4. Dificuldades na fase de avaliação

As fases de análise, projeto e avaliação de um sistema não são tarefas fáceis e tornam-se mais difíceis quando direcionadas para aplicações multi-usuárias.

A avaliação de um sistema de *groupware*, dentro de um campo qualquer, é muito complexa, devido ao número de pessoas que o analisam, a variedade na composição de cada grupo e o conjunto de fatores dentro do ambiente de

trabalho que influenciam na sua aceitação, tal como o treinamento de usuário, etc..

Todas essas dificuldades, encontradas na avaliação de um sistema de *groupware*, crescem, dramaticamente, dado à importância ao fornecimento de novas características e interfaces que variam de acordo com: o trabalho de cada usuário, níveis de conhecimento e suas preferências, como já citado anteriormente.

Como os demais problemas, a avaliação pode tornar-se menos difícil se a aplicação utilizar grupos menores e mais homogêneos, ao invés do envolvimento de indivíduos distribuídos por toda a organização.

II.2.5. Adaptação à organização

Uma organização pode se adaptar a um sistema de grande porte, mas é a aplicação de *groupware* que deve se adaptar à organização. Um sistema grande, por exemplo, pode suportar vários tipos de editores, mas todos os autores trabalhando juntos devem usar a mesma aplicação de co-autoria.

De fato, para o *groupware* obter sucesso, deve-se começar por entender melhor como os grupos e as organizações funcionam; ser mais sistemático na avaliação dos efeitos da introdução da tecnologia nos grupos; saber que as intuições passadas não funcionam mais quando evoluem-se das aplicações mono-usuárias e dos sistemas de grande porte para o novo mundo de minicomputadores e microcomputadores ligados em rede, que dão suporte aos

grupos. Precisa-se de ciências melhores e melhores intuições para esta nova área, porque o fracasso é muito caro e consome muito tempo.

II.3. A Importância da Comunicação, Colaboração e Coordenação

A maioria dos sistemas de *software* dá suporte apenas à interação entre um usuário e o sistema. Na edição de um documento ou na consulta a um banco de dados, o usuário interage somente com o computador. Mesmo os sistemas projetados para aplicações multi-usuária como, por exemplo, os sistemas de informação de escritório, fornecem um suporte mínimo à interação usuário-usuário. Este tipo de interação é extremamente necessário e importante, já que a maioria das atividades das pessoas ocorrem em grupo. Para entender a ação da interação do grupo, deve-se estudar a comunicação, a colaboração e a coordenação.

Segundo [Good87], as pessoas são capazes de trabalhar juntas, com mais prazer e mais eficiência, e a comunicação e as tecnologias computacionais podem contribuir para esse propósito. A interação informal entre as pessoas que trabalham juntas, por exemplo, numa conversa ou discussão dentro do escritório, é um importante recurso de informação e significa um tipo de coesão de grupo dentro do trabalho cooperativo. Assim como, também, a redução das necessidades de sincronismo na comunicação entre as pessoas é um impacto favorável na adoção do TCSC por uma organização. Muitas mensagens passam a ser trocadas sem que haja contato verbal direto entre os comunicadores.

Assim, por exemplo, mediante o uso do correio eletrônico (eventualmente até com voz) podem ser eliminados numerosos telefonemas improdutivos que assolam a vida das organizações.

A experiência tem mostrado que o *groupware* facilita as comunicações entre pessoas de níveis hierárquicos diferentes, desinibindo os que precisam se comunicar com um executivo que está a vários escalões acima, mas que deve intervir no processo administrativo em curso. Além disso, reequilibra as possibilidades de participação dos envolvidos num sistema de conferência, por evitar que a exuberância verbal de alguns abafe a voz de outros [Plon89].

Portanto, é fácil concluir que a eficiência da comunicação e a habilidade de uma organização em manipular sua tecnologia de comunicação, apropriadamente, influem, diretamente, no sucesso ou fracasso dessa organização em atingir seus objetivos.

Tal qual a comunicação, a colaboração é um elemento fundamental para o trabalho em grupo. A colaboração é uma atividade que recebe a participação de várias pessoas. Ela demonstra a influência de cada uma dessas pessoas no processo e envolve a troca recíproca da informação, onde pessoas compartilham informações de alguma forma, ocasionando mudanças nos pensamentos e nas ações das pessoas envolvidas.

O projeto de ferramentas para o trabalho cooperativo pode ser feito baseando-se somente no bom senso e nas boas intuições. Contudo, a história da tecnologia está repleta de exemplos de muitas teorias que ajudam no desenvolvimento de tecnologia de qualidade. É o caso da coordenação que

estuda como as pessoas trabalham juntas e como elas trabalhariam com o auxílio de novas tecnologias de informação.

Em geral, as pessoas possuem um senso intuitivo do que a palavra "coordenação" significa. Pode-se notar como bem coordenada as ações de um grupo de pessoas parece ser, ao presenciar a uma conferência bem conduzida, ou ao assistir a um time de basquetebol vencendo. No entanto, por outro lado, quase sempre uma boa coordenação parece ser invisível e, às vezes, passa-se a percebê-la apenas quando está prestes a fracassar. Por essas e outras razões, o efeito da má coordenação é motivo de muita preocupação.

III. EDITORES COOPERATIVOS E FERRAMENTAS DE SUPORTE À EDIÇÃO COOPERATIVA

Este capítulo é destinado à apresentação e à descrição de alguns editores cooperativos e outras ferramentas que dão suporte à edição cooperativa encontradas na bibliografia. O capítulo é organizado, de forma que, cada ferramenta contenha até cinco sub-itens: a apresentação, que faz uma breve descrição da ferramenta; as características, que o qualificam como um editor cooperativo, ou como uma ferramenta de suporte à edição cooperativa; o funcionamento, que cita alguns detalhes da ferramenta em execução; o software de suporte, que pode ser um sistema já existente, que é utilizado para acompanhar o seu funcionamento; e os trabalhos futuros, que mostram as perspectivas de uma nova versão.

III.1. DistEdit [Knis90]

III.1.1. Apresentação

O DistEdit, diferente de outros sistemas de edição cooperativa, não é um editor, mas uma ferramenta que pode ser usada para adaptar editores pessoais comuns à tarefa de edição em grupo.

O DistEdit fornece um conjunto de primitivas que são usadas para adicionar o suporte colaborativo aos editores. Estas primitivas são genéricas o bastante para dar suporte a editores com diferentes *interfaces*.

O processo funciona da seguinte maneira: editores familiares aos usuários são modificados para utilizarem o DistEdit. Os editores, então, permitirão os usuários a fazer alterações, enquanto outros usuários observam essas alterações, concorrentemente.

III.1.2. Características

As características da ferramenta DistEdit são as seguintes:

- i) O DistEdit dá suporte à colaboração multi-usuária, ou seja, os editores que utilizam a ferramenta DistEdit, permitem aos usuários editar arquivos de texto, colaborativamente, sem estarem próximos fisicamente;
- ii) Os protocolos de comunicação utilizados pelo DistEdit dão uma visão consistente dos arquivos para todos os usuários, num tempo razoável, tornando a edição em grupo eficiente;
- iii) É possível usar editores diferentes numa mesma sessão em grupo. As pessoas, geralmente, possuem seus próprios editores e seria desagradável forçá-las a usarem editores diferentes, quando forem participar de uma sessão em grupo;

- iv) Em uma sessão em grupo, o usuário continua a trabalhar sem perceber falhas em outras máquinas e movimentos de pessoas entrando e saindo de sessão;
- v) O editor, que participa do grupo de edição do DistEdit, não precisa ter o conhecimento de sistemas distribuídos como, por exemplo, protocolo de comunicação;
- vi) Qualquer número de usuários pode participar de uma sessão em grupo.

III.1.3. Funcionamento

O DistEdit é uma ferramenta que adapta editores diferentes numa sessão de edição cooperativa. Cada editor sofre algumas modificações, requeridas pelo seu código para serem utilizados numa mesma sessão de edição em grupo.

O DistEdit possui um servidor responsável em manter o estado do *buffer* do editor, garantindo a consistência entre as visões dos usuários, já que todos os comandos de edição passam pelo servidor compartilhado.

O DistEdit permite duas classes de usuários: mestre e observador. O mestre, que é o primeiro usuário a chamar o arquivo para a edição, é o único que tem a permissão de alteração do texto. Os outros usuários que entrarem em sessão serão os observadores, sem a permissão de alteração. Eles verão todas as alterações e o movimento do cursor feito pelo mestre. Eles podem, no entanto, salvar arquivos, mudar o tamanho da janela, manipular qualquer

comando do editor que não altere o texto, liberar seu cursor da dependência do cursor do mestre e movê-lo para qualquer lugar do texto. Qualquer mudança no texto feita pelo mestre é imediatamente atualizada nas telas de todos os usuários.

Em qualquer momento, o mestre pode se desligar. Assim, em todas as telas aparecerá a indicação de ausência de mestre. Uma vez sem mestre, qualquer observador pode tornar-se o mestre, acionando um comando de controle. O primeiro observador a acionar este comando, tornar-se-á o mestre. O DistEdit permite, também, a transferência de controle de edição, que é feita através de uma conversação por uma sessão *on-line* ou até pelo telefone.

III.1.4. Software de Suporte

O DistEdit utiliza as facilidades de comunicação da ferramenta ISIS [Birm89], que é utilizada para aplicações de programação distribuída.

III.1.5. Trabalhos Futuros

Inúmeras extensões ao DistEdit estão sendo exploradas, incluindo a permissão de atualizações concorrentes, suporte a documentos mais complexos, tais como documentos que incluem formatação (fontes, margens, espaçamento,...), e a introdução de segurança.

III.2. PREP (Work in Preparation) [Neuw90]

III.2.1. Apresentação

O Prep é um ambiente multi-usuário que dá suporte a uma variedade de colaborações, em particular co-autoria e relacionamentos através de comentários na comunicação escolar. Os sistemas com suporte à comunicação escolar são considerados como sistemas cooperativos, onde membros de um grupo compartilham objetivos comuns em estágios avançados do conhecimento. Estes grupos são organizados localmente, contudo podem interagir intensamente à distância.

III.2.2. Características

As características do editor Prep são as seguintes:

- i) Prep não permite interações feitas ao mesmo tempo;
- ii) Enfatiza a comunicação entre duas pessoas;
- iii) Para o editor Prep, a co-autoria representa um desafio no trabalho cooperativo dentro de uma rede; o comentário, um desafio na comunicação entre grupos de autores e entre leitores externos com esses grupos; e a comunicação escolar, um desafio na interação da comunidade escolar.

- iv) A maior ênfase dada no projeto da ferramenta Prep foi a *interface*, principalmente, a representação visual do rascunho.
- v) Um ciclo comum em co-autoria é "edição-revisão-incorporação", onde o autor edita um documento, dá o seu rascunho para um segundo autor revisar e, em seguida, incorpora o material. O Prep faz com que as revisões sejam distintas do rascunho. Estas revisões, através de versões, permitirão aos escritores utilizarem o seu rascunho sem medo de perder o material antigo.

III.2.3. Funcionamento

O editor Prep compartilha características básicas com muitos sistemas hipermídias. O sistema define troncos, que correspondem a idéias. Um tronco pode conter textos, malhas, árvores ou imagens arbitrárias. O sistema define, ainda, ligações (*links*) entre estes troncos. Os troncos são guardados num banco de dados e são compartilhados entre os colaboradores.

O Prep permite aos autores criar rascunhos. Um rascunho define um espaço na área de trabalho que possibilita o acesso e a comunicação de outros autores ou comentaristas. Esta área consiste de várias colunas. Uma coluna, por exemplo, pode conter o artigo; outra, o plano que guia na confecção do conteúdo; e uma terceira, o local onde são dispostos os comentários.

III.3. InterNote [Catl89]

III.3.1. Apresentação

O InterNote é uma ferramenta de suporte à colaboração através de anotações no ambiente Intermedia [Garr86].

