


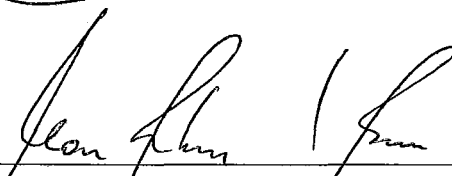
# DUETO: UM EDITOR COOPERATIVO DE TEXTOS TÉCNICOS.

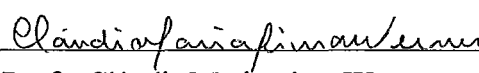
Marco Antonio Aniceto Vaz

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

Aprovada por:

  
\_\_\_\_\_  
Prof. Jairo Moreira de Souza, Ph.D.  
(Presidente)

  
\_\_\_\_\_  
Prof. Marcos Roberto da Silva Borges, Ph.D.

  
\_\_\_\_\_  
Prof. Cláudia Maria Lima Werner, D.Sc.

RIO DE JANEIRO, RJ - BRASIL

JULHO DE 1993

VAZ, Marco Antonio Aniceto

DUETO: Um Editor Cooperativo de Textos técnicos [Rio de Janeiro] 1993 vii, 147p.  
29,7 cm (COPPE/UFRJ, M.Sc., Engenharia de Sistemas e Computação, 1993)

Tese: Universidade Federal do Rio de Janeiro, COPPE

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

Para Valéria, Igor e Gabriel.

## **Agradecimentos:**

- . Agradeço à Valéria, Igor e Gabriel pela paciência(!!), força e pela compreensão nas longas ausências.
- . Agradeço à minha mãe e meu pai (In Memoriam) pela força e amor que sempre me dedicaram.
- . Agradeço ao Prof. Jano pela escolha do tema, orientação e, acima de tudo, pela paciência e compreensão.
- . Agradeço aos profs. Marcos R. da Silva Borges e Cláudia M. Lima Werner pela honra de participarem da minha banca de tese.
- . Agradeço à C.N.E.N. pela oportunidade e apoio.
- . Agradeço aos colegas da Superintendência de Informática da C.N.E.N. pela oportunidade, pela força e preocupação nos momentos mais difíceis.
- . Agradeço às bibliotecárias do C.I.N, em especial, Odete, Sonia, Eliane e Jane pela força e profissionalismo na busca de Informações, pois sem elas a pesquisa bibliográfica seria bem mais demorada.
- . Agradeço ao Galvão e a Anna Christina (Babi) pelo empréstimo do microcomputador, pois sem ele talvez não teria conseguido.
- . Agradeço ao Monnerat, Beth, Gilda e, mais uma vez, à Babi pela revisão do texto.
- . Agradeço ao Carlos B. Rocha pela editoração da tese.

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

DUETO: Um Editor Cooperativo de Textos Técnicos.

Marco Antonio Aniceto Vaz.

Julho de 1993

Orientador: Prof. Jano Moreira de Souza

Programa: Engenharia de Sistemas e Computação

Apresenta-se o **DUETO** - **DU**plo Editor de textos **T**ecnicamente **O**rientado, desenvolvido a partir de estudos realizados sobre os ambientes de trabalho cooperativo apoiado por computador (TCAC).

O **DUETO** é uma ferramenta cooperativa, usada para edição de textos técnicos, e possui características que permitem auxiliar grupos de autores nas tarefas de criação (autoria) e edição de textos técnicos em ambientes cooperativo. Algumas das características implementadas são baseadas nas características apresentadas pelos sistemas de hipertextos. O editor implementa facilidades que permitem o compartilhamento de um documento, a navegação não sequencial a partir da estrutura do documento, facilidades para a interação e troca de informações entre os diversos participantes, através de mensagens e da incorporação de anotações nos componentes do documento.

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

**DUETO: A Cooperative Editor for Technical Documents**

**Marco Antonio Aniceto Vaz**

July, 1993

Thesis Supervisor: Prof. Jano Moreira de Souza

Department: System Engineering and Computer Science

**DUETO** - A Technical Oriented Dual Editor, based on computer supported cooperative work (CSCW) environment is presented. **DUETO** is a cooperative tool used for editing technical texts with special features to aid writing groups in both creating (authorship) and editing tasks in a collaborative environments. The editor allows - document sharing, non-sequential navigation along the document structure, and interaction and exchange of information among different participants through the exchange of message and the incorporation of notes to the document components. Some of these implemented features are based on hypertext systems.

## ÍNDICE

<b>I - INTRODUÇÃO</b>	<b>1</b>
<b>II - AUTOMAÇÃO DE ESCRITÓRIOS</b>	<b>5</b>
<b>II.1 - Abordagem Geral</b>	<b>5</b>
<b>II.2 - Categorias dos Sistemas de AE</b>	<b>6</b>
<b>II.2.1 - Quanto às Atividades Desenvolvidas</b>	<b>6</b>
<b>II.2.2 - Quanto às Necessidades Individuais</b>	<b>7</b>
<b>II.3 - Integração dos Sistemas de AE</b>	<b>7</b>
<b>II.3.1 - Ambientes de Processamento distribuído</b>	<b>8</b>
<b>II.3.2 - Trabalho Cooperativo Apoiado por Computador (TCAC)</b>	<b>10</b>
<b>II.3.3 - Categorias de Sistemas de Groupware</b>	<b>13</b>
<b>II.3.3.1 - Sistemas de Mensagens</b>	<b>13</b>
<b>II.3.3.2 - Editores Cooperativos</b>	<b>14</b>
<b>II.3.3.3 - Sistemas de Suporte à Decisão em Grupos e Sala de Reuniões Eletrônica</b>	<b>14</b>
<b>II.3.3.4 - Conferências por Computador</b>	<b>15</b>
<b>II.3.3.5 - Agentes Inteligentes</b>	<b>15</b>
<b>II.3.3.6 - Sistemas de Coordenação</b>	<b>15</b>

<b>II.4 - Detalhes de Implementação em Sistemas de Groupware</b>	<b>16</b>
<b>II.4.1 - Interface para Trabalhos Cooperativo</b>	<b>17</b>
<b>II.4.2- Coordenação das Atividades</b>	<b>19</b>
<b>II.4.2.1 - Contrôles de Acesso</b>	<b>20</b>
<b>II.4.2.2 - Contrôles de Concorrência</b>	<b>21</b>
<b>II.4.3 - Contrôles de Versões</b>	<b>24</b>
<b>III - SISTEMAS DE EDIÇÃO DE TEXTOS</b>	<b>27</b>
<b>III.1 - Características</b>	<b>27</b>
<b>III.2 - Classificação dos Sistemas de Edição de Textos</b>	<b>30</b>
<b>III.2.1 - Editores orientados à Linha</b>	<b>30</b>
<b>III.2.2 - Editores de Edição Contínua (Streams Editors)</b>	<b>31</b>
<b>III.2.3 - Editores de Visualização Completa (Display Editors)</b>	<b>31</b>
<b>III.2.4 - Editores/Formatadores Gráficos Interativos</b>	<b>32</b>
<b>III.2.5 - Editores de Estrutura de Propósito Geral</b>	<b>32</b>
<b>III.2.6 - Editores Orientados à Sintaxe</b>	<b>33</b>
<b>III.2.7 - Processadores de Textos</b>	<b>34</b>
<b>III.2.8 - Processadores de Textos com Interfaces Gráficas</b>	<b>35</b>
<b>III.2.9 - Ambientes Integrados</b>	<b>36</b>



III.2.10 - Sistemas de Edição de Textos para TCAC	36
III.3 - Exemplos de Editores Cooperativos	38
III.3.1 - CES ( Collaborative Document Editing )	38
III.3.2 - VAX ACCORD	40
III.3.3 - Shared Books	41
III.3.4 - DistEdit	43
III.3.5 - GROVE (GRoup Outline Viewing Editor)	45
III.3.6 - MACE	46
III.3.7 - ShrEdit	48
III.4 - Resumo das Características dos Editores Cooperativos	49
IV - SISTEMAS DE HIPERTEXTOS.	50
IV.1 - Características	50
IV.2 - Navegação em Hipertextos	51
IV.3 - Hipertextos como Suporte a Edição Cooperativa	54
IV.3.1 - Características	54
IV.3.2 - Exemplos de Hipertextos para Edição Cooperativa	55
IV.3.2.1 - KMS	56
IV.3.2.2 - Notecards	57

IV.3.2.3 - Neptune	58
IV.3.2.4 - Quilt	60
IV.3.2.5 - Internote	61
IV.3.2.6 - Co-Author	63
IV.3.2.7 - Prep (Work in Preparation)	65
IV.3.2.8 - GroupWriter	66
IV.3.2.9 - SEPIA	68
IV.3.3 - Resumo das Características dos Hipertextos para Edição Cooperativa	70
<b>V - DUETO: UMA FERRAMENTA PARA EDIÇÃO COOPERATIVA DE TEXTOS TÉCNICOS</b>	<b>71</b>
V.1 - Introdução	71
V.2 - Especificação da Ferramenta	72
V.2.1 - Requisitos Funcionais do DUETO	72
V.2.2 - Modelo de Dados	83
V.3 - Público-Alvo	86
<b>VI - IMPLEMENTAÇÃO DO EDITOR DUETO</b>	<b>87</b>
VI.1 - Características do DUETO	87
VI.2 - Interface do DUETO	91

<b>VI.3 - Recursos Computacionais Utilizados</b>	<b>124</b>
<b>VI.4 - Arquitetura do DUETO</b>	<b>125</b>
<b>VI.5 - Armazenamento</b>	<b>126</b>
<b>VI.6 - Exemplos de Utilização</b>	<b>128</b>
<b>VII - CONCLUSÕES</b>	<b>136</b>
<b>BIBLIOGRAFIA</b>	<b>139</b>

## ÍNDICE DE FIGURAS

<b>Número</b>	<b>Título</b>	<b>Página</b>
III.4.1	Características dos Editores cooperativos	49
IV.1.1	Visão da estrutura de um Hipertexto e suas Ligações	50
V.3.3.1	Características dos Hipertextos para edição cooperativa	70
V.2.2.1	Modelo de Dados do DUETO	83
VI.2.1	Tela Principal do Editor DUETO	91
VI.2.2	Alteração de tamanho da área de edição	93
VI.2.3	Cardápio da opção Documento	94
VI.2.4	Tela para Criar Documento	95
VI.2.5	Tela de Abertura de um documento	96
VI.2.6	Tela com informações do documento	97
VI.2.7	Documento aberto no DUETO	97
VI.2.8	Tela da opção Documento/Fechar	98
VI.2.9	Tela da opção Documento/Exportar	99
VI.2.10	Cardápio da opção Edição	100
VI.2.11	Cardápio da opção Localizar	102
VI.2.12	Tela da opção Localizar/Procurar	103
VI.2.13	Tela da opção Localizar/Trocar	104
VI.2.14	Cardápio da opção Formatar	105
VI.2.15	Cardápio da opção Componente	106
VI.2.16	Tela de criação de um componente(Irmão)	108

VI.2.17	Visualização da estrutura com o componente criado	108
VI.2.18	Criação de um componente de nível hierárquico inferior (filho)	109
VI.2.19	Visualização da estrutura com o componente criado (movimentação à esquerda)	109
VI.2.20	Opção Componente/Editar	110
VI.2.21	Tela 1 da opção Componente/Apagar	111
VI.2.22	Tela 2 da opção Componente/Apagar	112
VI.2.23	Tela de inclusão de anotação	113
VI.2.24	Tela para recuperação das anotações ligadas a um componente	114
VI.2.25	Opção Componente/Conteúdo	115
VI.2.26	Cardápio da opção Utilitários	116
VI.2.27	Opção Utilitários/Autores	117
VI.2.28	Opção Utilitários/Listar (Participantes acessando o documento)	118
VI.2.29	Opção Utilitários/Mensagem(Envio de Mensagem)	119
VI.2.30	Tela da opção Utilitários/Trilha/Criar	120
VI.2.31	Tela da opção Utilitários/Contexto/Criar	121
VI.2.32	Cardápio da opção Tipos	123
VI.4.1	Arquitetura do DUETO	125
VI.6.1	Criação de um documento com os primeiros componentes propostos	129
VI.6.2	Usuário já definido pelo criador tentando acessar documento não liberado	129
VI.6.3	Recepção de mensagem com informação sobre a estrutura do documento	130

VI.6.4	Recepção de mensagem com aviso de movimentação de componente	130
VI.6.5	Tela que informação de bloqueio do componente a ser movido	131
VI.6.6	Tela para escolha de um novo componente raiz	131
VI.6.7	Tela com informação da movimentação do componente	132
VI.6.8	Acesso a um componente bloqueado por outro usuário (opção Componente/Editar)	132
VI.6.9	Visualização de um componente editado por outro usuário (opção Conteúdo)	133
VI.6.10	Inserção de comentários em um componente	133
VI.6.11	Inclusão do comentário no texto do componente	134
VI.6.12	Tela de confirmação para continuidade da função Configuração	134
VI.6.13	Informação da criação de uma nova configuração do documento	135

## I - INTRODUÇÃO

Os editores de textos interativos talvez tenham sido as primeiras ferramentas a entrarem no ambiente de escritórios, como parte do processo de automação. As facilidades apresentadas por estas ferramentas na manipulação de textos, permitiram um rápido avanço desta tecnologia, fazendo com que o pessoal de escritório envolvido diretamente com as tarefas de edição migrasse rapidamente das máquinas de escrever para os microcomputadores.

Estas ferramentas, originalmente desenvolvidas como um meio para manipulação de textos, foram estendidas para permitir a inclusão de objetos gráficos tais como: tabelas, gráficos, ilustrações, fotografias e mais recentemente animação e sequências de vídeo. Com isso, os editores tornaram-se ferramentas que permitem a produção de documentos bem mais elaborados, permitindo assim que outras pessoas (gerentes, técnicos, etc.) começassem a utilizar os editores de textos como ferramentas interativas para criação de documentos.

Mas, apesar de todas as facilidades oferecidas, os editores de textos convencionais não são boas ferramentas no auxílio à criação e estruturação de idéias (Autoria) para elaboração de documentos, pois não oferecem facilidades para permitir a exploração, organização e composição de textos, características indispensáveis para autoria de documentos.

O processo de autoria (Edição de Documentos) pode ser visto como uma atividade a nível de organização e estruturação de sentenças e palavras, mas poucas pessoas conseguem sentar-se à frente de um editor de textos e criar um texto acabado.

A mente humana não aprende nem age por meio de grandes saltos, mas através de pequenos passos organizados e estruturados, de tal maneira que cada passo dependa do anterior, construindo uma verdadeira hierarquia de processos ou ações.

Cada indivíduo desenvolve um certo repertório de capacidades de processos dos quais ele seleciona e adapta, utilizando métodos e estratégias próprias, aqueles que comporão o processo que está sendo executado. Este repertório representa uma estrutura integrada e hierárquica de processos [Eng88a].

A atividade do pensamento é um processo constante de criação, desenvolvimento e rejeição de idéias que ocorrem em diferentes níveis e pontos, cada uma dependendo e contribuindo para as outras, formando uma rede de idéias.

Os sistemas que enfatizam o processo de autoria permitem ampliar a capacidade de criação de novas idéias a partir da estruturação (análise) e combinação (síntese) de idéias pré-existentes, implícitas ou não. As pessoas não pensam em termos de telas de computador cheias de informação, elas pensam em termos de idéias, fatos e evidências, no sentido de construir novas relações.

As ferramentas de autoria, implementadas para ambientes cooperativos, podem oferecer melhores facilidades para transmissão das mensagens e intenções do autor. A integração de informações adicionais sobre as intenções do autor, juntamente com a estrutura do conhecimento, facilitam o processamento compreensivo por parte dos leitores.