Os principais objetivos do InterNote são fornecer um conjunto de ferramentas para a revisão, fornecer revisões integradas no mesmo instante e capacidades de manipulação de anotações.

Com o auxílio do Intermedia, que é a base do InterNote, é possível criar *links* em vários tipos de documentos e dar suporte aos *links* dos usuários num documento, simultaneamente.

III.3.2. Características

As características da ferramenta InterNote são as seguintes:

- i) O InterNote acrescenta a facilidade de anotação ao ambiente, já existente, Intermedia, possibilitando ao usuário fazer anotações em qualquer tipo de documento. Por exemplo, um anotador pode sugerir revisões num documento gráfico usando ferramentas de edição gráfica;
- ii) O usuário é capaz de aprender a utilizar o InterNote rapidamente;

- iii) A criação de anotações é feita em apenas um passo. Ao fazer uma seleção e acessar um simples comando, um anotador poderá sugerir um comentário textual ou propor mudanças em edições;
- iv) Os documentos do Intermedia podem ser atualizados, a qualquer hora, pelo autor original ou pelo colaborador, com a autorização apropriada;
- v) O sistema é capaz de resolver casos de colisões quando, por exemplo, um usuário atualiza um documento no mesmo instante que outro faz anotações;
- vi) O InterNote permite a vários usuários fazer anotações num mesmo documento, simultaneamente;
- vii) Um revisor tem o acesso a comentários do revisor anterior, a fim de evitar comentários duplicados. Enquanto, por outro lado, um anotador pode esconder suas anotações;
- viii) Os autores podem incorporar, automaticamente, as sugestões de outros usuários. Este é um mecanismo simples e rápido, já que existe documentos com mais de cem anotações;
- ix) O sistema permite ao autor percorrer as anotações que foram incorporadas;

- x) Um autor pode ver as anotações de um documento de várias maneiras: uma de cada vez, acessá-las por data, ou ver aquelas que ainda não foram incorporadas.

III.3.3. Funcionamento

As anotações para um documento Intermedia são feitas usando NOTAS. Para criar uma NOTA, o usuário faz uma seleção dentro do documento e aciona o comando "Criar anotação" no cardápio. Um documento que foi anotado é chamado de rascunho. Todas as anotações de um rascunho são colocadas, automaticamente, numa janela de notas associadas ao documento, chamada de *Note Folder*. Esta janela, também, possui informações adicionais sobre cada anotação, tais como: o tipo de anotação, o nome do anotador, a data e a hora que a anotação foi criada, a posição da anotação no documento e o estado da anotação (conectado ou desconectado ao texto).

O Intermedia permite que apenas uma pessoa edite o documento por vez, mas qualquer número de pessoas pode fazer comentários sobre um documento, simultaneamente.

A estrutura de uma NOTA permite ao anotador fazer tanto uma edição diretamente na cópia da seleção de um rascunho, quanto fornecer um comentário textual que possa explicar ou justificar as alterações. Por essa razão, uma NOTA consiste de uma janela dividida em duas áreas: área de incorporação e área de comentário.

Uma vez feita a anotação por vários usuários, o autor do documento pode incorporar quaisquer mudanças sugeridas na área de incorporação das NOTAS, utilizando o comando "Incorporar anotações".

Se um documento recebeu anotações, os autores podem revisar o documento baseado nas sugestões de alterações e comentários. O processo de revisão inclui ler as anotações, determinar quando incorporar as sugestões, e fazer a edição de alterações no documento, baseado nos comentários nas NOTAS.

Para olhar as anotações, o autor deve, primeiro, abrir um *Note Folder* correspondente. Uma vez aberto, as marcas de anotações aparecem no rascunho. As anotações podem ser acessadas acionando-se uma linha no *Note Folder* ou seguindo-se as marcas dentro do documento.

O autor pode incorporar uma sugestão de alteração transferindo os dados através dos *links* entre anotações e o rascunho.

O InterNote possui, ainda, uma janela chamada *Web View*, que mostra o histórico do caminho do usuário, através do conjunto de *links* dos documentos, e um mapa, que mostra o documento no seu estado atual e todos os outros documentos conectados a ele.

III.3.4. Software de Suporte

O InterNote fornece uma *interface* com o usuário bastante consistente para anotação de todos os tipos de documentos do Intermedia.

III.3.5. Trabalhos Futuros

A seguir, são descritas algumas perspectivas de trabalhos futuros da ferramenta InterNote:

- i) criação de anotações com diferentes estilos de *interface* como, por exemplo, desenhar linhas, círculos e setas, diretamente, no documento a fim de facilitar as sugestões dos revisores;
- ii) visão dos comentários de vários anotadores, simultaneamente;
- iii) comunicação síncrona com os usuários que trabalham na mesma tarefa, assim como o envio de mensagens assíncronas para os usuários que não estão trabalhando naquele momento;
- iv) edição em grupo.

III.4. Quilt [Fish88]

III.4.1. Apresentação

O Quilt é uma ferramenta computacional para a edição colaborativa. Esta ferramenta fornece anotações, mensagens, um sistema de conferências e facilidades para notificação, para o suporte à comunicação e ao compartilhamento de informações entre os colaboradores de um documento.

III.4.2. Características

As características do sistema Quilt são as seguintes:

- i) O Quilt foi projetado para dar suporte a editores familiares ao usuário e a outras ferramentas de manipulação do documento;
- ii) O Quilt permite os usuários compartilhar voz, mensagens de texto e documentos;
- iii) Um documento é formado por uma base e nós unidos a essa base, utilizando a metodologia de hipertexto. Cada nó pode conter um documento, sugestões para revisão e comentários em formas de texto e voz;
- iv) Ele possui três tipos de comentários: comentários privados, que são visíveis apenas pelo criador; comentários públicos, que podem ser lidos por qualquer pessoa; e mensagens diretas, que são mensagens direcionadas a um indivíduo ou a um grupo;
- v) O Quilt possui diferentes tipos de autorizações, ou seja, ele determina as atividades permitidas em cada tipo de nó, associados aos diferentes tipos de colaboradores;
- vi) O Quilt fornece mecanismos que facilitam o compartilhamento de informações e a coordenação das atividades entre os co-autores. Como exemplo, pode-se citar a atividade *log* para cada nó, que faz com que

haja uma gravação da interação entre os colaboradores de um documento;

- vii) O Quilt pode ser visto, também, como um sistema de conferência assíncrono.

III.4.3. Funcionamento

Um escritor, dentro do sistema Quilt, pode utilizar um editor qualquer. Esse editor, ao contrário do Quilt, sabe como e onde o texto é exposto e alterado. Essa sessão de controle do Quilt para o editor acarreta problemas, porque o Quilt necessita saber onde o texto, o cursor e o *mouse* estão localizados para interpretar os pedidos de adição e de acesso às anotações corretamente. Para resolver essas dificuldades, o Quilt possui um processo de monitoração transparente, que se interpõe entre o usuário e o editor. No entanto, a saída (*output*) do editor é interpretado pelo monitor para atualizar a imagem na tela. Daí os comandos do Quilt sabem onde o cursor e o *mouse* estão e podem usar esta informação para determinar o que está sendo referenciado.

III.4.4. Software de Suporte

Na sua implementação, o Quilt permite a utilização de uma variedade de processadores de textos, evitando a mudança para um outro editor não muito familiar. O sistema utiliza como sistema de banco de dados, o Orion [Bane87] e o X Window [Sche86] e XR toolkit, para a manipulação de janelas, tornando-o extensível e portátil.

III.5. Shared Books [Lewi88]

III.5.1. Apresentação

O Shared Books é uma aplicação desenvolvida para dar suporte à manipulação de publicações colaborativas no sistema de processamento de texto View Point (1) da Xerox.

Utilizando o Shared Books, duas pessoas podem editar, simultaneamente, diferentes partes de um mesmo documento.

III.5.2. Características

As características do editor Shared Books são as seguintes:

- i) O Shared Books dá suporte à recuperação e ao armazenamento de documentos de uma forma bastante eficiente;
- ii) Ele possui um histórico de revisões, possibilitando a reprodução de versões antigas de um documento;
- iii) Ele é capaz de recuperar informações no caso de falhas no computador ou na rede;
- iv) Evita que pessoas não autorizadas modifiquem uma informação;

1 View Point é a marca registrada da Xerox Corporation

- v) Evita que diferentes pessoas modifiquem a mesma informação no mesmo instante;
- vi) Inclui assistência, não só no planejamento das tarefas e responsabilidades, como também no acompanhamento do progresso do trabalho de publicações;
- vii) Inclui suporte para a estimativa de custos do trabalho;
- viii) Simples de usar já que é uma extensão das facilidades do View Point.

III.5.3. Funcionamento

Quando o usuário aciona o Shared Books, surge na tela uma janela que expõe todas as suas entradas. Cada entrada corresponde a uma parte de uma publicação. Mais informações sobre essas entradas também podem ser expostas. É através de comandos do View Point que os usuários autorizados fazem o acesso ao Shared Books.

Quando há uma alteração numa estação de trabalho, não é feita imediatamente a atualização nas outras estações. Essa atualização é disparada após alguma ação feita pelo usuário.

III.5.4. *Software* de Suporte

O Shared Books fornece suporte à colaboração para o sistema de processamento de texto View Point da Xerox. A instalação View Point consiste de várias estações de trabalho unidas através de um arquivo compartilhado, da comunicação e dos serviços de impressão. O View Point também possui um sistema de correio eletrônico distribuído.

III.5.5. Trabalhos Futuros

O plano para o futuro é trabalhar mais perto dos usuários do Shared Books para refinar mais as necessidades de manipulação das publicações; fornecer melhor suporte para o acesso às revisões da publicação e estudar mais sua *interface*.

III.6. Video Draw [Tang91]

III.6.1. Apresentação

O Video Draw é um protótipo de suporte à atividade de desenho colaborativo, que se baseia nas observações do estudo de um grupo de trabalho formado por antropólogos, analistas de sistemas e desenhistas, conhecido como *Designer Interaction Analysis Laboratory* (DIAL). O DIAL foi formado para estudar o trabalho colaborativo. Ele está voltado, principalmente, para as atividades que ocorrem quando duas ou mais pessoas trabalham juntas,

utilizando uma superfície para o desenho compartilhado como, por exemplo, papéis, quadro-negro e tela de computador.

O protótipo do Video Draw utiliza um *video*, permitindo que dois ou mais colaboradores participem da edição do desenho em salas geograficamente distantes, compartilhando, interativamente, uma mesma superfície de desenho. Cada colaborador vê a composição dos desenhos feita por todos os outros componentes do grupo. Os gestos com as mãos sobre esses desenhos são transmitidos em tempo real para todos os outros colaboradores. Através do compartilhamento das visões dos desenhos e gestos de cada colaborador, eles podem interagir através do Video Draw como se estivessem fisicamente perto um dos outros.

III.6.2. Características

As características do editor Video Draw são as seguintes:

- i) Os gestos com as mãos são usados freqüentemente e produtivamente;
- ii) Não possui problemas de tempo durante a interação;
- iii) Produz um novo senso de relacionamento espacial entre colaboradores e espaço de desenho;
- iv) Permite que muitos participantes tenham um acesso concorrente ao desenho.

III.6.3. Funcionamento

O Video Draw aceita que duas pessoas compartilhem a mesma superfície de desenho. Ele consiste de câmeras de vídeo direcionadas a uma tela de monitores de vídeo interconectados. Os participantes utilizam canetas para escrever, diretamente, na superfície da tela. À medida que cada colaborador desenha na tela, a câmera transmite o desenho e os gestos das mãos ao outro colaborador. A comunicação entre os participantes pode ser aumentada com uma câmera de vídeo e monitores fornecendo uma visão frontal dos colaboradores e, também, com a ajuda de um telefone e canais de áudio.

III.7. VMACS [Laki87]

III.7.1. Apresentação

VMACS é um editor gráfico que possui uma coleção de macros visuais para a utilização de chaves e botões do *mouse*. O VMACS é um editor direcionado a pessoas que gastam muito tempo em seus editores gráficos. Ele fornece facilidades para a manipulação de objetos texto e gráfico. Ele dá suporte a um grupo de 3 a 4 pessoas em torno de um terminal, ou a grupos maiores com a projeção em uma tela.

III.7.2. Características

As características da ferramenta VMACS são as seguintes:

- i) Fornece imagens estáticas e imagens dinâmicas (que são seqüências de imagens estáticas);
- ii) Os objetos texto são tipos de objetos gráficos;
- iii) Suas principais funções não são acessadas através de um cardápio, a fim de evitar dúvidas no acompanhamento da edição;
- iv) A manipulação de objetos selecionados inclui: mover pela tela (*dragging*) e apagar. Os objetos atômicos podem ser agrupados em padrões e os padrões podem conter outros padrões. Qualquer objeto pode ser selecionado, desde o menor átomo visual até o maior;
- v) Ele foi projetado para um operador, que tem como tarefa ouvir o quê o grupo diz e colocar as informações na tela.

IV. FLECHA: EDITOR GRÁFICO COOPERATIVO PARA O MODELO DE OBJETOS DO GEOTABA

IV.1. Apresentação

O Flecha é uma ferramenta de edição gráfica cooperativa para o Modelo de Objetos (MOO) do GEOTABA [Matt89], desenvolvido num ambiente multi-usuário. Seu objetivo é ajudar um grupo de usuários a trabalhar em conjunto na confecção de Diagramas de Objetos (isto é, da representação gráfica do MOO [Mont91]), utilizando computadores, interagindo na troca de idéias e sugestões, revisando contextos até integrarem diagramas complexos de uma forma rápida, produtiva e eficiente.

IV.2. Representação Gráfica para o Modelo de Objetos do GEOTABA

Conforme dito anteriormente, o Flecha foi projetado para ser um editor gráfico cooperativo da representação gráfica do Modelo de Objetos do Sistema de Gerência de Objetos GEOTABA. O Modelo de Objetos (MOO) [Mont90] é uma ferramenta conceitual com um alto nível semântico, utilizado na modelagem de informação e sistemas, que foi definido de modo a poder ser

implementado, diretamente, pelo GEOTABA [Mont91]. A representação gráfica, que simboliza os conceitos do MOO, que é apresentada nesta seção, é chamada de Diagrama de Objetos.

IV.2.1. Modelo de Objetos

A seguir são apresentados alguns conceitos envolvidos no Modelo de Objetos:

- i) Classes: Descrevem propriedades comuns dos objetos.
- ii) Coleções: É a reunião de vários objetos. As consultas, em geral, se referem às coleções.
- iii) Pseudoclasses: São falsas classes, ou seja, são classes que não possuem instâncias.
- iv) Associações constantes e variáveis: São símbolos associados aos objetos. Uma associação constante indica que o objeto é sempre o mesmo. Uma associação variável refere-se a objetos diferentes em diferentes instantes.
- v) Relacionamento: As variáveis de instância de um objeto criam relacionamentos entre objetos. Os relacionamentos podem ser atributos do objeto, componentes e mapeamentos.

- vi) Cardinalidade dos Mapeamentos: Os mapeamentos têm nome e cardinalidade mínima e máxima.
- vii) Domínios pré-definidos: Os mapeamentos podem ser feitos para objetos definidos no modelo ou para objetos de domínios pré-definidos.
- viii) Serviços: São os processamentos.
- ix) Herança: Onde as propriedades são herdadas de uma superclasse pela sua subclasse.
- x) Chave: Quando uma variável de instância de uma coleção é uma chave; esta não terá valores nulos, nem dois elementos nesta coleção têm valores iguais da variável de instância.
- xi) Nomes Implícitos: O modelo oferece muita flexibilidade quanto à necessidade de dar nomes às entidades.
- xii) Níveis de Visualização: O modelo oferece vários níveis de visualização.

Maiores detalhes sobre o Modelo de Objetos podem ser encontrados em [Mont90].

IV.2.2. Diagrama de Objetos

O Diagrama de Objetos é a representação gráfica do MOO, apresentado na seção IV.2.1. O Diagrama de Objetos é a linguagem gráfica implementada no Flecha.

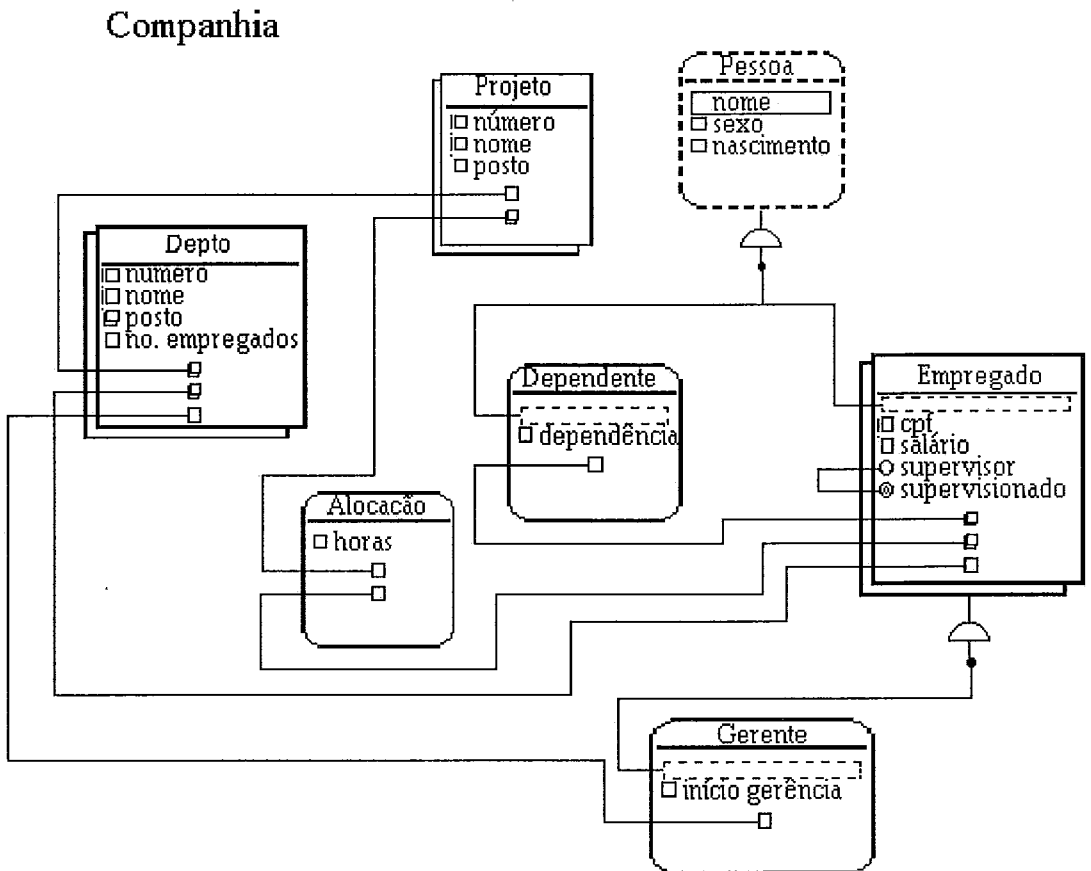
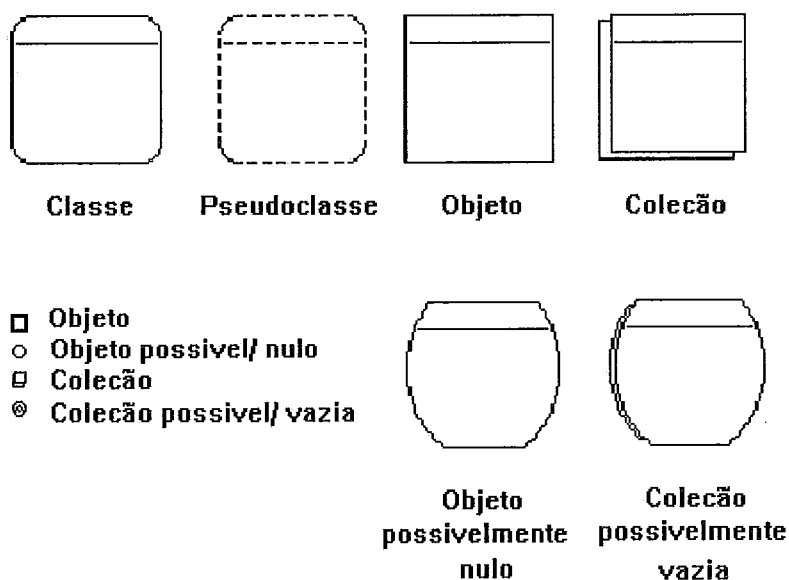


Figura IV.1 - Diagrama de objetos
para a Companhia [Mont91]

A figura IV.1 ilustra o esquema de objetos e classes para uma base de objetos COMPANHIA, através de um Diagrama de Objetos. A figura IV.2 mostra as notações de classes e objetos. Tendo como referência as duas figuras, serão apresentadas as representações gráficas associadas, fazendo um paralelo com os conceitos do Modelo de Objetos:



**Figura IV.2 - Ícones e Caixas para
Classes e Objetos [Mont91]**

- i) **Classes:** Na figura IV.1, as entidades Pessoa, Dependente, Alocação e Gerente são classes.

- ii) Coleção: As entidades Empregado, Projeto e Depto representam coleções.
- iii) Pseudoclasses: A entidade Pessoa é uma pseudoclasse.
- iv) Associações constantes e variáveis: como uma associação constante, pode-se citar o nome da coleção ou da classe, por exemplo, Empregado. O nome da coleção não mudará. Uma associação variável, por exemplo, são seus atributos: cpf, endereço, salário, supervisor e supervisionado, que podem ter seus valores alterados. O nome da Pessoa é uma associação constante, onde o nome do atributo é colocado dentro da representação da entidade.
- v) Relacionamento: Os relacionamentos são representados pelas ligações entre as entidades. No exemplo, o Empregado possui um relacionamento com o Depto e com a Alocação. Estes são chamados de relacionamentos binários sem atributos. No exemplo, existe, ainda, o auto-relacionamento Empregado-Empregado, que é representado pelos atributos supervisor e supervisionado. Este é o caso, onde os nomes dos mapeamentos são explícitos.
- vi) Cardinalidade dos Mapeamentos: No Diagrama de Objetos, a cardinalidade de um relacionamento é representado no início e no fim da ligação, de acordo com os símbolos da figura IV.2. Um objeto corresponde a um, e somente um objeto; um objeto possivelmente nulo corresponde a um ou nenhum objeto; uma coleção corresponde a um

ou mais objetos; e uma coleção possivelmente vazia corresponde a nenhum ou mais objetos.

- vii) Classes Implícitas: Os atributos cpf, endereço e salário de Empregado são considerados objetos de classes triviais e pré-definidas (texto, número, data, etc.). Por essa razão, estes atributos, também, representam mapeamentos, apesar de não existirem ligações dos atributos com as classes de seus elementos.
- viii) Herança: A classe Pessoa, na figura IV.1 é a superclasse de Dependente e Empregado. Da mesma forma, o Gerente é sub-classe de Empregado. O símbolo gráfico de um semi-círculo embaixo de um objeto e uma ligação partindo do semi-círculo, atingindo uma caixa pontilhada dentro de outro objeto, representa a herança do pai (primeiro objeto) para o filho (segundo objeto).
- ix) Chave: Os atributos chaves são representados precedidos por um sinal de exclamação (!). Na coleção Projeto, os atributos número e nome são chaves.

Vários conceitos do MOO não tiveram suas representações introduzidas no Diagrama de Objetos. É o caso da especificação de atributos componentes, da especificação de cardinalidade máxima e mínima, da utilização de objetos e classes pré-existentes ou padrões, e da especificação do protocolo de serviços dos objetos. Estas são consideradas as perspectivas futuras do Diagrama de Objetos, que é apresentado com mais detalhes em [Mont91].

IV.3. Requisitos

O Flecha é uma ferramenta de *software* utilizada para a modelagem conceitual de um sistema de objetos. Esta seção apresenta as funcionalidades do Flecha que o caracterizam como uma ferramenta cooperativa.