O processo de autoria de documentos técnicos pode ser considerado como uma forma de engenharia - Engenharia de Documentos. A aplicação de princípios de engenharia de software podem ser úteis no processo de autoria de documentos, principalmente nas áreas de autoria cooperativa e na manutenção de documentos.

A falta de disciplina na elaboração de documentos (autoria), em editores de textos que não permitem a estruturação dos documentos, pode ser comparada à criação de software nas décadas de 50 e 60, onde programas (códigos "Spaghetti") eram gerados desordenadamente e eram difíceis de serem mantidos.

Em ambientes de trabalho que envolvem várias pessoas trabalhando cooperativamente, tanto o desenvolvimento de software quanto o de documentos exigem um meio para a coordenação das tarefas entre os desenvolvedores trabalhando em locais diferentes. Como forma de aumentar a qualidade da produção, deve-se lançar mão de outras tecnologias que forneçam meios para coordenar as atividades ( p.ex. sistemas de mensagens, catálogos de componentes reusáveis etc.).

O conceito de modularização é tão importante para a engenharia de documentos quanto é importante para a engenharia de software. No desenvolvimento de software, a modularidade e abstração são largamente reconhecidas como fundamentais para gerenciar a complexidade dos programas [Walk88b].



Em ambientes de trabalho cooperativo, a modularização aumenta a cooperação entre os autores, porque os módulos podem ser utilizados para decompor o documento em partes independentes e dividi-lo entre os diversos autores.

A evolução da tecnologia da comunicação e suas aplicações permitiu a interligação de microcomputadores para compartilhamento de recursos e troca de informações, formando as chamadas redes de computadores, base para os sistemas de suporte a trabalhos cooperativos. Trabalho cooperativo apoiado por computador (TCAC) é um campo de pesquisas emergente, que se interessa pelo papel do computador para trabalhos em grupo.

A idéia da introdução dos sistemas de trabalho cooperativo apoiado por computador não tem como objetivo fazer com que os membros do grupo abandonem o seu trabalho individual, mas dispor de meios que permitam que a divisão entre o trabalho individual e de grupo se tornem transparentes.

Entre as pesquisas mais importantes na área da ciência da computação encontram-se os sistemas de hipertextos. Um sistema de hipertexto é uma ferramenta utilizada para organização e recuperação da informação de forma não convencional, que permite a “navegação” ao longo dos textos, seja na criação ou leitura, da forma mais livre possível.

O advento dos sistemas de hipertextos permitiu aliar as facilidades da edição de textos com o processo de autoria, provendo a ampliação da capacidade de gerar novas idéias. Por isso, esses sistemas podem apresentar-se como editores de textos com características mais flexíveis, facilitando o processo de criação de documentos.

Apesar dos sistemas de hipertextos terem sido projetados para servirem como ferramentas que viabilizam novos meios de estruturação de textos, poucos permitem a colaboração dos autores sobre o texto que estão escrevendo.

Portanto, um editor de textos cooperativo pode ajudar no processo de criação de documentos com multi-autoria, representando uma importante ferramenta para trabalhos cooperativos sobre uma rede de computadores, permitindo assim que os diversos participantes (autores) possam interagir simultaneamente, à distância, na criação e composição de documentos.

O presente trabalho tem como objetivo pesquisar as principais características dos ambientes para trabalhos cooperativos apoiados por computadores, com destaque para as ferramentas de suporte à edição cooperativa de textos, no sentido de propor e implementar uma ferramenta protótipo para edição cooperativa de textos técnicos - **DUETO** (**DU**plo **E**ditor **T**écnicamente **O**rientado).

O editor **DUETO** auxilia os co-autores no processo de criação e estruturação de idéias (Autoria) para elaboração de documentos, em ambientes de trabalho cooperativo, permitindo estruturar o texto de forma que a recuperação possa ser efetuada de uma maneira não-sequencial - característica indispensável no processo de autoria de documentos.

Além disso, a ferramenta permitirá a cooperação concorrente entre os diversos participantes, independente da sua localização física e deverá também fornecer meios à comunicação entre os co-autores de um documento, seja através de comentários inseridos no texto do documento, seja através de mensagens diretas.

Esta dissertação está sub-dividida em 7 (sete) capítulos. O capítulo I apresenta, na sua introdução, a motivação que deu origem ao presente tema, descrevendo, de forma sucinta, os objetivos e a organização da mesma. O capítulo II apresenta uma análise do processo de automação de escritórios com ênfase nos ambientes de trabalho cooperativo apoiado por computador (TCAC), com o objetivo de contextualizar o tema e examinar algumas informações relevantes dessa área. O capítulo III examina a evolução das ferramentas para edição de textos e apresenta alguns exemplos de editores de textos convencionais voltados para o trabalho cooperativo. O capítulo IV passa a tratar do tema hipertextos, descrevendo sucintamente algumas características destas ferramentas com objetivo de posicionar o leitor no tema. Em seguida, passa a tratar da utilização dos sistemas de hipertextos no processo de autoria de documentos, em especial no ambiente de multi-autoria, com a apresentação de alguns exemplos de sistemas de hipertextos voltados para edição cooperativa de documentos. O capítulo V apresenta o software proposto nesta dissertação, denominado **DUETO**. O capítulo VI aborda a sua implementação, onde também apresentamos um exemplo de utilização do software. Finalmente, o capítulo VII, apresenta as principais conclusões do trabalho e em seguida, as referências bibliográficas.

## II - AUTOMAÇÃO DE ESCRITÓRIOS

Este capítulo aborda, de uma maneira geral, o tema automação de escritórios, onde apresenta uma classificação dos diversos sistemas desenvolvidos nesses ambientes com intuito de levantar a importância da integração das atividades. A partir desta abordagem, passa-se a focalizar o tema que é inerente a esses ambientes - o trabalho cooperativo apoiado por computador (TCAC).

O propósito deste capítulo é introduzir o leitor no ambiente de escritórios, em particular, nas características apresentadas pelo TCAC.

### II.1 - Abordagem Geral

Um típico ambiente de escritório trata com uma variedade de informações, em sua maioria textuais e gráficas. As atividades desenvolvidas nos escritórios, em geral, são paralelas, distribuídas e interativas, necessitando de constante coordenação. Em virtude do caráter interativo das atividades, a comunicação entre os diversos elementos torna-se um fator fundamental nesses ambientes.

As atividades desenvolvidas num ambiente de escritório envolvem a manipulação e troca de informações, sejam elas internas ou externas. Os trabalhadores de um escritório devem saber gerenciá-las e transmiti-las efetivamente.

Diversas atividades do escritório podem ser automatizadas individualmente, mas em virtude do paralelismo das atividades e a constante necessidade de comunicação entre os diversos membros do escritório, torna-se extremamente necessário o desenvolvimento de sistemas de informação de escritórios que focalizem a integração e sincronização das atividades e dos indivíduos, no sentido de reduzir a complexidade e melhorar a eficiência do escritório.

A introdução de novos procedimentos no ambiente de escritórios necessariamente envolve hardware, incluindo estações de trabalho e microcomputadores interligados através de redes locais e equipamento de comunicação para acesso remoto a dados. Além disso, um complicado ambiente de software que deve integrar as várias classes de informação, incluindo textos (formulários, cartas, trabalhos técnicos etc.), informações gráficas e pictoriais, além dos arquivos tradicionais de dados[Salt89].

Portanto, Automação de Escritório (AE) pode ser definida como a utilização da tecnologia para integrar as atividades de comunicação e informação no ambiente de escritório. Os Sistemas de Automação de Escritório representam métodos estruturados que permitem processamento de textos, gerenciamento da informação e tomada de decisão, além da comunicação através de uma rede integrada, auxiliando o pessoal de escritório na execução de suas tarefas.

## **II.2 - Categorias dos Sistemas de AE**

Esta seção apresenta duas classificações dos sistemas de automação de escritórios, uma baseada nas atividades desenvolvidas nos escritórios e outra baseada nas necessidades individuais [Naff88].

### **II.2.1 - Quanto às Atividades Desenvolvidas**

#### **.Ferramentas de Ajuda Profissional**

Essas ferramentas tentam ajudar os usuários em tarefas específicas, permitindo aumentar a capacidade individual nos processos de tomada de decisões gerenciais, na resolução de problemas pouco estruturados. Dentro desta categoria, os sistemas mais conhecidos são as planilhas eletrônicas, agendas pessoais, sistemas de gerenciamento financeiro, utilizados no planejamento e previsão financeira.

#### **.Ferramentas de Produção de Documentos**

Essas ferramentas permitem a criação, edição, armazenamento, recuperação e impressão de material textual ou gráfico, para a geração de correspondências formais e informais. Dentro desta categoria os sistemas mais conhecidos são os sistemas de edição de textos.

#### **.Ferramentas de Ajuda à Comunicação**

Essas ferramentas permitem o estabelecimento de comunicações eletrônicas entre seus usuários, para transmissão de informações e documentos, através de uma rede integrada. Dentro desta categoria encontram-se os sistemas de Correio Eletrônico, Videoconferência, Teleconferência, etc.

## **II.2.2 - Quanto às Necessidades Individuais**

### **.Sistemas para Computadores Pessoais**

São aplicações de pequena escala desenvolvidas para micro-computadores sem qualquer capacidade para comunicação (mono-usuário). Essas aplicações, em geral são destinadas a dar assistência aos usuários no tratamento da informação pessoal, independentemente das regras e procedimentos corporativos.

### **.Sistemas de Trabalho em Grupo Integrado**

São aplicações desenvolvidas para micro-computadores /estações de trabalho geralmente interligados por uma rede de comunicação, permitindo que grupos de pessoas compartilhem serviços e informações nos diversos níveis corporativos, aumentando a eficiência na execução dos procedimentos administrativos.

### **.Sistemas Departamentais**

São sistemas baseados em computadores de grande porte que agem como sistemas coordenadores centrais, permitindo que grupos de pessoas acessem as mesmas informações, em geral gerenciadas por um grande SGBD corporativo.

## **II.3 - Integração dos Sistemas de AE**

A idéia básica da integração dos diversos sistemas num único pacote é combinar os diferentes tipos de informação, funções e dispositivos de hardware conectando-os através de redes de comunicação para transporte, armazenamento, processamento e apresentação das informações. Os sistemas cooperativos exigem uma ampla mistura de diversos requisitos, ainda não disponíveis num único ambiente de desenvolvimento. Na medida em que forem integrados num modelo de dados unificado e em ambientes de desenvolvimento e gerência, a tarefa de implementar sistemas de escritórios, voltados para ambientes de trabalho cooperativo, será simplificada, permitindo que novas aplicações sejam desenvolvidas e efetivamente utilizadas nas organizações.

A integração é um fator multidimensional, existindo através de múltiplas formas:

- .Integração de ferramentas;
- .Integração dos meios para manuseio da informação;
- .Integração de tecnologias;
- .Integração para permitir o acesso às informações, independente da sua localização;
- .Integração através do tempo, isto é, que permitam a convivência entre as diversas gerações de equipamentos;
- .Integração do sistema (Hardware + Software) com o ambiente Sócio-Técnico;
- .Integração da interface com o usuário.

### **II.3.1 - Ambientes de Processamento distribuído**

A evolução dos sistemas de processamento compartilhado para sistemas de processamento distribuído pode ser creditada, em especial, à mudança na estrutura organizacional que tem exigido mais flexibilidade dos sistemas para se encaixarem nas suas novas necessidades, mas deve-se levar em conta também o avanço tecnológico das comunicações e o barateamento dos custos e o aumento do poder de processamento dos computadores pessoais.

As atividades de grupo tem dominado os ambientes de trabalho. As médias e grandes instalações de ambientes de processamento serão dirigidas a uma crescente descentralização, através de poderosas estações de trabalho integradas no fluxo de trabalho das empresas e através da diminuição na taxa custo/benefício de operações remotas [Muh190].

A idéia fundamental num ambiente de processamento distribuído é a transparência da distribuição, isto é, a transparência da natureza distribuída dos sistemas e das aplicações. O uso parcial de sistemas operacionais distribuídos, banco de dados distribuídos e linguagens distribuídas orientadas a objetos, introduzem uma transparência parcial na distribuição.

Além disso, o compartilhamento de recursos oferecido pelos ambientes de processamento distribuídos e o compartilhamento das informações entre os diversos participantes acentua ainda mais o caráter cooperativo proporcionado por esses ambientes.

Esses fatores levaram, nos últimos anos, a um crescente interesse no papel do computador em grupos de trabalho, com intuito de se criar ferramentas que suportem o trabalho cooperativo.

Em ambientes de processamento que suportam o trabalho cooperativo entre as pessoas, a comunicação é de vital importância, pois permitem que as pessoas compartilhem e troquem informações por meio do computador.

Para que os elementos de computação possam cooperar, trocando informações, é necessário, embora não suficiente, que estejam interconectados por um meio físico de comunicação, formando redes de computadores.

Nos últimos anos, o aspecto da comunicação que tem sido estudado mais intensamente pelos pesquisadores é a comunicação de dados sobre uma rede de computadores, em especial as chamadas “redes locais” de computadores (LAN).

A arquitetura dos sistemas distribuídos pode ser construída de duas formas: Replicada e Centralizada.

Os sistemas de processamento baseados na arquitetura replicada (distribuída) permitem que uma instância da aplicação execute em diferentes estações e a comunicação e troca de informações é efetuada diretamente entre as estações, através de mensagens. Um dos problemas relacionados com este tipo de arquitetura é a dificuldade no processo de sincronização entre as aplicações.

Os sistemas de processamento baseados na arquitetura centralizada (cliente/servidor) referem-se fundamentalmente a um método de processamento de aplicações. Um cliente é o “Front-End” de uma aplicação, que roda nas estações de trabalho e interage com os usuários, enquanto que um servidor é o “Back-End” que executa a maior parte das tarefas de processamento, que deveriam ser manipuladas nas estações clientes, provendo serviços, tais como: impressão, gerenciamento da rede e de arquivos, etc.

O entendimento do que deve ser distribuído ou centralizado em uma rede é a chave para entender o que é a arquitetura centralizada (cliente/servidor) e como ela trabalha [Rose89].

Esse tipo de arquitetura é projetada para permitir que aplicações desenvolvidas para rodarem como “back-end” possam tirar o máximo proveito dos serviços oferecidos pela rede, permitindo uma forte integração do sistema operacional da rede e das aplicações.

Para Knister [Knis90], este tipo de arquitetura permite o desenvolvimento de aplicações com alto grau de tolerância a falhas, importante para trabalhos em grupo, pois a entrada, saída ou mesmo a queda de um dos participantes não afeta o trabalho porque as transações são centralizadas na máquina servidora. Apesar do alto grau de compartilhamento entre as diversas máquinas clientes, inclusive a nível de transações dos usuários, este tipo de arquitetura apresenta um ponto de vulnerabilidade que é a não tolerabilidade de queda da máquina servidora.

### **II.3.2 - Trabalho Cooperativo Apoiado por Computador (TCAC)**

O tema dominante nas sociedades industriais modernas é que as atividades em grupo são economicamente necessárias, eficientes meios de produção e reforçam os valores democráticos. A necessidade para as atividades de grupo sobre as individuais tem sido assegurada pelo moderno estado industrial que não sobrevive somente no esforço individual [Krae88].

Embora o estilo real de trabalho em muitas organizações toma a forma de um trabalho cooperativo, raramente existe um suporte por computador.

Os sistemas de Informação de Escritórios (SIE) têm a função de prover ferramentas para diferentes classes de pessoas, tais como: gerentes, pessoal técnico e administrativo, mas ainda voltado para a satisfação das necessidades individuais.