IV.3.1. Editor para a representação gráfica do MOO

Como editor da representação gráfica do MOO, o Flecha, de uma forma bastante simples, transforma a modelagem manual, já existente, numa modelagem assistida por computador.

IV.3.2. Multi-usuário cooperativo

O Flecha é um sistema multi-usuário e, portanto, pode atender a vários modeladores ao mesmo tempo, estimulando a cooperação e a comunicação entre estes, via diagrama ou outros mecanismos descritos a seguir.

IV.3.3. Diferentes tipos de autorizações

Cada usuário do Flecha possui um papel específico num diagrama particular e, a cada um, são associados alguns privilégios, ou melhor, diferentes autorizações. As autorizações regulam as diferentes visões dos usuários no diagrama.

Todo diagrama possui um gerente. O gerente é o responsável pelo trabalho final e seu andamento. Ele pode ser considerado como o primeiro modelador do diagrama. É aquele que cria e inicia a sua confecção e possui a responsabilidade de coordenar as atividades; ceder novas autorizações dentro do diagrama; dividir o diagrama em contextos e depois integrá-los novamente; alterar qualquer contexto; e fazer qualquer tipo de comentário. O poder do gerente pode variar, dependendo do grupo de trabalho, e o seu papel pode ser substituído por outro membro do grupo.

Os co-modeladores, também conhecidos como co-autores, são os usuários diretos do editor. Eles fazem parte da equipe liderada pelo gerente. A cada um é designado um ou mais contextos. Suas responsabilidades são: trabalhar no contexto e finalizá-lo na data prevista; revisar ou auxiliar outros co-modeladores. Dentro de um contexto, um co-modelador tem o direito de liberar o acesso de alteração para outro co-modelador dessa mesma equipe.

O último tipo de usuário do editor, o leitor, apenas se inteira do que está sendo feito pelos co-modeladores, participa com comentários e mensagens diretas, sem a permissão de alteração.

As autorizações, como foi mencionado anteriormente, são definidas pelo gerente, antes de qualquer coisa. Através de uma função no cardápio, o gerente fornece ou altera autorizações. Num dado diagrama, um mesmo usuário pode representar um co-modelador, um leitor e até mesmo um gerente.

IV.3.4. Comentários e Anotações

Uma característica importante é a facilidade de criar e acessar as anotações e os comentários dentro do ambiente do Flecha. Eles são expostos numa janela e podem conter textos ou gráficos. Um usuário tem o acesso a três tipos de comentários:

- i) Públicos: são comentários que podem ser lidos por qualquer pessoa;
- ii) Privados: são comentários visíveis apenas pelo criador;
- iii) Mensagens Diretas: são mensagens dirigidas a um indivíduo ou a um grupo.

O papel dos comentários dentro da edição cooperativa é muito importante. Ele funciona como uma revisão feita por um colega experiente, que também possui o conhecimento geral do que está sendo feito. Portanto, é fundamental que a utilização direta dos comentários num contexto (ou diagrama) seja feita de uma forma bem fácil, ou seja, os comentários gráficos devem poder ser adaptados e reaproveitados imediatamente.

IV.3.5. Atualização imediata de objetos bloqueados em outros contextos

O diagrama de objetos pode ser dividido em vários contextos, onde cada contexto é uma lista de objetos. Portanto, um contexto pode ser considerado como uma parte do diagrama geral, que será designado a um co-modelador. Este trabalhará no seu contexto até integrá-lo aos outros contextos.

É comum objetos do diagrama pertencerem a mais de um contexto ao mesmo tempo. Assim, o primeiro contexto que entrar em sessão bloqueará a permissão de alteração dos seus objetos por outros contextos.

Por exemplo, um certo diagrama é dividido em vários contextos, entre eles os contextos 1 e 2, onde ambos possuem o objeto X. Num certo instante, um co-modelador entra em sessão com o contexto 2 e, em seguida, outro co-modelador entra em sessão com o contexto 1. O primeiro co-modelador tem livre acesso a todos os objetos contidos no contexto 2, enquanto na tela do segundo co-modelador, o objeto X aparece em vermelho, ou seja, bloqueado para alteração. Nesse caso, o segundo co-modelador poderá apenas observar as atualizações feitas naquele momento no objeto X pelo primeiro co-modelador e poderá participar trocando mensagens, se uma alteração indesejada tiver sido feita.

IV.3.6. Integração de contextos

O diagrama pode ser dividido em vários contextos, cada um designado a um co-modelador. Estes contextos são trabalhados durante algum tempo, comentados, criticados e revisados por outras pessoas como, por exemplo, o gerente, leitores externos, etc., até ficarem prontos para a integração com outros contextos no diagrama geral. Com as características de edição cooperativa fornecida pelo Flecha, é esperado que a frequência dessas integrações seja tão grande quanto possível. Esta fase de integração é feita em etapas e é realizada apenas pelo gerente, ou por um co-modelador autorizado. A seção V.1.7 descreve com mais detalhes o processo de Integração de Contextos.

IV.3.7. Histórico de Contextos e Diagramas

O Flecha mantém um histórico de todos os contextos e diagramas que estão, atualmente, sendo modelados. Este inclui informações sobre os modeladores responsáveis; quando um contexto foi criado; quando foi concluído; e indica a presença de comentários ou anotações. As informações podem ser úteis como parte da história do projeto, para futura reutilização, manutenção, coordenação das tarefas e para ajudar nas atividades dos co-modeladores.

IV.3.8. Comunicação com o PROTOGEO

O Flecha prevê ainda uma facilidade para comunicação com o protótipo do GEOTABA, PROTOGEO [Mont92]. Essa comunicação será feita através de uma ferramenta de conversação entre a representação gráfica interna dos diagramas do Flecha e o esquema de classes do PROTOGEO. Esta ferramenta, que ainda está em fase de concepção (especificação), possui a tarefa de gerar, a partir da modelagem do Flecha, um documento com o esquema de classes para o PROTOGEO, e vice-versa, ou seja, a partir deste documento criado pelo PROTOGEO, gerar a representação gráfica no Flecha.

V. IMPLEMENTAÇÃO DO FLECHA

Muitas pessoas que trabalham com modelagem de objetos já utilizam uma variedade de editores gráficos e algumas ferramentas de comunicação; os colaboradores, por sua vez, freqüentemente usam diferentes ferramentas num único diagrama. Por essa razão, uma das preocupações na implementação do Flecha foi minimizar o tempo de aprendizado na sua utilização.

O Flecha se compara a um editor gráfico comum, com funções simples de serem acessadas e manipuladas. O editor possui cardápios e janelas bastante conhecidas, pois utiliza a *interface* do MS-Windows [Micr87]. Suas facilidades cooperativas baseiam-se na comunicação entre processos feita através da rede Novell [Nove88].

Uma boa *interface* foi a meta principal do Flecha. Este editor foi, desde o princípio elaborado e, depois, implementado tendo como objetivo tornar-se uma ferramenta simples de aprender e, conseqüentemente, fácil de manipular. Desta forma foi fácil constatar a boa aceitação do protótipo do Flecha, pois os primeiros usuários não tiveram grandes dificuldades em manipulá-lo e ficaram motivados para utilizar todas as suas funções.

O Flecha foi implementado na linguagem Actor [Whit91]. Toda a sua *interface* e a parte gráfica foram simplificadas por essa linguagem, que é executada

sob o ambiente Windows [Micr87]. O Actor é uma linguagem de programação puramente orientada a objetos, cuja filosofia é bastante semelhante ao Smalltalk [Gold84]. Sua programação é toda feita em termos de objetos e das mensagens que estes objetos enviam e recebem.

A rede Novell, versão 3.11, é utilizada para fazer o compartilhamento entre os diversos arquivos, tornando possível a comunicação entre co-modeladores e a interação através de contextos.

V.1. Funcionamento

No ambiente do MS_Windows, o Flecha aparece como um ícone na janela de aplicações. Qualquer pessoa dentro da rede tem acesso a esse ícone.

V.1.1. Histórico

Ao ser iniciada a ferramenta Flecha, o nome e a senha do usuário são solicitados para identificação de seu tipo de autorização. Em seguida, é exposto o "Histórico do Flecha" que é uma tabela utilizada para fazer o acesso a um diagrama, ou a um contexto, anteriormente, criado. Cada entrada na tabela é constituída das seguintes informações:

- i) **diagrama:** nome ou identificador do diagrama;
- ii) **contexto:** nome ou identificador do contexto;
- iii) **modelador:** nome do modelador responsável;
- iv) **data inicial:** data da última atualização;

- v) **data final:** data do término da edição;
- vi) **comentários:** indicam a presença, ou não, de comentários.

V.1.2. Tela Principal

A figura V.1 mostra a tela principal do editor. Ela é composta do título do diagrama, ou do contexto, do cardápio de suas principais funções, de uma palheta de ícones, que correspondem às representações do Diagrama de Objetos, e uma área onde é feita a modelagem.

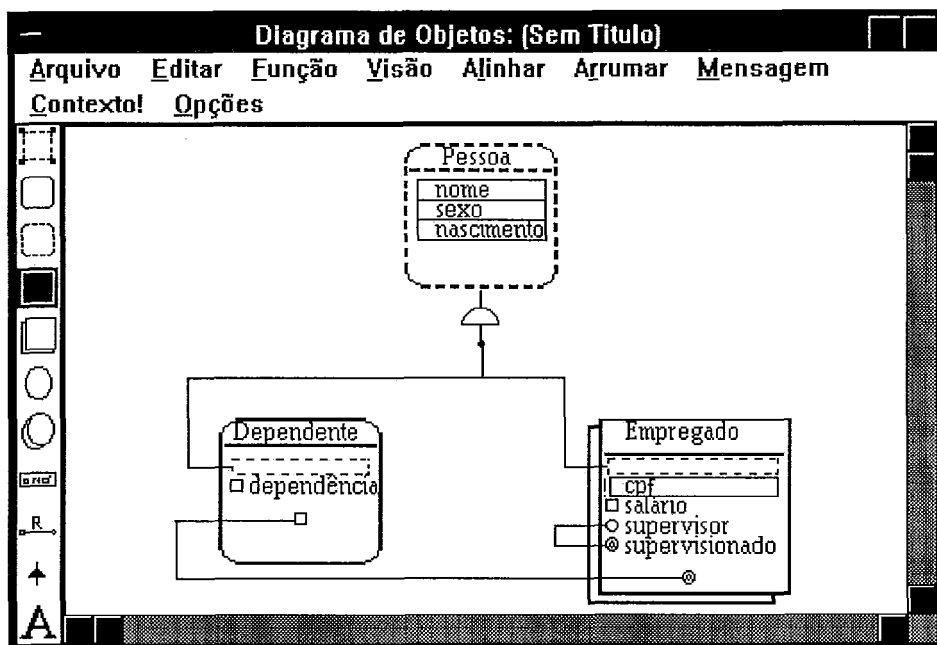


Figura V.1 - Tela Principal

V.1.2.1. Comandos

O Flecha fornece um conjunto de comandos para criar, manipular e guardar diagramas e contextos em arquivos. Estes comandos são selecionados através do acesso a cardápios de comandos e da palheta de ícones. A maioria destes comandos correspondem, diretamente, ao comportamento de um objeto no Flecha. Eles também ilustram o poder das facilidades oferecidas pela ferramenta.

V.1.2.2. Funções dos Ícones

Os ícones são símbolos gráficos que são sempre apresentados na esquerda da área do desenho, onde cada símbolo representa um objeto gráfico do Flecha. A seleção de uma representação do Diagrama de Objetos pode ser feita através desses ícones. A função selecionada torna-se marcada de preto. Em ordem de apresentação dos ícones, as funções são: seleção, classe, pseudoclasse, objeto, coleção, objeto possivelmente nulo, coleção possivelmente vazia, variável de instância, relacionamento, herança e título do diagrama.

A seguir, é feita a descrição das funções do Flecha:

i) Seleção

Quando a função Seleção é escolhida, existem várias maneiras de selecionar um objeto ou um grupo de objetos. A maneira mais simples é mover o cursor seletor para qualquer ponto dentro do polígono envolvente (*bounding-box*) do objeto e apertar o botão esquerdo do *mouse*; assim, seleciona-se aquele objeto. Para selecionar um grupo de objetos, aciona-se o botão esquerdo do *mouse*, movendo-o

sem largar o botão. Será desenhado, então, um retângulo feito com linhas pontilhadas. Quando o botão do *mouse* é solto, qualquer objeto que estiver completamente dentro do retângulo é selecionado.

NOTA: Se um objeto já estiver selecionado, pode-se movê-lo através da movimentação do *mouse* com o botão esquerdo apertado.

ii) Classe, pseudoclasse, objeto, coleção, objeto possivelmente nulo, coleção possivelmente vazia

Cada uma dessas funções desenha o objeto correspondente na posição escolhida. Depois de feita a escolha do objeto, acionando o botão esquerdo do *mouse*, aparece na tela a sombra do objeto. Esta sombra acompanhará os movimentos do *mouse* até que o botão seja solto. Depois de fixado o objeto, aparece uma janela para o fornecimento de seu nome. Este nome poderá ser alterado posteriormente, se necessário.

iii) Variável de Instância

Esta função preenche a lista de variáveis de instância de um objeto. Depois da seleção do objeto e da escolha da função Variável de Instância, é exibida na tela uma janela, que é mostrada na figura V.2. A janela é composta de um campo de edição de atributos, de campos de escolha de tipos e chave, de botões e de um campo destinado à lista de variáveis de instância (retângulo maior). Suas principais funções são:

Variáveis de Instância de: Empregado

OK
Cancela
Remove

nome
endereço
naturalidade
nacionalidade

Insere
no Fim
Estrutura

Nome: nacionalidade

Tipo: Objeto

Chave: Sim Nao

Figura V.2 - Variável de Instância

- nome:** edição do nome do atributo;
- tipo:** seleção do tipo de atributo: objeto, objeto possivelmente nulo, coleção, coleção possivelmente vazia e constante;
- chave:** indicação do atributo como chave, ou não;
- estrutura:** botão que pede a lista de variáveis de instância que farão parte da estrutura do atributo selecionado na lista de variáveis de instância;
- insere:** botão que insere o atributo na lista de variáveis de instância. O novo atributo ocupará a posição acima do atributo já selecionado na lista de variáveis de instância;
- insere no fim:** botão que insere o atributo no fim da lista de variáveis de instância;
- remove:** remove o atributo selecionado na lista de variáveis de instância;

- cancela:** cancela a operação;
OK: completa a operação.

Depois da operação completa, o objeto será redesenhado e preenchido com a lista de variáveis de instância.

iv) Relacionamento

Esta função desenha o relacionamento entre dois objetos. Depois da seleção do objeto e da escolha da função Relacionamento, é exibida na tela uma janela origem, que é mostrada na figura V.3. A janela é composta, na sua esquerda, da lista de variáveis de instância do objeto e, na sua direita, dos tipos associados às variáveis. Conforme foi visto no capítulo anterior (seção IV.2.2), existem dois tipos de relacionamentos: com atributo e sem atributo. Para o relacionamento com atributo, seleciona-se o atributo na lista da esquerda e, automaticamente, é marcado o tipo associado. Para o relacionamento sem atributo, seleciona-se o tipo na lista da direita.

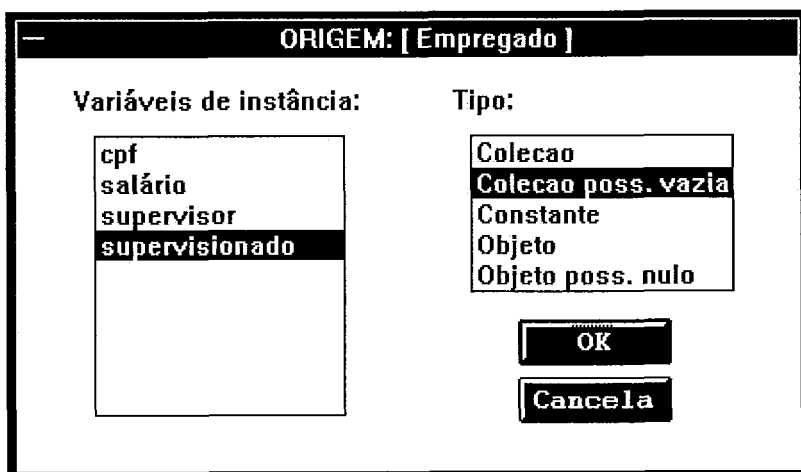


Figura V.3 - Relacionamento

Os botões OK e Cancela completam ou cancelam, respectivamente, a operação.

Depois de selecionada a origem, deve-se percorrer com o *mouse*, desenhando o relacionamento até se atingir um outro objeto. Ao alcançar o objeto destino, a janela da figura V.3 reaparece, pedindo o tipo do relacionamento destino. Depois de completa a operação, o relacionamento calculado é desenhado na tela. Esta função oferece outras duas facilidades:

iv.1) Malha

Para facilitar o caminho do relacionamento, é desenhada uma malha de pontos na tela. No fim de toda a operação, esta malha desaparece.

iv.2) Voltar

A função oferece a opção de voltar no caminho da malha. Para acessar esse comando, deve-se acionar o botão direito do *mouse*.

v) Herança

Esta função desenha a herança entre dois objetos. Depois da seleção do objeto pai e da escolha da função Herança, o objeto pai ganha um conector que o ligará a seus filhos. Este conector ficará localizado abaixo do objeto pai. Deve-se, então, mover o *mouse* com o botão esquerdo acionado até localizar o objeto filho. Em seguida, o Flecha completará a ligação entre pai e filho.

vi) Título do diagrama

Esta função desenha o título do diagrama no topo esquerdo da página.

V.1.2.3. Funções do Cardápio de Comandos

i) Arquivo

O Flecha dá suporte ao salvamento, carregamento e impressão de gráficos, através de uma função Arquivo do cardápio do Windows. Os comandos Novo, Abrir, Salvar e Salvar como, funcionam como nos demais aplicativos Windows, assim como os comandos Imprimir e Set Up da Impressora. Os diagramas são guardados no formato do Flecha com a extensão ".DOB".

ii) Editar

O Flecha fornece suporte à área de transferência (*clipboard*) do Windows, através da função Editar do cardápio. Cortar e Copiar põem a figura selecionada na área de transferência. A função Cortar apenas remove a figura selecionada da tela. A função Colar traz o conteúdo da área de transferência para a tela na posição definida, anteriormente, pelo *mouse*.

A função Limpar Tela remove todos os objetos presentes na tela.

iii) Funções

As funções correspondem, diretamente, aos ícones expostos no lado esquerdo da tela. Escolher uma função possui o mesmo efeito de selecionar um ícone.

iv) Visão

Esta função do cardápio reúne os comandos de alteração da aparência geral de uma imagem. Selecione Normal para mostrar a imagem próxima à sua real dimensão. Página Inteira reduz a imagem para mostrar a página inteira correspondente dentro da janela.

Para que a função Zoom seja acionada, é necessário fazer a seleção de um retângulo, no qual tudo que estiver contido será aumentado.

Selecionando Mostra Malha, aparece na tela uma malha na área de desenho em forma de uma coleção de pontos equidistantes.

v) Alinhar

Os objetos selecionados podem ser alinhados, acionando-se a função Alinhar no cardápio. Esta função oferece as opções Esquerda, Centro e Direita, e Cima, Meio e Baixo, esses alinhamentos são feitos tendo como referência a caixa que envolve as figuras selecionadas. Selecionando Esquerda, por exemplo, moverá todos os objetos que estão dentro da seleção para a esquerda do polígono envolvente do retângulo de seleção.

vi) Arrumar

Esta função permite a reorganização das posições dos objetos numa certa ordem. Ela funciona depois de feita a seleção de objetos.

Os comandos Trazer para Frente e Mover para Trás controlam a posição da seleção numa ordem de objetos. Os objetos na frente cobrem objetos que estão atrás.

Selecionando Alinhar na Malha, alinha-se as origens de todos os objetos selecionados na malha ativa.

vii) Mensagem

Esta função envia e acessa as mensagens ou comentários entre os usuários do Flecha. Ela possui os comandos: Comentário Público, Comentário Privado e Mensagem Direta.

viii) Contexto!

Esta função divide o diagrama corrente em contextos. Esta função só pode ser utilizada no modo gerente e depois de feita a seleção de alguns objetos (ver seção V.1.6).

ix) Opções

É a função onde são, ou serão, colocados os comandos que fazem parte de uma miscelânea. A atual versão da ferramenta possui apenas o comando Tempo de Atualização, que muda o tempo de atualização de tela e arquivo. Outros comandos poderão ser incluídos futuramente como, por exemplo, liberação de objetos bloqueados, mudança de autorizações e manipulação de comentários e mensagens.

x) Outras funções

O Flecha apresenta, ainda, outras funções:

x.1) Renomear um objeto

Esta função altera o nome de um objeto que foi desenhado na tela. Este comando funciona depois de feita a seleção de um objeto e acionado o botão direito do *mouse*.

x.2) Apagar um objeto

Esta função apaga um objeto da tela. Este comando funciona depois de feita a seleção deste objeto, ou objetos, e acionada a tecla Del do teclado.

x.3) Mandar mensagens de advertência

O Flecha evita que o usuário cometa erros, mandando mensagens de advertência.

V.1.3. Modo Leitor, Gerente e Modelador

O acesso a um diagrama, ou a um contexto, só é feito pelos usuários com a devida autorização. Uma vez autorizado, o usuário pode fazer o acompanhamento no editor através de três modos: Modo Leitor, Modo Modelador ou Modo Gerente. Por exemplo, um modelador **A** é autorizado a alterar o contexto **A** e não é autorizado a alterar o contexto **B**. Caso ele solicite acessar o contexto **B**, ele participará através do Modo Leitor.

i) Modo Leitor

O Modo Leitor corresponde ao modo onde o usuário não tem a permissão de alteração no contexto, ou no diagrama. No entanto, ele pode acessar algumas funções do cardápio: Novo, Abrir, Imprimir, Set Up Impressora, Limpar Tela, Visão e Mensagens, enviando mensagens diretas, lendo e enviando comentários públicos.

Pode acontecer que o contexto citado esteja, no mesmo instante, sendo alterado pelo seu modelador, então, o usuário leitor poderá ver essas alterações "imediatamente".

ii) Modo Gerente

O gerente é um modelador com mais direitos. Ele pode ser considerado o primeiro modelador. Qualquer usuário pode ser um gerente, basta que ele crie um novo diagrama. Assim, imediatamente, é associado a ele a autorização de gerente daquele diagrama.

O gerente cria um novo diagrama e pode dividi-lo em contextos. Cada contexto é associado a um usuário cadastrado no Flecha. Estes usuários serão os co-modeladores.

Uma vez feita a divisão de contextos e depois destes serem modelados e revisados, o gerente pode integrá-los novamente. A integração é feita da seguinte forma: aparecerão todos os contextos na tela. Estes serão estudados e retocados, até formarem um só todo (ver seção V.1.7).

iii) Modo Modelador

O co-modelador terá a ele associado um contexto. Neste caso, ele tem livre acesso às alterações.

Pode acontecer de dois co-modeladores estarem em sessão com dois contextos que possuem objetos comuns. Um contexto terá o acesso livre a esses objetos, enquanto o outro contexto não poderá alterá-los. Os objetos bloqueados aparecerão na tela em vermelho. Através das trocas de mensagens, um co-modelador pode ceder o acesso dos seus objetos para um outro co-modelador.

V.1.4. Envio de Mensagens

Para enviar uma mensagem, o usuário deve acessar à função "Mensagem" no cardápio. Em seguida, deve escolher entre comentário público, privado ou mensagem direta. Depois de escolher entre comentário público ou privado, faz-se uma seleção, através do *mouse*, no contexto ou diagrama. Neste instante, aparecerá na tela uma janela.