King em [Grun91] sumariza que a computação cooperativa não é um campo, ela é uma arena na qual as pessoas de vários outros campos se juntam para demonstrarem seus trabalhos e aprenderem com os outros.

Os sistemas para trabalho cooperativo apoiado por computador (STCAC) ou GROUPWARE, estendem os conceitos dos SIE's, pois além de focalizarem as necessidades dos ambientes enfatizam os modelos especializados para trabalhos em grupo.



Para Ellis [Elli91], GROUPWARE pode ser definido como sistemas baseado em computador que suportam grupos de pessoas engajados numa tarefa (objetivo) comum e que provem uma interface para ambiente compartilhado, onde as ações dos usuários devem ser propagadas rapidamente para os outros usuários. Segundo os autores, o objetivo é assistir os grupos na comunicação, na colaboração e na coordenação de suas atividades.

Nesta monografia usaremos o termo GROUPWARE para designar sistemas para TCAC. Recentemente, está surgindo um consenso entre os pesquisadores de utilizar o termo GROUPWARE para denotar apenas o software e hardware de suporte aos sistemas de TCAC.

Os sistemas de groupware têm a intenção de criar um espaço compartilhado que suporte a colaboração dinâmica, em um grupo de trabalho, acima das restrições de espaço e tempo. Para se obter benefícios coletivos do uso de sistemas de groupware, estes devem ser aceitos pela maioria dos indivíduos como uma ferramenta comum [Ishi91].

Os sistemas de groupware não têm a intenção de fazer com que os membros de um grupo de trabalho abandonem seus trabalhos individuais, mas introduzir ferramentas que permitam que o trabalho individual e de grupo se tornem transparentes. A idéia é desenvolver ambientes que permitam tanto o trabalho individual quanto o de grupo.

Os sistemas de groupware se diferenciam dos outros sistemas multi-usuários, por proverem notificações das ações dos usuários para os outros participantes, com as mudanças sendo refletidas na interface de cada um.

Os sistemas GROUPWARE em tempo real são caracterizados por serem [Tane89] [Grei92]:

**.Altamente Interativos**

Rapidez para refletir as ações de um usuário na sua própria interface (tempo de resposta);

**.Tempo real**

Rapidez para refletir as alterações de um determinado usuário, na interface dos outros participantes (tempo de notificação);

**.Distribuídos**

Permite que todos os usuários conectados a uma rede compartilhem as informações independentes de sua localização física;

**.Voláteis**

Liberdade para os participantes entrarem e saírem durante uma sessão;

**.Ad hoc**

Capacidade de recuperar quaisquer informações dos usuários, sem qualquer pré-planejamento;

**.Controladores**

Durante uma sessão há um alto grau de conflitos de acessos enquanto os participantes trabalham e modificam os mesmos dados;

**.Conectivos**

Permitir a conexão entre participantes, por canais externos, como uma ligação de áudio e de vídeo;

**.Reconfiguráveis**

Permitir que novos pontos possam ser adicionados e retirados quando mudar a configuração do hardware.

O trabalho cooperativo apoiado por computador (TCAC) ou Groupware é um campo de pesquisas emergente que se interessa pela colaboração organizacional, visando trazer clareza e simplicidade à coordenação da ação humana. As pesquisas nesta área são de natureza interdisciplinar englobando aspectos provenientes da sociologia, antropologia, projetos organizacional e de gerenciamento, inteligência artificial, comunicação, sistemas distribuídos e a interação homem-máquina. Em resumo, Groupware é um campo de pesquisas que envolve o ambiente como um todo.

O avanço na tecnologia das comunicações e de suas aplicações aumentou o interesse de vários pesquisadores com intuito de criar sistemas que possam auxiliar as diversas pessoas a trabalharem em grupos. Tais sistemas permitem a comunicação entre as pessoas utilizando-se do computador como um meio de recepção/envio de mensagens.

Em virtude disto, pode-se dizer que TCAC abrange um amplo espectro que varia desde os sistemas de correio eletrônico, que transmitem mensagens mas provêm pouca informação ambiental (p.ex. Informação sobre os participantes, estado corrente do projeto, a atmosfera social, etc.), até os sistemas de “Sala de Aula Eletrônica”, que usam múltiplas janelas para apresentar a informação sobre o objeto que está sendo ensinado e sobre o ambiente. Dentro desse espectro encontram-se os sistemas de conferência por computador, sistemas de suporte a decisão em grupo, sistemas para co-autoria e sistemas de gerenciamento de projetos (descritos em II.3.3).

O meio eletrônico constitui-se no maior componente dos meios de produção para um trabalho em grupo com troca intensiva de informação, pois geram uma mudança tanto qualitativa quanto quantitativa em relação aos trabalhos desenvolvidos tradicionalmente. Isto resulta em significantes investimentos de tempo e energia por parte de seus membros, além de um nível relativamente alto de assistência durante o processo de aprendizagem [Evel88].

A idéia é permitir que os membros dos grupos coordenem suas próprias ações no sentido de capacitar a cooperação [Grei88].

### **II.3.3 - Categorias de Sistemas Groupware**

Esta seção pretende apresentar uma classificação dos sistemas considerados como groupware no sentido de demonstrar a abrangência desses sistemas [Elli91].

A classificação está baseada nas funcionalidades dos softwares a nível de aplicação. Muitas categorias podem se sobrepor, devido à demanda para aumentar a integração dos sistemas, ocorrendo uma diversificação nas funcionalidades dos sistemas.

Os sistemas groupware foram classificados em seis diferentes categorias, descritas a seguir:

#### **II.3.3.1 - Sistemas de Mensagens**

São sistemas de groupware, que suportam a troca assíncrona/síncrona de mensagens textuais entre os diversos grupos de usuários. Nesta categoria estão incluídos os sistemas de correio eletrônico e os sistemas de “Quadro de Avisos”.

A grande quantidade de sistemas existentes tem gerado o fenômeno denominado de “Sobrecarga de Informação”, fazendo com que os sistemas mais recentes adicionem “Inteligência” para suportar o compartilhamento de informação e a coordenação das ações, ajudando a filtrar e a priorizar as informações úteis que cada usuário recebe [Lee88].

### **II.3.3.2 - Editores Cooperativos**

São sistemas que permitem que diversas pessoas ou grupos possam, em tempo real ou não, compor e editar documentos (Editores de textos cooperativos) ou gráficos (Editores gráficos cooperativos) [Cama92].

Os editores síncronos (tempo real) permitem que os objetos que estão sendo criados possam ser acessados concorrentemente, sincronizando transparentemente as ações dos diversos usuários. Alguns editores provêm notificações explícitas das ações dos outros usuários.

Os editores assíncronos permitem a avaliação dos comentários anexados a um documento, em tempos diferentes, pelo revisor do documento, antes da sua forma final.

### **II.3.3.3 - Sistemas de Suporte à Decisão em Grupos e Sala de Reuniões Eletrônica**

Os sistemas desta categoria se utilizam das facilidades do computador para auxiliar um conjunto de pessoas, trabalhando em conjunto, na exploração de problemas não-estruturados tais como: geração de idéias, sistemas de voto [Krae88]. O objetivo de tais sistemas é aumentar a produtividade das reuniões para tomadas de decisão, através do aumento da velocidade ou pelo aumento da qualidade das decisões resultantes.

Para Halonen [Halo90], muitos desses ambientes podem ser implementados por um modelo de hardware simples, onde um simples computador equipado com um monitor de grande dimensão, controlado por um técnico, está acessível durante a reunião. O computador pode ser usado para mostrar agenda de reuniões, documentos relevantes, planilhas eletrônicas e etc. Embora este modelo capacite o grupo como um todo, ele apresenta uma desvantagem pois não permite que os participantes tenham igual acesso ao computador, nem usem o computador de forma privada durante a reunião.

Outros ambientes são implementados pelo modelo de groupware que, ao contrário do modelo anterior, focalizam o desenvolvimento de ferramentas de software específicas para suportar as atividades que ocorrem em reuniões. Esses ambientes podem ser vistos como Salas de Reuniões Eletrônica que contêm diversas estações de trabalho interconectadas através de uma rede que ajudam os participantes a estruturar as decisões, seja através de ferramentas para votação ou através da geração de idéias ou pela análise de aspectos [Ohku90].

#### **II.3.3.4 - Conferências por Computador**

Esta categoria de sistemas permite que grupos de pessoas, juntos em uma sala de reuniões eletrônica ou mesmo fisicamente dispersos (metáfora da sala de reuniões “virtual”) [Ahuj88], possam interagir sincronamente através de suas estações ou terminais.

Essa classe engloba vários tipos de sistemas de conferências por computador, que vão desde sistemas que oferecem capacidade de vídeo e áudio, mas não permitem o compartilhamento de textos e gráficos (aplicações) entre os participantes (Teleconferência), até sistemas que se utilizam de estações de trabalho como interface para conferência, permitindo o compartilhamento das aplicações entre os participantes e suportando múltiplas janelas por estação (Desktop Conferencing) [Ahuj88].

#### **II.3.3.5 - Agentes Inteligentes**

Nesta categoria estão incluídos os sistemas que se utilizam de participantes não-humanos (Agentes Inteligentes), como responsáveis por um conjunto específico de tarefas em que a interface do usuário torna suas ações parecidas com as dos outros usuários.

#### **II.3.3.6 - Sistemas de Coordenação**

Os sistemas de Coordenação discutem a integração e o ajuste harmonioso do esforço individual em direção à consumação de um grande objetivo. Estes sistemas permitem que o trabalho seja organizado como uma rede de ações interligadas, onde os usuários visualizam suas ações, bem como as ações relevantes dos outros usuários, dentro do contexto do objetivo global.

Sistemas de Coordenação podem ser categorizados por quatro tipos de modelos:

**Modelos Orientados a Formulários** - focalizam a rota (direção) dos documentos (formulários) nos procedimentos organizacionais.

**Modelos Orientados a Procedimentos** - veem os procedimentos organizacionais como processos programáveis.

**Modelos Orientados a Conversação** - são baseados na observação de que as pessoas coordenam suas atividades via sua conversação. A habilidade de afetar e antecipar o comportamento de outros através da linguagem é uma condição importante da ação humana. A teoria base para muitos sistemas que seguem o modelo de conversação é a teoria dos atos da fala ("Speech Acts"). Esta teoria revela a dimensão linguística que supera a visão da linguagem como um sistema apenas para representação da palavra e transmissão de pensamento e informações [Flor88].

**Modelos Orientados a Estrutura de Comunicação** - descrevem as atividades organizacionais em termos do papel do relacionamento.

## II.4 - Detalhes de Implementação em Sistemas Groupware

Essa seção discute algumas questões que ainda permanecem como objeto de estudo na área de trabalho cooperativo apoiado por computador e que devem ser consideradas no desenvolvimento de sistemas cooperativos.

Uma das questões relaciona-se com a interação homem-máquina em um contexto de interfaces de grupo ou multi-usuários. As interfaces de grupo introduzem problemas não apresentados em interfaces mono-usuário, pois necessitam de um complexo gerenciamento devido ao alto grau de concorrência das atividades que um grupo de participantes pode produzir [Elli91].

Outra questão relaciona-se com a necessidade de coordenação das diversas atividades entre os participantes de um grupo de trabalho, visto que tais atividades impoem obrigações aos participantes e o tempo de resposta das atividades é variado.

Uma outra questão importante, especialmente em ambientes de edição cooperativa, é a manutenção de diferentes versões dos objetos de maneira que os participantes possam traçar a evolução dos documentos.

A seguir, descrevemos com mais detalhes cada uma das questões apresentadas:

#### **II.4.1 - Interface para Trabalhos Cooperativos**

As interfaces são mecanismos desenvolvidos para que usuários e sistemas dialoguem entre si, sem necessidade de intermediário. Podemos dizer que uma interface se destina principalmente a facilitar a conversação entre sistema/usuário.

Hoje já não se pode avaliar a potencialidade dos sistemas somente pelas suas características internas de processamento, mas depende também da qualidade das operações que o sistema apresenta para subsidiar a troca de informações.

Para Furuta [Furu82], a qualidade da interface apresentada para os usuários pode ser julgada, em parte, pelos seguintes critérios:

- .A quantidade de detalhes que um usuário deve memorizar em ordem, para usar o sistema;
- .A quantidade de esforço mental que é exigido para executar funções comuns;
- .O número médio de erros feito pelos usuários, especialmente incluindo erros cuja a recuperação é difícil;
- .A quantidade de tempo de espera para que o sistema execute funções requeridas pelos usuários.

Em um ambiente cooperativo, um dos aspectos mais importantes é prover uma interface coordenada para todos os participantes no intuito de facilitar a execução dos comandos dos usuários, bem como a interação entre os mesmos.

O grande problema para gerenciamento da interface é suportar as diversas atividades de alto nível e a concorrência entre as atividades, que múltiplos usuários podem produzir.

Apresentar a informação compartilhada para todos participantes, através de imagens consistentes, permite criar a impressão de que os membros do grupo estão interagindo com objetos tangíveis e compartilhados.

O termo WYSWYG (What You See Is What You Get) é usado para descrever a interface dos sistemas para edição de textos, onde as mudanças no texto devem ser refletidas interativamente na tela do computador, para permitir a imediata visualização do efeito das mudanças. Semelhante a este termo, a área de TCAC, define o termo WYSIWIS (What You See Is What I See) para as interfaces de ferramentas de suporte a grupos ("Meeting Tools") que permite aos participantes tomarem conhecimento (visualizarem) do que cada um está fazendo, implicando numa frequência de atualização em tempo real.

O ideal é que não haja uma mudança brusca na visualização das modificações resultantes das ações de outros usuários. Uma solução apresentada é promover um anúncio da modificação, a priori, como por exemplo através de uma materialização ou mudança de coloração gradual, na visualização da ação [Elli91].

Na edição cooperativa (co-autoria) quando as pessoas estão editando a mesma coisa, ao mesmo tempo, pode ser importante a visualização das ações de cada usuário. Um exemplo seria mostrar onde o cursor de cada um está e a quem pertence, ou mostrar as ações através da materialização gradual ou da mudança de coloração do texto, à medida em que vai sendo editado. Todavia, quando a edição ocorre em tempos diferente pode ser importante destacar as mudanças que ocorreram desde o último acesso do usuário.

A interface WYSIWIS é limitante em termos práticos por isso os autores defendem o relaxamento de algumas regras [Ohku90].

Um dos relaxamentos diz respeito ao processo de sincronização das ações paralelas, isto é, a visualização das operações efetuadas por um usuário não devem ser apresentadas a outros usuários nem rapidamente (a nível de tecla) nem muito posterior à ação (P.ex. após a gravação).

Outro ponto é que grupos de pessoas serão afetadas por, ou estão envolvidas com, essas operações.



Outros tipos de relaxamentos dizem respeito à existência ou não de uma janela particular para cada usuário e à permissão para reposicionamento e variação do tamanho das janelas, tempo de apresentação [Stef88] [Newm91].

Um outro termo conhecido em ambientes que suportam o trabalho cooperativo é a interface WYSIWIMS (What You See Is What I May See) que provê dois ambientes de visualização para os usuários. Um ambiente privativo onde só o usuário tem acesso e um ambiente compartilhado que permite diferentes perspectivas de visualização. Ao usuário é dada a opção de “ver” somente o que ele quer da porção instável dos dados compartilhados, isto é, os usuários não necessitam ter as mesmas partes dos objetos compartilhados em suas telas nem querer ver as mesmas versões dos objetos compartilhados [Newm91].

A co-existência de duas visões (privativa e compartilhada) pode criar obstáculos na troca de ambientes e conseqüentemente pode confundir os usuários.

Estas questões ainda continuam objetos de discussão, por parte dos pesquisadores, no sentido de se obter uma solução comum.

#### **II.4.2 - Coordenação das Atividades**

Num ambiente de trabalho cooperativo, a coordenação das atividades aparece no sentido de gerenciar as tarefas executadas pelos usuários ( Que tarefas podem ser executadas e por quem ?) e garantir a consistência nos dados acessados.

Os dois aspectos mais importantes na coordenação das atividades em ambientes de TCAC, são o controle de acesso e a sincronização das ações concorrentes (Controle de Concorrência).

Num ambiente de TCAC, o controle de concorrência e o controle de acesso estão intimamente relacionados, em função das operações executadas pelos usuários e dos papéis permitidos para a execução das operações, no sentido de prover meios para a coordenação das atividades entre os participantes, prevenindo colisões e capacitando a cooperação.

### II.4.2.1 - Controle de Acesso

Um importante ponto na administração de sistemas está relacionado à especificação das ações que os usuários podem executar nos objetos compartilhados. Essa especificação, denominada de controle de acesso, deve ser implementada para coordenar as atividades do escritório.

Em sistemas de groupware é importante que se tenha um efetivo controle de acesso, pois estes sistemas tendem a focar nas atividades e aumentar a probabilidade de interferência usuário-usuário [Elli91].

A coordenação das ações pode ser suportada provendo um espaço de informação compartilhado em que as pessoas adicionam ou retiram as informações da maneira que elas definiram, ou através de um sistema que mantém informações sobre os diferentes papéis ou procedimentos dentro do grupo ou organização [Ishi91].

Os ambientes cooperativos introduzem novos requisitos para controle de acesso que não são atingidos pelo uso dos modelos existentes para os domínios não-cooperativos [Shen92]. O uso de critérios convencionais de acesso, baseados em dono (owner), grupo e público, raramente são suficientes para implementar uma política desejada nos sistemas de trabalhos cooperativos. Em ambientes cooperativos, as operações (leitura, gravação ou qualquer outra) em que seus resultados possam afetar múltiplos usuários, estas devem ser protegidas pelos direitos de acesso.

Um dos pontos mais importantes que devem ser considerados nos ambientes groupware, refere-se aos aspectos sociais da colaboração e coordenação entre as diversas pessoas envolvidas.

Segundo Neuwirth [Neuw90], uma resposta para o problema da coordenação é suportar a definição de papéis sociais. A definição de papéis reduz o problema da coordenação, através de especificação de “Funções próprias” (por ex. Responsabilidades e Padrões de interação) dos vários colaboradores.

#### **.Papéis**

Papéis são atributos dos usuários sobre determinadas operações, em determinados objetos do sistema. Isto reflete a idéia de que usuários com atividades semelhantes podem ter diferentes privilégios, dependendo dos papéis correntemente definidos.

Para implementação dos papéis, muitos sistemas utilizam listas com os nomes dos usuários ou grupos juntamente com seus direitos de ações nos objetos (Leitura/Edição). Essas listas podem ter diferentes granularidades dependendo do nível de controle que os sistemas desejam atingir [Grei88] [Lewi88].

Apesar dos potenciais problemas existentes nos sistemas que suportam a definição de papéis sociais, é importante a utilização dessa estratégia para gerenciar os problemas de coordenação e acesso [Neuw90]. Para Shen [Shen92], os sistemas devem saber inferir os direitos de acessos aos objetos a partir dos papéis definidos para cada usuário e devem permitir a mudança de papéis nas diferentes fases de colaboração.

#### **II.4.2.2 - Controle de Concorrência**

Em um ambiente de trabalho cooperativo, onde a maioria das ações são tomadas em paralelo, torna-se importante a criação de mecanismos de controle de concorrência de maneira a evitar situações de conflitos, mantendo-se a consistência dos objetos compartilhados.

Os sistemas de Banco de Dados convencionais para manterem a consistência, face aos acessos concorrentes, implementam um modelo de maneira que cada transação por si só preserve a consistência e que os efeitos de múltiplas transações, executadas em paralelo, devem ser os mesmos como se elas estivessem sendo executadas uma de cada vez [Grei88].

Para Ellis [Elli91], os modelos apresentados pelas aplicações de Banco de Dados podem ser inapropriados em ambientes cooperativos pois geralmente impedem a interação entre os usuários, importante para um perfeito trabalho em equipe.

Um dos aspectos relacionados à concorrência das ações em tempo real é com relação a rapidez no tempo de resposta e notificação aos usuários, de modo a manter a sincronização e cadência do grupo. Mesmo em casos onde as transações são longas, devem existir mecanismos que tornem os efeitos das transações permanentes e atualizados, mesmo que as transações subsequentes abortem [Grei92]. Uma longa interação deve ser bem gerenciada, pois pode impor limites severos na concorrência devido a um determinado dado que possa estar “preso” (“Hold”) a um determinado usuário, impedindo seu acesso por tempos muito longos.

Em ambientes de edição que suportam múltiplas versões de documentos, o método de controle de concorrência pode ser efetuado diretamente, isto é, um usuário visualizará somente a última versão liberada para o documento, enquanto este estiver sendo utilizado por outros usuários [Woel86].

A seguir descrevem-se alguns métodos utilizados pelos sistemas para implementação do controle de concorrência, a saber [Elli91]:

### **.Bloqueio (Lock)**

É o método mais utilizado e conhecido para controle de concorrência e consiste em bloquear um determinado objeto, impedindo o seu acesso por outros usuários, antes que o objeto seja alterado.

Em geral, para sistemas de groupware em tempo real esse método apresenta três tipos de problemas: O primeiro é a sobrecarga que o sistema recebe no requerimento e obtenção do bloqueio, incluindo o tempo de espera caso o dado já esteja no estado de bloqueio, causando uma degradação no tempo de resposta. O segundo refere-se à granularidade do bloqueio, pois sistemas com ótima granularidade (por ex. bloqueio a nível de palavras, em sistemas de edição de textos) podem causar uma considerável sobrecarga de processamento. O terceiro problema envolve o tempo de requisição ou liberação do bloqueio pois, em sistemas de groupware, fica difícil saber quem deve controlar o período de bloqueio (sistema ou usuário).

Em um ambiente cooperativo, é útil informar aos usuários que objetos estão sendo manipulados por um determinado usuário. Essa sinalização pode ser efetuada através de sinais gráficos na tela ou permitir somente a visualização do objeto que está com o bloqueio, da maneira não usual (tonalidade de cinza ou outra cor).

### **.Gatilhos (“Triggers”)**

São ações arbitrárias associadas a determinados objetos que, baseadas no tempo ou em algumas condições, são disparadas (executadas) pelos sistemas. Em sistemas de groupware estas ações poderão ser notificações relevantes aos participantes.

## **.Transações**

Uma transação é um conjunto de operações consideradas como uma unidade de trabalho. O mecanismo de transações tem sido usado em sistemas interativos multi-usuários e consiste em bloquear objetos enquanto uma transação está sendo executada, garantindo a consistência do conjunto de operações.

Este método, quando adotado em sistemas de groupware, apresenta uma série de problemas relacionados aos algoritmos utilizados para controle e o subsequente custo no tempo de resposta, problemas nos métodos para bloquear os objetos e recuperação após falhas do sistema, com subsequente retorno ao estado inicial.

Um outro aspecto relevante é que em sistemas de groupware torna-se importante a visibilidade das ações por outros usuários, fazendo com que este mecanismo se oponha aos objetivos dos sistemas interativos de groupware.

## **.Controle de Área (“Turn-Taking Protocols”)**

Este método consiste no gerenciamento de áreas compartilhadas por diversos participantes, onde a cada um é dado o controle da área compartilhada por um certo período de tempo [Gree90]. Em geral estes mecanismos não se adaptam bem a sistemas em tempo real, pois inibem o fluxo livre e natural da informação.

## **.Detecção de Dependência**

O método de Detecção de Dependência consiste na detecção de conflitos entre as operações, que são resolvidos manualmente pelos participantes.

Esse tipo de método que envolve usuários é, em geral, apropriado para sistemas de groupware, pois além das vantagens de terem bom tempo de resposta, as operações não conflitantes são executadas imediatamente. Entretanto, qualquer operação que necessite a intervenção dos usuários para assegurar a integridade dos dados é suscetível a erros.

## **.Execução Reversível**

Os mais promissores métodos de controle de concorrência caem nessa categoria, que consiste em definir uma ordem temporal global para as ações. Quando uma ou mais operações executadas concorrentemente interferem entre si, uma (ou mais) destas operações é desfeita e reexecutada numa ordem correta. A desvantagem apresentada por esse método é a necessidade da globalização da ordem das operações.

## **.Contextos**

Contextos consistem em partições de um hiperdocumento que podem ser usadas em áreas de trabalho privadas ou mesmo públicas a um conjunto de pessoas.

Este método pode implementar o controle de concorrência, pois no intuito de manter a consistência do hiperdocumento global, as partes do hiperdocumento movidas para um determinado contexto são bloqueadas para alteração, até que sejam liberadas pelo criador do contexto.

O sistema de hipertexto Neptune [Des187] [Des189] (descrito IV.4.2.3) adota esse tipo de esquema de particionamentos, para suportar o processo de autoria cooperativa.

O editor gráfico cooperativo - FLECHA, desenvolvido na COPPE-UFRJ, também utiliza o esquema de contextos, permitindo que cada co-modelador possa particionar o diagrama geral de objetos em partes que contêm apenas os objetos que serão atualizados. Com isso, o editor impede o acesso de outros co-modeladores para alteração dos objetos constantes dos contextos, permitindo somente a visualização [Cama92].

O sistema Intermedia [Yank88] adota um esquema semelhante de contexto ("Webs"), sobre o qual é construído o mecanismo de controle de concorrência do sistema.

### **II.4.3 - Controle de Versões**

O conceito de versões implica no armazenamento de diferentes representações de um mesmo objeto. Um conjunto de versões linear representa uma coleção ordenada onde cada versão do objeto tem um único predecessor, à exceção da mais antiga, e um único sucessor, à exceção da mais nova [Davi86].

O controle automático de versões é uma importante característica, em ambientes cooperativos, tornando-se imprescindível na representação e gerenciamento das informações, onde a ênfase está no compartilhamento e na colaboração.

Em ambientes que necessitem manter maior controle de versões, pode-se identificar três tipos principais de versões:

**.Históricas** - Permitem manter a evolução temporal dos objetos.

**.Equivalentes** - Permitem manter outras porções dos objetos em diferentes representações [Katz86]. Em ambiente para edição cooperativa de documentos, podem corresponder a versões por autor.

**.Alternativas** - Representam diferentes implementações de um mesmo objeto. Correspondem a versões paralelas de um objeto, isto é, derivadas de um mesmo objeto pai. Cada alternativa pode evoluir como uma cadeia de versões, que podem formar novas alternativas. Para a manutenção de versões alternativas os sistemas devem criar mecanismos para manter simultaneamente diversas instâncias (versões) de um mesmo objeto competindo, após a divisão de um conjunto de versões em dois ou mais caminhos [Baum88]. Esses sistemas necessitam sempre manter diferentes “estágios” ou “alternativas” de um objeto e, por isso, devem prover mecanismos para permitir a manipulação e o armazenamento destes objetos.

O acesso aos estados passados de objetos possibilita aos usuários explorarem alternativas e conhecerem a evolução do desenvolvimento dos objetos, permitindo o retorno a versões anteriores se as mudanças subsequentes são consideradas inadequadas.

A informação histórica torna-se mais importante quando múltiplos usuários compartilham a mesma informação, pois o risco de modificação acidental é alto e um usuário pode querer ver os objetos num estado estável, mesmo que outros usuários os tenham modificado [Grei88] [Furu82].

Configurações podem ser vistas como agrupamentos lógicos de versões específicas de objetos. As configurações podem ser consideradas como objetos e a interligação entre as diferentes versões constituem uma árvore de configurações que permite construir a história da evolução dos objetos.

A configuração particular de um objeto deve permanecer estável, isto é, as versões dos objetos componentes deve ser imutável. Tipicamente, somente as versões dos objetos e a configuração correntes do projeto devem ser permitido alterar [Grei88].

Uma questão crítica em mecanismos de controle de versões é o problema de propagação de versões dos componentes. Por exemplo, uma mudança significativa efetuada num componente pode implicar na criação de uma nova versão, enquanto que apenas uma correção num determinado parágrafo do componente pode não justificar a criação de uma nova versão [Hala88].

Em ambientes de edição de documentos, o controle da evolução de documentos é parte integral da visão conceitual destes ambientes de aplicação [Woel86].

Para sistemas baseados em textos, um dos requisitos mais importantes para controle de versões é minimizar os requisitos de armazenamento, mantendo somente as diferenças (deltas) entre as versões de um mesmo objeto, que podem se referir à versão prévia ou à versão subsequente [Prev90].

Em sistemas multi-usuários é importante registrar mais que uma versão de objetos, isto é, deve-se considerar a informação contextual tal como quem criou uma versão (autor) e quando (histórica). Isto permitiria a seleção da versão do objeto com base nas propriedades da interação de grupos.

Uma importante característica que os mecanismos de controle de versões devem conter é o suporte a múltiplas versões de nós e múltiplas configurações (diferentes arranjos das ligações), permitindo o grupamento (partições) das modificações de nós e da rede, de maneira que possam ser gerenciadas como uma entidade. Além disso, devem permitir que os usuários possam trabalhar em diferentes partições (Contextos) ou planos de versões dos hipertextos e, posteriormente, juntá-los de volta ao hipertexto.

Essas facilidades são apresentadas pelos sistemas de hipertextos Neptune (descrito em IV.4.2.7) e Intermedia [Yank88].

Um esquema semelhante é apresentado em [Prev90] e [Tsic90] baseado nos conceitos de organização dos hipertextos. O esquema permite a construção de “Overlays”, que são conjuntos de modificações relacionadas ao hipertexto que implementam tarefas específicas; e de “Perspectivas”, que são criadas a partir da combinação de overlays sobre o hipertexto original, constituindo a visão particular do usuário sobre o hipertexto. Múltiplas visões do hipertexto podem ser criadas a partir de diferentes combinações de overlays.



### III - SISTEMAS DE EDIÇÃO DE TEXTOS

A proposta deste capítulo é fazer uma abordagem dos sistemas de edição de textos, onde apresenta-se uma classificação para estes sistemas, caracterizando o ambiente de edição de textos voltados para o trabalho cooperativo. Apresenta-se, também, uma descrição das principais características de alguns sistemas de edição de textos cooperativos. Ao final, essas características encontram-se resumidas em uma tabela (figura III.4.1).

#### III.1 - Características

Os Editores de textos interativos talvez tenham sido as primeiras ferramentas a entrarem no ambiente de escritórios como parte do processo de automação. As tarefas de edição de textos através do uso do computador transformaram as máquinas de escrever em meros figurantes, uma vez que a utilização destas ferramentas permite que a edição do texto seja executada apenas uma vez, com a possibilidade de tê-lo sempre disponível para quantas revisões forem necessárias, sem precisar editá-lo novamente.

A pesquisa na área de sistemas de processamento de textos renovou o interesse por parte dos pesquisadores, principalmente nos fins dos anos 60 e início dos anos 70, quando surgiram os editores/formatadores de textos interativos e estruturados. Até então, os editores existentes eram não-interativos e suas unidades básicas de manipulação eram linhas [Meyr82b].

Os editores interativos começam a inovar na maneira de tratamento dos textos, não sendo mais vistos como um conjunto de linhas e sim como completas unidades estruturadas. Assim, os documentos começam a ser visualizados e manipulados em hierarquias de objetos (p.ex. capítulos, seções, parágrafos, sentenças e palavras) compondo árvores abstratas.