A janela é dividida em duas partes. Os objetos selecionados, anteriormente, estarão na parte de cima da janela e a parte de baixo estará reservada para os comentários. Se a escolha for Mensagem Direta, aparecerá uma janela para a edição do texto mensagem e para a escolha do destinatário.

Nas telas para onde foram destinadas as mensagens, aparecerá um ícone que indica a presença da janela de mensagem enviada, de uma forma minimizada, ou seja, para ler a mensagem, deve-se acessar esse ícone.

V.1.5. Atualização Imediata

A atualização imediata é a atualização dos objetos bloqueados, num contexto, enquanto estão sendo alterados no mesmo instante, em outros contextos. O Modo Leitor é um caso particular onde todos os objetos do contexto estão bloqueados.

Na divisão de contextos feita pelo gerente, pode acontecer que dois contextos possuam objetos em comum. Por exemplo, o diagrama Companhia possui vários objetos. Na divisão, o contexto 1 ficou com os objetos: Pessoa, Empregado e Dependente, e o contexto 2 ficou com os objetos: Empregado,

Alocação, Projeto e Departamento. No caso, ambos os contextos possuem o objeto Empregado. Se um modelador entrar em sessão primeiro com o contexto 1 e, em seguida, outro modelador entrar em sessão com o contexto 2, o objeto Empregado aparecerá no contexto 2 bloqueado, na cor vermelha para diferenciar dos outros objetos em preto. Este objeto não poderá ser selecionado, nem alterado. No entanto, as alterações feitas pelo primeiro modelador serão percebidas "imediatamente" na sua tela.

O segundo modelador pode não concordar com as alterações que foram feitas pelo primeiro modelador no objeto. Eles poderão se comunicar através de mensagens. Eles poderão até ceder objetos bloqueados de um para o outro, ou seja, o primeiro modelador poderá ceder o objeto Empregado ao segundo modelador.

V.1.6. Divisão de Contextos

Esta é uma função acessada apenas pelo gerente. Quando um novo diagrama é criado, o Flecha, automaticamente, passa para o modo gerente. O gerente, no caso, modela alguns objetos neste diagrama. No exemplo do diagrama Companhia, o gerente poderia ter desenhado os objetos: Empregado, Departamento, Alocação e Pessoa. Para dividir esses primeiros objetos em diferentes contextos, inicialmente, ele deve selecionar os objetos candidatos ao primeiro contexto. Em seguida, acessar o comando Contexto! no cardápio do Flecha. Desta forma, aparecerá uma janela pedindo o contexto que deve ser aberto, ou se é o caso de criar um novo contexto. Se o nome do diagrama é "COMPANH.DOB", o primeiro contexto será "COMPANH1.CTX", o segundo "COMPANH2.CTX" e assim por diante. A janela seguinte exibirá a lista de objetos

que foi selecionada. Se for o caso, alguns objetos poderão ser removidos da lista. Continuando, o Flecha pedirá os nomes dos modeladores autorizados àquele contexto.

Estas operações serão repetidas até que todos os objetos sejam distribuídos em contextos. Nada impede que diferentes contextos possuam objetos comuns.

O acesso a esses contextos poderá ser feito por qualquer usuário do Flecha. No entanto, o direito a alteração será do modelador responsável.

V.1.7. Integração de Contextos

Esta função também só pode ser acessada no modo gerente. Depois dos modeladores terem modelado e revisado os seus contextos e, também, os de outros modeladores, fazendo críticas e comentários, o diagrama estará pronto para ser integrado.

A integração de contextos é feita, automaticamente, quando o gerente abre o diagrama. O Flecha acessa a todos os objetos pertencentes àquele diagrama que foram criados ou alterados em cada contexto. Os objetos são expostos numa mesma página. O gerente tem o trabalho de organizar os objetos na tela, acrescentar relacionamentos ou atributos novos e apagar objetos, se necessário.

O gerente, também, pode dividir o diagrama em novos contextos designando-os a novos modeladores. Estas operações podem ser repetidas até que se atinja o diagrama desejado.

V.2. Arquitetura do Flecha

A arquitetura do Flecha apresenta-se conforme a figura V.4.

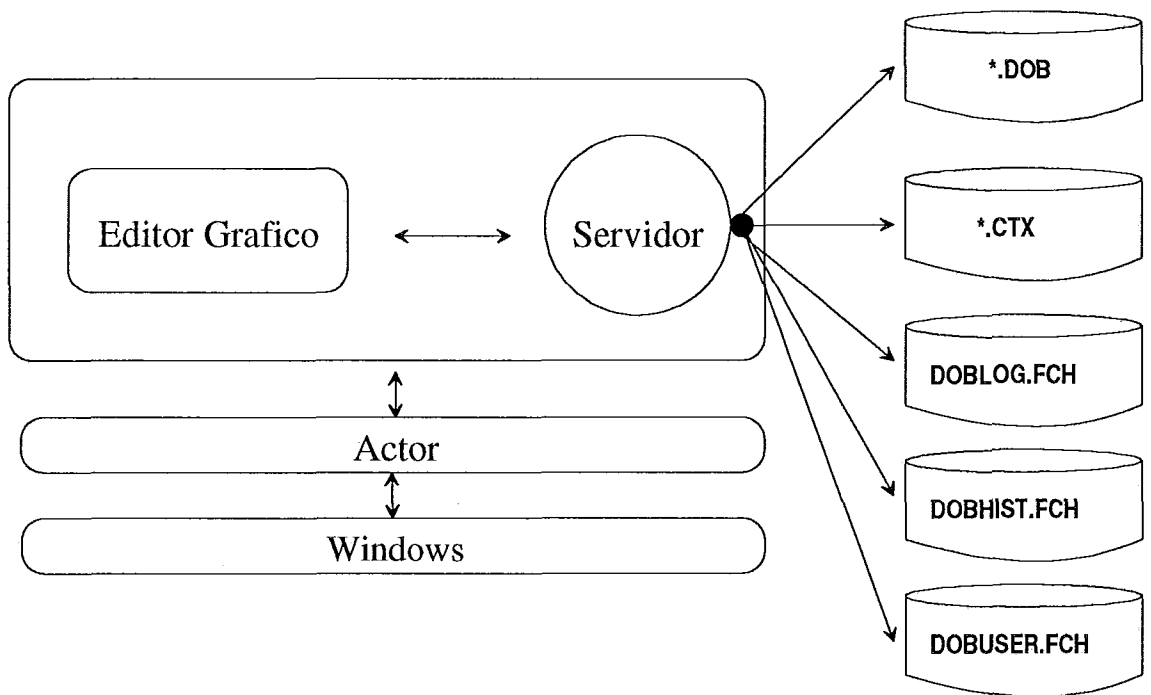


Figura V.4 - Arquitetura do Flecha

O Flecha é composto de um editor gráfico e de um servidor, que juntos tornam-no um editor gráfico cooperativo. O servidor coordena as atividades de todos os modeladores, permite o acesso através de autorização, o envio de mensagens e o compartilhamento de objetos, ou seja, controla todo o processo cooperativo. É ele também que gerencia os arquivos. Estes são:

- i) **arquivo com extensão DOB:** são os arquivos de diagramas modelados, que contém todas as informações sobre os diagramas durante a modelagem. Informações estas que incluem detalhes sobre cada objeto no diagrama, o usuário responsável e quais os objetos que estão bloqueados ou não. O nome do diagrama é dado pelo gerente e este nome poderá conter até sete caracteres.

- ii) **arquivo com extensão CTX:** são os arquivos que correspondem aos contextos de um diagrama. Os objetos contidos num contexto estão associados diretamente ao diagrama, ou seja, alterar ou criar um objeto num contexto significa alterar ou criar um objeto no seu diagrama. Os nomes dos contextos são formados pelo nome do diagrama mais um número.

- iii) **arquivo "DOBLOG.FCH":** é o arquivo que corresponde ao *log* do Flecha. Ele informa os usuários que estão utilizando o Flecha no momento, mantendo uma ordem de chegada.

- iv) **arquivo "DOBHIST.FCH":** é o arquivo que indica o histórico de todos os diagramas e contextos do Flecha.

- v) **arquivo DOBUSER.FCH":** é o arquivo de cadastro de todos os usuários do Flecha. Ele contém suas respectivas senhas.

V.3. Esquema de Classes

O esquema de classes do Flecha é apresentado nas figuras V.5a, V.5b e V.5c.