Além disso, permitem a inclusão e manipulação de objetos ambientais associados com o material de texto (p.ex. autores, índices, anexos, apêndices, etc.); informações gráficas (p.ex. desenhos para ilustrações); tabelas e formulários que incluem áreas que necessitam ser manipuladas; fotografias; som e voz.

As operações globais podem ser aplicadas a um elemento estrutural particular ou a um conjunto de elementos pertencentes a um mesmo nível hierárquico e sua implementação pode diferir dependendo do tipo de dado que está sendo manipulado [Salt89] [Naff88].

Outra questão importante tratada pelos editores interativos, em relação aos editores de linha, foi a utilização de toda a tela (“Displays Full-Screen”) para apresentação e edição dos textos, através de interfaces gráficas flexíveis, sob o princípio do WYSIWYG (What You See Is What You Get).

O avanço nos sistemas de processamento de textos permitiu a incorporação e manipulação de figuras gráficas juntamente com textos, além da utilização de menus para permitir a escolha das funções, sem a necessidade da utilização de comandos.

Mesmo com todo esse avanço, com raras exceções (Sistemas de Hipertextos), os sistemas de processamento de textos ainda continuam a ser utilizados somente como ferramentas para edição/formatação de documentos. Conforme introduzido por Engelbart [Eng88], a idéia básica para os sistemas de processamento de textos é a sua transformação em verdadeiros ambientes para a criação e autoria de documentos, permitindo unir as facilidades para o ordenamento lógico de idéias com os mais sofisticados recursos de edição/formatação.

Para Meyrowitz [Meyr82], o processo de edição de textos é um diálogo interativo usuário-computador que:

- .Permite a seleção de parte do objeto (Texto) a ser visto e manipulado;
- .Permite a determinação do formato a ser visualizado;
- .Permite a especificação e execução de operações que modifiquem o documento alvo;
- .Permite a atualização do objeto que está sendo visualizado.

Os sistemas de processamento de textos consistem de equipamentos e procedimentos para preparação, alteração, formatação e armazenamento de textos, de modo a permitir que os vários grupos envolvidos com a tarefa de edição de textos possam ter um sistema poderoso o bastante para executar todas as transformações nos textos, de maneira rápida, e que apresentem uma interface fácil de se utilizar [Salt89].

O processo de edição de textos em computador engloba tanto as facilidades de edição quanto as facilidades de formatação e, nesse processo destacam-se algumas funções que são encontradas na maioria dos editores de textos, tais como:

- . Movimentar-se (“Travelling”) de uma localização a outra dentro de um texto, ou mesmo entre dois ou mais documentos.
- . Pesquisar palavras de um texto a partir de um padrão (“Pattern”).
- . Folhear (“Browsing”) através de um documento estruturado, seguindo de um bloco de textos a outro.
- . Visualizar porções de textos desejadas (“Filtering”), indicando o nível de detalhe na hierarquia a ser mostrado.
- . Formatar o texto, incluindo espacejamento de texto com justificação automática, mudança de fonte, hifenação e paginação, tanto para visualização da forma final quanto para impressão.
- . Criar novos documentos a partir de outros já existentes, através da inserção de novas porções de textos.
- . Eliminar, modificar ou substituir porções de textos, preservando-se o texto original que pode ser novamente chamado, se necessário.
- . Desfazer (“Undo”) ou cancelar operações de edição previamente executadas, ou interromper uma sessão de edição salvando o conteúdo para recomeçar posteriormente.
- . Mover ou copiar pedaços de texto de uma localização para outra, dentro de um documento ou entre documentos, quando o editor permitir multi-janelas.
- . Prover operações (“Macros”) representando scripts de edição para operações complexa de transformações de textos.

- . Prover facilidades de ajuda (“Help”) interativamente para usuários que necessitem. As facilidades de ajuda podem dar uma explicação expandida de uma mensagem de erro, ou um pequeno sumário de uma sintaxe de comando, ou mesmo um acesso a uma versão interativa do manual de utilização.

## **III.2 - Classificação dos Sistemas de Edição de Textos**

A variedade dos sistemas de processamento de textos é tão grande que abrange um espectro que vai desde os editores/formatadores não-interativos até os sistemas integrados para edição/formatatação de textos.

Uma classificação dos sistemas de processamento de textos é apresentada, mas uma taxonomia padrão ainda está longe de ser conseguida porque, historicamente, o desenvolvimento dos sistemas de processamento de textos contém muitos paralelos e cruzamentos de idéias que impossibilitam uma estrita ordenação ou categorização [Meyr82b].

Os sistemas de processamento de textos são divididos em oito grandes classes: Editores orientados a linha; Editores de edição contínua (“Streams Editors”); Editores de visualização total (“Displays Editors”); Editores/Formatadores gráficos interativos; Editores de estrutura de propósito geral; Editores direcionados a sintaxe; Processadores de Texto; e Sistemas integrados.

### **III.2.1 - Editores orientados a Linha**

São editores de textos nos quais o modelo conceitual está baseado na edição virtual das imagens de cartão. Esses editores podem permitir a manipulação de linhas de caracteres em um arquivo documento, ou a manipulação de caracteres (palavras) dentro de uma linha ou grupo de linhas [Jame85].

Essas linhas de edição/visualização podem ter tamanhos fixos ou variáveis, dependendo do editor. Em geral, apresentam uma sintaxe de interação com comandos pré-fixados, onde se especifica a operação seguida do(s) elemento(s) que serão afetados pela operação.

Apesar da restrição, edição/visualização de uma linha de cada vez, alguns editores permitem que um comando possa ser expandido no seu escopo de uma única linha para um conjunto de linhas, indicando o intervalo de abrangência do comando.

### **III.2.2 - Editores de Edição Contínua (“Streams Editors”)**

São editores de textos baseados num modelo conceitual que considera o documento como uma cadeia contínua e indefinida de caracteres. Com isso, evitam os problemas que ocorrem nos editores de linha tais como: truncamento nas linhas e inabilidade de se executar pesquisas entre as linhas.

A interface destes editores, como no caso do TECO (“Text Editor COrretor”) [Meyr82b], é baseada numa sintaxe de comando consistindo de um único caracter. Estes comandos podem ser combinados para formarem sequências de comandos.

Apesar da orientação a caracter, permitem a movimentação edição do documento em termos de um modelo de linha. O texto é armazenado como uma simples sequências de caracteres que são interpretadas como linhas e, na sua recuperação, um filtro compreende a existência do delimitador de fim de linha.

### **III.2.3 - Editores de Visualização Completa (“Display Editors”)**

Nesta categoria estão incluídos os editores baseados em um modelo que considera os textos como sequências ilimitadas (limitado no tamanho do arquivo) de caracteres que se estendem, indefinidamente, tanto em largura quanto em comprimento. Apresentam grande flexibilidade de movimentação pois, a qualquer momento, os usuários podem visualizar qualquer porção de texto.

Estes editores consideram como válido qualquer caracter digitado e os comandos são acionados via teclas de funções, através de caracteres de controle ou editando uma linha de comando na base da tela. Em geral, apresentam uma sintaxe de interação com comandos simples pós-fixados, onde se especifica o(s) elemento(s) que serão afetados pela operação e em seguida a operação.

A maioria dos editores desta categoria já apresentam a espinha dorsal do comando “desfazer” (“undo”), onde é mantido uma arquivo temporário em disco para as atualizações.

Além disso, alguns dos editores oferecem facilidades de visualização do documento em várias janelas de uma vez, permitindo manter, como no caso do EMACS [Meyr82b], vários arquivos abertos simultaneamente.

### **III.2.4 - Editores/Formatadores Gráficos Interativos**

Nesta categoria encontram-se os editores de textos baseados em monitores de alta-resolução, com um modelo conceitual do ambiente que implementa uma metáfora eletrônica do escritório (“Desktop”), através da qual múltiplos objetos (arquivos, pastas, documentos) podem ser manipulados simultaneamente. Além disso, o modelo visualiza os documentos através de rolamentos (“Scroll”) contínuos de textos que podem ser paginados a qualquer momento.

A tela do computador é sub-dividida em diversas janelas (Bravo da Xerox Parc) com menus específicos por janela, onde os comandos podem ser executados via mouse ou teclas de funções (Xerox Star) [Meyr82b].

A interface destes editores é baseada numa sintaxe de comando que varia desde Pré-fixada, como no caso do ETUDE [Meyr82b], onde o projetista acredita que esteja mais próxima da linguagem natural (ex. DELETE 3 WORDS), até Pós-fixada/Infixada, como no caso do STAR. Além disso, o projeto da interface já considera os diversos níveis de conhecimento dos usuários.

Esses editores permitem facilidades de formatação e justificação interativa, onde as mudanças são refletidas imediatamente.

### **III.2.5 - Editores de Estrutura de Propósito Geral**

Essa categoria inclui os editores que exploram a estrutura dos documentos (p.ex. um documento, em geral, é composto por capítulos, seções, parágrafos, sentenças e palavras), como uma ordenação natural para simplificar a edição. A representação mais comum é uma hierarquia de elementos estruturados num modelo em árvore.

Engelbart [Eng88] com o sistema NLS foi o pioneiro neste tipo de editores que introduzem a noção do modelo conceitual para autoria e edição estruturada de documentos, com sofisticados

mecanismos de visualização e navegação (“browsing”), misturando textos e gráficos e, além disso, suportando múltiplos usuários.

Estes editores permitem a “ligação” entre as diversas estruturas dos documentos, formando uma “rede”, onde os usuários podem navegar ao longo das ligações (Sistemas de hipertextos).

Semelhantes aos editores/formatores gráficos interativos, os comandos podem ser executados usando o mouse para seleção de opções de um menu ou usando o teclado para edição dos comandos ou através de teclas de funções. Além disso, os comandos são divididos em duas categorias (comandos que operam no texto ou na estrutura do documento) dependendo do contexto em que estão inseridos.

### **III.2.6 - Editores Orientados à Sintaxe**

Essa classe inclui os editores de textos que conceitualmente são semelhantes aos editores de estruturas mas estão limitados a uma estrutura particular de uma linguagem de programação.

Além disso, foram projetados com intuito de aumentar a produtividade do programador, através da diminuição do tempo gasto com o processo de remoção de erros de sintaxe em programas.

Esses editores também são ferramentas para os compiladores/interpretadores pois interpretam a entrada do usuário, transformando-a numa forma intermediária que pode ser usada para gerar códigos.

Para evitar o tempo gasto com a digitação, esses editores apresentam padrões (“templates”) dos construtores da linguagem, frases prontas para preenchimento ou marcadores de lugar, que são etiquetas (“tags”) em um padrão que descrevem partes que necessitam ser preenchidas.

Alguns editores já encontram-se integrados num ambiente de desenvolvimento de programas, evitando o tempo gasto com o processo de compilação/remoção de erros pois, a compilação é efetuada incrementalmente onde as mensagens de erros são interativas permitindo a visualização, para correção, dos componentes que originaram os erros.

### III.2.7 - Processadores de Textos

Esta categoria inclui os sistemas destinados a, rapidamente, criar e modificar variados documentos textos, tais como: memorandos, livros e trabalhos técnico-científicos.

São sistemas projetados ao redor da metáfora da máquina de escrever, onde ainda não é permitido a visualização simultânea da base e topo de uma mesma página, tendo sua largura simulada (Word Perfect) [Mend90] como se fosse uma página impressa.

Os processadores de textos têm crescido em poder e flexibilidade para permitir o processo de revisão e a impressão de documentos com elegância e complexidade só visualizadas em impressões profissionais.

O modelo conceitual adotado é semelhante ao dos editores de edição contínua, onde os textos são vistos como contínuas cadeias de caracteres. Além disso, baseiam-se na tecnologia WYSIWYG (What You See Is What You Get), permitindo a visualização, na tela do computador, de uma representação exata da página impressa. Com isso, qualquer alteração (p.ex. tamanho e tipo do fonte, formatação da linha) é imediatamente visualizada na tela.

A maioria dos processadores apresentam variados tipos e tamanhos de fontes e alguns editores, específicos para edição de trabalhos técnico-científicos, possuem fontes para edição de fórmulas e símbolos matemáticos (p.ex. ChiWriter) [Mend90].

Alguns processadores apresentam uma interface de interação baseada em teclas de controle (“Control Key”) como o Word-Star, outros baseiam-se em teclas de funções como o Word-Perfect e outros são baseados em hierarquias de menus como, por exemplo, Microsoft Word, Ami Professional e ChiWriter, entre outros [Mend90].

Muitos desses sistemas permitem a associação de cabeçalhos, notas de rodapé e comentários nas páginas dos textos. As versões mais recentes apresentam diversas características, tais como: corretor automático de textos; “thesaurus”; edição de caracteres super e subscritos; hifenação automática; formatação automática que pode ser definida através de combinações de teclas (Microsoft Word); importação de gráficos, planilhas e figuras; revisão da página inteira antes da impressão; criação de macros com comandos definidos pelos usuários [Meyr82b] [Mend90] [Eglo90].



### III.2.8 - Processadores de Textos com Interfaces Gráficas

Essa classe engloba os chamados processadores com interfaces gráficas (GUI Word Processors) que permitem a geração e manipulação de documentos, figuras gráficas e lay-outs de páginas, num grau ainda não conseguido pelos editores baseado no ambiente DOS. Nessa classe destacam-se os processadores de textos baseados em ambiente WINDOWS.

Muitos processadores desenvolvidos para o ambiente DOS diferem dramaticamente em termos de estrutura de comandos e no modo de formatação de páginas. Por outro lado, muitos processadores Windows trabalham de modo similar que se estende além de uma estrutura de menus compartilhada.

Muitos programas aplicam a formatação através de estilos, característica essa herdada do processador WORD da MICROSOFT para DOS. Um estilo é um conjunto repetitivo de comandos de formatação aplicados, através de poucas teclas, ao parágrafo corrente ou ao texto selecionado. A formatação é efetuada em blocos de texto sem a necessidade de inserção de códigos de formatação visíveis, o que acontecia com a maioria dos processadores desenvolvidos para o ambiente DOS.

Estes processadores permitem a edição de textos e gráficos numa imagem da página, semelhante ao que será impresso, com cabeçalhos e notas de rodapé. Esta capacidade é importante quando existe a necessidade de formatação da estrutura de uma página, pois permite a manipulação e visualização completa do texto e gráficos na página.

Os processadores de textos Windows podem ser subdivididos em duas sub-categorias: Programas de características completas e Programas com características limitadas.

Os programas de características completas permitem a abertura de vários arquivos ao mesmo tempo, a criação e uso de macros, além de criarem e editarem gráficos. Estes programas podem mostrar diferentes visões de uma página, completamente formatada (visão WYSIWYG) ou uma visão de edição que suprime as margens de topo e de base.

Esta categoria engloba os editores Ami-Pro, Word for Windows e WordPerfect 5.1 for Windows [Mend92].

Os programas com características limitadas são distintos dos anteriores pois só permitem a edição de um arquivo por vez e não permitem a edição nem o uso de macros. Esta categoria engloba os editores WordStar for Windows, Professional Write plus e Just Write [Mend92].

A maioria dos processadores Windows apresentam grande habilidade na manipulação de grandes documentos, através da criação de um documento “Master” no qual estão contidos documentos menores que podem ser editados separadamente. Esses pequenos documentos podem ser combinados no documento “Master” para criação de índices e tabelas de conteúdo [Mend92].

### **III.2.9 - Ambientes Integrados**

A inclusão destes ambientes nessa classificação é devido ao modelo de interface em que estão baseados, onde o editor de textos é parte integrante da interface [Meyr82b].

A idéia principal é dar aos usuários uma interface onde tudo na tela pode ser editado (textos, programas, comandos etc), fazendo com que o usuário tenha um completo intervalo de escolhas em qualquer ponto na sessão de edição (Smalltalk).

A funções de edição seguem o modelo adotado pelos editores de visualização completa, como descritos anteriormente, onde a manipulação de itens é efetuada em blocos de textos (tela).

O processo de edição nestes ambientes faz um uso extensivo do mouse para marcar pedaços de textos, movimentar-se entre caracteres e utilizar menus complementares.

Como nos editores gráficos, as janelas também simulam a mesa do escritório, onde os diversos objetos podem ser manipulados simultaneamente.

### **III.2.10 - Sistemas de Edição de Textos para TCAC**

Uma aplicação cooperativa pode ser uma extensão lógica de um processador de textos mono-usuário para um sistema de co-autoria. Um editor de textos não se torna um sistema cooperativo simplesmente por permitir que diferentes usuários acessem e utilizem o mesmo documento

simultaneamente. Portanto, novos conceitos devem ser pesquisados de maneira a definir o ambiente cooperativo, onde os aspectos interpessoais são mais complexos.

Para [Malc91] os requisitos para os sistemas de edição em grupo devem ir além dos sistemas de edição mono-usuário, a saber:

- . Devem permitir traçar a evolução do documento e, se possível, poder reverter a versões anteriores do documento, se as alterações na versão atual foram inadequadas;
- . Devem manter múltiplas versões paralelas de um documento;
- . Devem permitir a comparação entre as diferentes versões do documento;
- . Devem permitir o armazenamento e acesso de comentários e sugestões de revisões dos autores, efetuados por membros do grupo, para um documento;
- . Devem ser compatíveis com os editores de textos existentes;
- . Devem ter um uso simples e natural compatível com os editores já existentes;
- . Devem ter facilidades para a comunicação entre os usuários (sistema de correio eletrônico, comunicação por voz, etc);
- . Devem ter um sistema de armazenamento e recuperação confiável, mesmo na presença de falhas de hardware;
- . Devem impedir o acesso a pessoas não autorizadas.

O resultado do processamento de informações em escritórios quase sempre resulta num documento. O gerenciamento destes documentos sempre exige esforço extra por parte das pessoas que trabalham no escritório, porque eles foram produzidos por várias pessoas ou porque eles são revisados frequentemente.

Quando diversas pessoas trabalham juntas para produzirem um documento, o acompanhamento do trabalho e a coordenação se faz necessário, no sentido de tornar o sistema flexível o bastante para que possa acomodar os diferentes estilos de trabalho, de maneira a reduzir o número de erros envolvidos na criação, revisão e gerenciamento de documentos multi-partes [Lewi88].

A coordenação das atividades, o compartilhamento da informação e a supervisão conjunta entre os autores dependerá da natureza do relacionamento interpessoal, o estágio do projeto e seu papel social. Tudo isso implica que qualquer ferramenta para suporte à edição cooperativa não deve suportar somente a comunicação entre os autores e de prover uma interface com capacidades para edição e composição de textos mas devem, também, suportar a definição de papéis sociais, com intuito de permitir o gerenciamento dos problemas de coordenação [Fish88].

Em um ambiente de edição cooperativa, compartilhar informações significa que alguma informação que teria permanecido implícita num projeto de edição individual deve agora torna-se explícita, de maneira que possa ser comunicada aos co-autores. Para Fish [Fish88], os co-autores se comunicam para tomarem ciência sobre o conteúdo de seus trabalhos (p.ex. O que estão atualmente escrevendo), os procedimentos de seus trabalhos (p.ex. Quem necessita fazer o que e quando) e, talvez mais sutilmente, o relacionamento interpessoal em que se baseia o projeto de trabalho (p.ex. A possibilidade de um parceiro não estar executando o compartilhamento corretamente).

### **III.3 - Exemplos de Editores Cooperativos**

A seguir apresentamos a descrição das características dos editores cooperativos de textos selecionados na literatura:

#### **III.3.1 - CES ( Collaborative Document Editing ) [Grei86] [Grei88] [Grei92]**

O CES é um sistema cooperativo de edição de documentos destinado a grupos de Co-autores trabalhando assincronamente em um documento. Foi desenvolvido como um protótipo de pesquisa da área TCAC no MIT Laboratory.

Esse sistema foi desenvolvido para ser usado em um ambiente distribuído, permitindo o compartilhamento entre os vários autores.

## **Características:**

- Um documento CES é formado por um componente estrutural, visto pelo autor como um rascunho, e um conjunto de componentes textuais denominados de seções (nós) do documento. As seções são blocos de textos ordenados, formados por um corpo e um título;

- O sistema implementa uma política tradicional de controle de acesso, onde cada documento tem um conjunto de co-autores com acesso específico (leitura ou edição) para o esboço ou para as seções do documento;

- Todos os co-autores compartilham o acesso à estrutura do documento, mas cada autor é “dono” de uma seção. A seção do autor reside na sua própria estação, de maneira que o texto do documento fique distribuído fisicamente entre as estações de todos os co-autores;

- O sistema provê operações para criação e manipulação das estruturas dos documentos independentes do texto.

- O CES coordena as atividades dos autores e tenta minimizar os atrasos na atualização das versões disponíveis para a leitura. Por exemplo, se alguém está escrevendo uma seção, o leitor terá acesso a uma versão um pouco desatualizada;

- O sistema CES utiliza o mecanismo de Transações, através do conceito de bloqueio, que é dado em um objeto quando um autor inicia a edição. É garantido que os objetos que estejam sendo editados (bloqueados) sejam gravados, seja após certos períodos de inatividade (“Thickle”) ou seja após cada comando significativo (p.ex. apagar uma palavra ou a cada nova linha). O sistema permite que outros autores visualizem os objetos que estão sendo editados e esta visão é sempre atualizada quando o objeto é atualizado;

- O CES coordena a cooperação entre os co-autores através de uma “reunião” em tempo real, na qual um grupo de co-autores, examinando uma seção do documento, conversam entre si, através de uma conexão por voz, enquanto vêem o documento nas suas estações;

- No sentido de manter as visões dos usuários sempre consistentes, o CES implementa um mecanismo de controle de versões das seções dos documentos que estão sendo modificadas. O sistema mantém uma pilha de versões das seções e operações específicas que garantem a recuperação da versão, pelo menos da anterior à mais atual;

- O CES implementa facilidades de “desfazer” (“undo”) alterações nas versões das seções do documento, através de um mecanismo de testes (“checkpoint”).

### III.3.2 - VAX ACCORD [Kenn88]

O VAX ACCORD é um sistema projetado para suportar várias pessoas trabalhando assincronamente na construção de um documento, num ambiente de co-autoria. O sistema roda em ambiente VAX/VMS e utiliza para edição de textos o editor LUDWIG, da Universidade de Adelaide.

#### **Características:**

- Um documento produzido pelo ACCORD está sempre alocado a um usuário criador, denominado de Super-Editor. O Super-Editor é um usuário que tem domínio completo sobre o documento com poderes para alocar responsabilidades sobre os componentes do documento para os outros usuários;

- Para um determinado usuário acessar o documento, o seu nome deverá estar contido numa lista de usuários criada pelo editor. Neste momento, o usuário pode apenas visualizar ou fazer anotações;

- Um usuário responsável por um determinado componente tem capacidade de edição sobre o componente, bem como pode alocar novos componentes inferiores hierarquicamente ao seu, para outros editores, tornando-se um Super-Editor destes usuários. O acesso dos editores está sempre restrito aos domínios sob sua responsabilidade;

- O ACCORD permite dois tipos de anotações: Anotação Contextual e Anotação não-Contextual. Anotações Contextuais são anotações ligadas a uma determinada versão do documento só podendo ser visualizada no contexto da versão e Anotações não-Contextuais são comentários gerais ligados a um componente particular do documento;

- A visualização dos comentários e dos componentes são efetuadas por um utilitário ("SCROLL") que assegura que os usuários não modifiquem os componentes e comentários de outros editores;

- O ACCORD implementa um mecanismo de controle de versões que a cada modificação num componente do documento cria uma nova versão deste, compondo uma árvore de configurações, para permitir o retorno às versões anteriores. O sistema só armazena as diferenças entre a versão atual e anterior, permitindo que se construa a história do documento;

- Cada versão dos componentes são numeradas de acordo com uma menor e maior versão do componente. As versões maiores representam o estágio de desenvolvimento no qual considera-se que o documento atingiu um marco. Neste estágio o documento é “congelado” e são removidas todas as versões anteriores, iniciando-se um nova lista de versões;

- O arranjo da configuração em uma árvore binária determina a ordem dos componentes dentro de um particular componente do documento;

- O sistema provê um sistema de correio eletrônico para troca de mensagens entre os participantes;

- Futuras extensões permitirão que o sistema tenha seu próprio editor, que será baseado num sistema de janelas e um completo conjunto de comandos de edição.

### **III.3.3 - Shared Books [Lewi88]**

O Shared Books é um sistema que permite suportar cooperativamente o gerenciamento de publicações no sistema de processamento de documentos VIEW POINT da Xerox.

Uma instalação VIEW POINT consiste de diversas estações interligadas para compartilharem arquivos, comunicação e impressão. Cada estação tem seu sistema de arquivos local e para o compartilhamento no servidor de arquivos, os usuários devem efetuar cópias dos arquivos locais para o servidor.

O VIEW POINT possui também um sistema de correio eletrônico distribuído.

O objetivo do Shared Books é ajudar os usuários a se concentrarem mais nas suas publicações do que no processo de produção.

## **Características:**

- O Shared Books provê um suporte eficiente na recuperação, armazenamento e pesquisa de todas as informações referentes aos documentos armazenados. O sistema gerencia todas as informações necessárias à reprodução dos documentos e suas informações relacionadas (p.ex. contratos, conogramas,...);

- O Shared Books permite que as publicações tenham sua história de revisões, permitindo o acesso a revisões passadas das publicações. Além disso, o sistema provê meios de armazenar versões paralelas independentemente da história das revisões;

- A janela do Shared Books engloba uma tabela de conteúdos onde são visualizadas as entradas das publicações. As entradas de uma publicação podem ser de dois tipos: (1) Corpo, provê o conteúdo da publicação e (2) Auxiliar, mantém as informações adicionais (p.ex. correspondências). Além disso, a janela inclui informações adicionais, tais como: estado corrente da publicação, mostrando o número da revisão, indicadores de bloqueios, data da criação, status de comentários e identidades dos usuários que mantêm os bloqueios;

- O Shared Books ajuda na unificação das entradas das publicações, através de numerações automáticas e referências cruzadas entre as partes da publicações;

- As mudanças ocorridas numa determinada estação não são refletidas imediatamente nas outras estações, sendo efetuadas após um novo comando;

- O sistema provê facilidades para inclusões de notas e justificativas que podem ser utilizadas para armazenar informações procedurais (referem-se ao processo de produção) e informações de anotações (referem-se ao conteúdo da publicação);

- O Shared Books coordena as atividades concorrentes, através do mecanismo de bloqueio, dando aos usuários direitos exclusivos sobre uma entrada. Nenhum limite de tempo é imposto no bloqueio e este é garantido ao primeiro que acessar a entrada.

- As atualizações nas entradas da publicação causam um bloqueio automático e transparente. Se for invocado um comando que irá modificar os dados, o sistema tentará adquirir o bloqueio, caso bem sucedido, o comando poderá ser executado senão o sistema informa que o dado está bloqueado e exhibe o nome de quem está com o bloqueio;



- O controle de acesso é implementado de duas maneiras, uma para a publicação como um todo (Shared Books), e outra a nível de entradas (tópicos) da publicação. O sistema mantém uma lista de acesso para cada publicação (Shared Book) onde são estabelecidas as permissões de acesso padrão até o máximo de permissão para todas as entradas da publicação. O nível de acesso de cada usuário está ligado à sua identidade no sistema.

- O sistema contém mecanismos que impedem o acesso simultâneo a uma mesma informação e que pessoas não autorizadas acessem as publicações;

- A lista de acesso e a identidade dos usuários no sistema podem ser usadas para suportar diferentes “papéis” para os usuários. Um usuário pode ter diversas identidades, onde cada uma tem diferentes direitos de acesso à publicação. Uma identidade (“Editor de Produção”) pode ser capaz de modificar a lista de acesso da publicação, outra identidade (“Autor”) pode ser capaz de editar as entradas da publicação e uma terceira (“Leitor”) só pode ler as entradas.

### **III.3.4 - DistEdit [Knis90]**

Diferentemente de outros sistemas de edição de textos cooperativos, o DistEdit é uma ferramenta que provê um conjunto de primitivas que podem ser usadas para adicionar suporte à colaboração a editores de textos convencionais, não cooperativos.

Com o DistEdit é possível a utilização de diferentes editores, rodando em diferentes tipos de hardware em uma mesma sessão. As primitivas fornecidas pela ferramenta são genéricas o bastante para dar suporte a editores com diferentes interfaces.

Os protocolos de comunicação utilizados com o DistEdit devem ter performance razoável com pequenos atrasos e devem ser transparentes para os usuários, não exigindo nenhum conhecimento dos aspectos relacionados à sistemas distribuídos.

#### **Características:**

- O DistEdit implementa um controle de concorrência, baseado no método de controle de área, permitindo que somente um usuário tenha capacidade de edição (Master) num determinado texto,

enquanto que aos outros só é permitido observar (Observador) o texto que está sendo alterado/modificado;

- O usuário Master pode liberar o controle de edição do texto através de um comando de chaveamento de controle, tornando-se um usuário observador. Após a liberação, qualquer usuário observador pode requerer o controle do texto, tornando-se Master;

- Os usuários podem entrar e sair de uma sessão de edição a qualquer momento, sem afetar os demais;

- Um Observador pode ter controle do cursor de sua estação mas não pode alterar o texto, que é permitido somente ao Master;

- Todos os usuários em uma sessão terão seus buffers de textos idênticos. Cada usuário pode ter visões levemente diferentes, dependendo das janelas de seus editores;

- Do ponto de vista dos editores de texto, a adição do DistEdit só modificará na adição dos indicadores de status dos comandos de acesso (Master/Observador) e de cursor (Livre/Protegido);

- O DistEdit provê três tipos de primitivas: (1) primitivas de atualização, executam operações no texto; (2) primitivas de notificação da posição do cursor, informam a posição corrente do cursor do Master; e (3) primitivas de controle, controlam a transferência de status de acesso e do cursor;

- Futuras extensões permitirão suportar bloqueios, a nível de regiões do texto, fazendo com que vários autores tenham capacidade de edição simultânea. Também está prevista a inclusão de mecanismos que garantam a segurança dos dados.

### III.3.5 - GROVE (GRoup Outline Viewing Editor) [Elli91]

O GROVE é um editor de esboços, implementado pelo Software Technology Program no MCC, que permite a utilização de grupos de pessoas trabalhando sincronamente em um esboço textual.