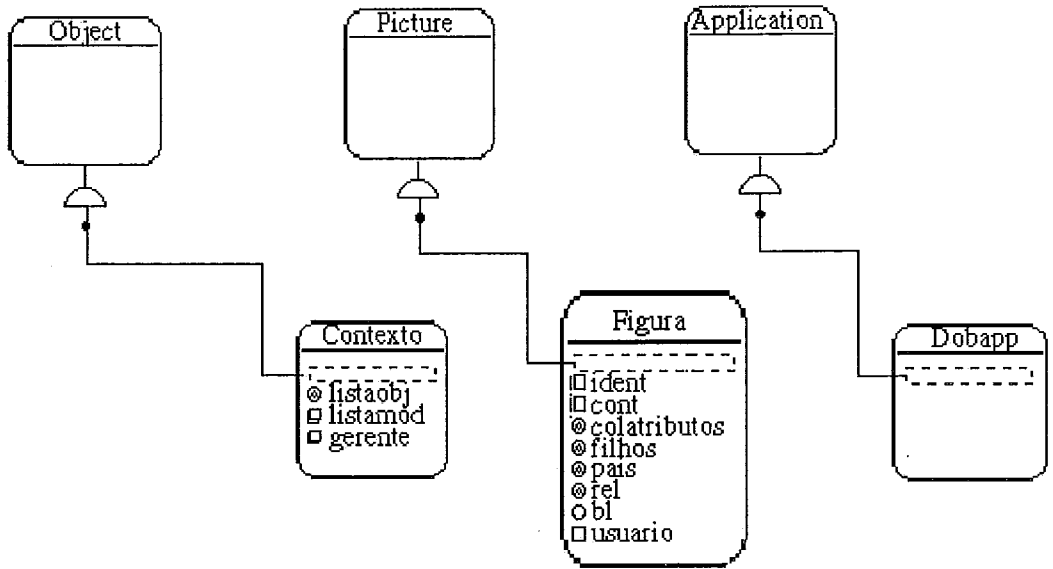


Figura V.5a - Esquema de Classes do Flecha

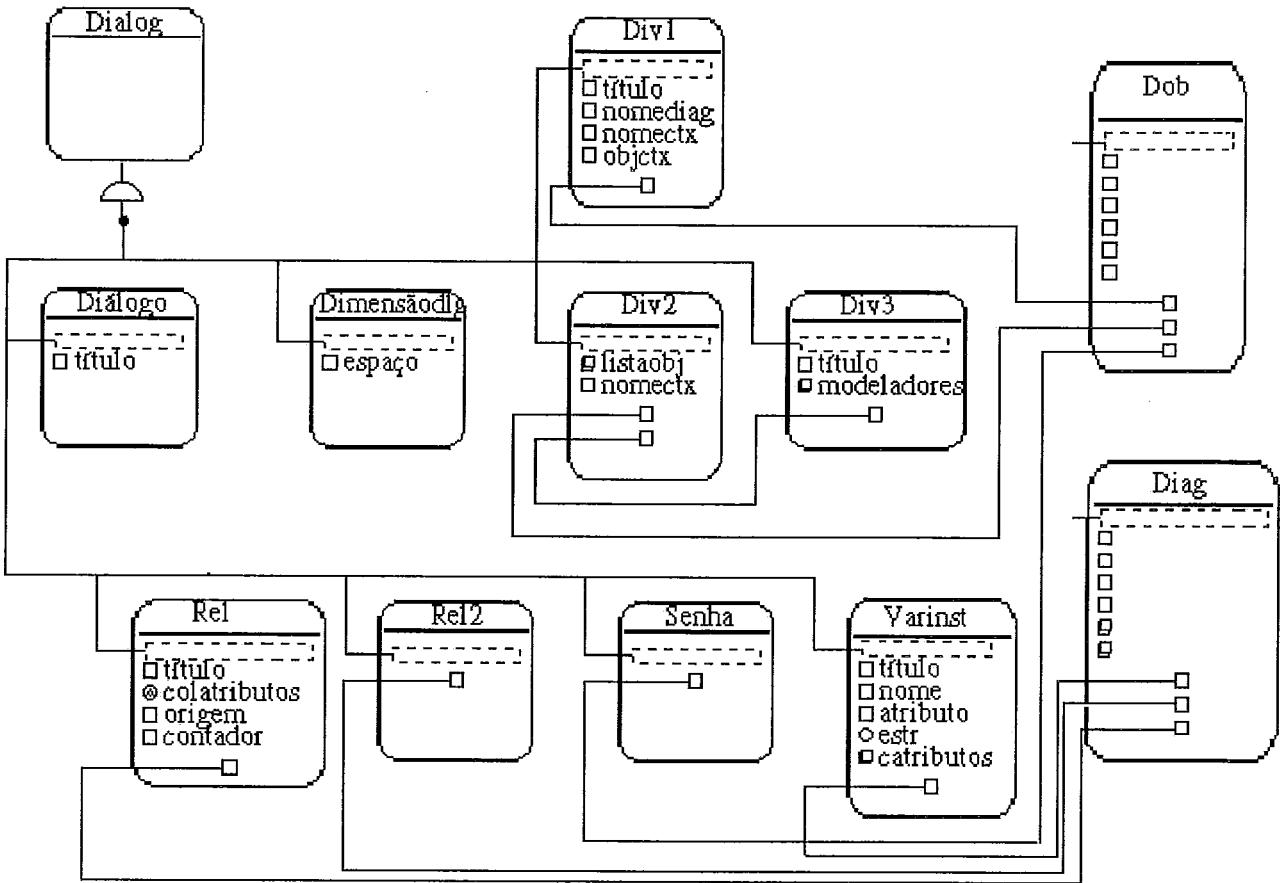


Figura V.5b - Esquema de Classes do Flecha

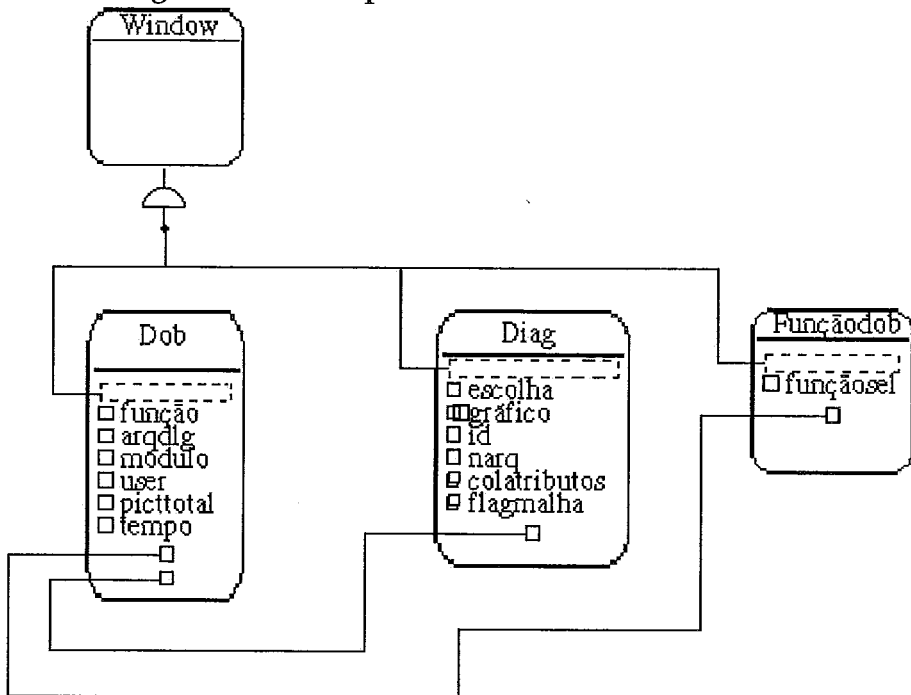


Figura V.5c - Esquema de Classes do Flecha

V.3.1. Classe de Aplicação

i) **DOBApp**: Inicializa o sistema

V.3.2. Classes de janelas

- i) **DOB**: É a janela mãe. Manipula os cardápios, todos os diálogos, cria e inicializa as janelas filhas: **Diag** e **FunçãoDOB**.
- ii) **Diag**: Onde são desenhados os objetos do diagrama. Ele permite o usuário criar, selecionar, mover, modificar, alinhar, cortar e copiar objetos. Mostra o diagrama no tamanho real, no tamanho da janela ou em *zoom*. Move o diagrama horizontalmente e verticalmente (*scroll*). Ele salva, abre e imprime o diagrama.
- iii) **FunçãoDOB**: É a classe que posiciona e mostra os ícones que representam as funções do Flecha. Ela também notifica à janela mãe sobre qual foi a seleção feita pelo usuário.

V.3.3. Classes de Diálogo

- i) **Diálogo**: Corresponde aos diálogos mais simples do editor.
- ii) **DimensãoDlg**: Especifica o espaço gráfico.
- iii) **Rel** e **Re2**: Responde aos comandos do Relacionamento.

iv) **Senha:** Pede o nome e a senha do usuário.

v) **VarInst:** Diálogo para a função Variável de Instância.

vi) **Div1, Div2 e Div3:** Primeiro, segundo e terceiro diálogo da divisão de contextos.

V.3.4. Classe da Figura

i) **Figura:** Corresponde ao desenho do objeto. Acrescenta à classe Picture, já existente, variáveis de instância particulares a esses objetos.

V.3.5. Classe de Registro

i) **Contexto:** Registro do arquivo Contexto

V.4. Futuras Implementações

Nesta seção, são apresentadas algumas idéias que contribuirão para uma versão mais sofisticada da ferramenta Flecha.

A adaptação do comentário gráfico à nova versão seria parecida com o comentário tipo texto. Primeiro, seria feito uma seleção dos objetos que seriam comentados, esses objetos apareceriam numa outra janela, que teria as mesmas funções da tela principal do Flecha. Daí, o co-modelador poderia acrescentar

objetos, relacionamentos, etc., tal qual é feito normalmente no editor. Estas alterações apareceriam em outra cor ressaltando estes comentários.

Outras funções para fazer comentários poderiam, também, estar disponíveis como, por exemplo, a possibilidade de escrever um texto em qualquer parte do diagrama e desenhar setas e círculos que ajudariam na sua revisão.

Todo comentário, textual ou gráfico, poderia ser manipulado em forma de arquivo: imprimindo, guardando e recuperando. A manipulação de comentários gráficos poderia ser feita, também, na incorporação do mesmo ao diagrama ou contexto corrente.

A inserção de arquivo seria um comando simples que teria como função incorporar um outro diagrama, ou parte dele, no diagrama corrente.

A próxima versão do Flecha poderia incluir o desenho automático do relacionamento entre dois objetos. Depois de indicar a origem e o destino do relacionamento, o Flecha escolheria o melhor caminho entre esses dois pontos e o traçaria automaticamente.

Devido à sua característica cooperativa, o Flecha precisa, constantemente, rearrumar as linhas de relacionamento e herança na integração de contextos. Nesta fase, uma função para arrumar relacionamentos e heranças seria bastante útil para o gerente.

A comunicação com o PROTOGEO seria o item mais importante e complexo. Sendo necessário um estudo maior de comunicação entre essas duas partes: o Flecha e o PROTOGEO.

O Flecha, inicialmente, foi implementado utilizando arquivos para fazer a comunicação entre duas máquinas conectadas pela rede local. A sugestão para uma nova versão do Flecha seria a utilização das próprias rotinas da rede Novell, junto com o ambiente Windows. Desta forma, o Flecha se tornaria mais rápido e o processo cooperativo mais eficiente.

Mais facilidades na fase de integração de contextos seria uma outra sugestão. Como exemplo de um inconveniente existente na atual versão da ferramenta, tem-se a sobreposição de objetos criados em diferentes contextos. Isto seria resolvido limitando-se o espaço de um contexto, ou seja, seria sugerido um espaço que estaria livre até aquele momento. Numa página, o espaço ocupado por objetos de outros contextos apareceria em forma de uma sombra cinza. Assim, o co-modelador poderia evitar esta área, se assim desejasse.

Finalmente, a próxima versão do Flecha poderia facilitar o acesso a diagramas, contextos e até comentários dentro do histórico, tornando cada vez mais rápida a sua atualização.

VI. CONCLUSÕES

Este trabalho teve como objetivo pesquisar aspectos sobre as características e o funcionamento de editores cooperativos e ferramentas de suporte à edição cooperativa, com a finalidade de propor e desenvolver um protótipo de um editor gráfico cooperativo.

O Flecha foi planejado para participar do conjunto de ferramentas cooperativas da Estação de Desenvolvimento de *Software* - TABA, como um editor gráfico cooperativo para o Modelo de Objetos do GEOTABA.

Dada a falta de artigos na bibliografia, que mostrassem a experiência na construção de editores gráficos cooperativos, o Flecha veio agregar as características de editores de texto a uma ferramenta gráfica.

Destas características, algumas foram, imediatamente transportadas, contudo outras tiveram que ser bastante trabalhadas como, por exemplo, o compartilhamento de figuras e divisão e integração de contextos. Este foi o desafio principal da tese, ou seja, tornar um editor gráfico como o Flecha, cooperativo. Outro ponto importante foi tentar solucionar dificuldades relacionadas a edição gráfica cooperativa, sem prejudicar sua *interface*, nem sua facilidade de manipulação e de aprendizado.

O ambiente de programação escolhido para a implementação da ferramenta foi o ambiente **Actor** (versão 3.0). Deste modo, com a *interface* e as classes gráficas do **Actor**, foi fácil diminuir o tempo de implementação do editor, propriamente dito, ou seja, o editor sem as características cooperativas. O tempo gasto nesta fase foi grande, contudo sem ele seria impossível a aplicação de cooperação.

REFERÊNCIAS BIBLIOGRÁFICAS

-
- [Ahuj88] Ahuja, S. R.; Ensor, J. R.; Horn, D. N.; *The Rapport Multimedia Conferencing System*; ACM SIGOIS Bulletin; V.9 N.2&3; Mar 1988; pp 1-8
- [Bane87] Banerjee, J.; Chou, H. T.; Garza, J. F.; Kim, W.; Woelk, D.; Ballou, N.; Kim, H. J.; *Data model issues for object-oriented applications*; ACM Transactions on Office Information Systems; V5; 1987; pp 3-26
- [Bear90] Beard, D.; Palaniappan, M.; *A Visual Calendar for Scheduling Group Meetings*; Proceedings of the Third Conference on Computer Supported Cooperative Work (Los Angeles, California); Out 1990; pp 279-290
- [Birm89] Birman, K. P.; Cooper, R.; Joseph, T.; Kane, K.; Schmuck, F.; *The ISIS System Manual*; Jun 1989
- [Bly 90] Bly, S. A.; Minneman, S. L.; *Commune: A Shared Drawing Surface*; Conference of Office Information Systems (Cambridge, Massachusetts); SIGOIS BULLETIN; V.11 N.2&3; Abr 1990; pp 184-192
- [Bull90] Bullen, C. V.; Benett, J. L.; *Learning from User Experience with Groupware*; Proceedings of the Third Conference on Computer Supported Cooperative Work (Los Angeles, California); Out 1990; pp 291-302
- [Cama92] Camargo, C. S. P.; Souza, J. M.; *Flecha: A Cooperative Graphic Editor for the Object Data Model of GEOTABA*; XII International Conference of the Chilean Computer Science Society; Santiago, Chile; Out 1992; pp 195-206
- [Catl89] Catlin, T.; Bush, P.; Yankelovich, N.; *Internote: Extending a Hypermedia Framework to Support Annotative Collaboration*; Hypertext'90 Proceedings, ACM Pittsburg, PA; Nov 1989; pp 365-378
- [Chen76] Chen, P. P.; *The Entity-Relationship Model Toward a Unified View of Data*; ACM Transactions on Database System; V1 N1; mar 1976; pp 9-36
- [Clem90] Clement, A.; *Cooperative Support for Computer Work: A Social Perspective on the Empowering of End Users*; Proceedings of the Third Conference on Computer Supported Cooperative Work (Los Angeles, California); Out 1990; pp 223- 236