#### Características:

- O GROVE suporta visões múltiplas do texto e cada uma delas é apresentada numa “janela de grupo” que é replicada sobre todas as estações participantes, informando quem está usando a janela;

- Uma janela pode ser privada, compartilhada ou pública;

- O sistema exibe, na base da janela, os usuários que estão participando da sessão de edição e que está utilizando a janela;

- A estrutura do esboço é apresentada na janela juntamente com o direito de acesso correspondente ao item da estrutura. Os participantes podem modificar os direitos de acesso (ler/gravar) nos itens do esboço;

- Os itens do esboço podem ser abertos ou fechados utilizando-se botões, visualizados no lado esquerdo de cada item;

- As operações de edição são executadas concorrentemente, por qualquer participante, nos itens do esboço, e são imediatamente visualizadas nas janelas dos outros participantes;

- O mecanismo de controle de concorrência utilizado pelo GROVE é o de Detecção de Dependência com resolução de conflitos automática . Esta técnica permite que as operações requisitadas sejam executadas imediatamente na cópia local do usuário e, quando terminada a operação, esta seja enviada a cada participante juntamente com um “Vetor de Estado” que indica quantas operações foram processadas recentemente. Cada estação tem seu próprio vetor de estado que é comparado com os vetores recebidos. Se os dois vetores (Recebido e Local) são iguais, a operação enviada é executada como requisição, do contrário, a operação é “Transformada” antes da execução. A operação de transformação é dependente do tipo de operação ( p.ex., uma operação de inserir ou deletar).

### III.3.6 - MACE [Newm91]

O MACE é um sistema cooperativo de edição de textos que permite o compartilhamento dos documentos concorrentemente. O nível de compartilhamento é controlado consensualmente entre os usuários da forma que desejarem, inclusive a visualização das alterações em tempo real.

O editor foi projetado para ser um módulo para edição de documentos do sistema de conferências DCS desenvolvido pela Universidade da Flórida.

#### **Características:**

- O sistema permite que a visualização de um mesmo documento ou de diferentes documentos, possa ser efetuada em múltiplas janelas;

- O MACE é um editor modal que possui três modos de interação: Rolamento ("Scroll"), Edição ("Edit") e Atualização ("Update");

- O modo rolamento é o modo natural de exibição na janela do usuário quando o documento é carregado pelo sistema. Esse modo permite que o usuário tenha liberdade de visualizar qualquer parte do documento;

- O modo de edição é atingido quando o usuário obtém um par (superior e inferior) de bloqueio, que engloba a seção do documento a ser editada. A seção do documento bloqueada só poderá ser vista por outros usuários no modo rolamento, enquanto for mantido o bloqueio;

- O modo de atualização é obtido no momento que o usuário desejar salvar as alterações na seção do documento bloqueada. Esta ação poderá resultar na liberação do bloqueio e o retorno ao modo rolamento. Após a atualização o sistema exibirá, nas janelas dos outros usuários, a nova versão da seção;

- O sistema somente permite que seja bloqueada uma seção por janela de edição. Se os usuários desejarem bloquear outras seções dos documento, isto deverá ser efetuado em outras janelas de edição que poderão conter o mesmo documento;

- Os usuários, no modo rolamento, são informados da existência de uma edição (bloqueio) remota, quando esta ocorrer na seção do documento que está sendo exibida em sua janela. O sistema provê opções que informam a identificação de quem está em edição remota e três opções de atualização da janela: Atualização simples, Atualização tempo real e Cópia extra;

- A atualização simples permite que o usuário visualize a janela do editor e retorne novamente ao modo rolamento. A atualização em tempo real permite uma sincronização na visualização, de maneira que a janela do usuário seja atualizada a cada alteração efetuada na janela de edição. Nessa opção, o usuário é livre para retornar ao modo de rolamento. A opção cópia extra ainda não está implementada, mas permitirá a edição simultânea da seção bloqueada, criando uma cópia extra que poderá ser enviada aos outros usuários;

- O MACE pode ter três modos de visualização: Estático, Instantâneo e Síncrono. O modo estático (rolamento) permite que o usuário tenha liberdade de visualizar qualquer parte do documento, mas a frequência de atualização é controlada pelo editor na atualização do arquivo e a granularidade se limita ao tamanho da seção do documento que está sendo alterada. O modo instantâneo (atualização) permite que no momento da atualização as janelas do editor e visualizador estejam do mesmo modo e tamanho. A frequência de atualização é controlada pelo visualizador e a granularidade limita-se a uma página do documento. No modo síncrono, a visualização é semelhante ao modo estático, só que a frequência de atualização é efetuada em tempo real de maneira que todas as alterações sejam refletidas imediatamente;

- O MACE implementa o controle de acesso através da definição de papéis, para gerenciar o acesso aos arquivos compartilhados e através do mecanismo de votação, para remoção de bloqueios quando alguns usuários deixaram de liberá-los ou ficaram bloqueados por longos períodos;

- A próxima versão do MACE já incluirá a opção Cópia extra, em conjunto com o mecanismo de voto do DCS para permitir o gerenciamento das possíveis colisões que vierem a ocorrer. Também está planejada a inclusão de um mecanismo de controle de versões e anotações nos documentos;

### III.3.7 - ShrEdit [Dour92]

O ShrEdit é um editor cooperativo de textos que permite suportar um grupo de usuários editando um conjunto de documentos sincronamente. O editor foi projetado como uma ferramenta para exploração do suporte de reuniões de projetos.

#### Características:

- O editor provê dois tipos de janelas: Uma janela compartilhada que permite a edição de textos concorrentemente; Uma janela privativa onde somente um usuário visualiza e edita o documento, sendo usada também para criar anotações ou criar textos que serão posteriormente inseridos num documento compartilhado;

- O editor permite várias janelas (compartilhadas e privativas) abertas simultaneamente. As visões dos documentos são únicas para cada usuário, podendo ter diferentes tamanhos e alinhadas em diferentes partes do documento;

- O ShrEdit bloqueia a janela compartilhada a nível de seleção de textos, impedindo o acesso de outros usuários ao texto selecionado. O sistema não permite que dois usuários tenham os seus cursores de edição na mesma posição do texto. As colisões de cursores são indicadas por sinal de áudio e uma janela que se abre indicando a colisão. As ações de edição dos usuários são visualizadas em todas as janelas compartilhadas;

- O sistema também possui uma janela de controle associada com cada janela de edição, que exibe o nome dos participantes na sessão. Esta janela permite também que um usuário visualize o texto selecionado para edição na janela de outros usuários, inclusive com a posição do cursor de edição;

- Cada janela de controle registra a seleção dos participantes, indicando também se esta seleção está sendo visualizada por outros;

- O ShrEdit não impõem nenhuma estrutura sobre as atividades dos participantes, permitindo o igual acesso para as janelas compartilhada do documento. O sistema permite que os usuários adotem seu próprio estilo de trabalho.

### III.4 - Resumo das Características dos Editores Cooperativos

A seguir, apresentamos um quadro comparativo (figura III.4.1) dos editores descritos em III.3, com relação a algumas características consideradas importantes para editores cooperativos. Tais características incluem: Os tipos de interação entre um grupo de usuários e um dado documento; Os mecanismos implementados para controle de concorrência, controle de acesso e controle de versões; Os meios utilizados pelos editores para comunicação entre os usuários; A permissão para incorporação, ao longo do texto do documento, de anotações para revisão.

	<u>Tipos de Interação</u>	<u>Controle de Concorrência</u>	<u>Controle de Acesso</u>	<u>Controle de Versões</u>	<u>Comunicação Externa</u>	<u>Anotações</u>
<u>CES</u>	Assíncrona	Transações Curtas	Leitor / Editor	Não	Conexão por Voz	Não
<u>Vax ACCORD</u>	Assíncrona	Não	Lista de Acesso	Sim	Correio Eletr.	Sim
<u>Shared Books</u>	Assíncrona	Bloqueio	Papéis	Sim	Não	Sim
<u>DistEdit</u>	Assíncrona	Contrôle de Área	Master / Observador	Não	Não	Não
<u>GROVE</u>	Síncrona	Deteção de Dependência	Ler / Gravar	Não	Não	Não
<u>MACE</u>	Síncrona	Bloqueio	Papéis / Votação	Não	Conferência Eletr.	Não
<u>ShrEdit</u>	Síncrona	Bloqueio	Papéis	Não	Não	Sim

Figura III.4.1 - Características dos Editores Cooperativos

## IV - SISTEMAS DE HIPERTEXTOS

Este capítulo introduz as principais características dos sistemas de hipertextos, com o objetivo de familiarizar o leitor nos principais conceitos desta tecnologia. Para maiores detalhes ver [Conk88], [Niel90b], [Scac89], [Parsa89] e [Mend92b].

### IV.1 - Características

Credita-se a Vanevar Bush a primeira proposta sobre os sistemas de hipertextos. Em 1945, Bush propôs um sistema mecanizado denominado “Memex”, que seria um sofisticado sistema de biblioteca de literatura científica. Esse sistema permitiria o estabelecimento de ligações entre dois pontos quaisquer de documentos (Livros, Registros e Comunicações) armazenados no sistema e que pudessem ser consultados com rapidez e flexibilidade extraordinária [Bush88].

No início da década de 60, começaram a surgir algumas propostas como as de Engelbart (63) e Nelson (67), mas o termo “Hipertextos” somente surgiu em 1967, quando Ted Nelson definiu esse nome para sistemas com características semelhantes ao que estava sendo desenvolvido por ele - Xanadú [Pime89].

A idéia principal de um sistema de hipertextos é permitir a criação e manutenção de trechos de textos de uma forma não linear, formando uma rede na qual cada nó contém um trecho de informações e cada elo de ligação entre dois nós representa um relacionamento entre as informações contidas nos nós (figura IV.1.1).

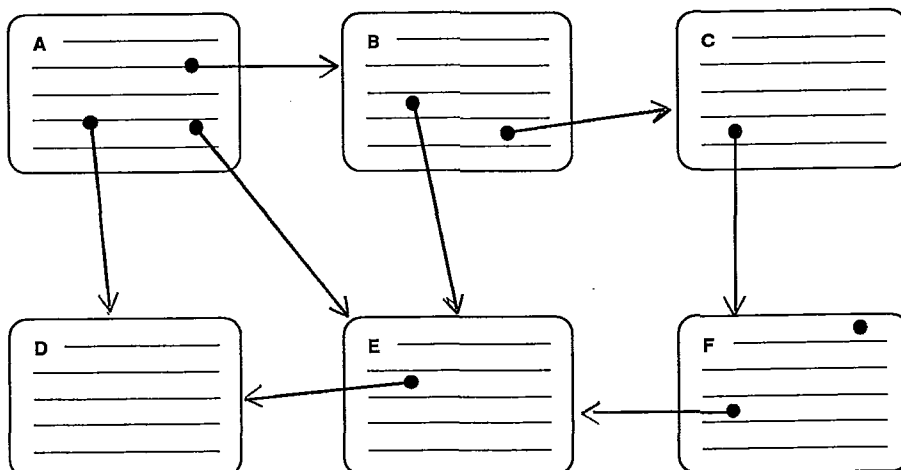


Figura IV. 1.1 - Visão da estrutura de um Hipertexto e suas Ligações



O conteúdo das informações, num sistema de hipertexto, não está restrito a textos somente, podendo representar imagens gráficas, textos combinados com gráficos, fotos e sons. Nestes casos, os sistemas são conhecidos como “Hipermeios”.

Um hiperdocumento pode ser caracterizado como uma rede de nós interligados que possuem trechos de informação e cada ligação representa o relacionamento entre os trechos de informações (no mesmo nó ou em nós diversos). Os trechos de informações podem representar qualquer objeto do documento (p.ex. páginas, seções, parágrafos), dependendo do projeto de implementação do hiperdocumento [Conk88].

Os nós são as unidades básicas de armazenamento de informações textuais ou gráficas, nos sistemas de hipertextos. Dependendo do sistema, os nós podem ser de tamanho fixo ou de tamanho variável.

As ligações permitem representar o relacionamento entre objetos de informações dos hipertextos. Além disso, fazem com que os usuários tenham liberdade na realização de consultas aos documentos, pois podem optar por seguir ou não a estrutura sequencial dos documentos.

Além da função de ligação entre os nós de texto de um mesmo documento, as ligações permitem outras funções, tais como:

- .Conectar um comentário ou anotação a um trecho de texto que está sendo escrito;
- .Prover informação organizacional;
- .Conectar dois pedaços sucessivos de texto ou pedaço de texto e todos seus descendentes imediatos;
- .Conectar entradas em tabelas ou figuras às descrições ou a outras tabelas ou figuras [Conk88].

## IV.2 - Navegação em Hipertextos

Um dos aspectos mais importantes dos sistemas de hipertextos é a facilidade de movimentação ao longo das ligações (Navegação), permitindo o acesso às informações (Nós) armazenadas.

Quando um texto é armazenado num formato não-linear existem diversas possibilidades de movimentação ao longo do texto e estas são produtos da combinação de três tipos de movimentos [Trig86]:

### **. Movimentação Horizontal**

Permite o acesso aos nós vizinhos ao nó corrente. O próximo nó a ser visitado pode seguir linearmente o texto ou pode ser escolhido de um grande conjunto de nós vizinhos potenciais. Uma viagem “lateral” ortogonal ao encadeamento de idéias é, em geral, necessária aos novatos para obtenção de exemplos e explicações e para permitir a obtenção de mais detalhes para os usuários mais experientes.

### **.Movimentação Vertical**

Permite o acesso dos usuários, a partir de uma tabela de conteúdos, dos mais altos aos mais baixos níveis de abstração. A tabela de conteúdos impõe o acesso através de uma hierarquia.

### **.Movimentação por Pulos ou Saltos**

Permite o acesso arbitrário ao longo do texto, em geral efetuado através de índices (Palavras-chave ou padrões).

A liberdade de navegação ao longo da estrutura dos hiperdocumentos pode trazer riscos para os usuários com relação a sua localização, fazendo com que se sintam “desorientados” e fazendo com que os usuários tenham um esforço a mais para manterem-se informados sobre o caminho percorrido.

Por causa disto, a maioria dos sistemas de hipertextos tem tentado desenvolver mecanismos adicionais para auxiliar na orientação dos usuários ao longo da navegação nos hiperdocumentos. Alguns destes métodos serão descritos a seguir:

## **.Diagramas para visualização gráfica dos dados**

São diagramas que exibem a estrutura do hiperdocumento (nós e ligações) na forma de grafos direcionados. O problema deste tipos de estrutura é o grande número de nós e suas diversas conexões com quaisquer outros nós, que pode comprometer a legibilidade quando da apresentação aos usuários.

Alguns sistemas tentam resolver este tipo de problema apresentando a estrutura do hiperdocumento seletivamente, reduzindo assim o número de objetos visíveis em cada nível de apresentação. Um exemplo é apresentado em Nielsen [Niel90], onde a estrutura do hipertexto é visualizada em dois níveis de diagrama que provêm aos usuários um sentido de localização “grosseiro” dentro do espaço global e um sentido de localização mais “fino” (“Fisheye View”) na vizinhança local do nó corrente.

## **.Contextos**

Permitem a visualização de um conjunto de nós e ligações que apresentam certa proximidade, dependendo de certos critérios (p.ex. assunto, tipos de nós,..).

A noção de proximidade é dependente do contexto, portanto o conjunto de objetos (Nós e Ligações) pode ser visualizado diferentemente dependendo do contexto.

## **.Marcadores**

Marcadores (“checkmarks ou bookmarks”) são mecanismos utilizados para sinalizar os nós ou páginas em que o leitor já esteve, permitindo a sua utilização como caminhos nos diagramas para um provável retorno [Bern88]. Outro tipo de marcadores, denominado de Tabuladores (“thumb tabs”), permitem a criação de caminhos constantes e não modificáveis para os leitores, permitindo rápido acesso para as partes importantes dos documentos.

## **.Listas de visita**

Permitem manter a informação sobre o caminho (trilha) que o usuário escolheu, permitindo que o mesmo possa retornar à origem de onde partiu.

Um exemplo dessa característica é encontrado em Trigg [Trig86] onde o sistema mantém uma lista ordenada de todas as ligações, denominada de caminho ("path"), percorrida pelos leitores. Além disso, o sistema permite a criação, por parte dos autores, de caminhos padrões para leitores inexperientes. Outro exemplo é apresentado em Nielsen [Niel90], onde além de utilizar uma lista como explicado anteriormente, o sistema mantém uma lista com a história de todos os nós visitados, numa ordem sequencial de tempo de visita.