- [Cook87] Cook, P.; Ellis, C.; Graf, M.; Reif, G. Smith, T.; *Project Nick: Meetings Augmentation and Analysis*; ACM Transactions on Office Information Systems; V.5 N.2; Abr 1987; pp 132-146
- [Cros88] Crowston, K.; Malone, T. W.; *Intelligent Software Agents*; Byte; V.13 N.13; Dez 1988; pp 267-282
- [Crow90] Crowley, T.; Milazzo, P.; Baker, E.; Forsdick, H. Tomlinson, R.; *MMCONF: An Infrastructure for Building Shared Multimedia Applications*; Proceedings of the Third Conference on Computer Supported Cooperative Work (Los Angeles, California); Out 1990; pp 329-342
- [Elli91] Ellis, C. A.; Gibbs, S. J.; Rein, G. L.; *Groupware: Some Issues and Experiences*; Communications of the ACM; V.34 N.1; Jan 1991; pp 39-58
- [Enge87] Engelbart, D. C.; *Authorship Provisions in Augment*; in Computer-supported Cooperative Work: A Book of Readings edited by Irene Greif; Morgan Kaufmann Publishers, Inc.; 1987; pp 107-126
- [Enge88] Engelbart, D.; Lehtman, H.; *Working Together*; Byte; V.13 N.13; Dez 1988; pp 245-252
- [Feld87] Feldman, M. S.; *Electronic Mail and Weak Ties in Organizations*; Office: Technology and People; V.3 N.2; Ago 1987; pp 83-101
- [Fish88] Fish, R. S.; Kraut, R. E.; Leland, M. D. P.; Cohen, M.; *Quilt: A Collaborative Tool for Cooperative Writing*; ACM SIGOIS Bulletin; V.9 N.2&3; Mar 1988; pp 30-37
- [Fran90] Franz, M.; *Object-oriented Programming, Featuring Actor*; Scott, Foresman and Company, Glenview, Illinois, London, 1990
- [Gale90] Galegher, J.; Kraut, R. E.; *Computer-mediated Communication for Intellectual Teamwork: A Field Experiment in Group Writing*; Proceedings of the Third Conference on Computer Supported Cooperative Work (Los Angeles, California); Out 1990; pp 65-78
- [Garr86] Garret, L.; Smith, K.; Meyrowitz, N.; *Intermedia: Issues, Strategies and Factias in the Design of a Hypermedia Document System*; Proceedings of the Conference on Computer-supported Cooperative Work; Austin, Texas; Dez 1986;
- [Gold84] GoldBerg, A.; *Smalltalk-80: The Interactive Programming Environment*; Addison-Wesley; 1984
- [Good87] Goodman, G. O.; Abel, M. J.; *Communication and Collaboration Facilitating Cooperative Work Through Communication*; Office: Technology and People; V.3 N.2; Ago 1987; pp 129-145
- [Grei87a] Greif, I.; *Overview*; in Computer-supported Cooperative Work: A Book of Readings edited by Irene Greif, Morgan Kaufmann Publishers, Inc. 1987; pp 5-12
- [Grei87b] Greif, I.; Sarin, S.; *Data Sharing in Group Work*; in Computer-supported Cooperative Work: A Book of Readings edited by Irene Greif, Morgan Kaufmann Publishers, Inc. 1987; pp 477-508
- [Grud88] Grudin, J.; *Perils and Pitfalls*; Byte; V.13 N.13; Dez 1988; pp 261-264

- [Grud89] Grudin, J.; *Why Groupware Applications Fail: Problems in Design and Evaluation*; Office: Technology and People; V.4 N.3; 1989; pp 245-264
- [Hilt81] Hiltz, S. R.; Turoff, M.; *The Evolution of User Behavior in Computerized Conferencing System*; Communications of the ACM; V.24 N.11; Nov 1981; pp 739-751
- [Hilt85] Hiltz, S. R.; Turoff, M.; *Structuring Computer-mediated Communication Systems to Avoid Information Overload*; Communications of the ACM; V.28 N.7; Jul 1985; pp 680-689
- [Horn91] Hornung, C.; *Cooperative Hypermedia Systems and Applications*; VI Simpósio Brasileiro de Computação Gráfica e Processamento de Imagens; Jul 1991
- [Karb90] Karbe, B.; Ramsperger, N.; Weiss, P.; *Support of Cooperative Work by Electronic Circulation Folders*; Conference on Office Information Systems ACM Press; Abr 1990; pp 109-117
- [Kies87] Kiesler, S.; Siegel, J.; McGuire, T. W.; *Social Psychological Aspects of Computer-mediated Communication*; in *Computer-supported Cooperative Work: A book of Readings* Edited by Irene Greif; Morgan Kaufmann Publishers, Inc.; 1987; pp 657-682
- [Knis90] Knister, M. J.; Prakash, A.; *DistEdit: A Distributed Toolkit for Supporting Multiple Group Editors*; Proceedings of the Third Conference on Computer Supported Cooperative Work (Los Angeles, California); Out 1990; pp 343-355
- [Laki87] Lakin, F.; *A Performing Medium For Working Group Graphic*; in *Computer-supported Cooperative Work: A book of Readings* Edited by Irene Greif, Morgan Kaufmann Publishers, Inc 1987; pp 367-396
- [Lewi88] Lewis, B.T.; *Shared Books: Collaborative Publication Management for an Office Information System*; ACM SIGOIS Bulletin; V.9 N.2&3; Mar 1988; pp 197-204
- [Lisb90] Lisboa, J.; Monte, L. C. M.; *O Nível Lógico do GEOTABA*; Workshop de Banco de Dados não Convencionais; COPPE/UFRJ; Programa de Engenharia de Sistemas e Computação; Jul 1990; pp 18-21
- [Mads89] Madsen, C. M.; *Using Persistent Objects to Implement an Environment for Cooperative Work*; ACM SIGOIS Bulletin; V.10 N.4; Dez 1989; pp 11-20
- [Malo90] Malone, T. W.; Crowston, K.; *What is Coordination Theory and How Can It Help Design Cooperative Work Systems?*; Proceedings of the Third Conference on Computer Supported Cooperative Work (Los Angeles, California); Out 1990; pp 357-370
- [Mark90] Markus, M. L.; Cronolly, T.; *Why CSCW Applications Fail: Problems in the Adoption of Interdependent Work Tool*; Proceedings of the Third Conference on Computer Supported Cooperative Work (Los Angeles, California); Out 1990; pp 371-380

- [Matt89] Mattoso, M. L. Q.; Souza, J. M.; Gonçalves, L.; Degrazia, C.; Rossatto, M.; *GEOTABA: O Sistema de Gerência de Objetos da Estação TABA*; Relatório Técnico ES-179/88 do Programa de Engenharia de Sistemas e Computação - COPPE/UFRJ; 1989
- [Mend92] Mendes, E. X.; *HIPPEA: Um Hipertexto Cooperativo Educacional*; Tese de Mestrado em desenvolvimento na COPPE Sistemas/UFRJ; 1992
- [Micr87] Microsoft Corporation; *Microsoft Windows User's Guide*; Washington; U.S.A.; 1987
- [Mont90] Monte, L. C. M.; *O Modelo de Objetos do GEOTABA*; Exame de Qualificação para Doutorado em Engenharia de Sistemas e Computação; COPPE/UFRJ; Rio de Janeiro; 1990
- [Mont91] Monte, L. C. M.; *Uma Representação Gráfica para o Modelo de Objetos do GEOTABA*; XVII Conferência Latinoamericana de Informática (Caracas, Venezuela); Jul 1991; pp 697-711
- [Mont92] Monte, L. C. M.; Filho, J. L.; Mattoso, M. L. Q.; Souza, J. M.; *Contribuição do PROTOGEO ao projeto do SGBDOO GEOTABA*; 7o. Simpósio Brasileiro de Banco de Dados; Porto Alegre; Mai 1992; pp 125-139
- [Mora90] Moran, T. P.; Anderson, R. J.; *The Workaday World as a Paradigm for CSCW Design*; Proceedings of the Third Conference on Computer Supported Cooperative Work (Los Angeles, California); Out 1990; pp 381-393
- [Neuw90] Neuwirth, C. M. ; Kaufer, D. S. ; Chandhok, R.; Morris, J. H.; *Issues in the Design of Computer Support for Co-authoring and Commenting*; Proceedings of the Third Conference on Computer Supported Cooperative Work (Los Angeles, California); Out 1990; pp 183-195
- [Nove88] Novell Inc.; *NetWare User's Guide*; Utah, U.S.A.; 1988
- [Nuna91] Nunamaker, J. F.; Dennis, A. R.; Valacich, J. S.; Vogel, D. R.; George, J. F.; *Electronic Meeting Systems to Support Group Work*; Communciations of the ACM; V. 34 N. 7; Jul 1991; pp 40-61
- [Olso87] Olson, M. H.; *Computer-supported Cooperative Work*; Office: Technology and People; V.3 N.2; Ago 1987; pp 77-81
- [Oppe88] Opper, S.; *A Groupware Toolbox*; Byte; V.13 N.13; Dez 1988; pp 275-282
- [Plon89] Plonski, G. A.; Sznifer, M.; *Groupware: Trabalho em Equipe Facilitado pela Informática*; XXII Congresso Nacional de Informática, SUCESU; São Paulo; Set 1989; pp 798-802
- [Roch91] Rocha, A. R. C.; Souza, J. M.; *O Projeto TABA*; Relatório Técnico TABA RJ-7/91; COPPE/UFRJ; Programa de Engenharia de Sistemas e Computação; Jul 1991
- [Sche86] Scheifler, R. W.; Gettys, J.; *The X Window System*; ACM Transactions on Computer Graphics; V5; 1986; pp 79-109

- [Stef87a] Stefik, M.; Foster, G.; Bobrow, D. G.; Kahn, K.; Lanning, S.; Suchman, L.; *Beyond the Chalkboard: Computer Support for Collaboration and Problem Solving in Meetings*; in *Computer-supported Cooperative Work: A Book of Readings* edited by Irene Greif, Morgan Kaufmann Publishers, Inc. 1987; pp 335-366
- [Stef87b] Stefik, M.; Bobrow, D. G.; Foster, G.; Lanning, S.; Tatar, D.; *WYSWIS Revised: Early Experiences with Multiuser Interfaces*; ACM Transactions on Office Information Systems; V.5 N.2; Abr 1987; pp 147-167
- [Tang91] Tang, J. C.; Minneman, S. L.; *Video Draw: A Video Interface For Collaborative Drawing*; ACM Transactions on Information Systems; V.9 N.2; Abr91; pp 170-184
- [Torn92] Tornaghi, A.; *MULEC: Multi Editor Livre para Ensino Cooperativo*; Tese de Mestrado em desenvolvimento na COPPE Sistemas/UFRJ; 1992
- [Vaz 92] Vaz, M. A.; *Dueto: Um Editor Cooperativo de Textos Técnicos*; Tese de Mestrado em desenvolvimento na COPPE Sistemas/UFRJ; 1992
- [Whit91] The Whitewater Group, Inc.; *Actor User's Manual*; Evanston, U.S.A.; 1991
- [Wino87] Winograd, T.; *A language/Action Perspective on the Design of Cooperative Work*; in *Computer-supported Cooperative Work: A book of Readings* Edited by Irene Greif, Morgan Kaufmann Publishers, Inc 1987; pp 623-653
- [Wino88] Winograd, T.; *Where the Action Is*; Byte; V.13 N.13; Dez 1988; pp 256-258