## **IV.3 - Hipertextos como Suporte à Edição Cooperativa**

### **IV.3.1 - Características**

Criar um documento técnico é uma tarefa raramente executada por apenas uma pessoa, pois a produção de boa documentação, mesmo quando existe apenas um único autor, exige a participação direta de várias outras pessoas como revisores, editores [Deli89].

Os sistemas de hipertextos podem prover facilidades para suportar o processo de edição cooperativa ou co-autoria, pois o processo de criação de documentos, de manutenção de múltiplas organizações de um simples conjunto de material e a transferência de mensagens entre usuários são espécies de atividades que exigem um esforço colaborativo.

O modelo de grupo de co-autores é também caracterizado por uma divisão em três maiores camadas. O trabalho de grupo inicia com a "geração" de contribuições individuais, tais como idéias, propostas de esboços e porções específicas de textos. Cada objeto emergente desta fase está aberto a "comentários" pelos membros do grupo (p.ex. comentários críticos, argumentação por alternativas).

Após a coleta da concepção do grupo de contribuições individuais, estas unidades tem de ser combinadas de acordo com os requisitos temáticos e formais do documento todo. Isto é essencialmente uma tarefa de "reconfiguração", colocando pedaços de idéias juntos - relacionando idéias simples

para formar um agregado de idéias estruturadas, estruturar propostas de esboço em termos de planejar um texto compreensivo, englobar pedaços de texto de forma a construir um documento coerente. Durante esta fase, surgem em diferentes níveis do processo de co-autoria especificações incompletas, múltiplas e inconsistentes soluções.

Entretanto, estratégias baseadas na negociação têm de suportar as resoluções de conflitos conceituais, seleção/adaptação de alternativas temáticas e conceituais, etc. Como resultado, tópicos, esboços e passagens de textos podem ser confirmadas, modificadas ou retiradas, levando a um reajustamento global das contribuições individuais [Hahn89].

Segundo Stefik [Stef88], o acesso serial obstrui a participação equalitária que geralmente caracteriza a cooperação, particularmente na tarefa de edição, onde a colaboração parece não envolver qualquer divisão de trabalho fixa ou pré-determinada entre os participantes.

As facilidades apresentadas pelos sistemas de hipertextos (p.ex. controle de versões de documentos, interface multi-janelas que permite a visualização de várias partes do documento), juntamente com um forte mecanismo de gerenciamento de grupos e com estratégias que melhorem a comunicação entre os participantes, podem torná-los importantes ferramentas para o desenvolvimento de sistemas em ambientes distribuídos, em especial para a edição de documentos com multi-autoria.

As ações paralelas exigem que os sistemas adotem políticas para o controle de acesso e concorrência aos dados compartilhados, como descrito na seção II.4.

#### **IV.3.2 - Exemplos de Hipertextos para Edição Cooperativa**

Esta seção descreve alguns exemplos de hipertextos voltados para edição cooperativa de documentos. Cabe ressaltar que alguns destes exemplos podem não ser totalmente classificados como hipertextos, mas se utilizam desta metodologia como forma de estruturar o documento. Ao final, as principais características estão resumidas numa tabela (figura IV.3.3.1).

#### IV.3.2.1 - KMS [Aksc87]

O KMS é um sistema de hipermeios comercial desenvolvido pela Knowledge Systems para redes de estações heterogêneas. O objetivo do sistema é suportar a colaboração em larga escala, abrangendo um grande número de aplicações entre elas, publicações eletrônicas e documentação interativa.

##### **Características:**

- O KMS consiste de um conjunto de áreas de trabalho, denominadas de “Frames”. Um frame corresponde a um nó do hiperdocumento e seu tamanho é limitado ao tamanho da tela do computador;

- Um frame pode conter vários tipos de informação, tais como: textos, imagens e gráficos. Um frame pode ser ligado a qualquer outro frame. Um frame pode conter associado itens especiais para diversas propostas, tais como: Comentários, Notas, etc;

- As ligações entre os frames permitem a representação de uma grande variedade de estruturas de conhecimento (p.ex. documentos, projetos, programas,...);

- O modo de navegação e o de edição de documentos são acessíveis simultaneamente sem necessidade de transição entre eles. Para edição basta mover o cursor para a posição desejada e editá-lo;

- A edição no KMS é dominada pela manipulação de itens individuais e a maioria das operações podem ser invocadas diretamente. O item apontado pelo cursor permite definição de um conjunto de operações que podem ser realizadas;

- O KMS permite múltiplos usuários trabalhando simultaneamente em um projeto comum, onde cada usuário pode visualizar o que os outros estão fazendo, fazer comentários, imprimir qualquer parte do material, etc.;

- O controle de concorrência implementado pelo KMS é baseado no modelo “otimista”, que supõe ser raro o conflito entre dois usuários editando simultaneamente um mesmo frame;

- O sistema garante que as alterações efetuadas num frame não poderão ser modificadas por outros usuários, naquela versão;

- O sistema permite também que um usuário proteja os frames que estão sendo modificados por ele, utilizando um frame “informal” que avisa aos outros usuários que uma alteração está sendo executada;

- O KMS suporta um tipo de controle de versões efetuado através de um utilitário, que permite “congelar” todos os frames de uma hierarquia, como pertencentes a uma determinada versão. Quando um frame congelado é alterado, seu conteúdo é salvo automaticamente num novo frame que pode ser mapeado para uma nova versão do frame original. Os usuários podem acessar versões específicas de um frame;

- O KMS implementa um tipo de proteção individual de frames. Cada frame tem seu “dono”, que é o criador do frame, e este pode protegê-lo contra o acesso de outros usuários ou pode permitir o acesso mas sem modificá-lo.

#### **IV.3.2.2 - Notecards [Hala87] [Hala88]**

O Notecards é um sistema de hipertexto para auxiliar diversos usuários (autores, pesquisadores) a trabalharem, de um modo geral, no processamento de idéias.

Inicialmente, o Notecards foi projetado para ser um sistema mono-usuário mas, a partir da observação de que as tarefas de processamento de idéias e gerenciamento de informação eram inerentemente colaborativas, o sistema sofreu extensões para suportar o trabalho cooperativo.

##### **Características:**

- O sistema fornece ao usuário uma rede de cartões eletrônicos de anotação interligados por ligações tipadas. Esta rede serve como meio na qual o usuário pode representar coleções de idéias relacionadas. O Notecard possui facilidades para visualização, modificação, manipulação e navegação através da rede de cartões e ligações;

- O sistema está baseado em quatro construtores básicos: (1) Notecard , é o nó do sistema utilizado para anotações, gráficos e outros estilos de nós; (2) Filebox, é um tipo especial de cartão utilizado para organizar e criar categorias, a partir de coleções de cartões relacionados; (3) Ligação, são tipadas e bi-direcionais usadas para interconectar cartões em rede ou estruturas de cartões relacionados; e (4) Browser, é um notecard que contém o diagrama estrutural do hiperdocumento;

- O Notecards provê um compilador de documentos que permite linearizar a rede de cartões criados para um documento. Um simples cartão contém todo o texto e gráficos do documento na ordem apropriada, sem qualquer estrutura hierárquica. Após a compilação, a visualização só ocorre num único nível, não sendo mais permitido a visualização do documento a partir da estrutura;

- O mecanismo de compartilhamento de “notefiles” (Banco de dados de Notecards e ligações) ainda é limitado, pois não permite múltiplos usuários acessando simultâneamente um notefile. O compartilhamento é executado utilizando-se alguma forma de “passagem de esboço”. Um dos problemas deste processo é que exige que os usuários gerenciem uma logística de controle, incluindo a manutenção do caminho da versão “verdadeira” do notefile enquanto é copiado de um lugar para outro;

- Os colaboradores coordenam a transferência através de correio eletrônico ou interação face-a-face e a transferência física do notefile é realizada através da cópia dos arquivos entre o servidor e as estações;

- O sistema permite que os colaboradores “quebrem” um notefile em pedaços, permitindo o acesso de várias pessoas ao mesmo tempo;

- Futuros desenvolvimentos incluirão facilidades de cartões de história para ajudar os usuários a adicionarem tanto uma base lógica de seus trabalhos quanto anotações gerais para histórias, formando uma história coerente das mudanças nos notefiles.

#### **IV.3.2.3 - Neptune [Deli87] [Deli89]**

O sistema de hipertexto Neptune foi desenvolvido para suportar o trabalho cooperativo de multi-autoria em projetos de engenharia, em ambiente de rede local.



## **Características:**

- O Neptune é composto de um conjunto de nós com variados tipos de informações (textos, gráficos, imagens) e ligações entre os nós. As ligações são utilizadas como dispositivos de referências cruzadas, permitindo que uma parte do documento possa ser referenciada ou acessar partes de outros documentos;

- O sistema permite que nós contendo anotações, notas de pé de páginas e referências sejam ligadas ao documento;

- O Neptune provê um suporte especial para a edição da hierarquia do documento, através de comandos de manipulação de árvores num mapa gráfico e através de listas orientadas ao esboço;

- O Neptune implementa um mecanismo de gerenciamento de transações que assegura a consistência das informações caso ocorram falhas do sistema;

- O sistema permite a recuperação de um conjunto de nós e ligações baseados em valores de atributos associados aos nós;

- O Neptune inclui um sistema de gerência de armazenamento que provê um repositório para todas as informações e documentação associada com um projeto, juntamente com suas configurações e a história de versões;

- O sistema implementa o conceito de Contexto (descrito em II.4), permitindo que usuários criem visões (conjunto de nós e ligações) privadas ou públicas a um conjunto de autores, a partir do hipertexto global (visão ou contexto Master) acessível a todos os usuários;

- O sistema assegura a consistência do contexto Master, quando parte do documento foi copiada para um contexto privado, impedindo o acesso para modificação por outros usuários, garantindo que somente um autor fique responsável por uma determinada porção do documento que está sendo modificada. O sistema suporta esse protocolo através do bloqueio dos nós e ligações do contexto Master, que foram copiados para um contexto privado;

- Os contextos formam mundos privados onde os autores podem explorar alternativas e as revisões em progresso, que só são visíveis por outros autores após sua liberação via a operação Fundir (“Merge”) entre as visões “Privada” e “Master”. Diferentes versões de um mesmo nó e ligações podem co-existir simultaneamente em diferentes contextos. Os contextos de revisões podem ser destruídos após a fusão no contexto master ou podem ser armazenados como permanentes, registrando que alternativas foram consideradas;

- O Neptune provê um mecanismo de pesquisa para recuperação de contextos a partir de nós criados pelos autores que identificam os contextos. O sistema permite a visualização gráfica da árvore de contextos;

- O sistema permite a definição de heurísticas modificáveis pelos usuários, para extração de situações de conflitos que possam ocorrer, quando dois contextos contêm partições não disjuntas do hiperdocumento;

- O sistema inclui diversos mecanismos para armazenar e manipular versões. Todas as alterações efetuadas nos nós e ligações são registradas e versões antigas podem ser acessadas em qualquer momento;

- O mecanismo de controle de versões inclui dois tipos de ligações: Ligações para versão corrente e Ligações para versão específica. Configurações podem ser construídas utilizando-se um nó para representar uma configuração e os itens da configuração são representados no nó pela ligação para a versão específica do nó atual, representando as versões desejadas. Desta maneira diferentes versões dos componentes do sistema podem ser unidas.

#### **IV.3.2.4 - Quilt [Fish88]**

O Quilt é uma ferramenta computacional assíncrona que provê meios para um sistema multimeios de mensagens, um sistema de conferências, anotações e facilidades de notificações para suportar a comunicação e o compartilhamento de informações aos colaboradores de um documento.

### **Característica:**

- O Quilt possui três tipos de comentários: Privados, visíveis apenas pelo criador; Públicos, lidos por qualquer usuário e; Mensagens diretas, direcionadas a um indivíduo ou grupo de indivíduos;

- A coordenação das atividades é efetuada de duas maneiras, informando aos participantes sobre o que cada um está fazendo ou através da recuperação das anotações dos outros participantes da história do documento;

- No primeiro caso, o sistema mantém um arquivo de história (“log”) das atividades para cada nó, com um registro da interação dos colaboradores com o documento. As entradas do log podem ser efetuadas pelos participantes, pela máquina ou pela combinação dos dois. As entradas são descrições abstratas das intenções ou execuções e o sistema as utiliza para alertar os participantes, caso tentem modificar um objeto, que está sendo modificado por outro;

- No segundo caso, o sistema utiliza “gatilhos” (“triggers”) que são mensagens semi-estruturadas ou formulários, preenchidos com um repertório padrão, para notificação aos participantes;

- O Quilt pode ser usado como um sistema de conferências por computador no sentido de facilitar a comunicação no processo de edição cooperativa. O sistema provê um meio através do qual os colaboradores podem ter discussões planejadas, numa forma que facilmente levaria a um projeto de edição. Repostas às mensagens anteriores poderão ser efetuadas através de comentários que podem ser públicos, privados ou mensagens diretas.

#### **IV.3.2.5 - Internote [Cat189]**

O Internote é uma ferramenta que estende o ambiente Intermedia [Yank88], com objetivo de dar suporte à colaboração através de anotações. O Internote é uma ferramenta intuitiva para revisão integrada com capacidade de gerenciamento de anotações.

Com o Intermedia como base para o sistema, tem-se a facilidade para a criação de ligações entre seleções nos vários tipos de documentos e suporte a ligações de múltiplos usuários de e para um documento, simultaneamente.

## **Características:**

- O sistema permite anotações em qualquer tipo de documento. As anotações são sub-divididas em dois tipos principais: Sugestões de mudanças no documento e Comentários textuais para notas, comentários, sugestões não específicas e justificações;
- O sistema provê uma interface consistente para anotações em todos os documentos Intermedia, com dois tipos de comandos: Criar Anotação, liga uma anotação em qualquer documento, e Incorporar Anotação, incorpora o conteúdo da anotação ao documento;
- O sistema permite que os usuários percorram, através das ligações, as anotações incorporadas;
- As anotações são efetuadas utilizando “notas”. Uma nota consiste de dois quadros: Quadro de Incorporação, para descrever as anotações que serão incorporadas nos documentos e Quadro de Comentários, para comentários de sugestões ou de explicações das mudanças sugeridas;
- O Internote permite que um autor e um anotador visualizem uma anotação simultaneamente e provê meios para gerenciar as colisões existentes, quando um usuário está atualizando um documento no momento que outro faz anotações;
- O sistema permite que vários usuários façam anotações simultâneas num mesmo documento;
- Um documento anotado é denominado de rascunho e todas as anotações pertencentes a um rascunho são, automaticamente, colocadas em uma janela associada ao documento denominada de “Note Folder”;
- A ligação das notas ao documento pode ser efetuada simultaneamente por diversos usuários;
- Os autores podem revisar um documento baseados nas sugestões de alterações e comentários. O processo de revisão inclui ler as anotações, determinar quando incorporar uma sugestão e fazer a edição das alterações no documento, baseada nos comentários das notas;
- Um anotador é capaz de esconder as anotações, efetuadas por outros participantes, para trabalhar numa cópia “limpa” (sem comentários anexos) do documento;

- As anotações podem ser armazenadas, juntamente com os nós e ligações, em contextos denominados de “Webs”. O sistema permite que múltiplos usuários trabalhem no mesmo Web mas com específicos direitos de acesso (ler/gravar e anotar);

- O Internote possui uma janela denominada de “Web View” que mostra o histórico do caminho efetuado pelo usuário, através do conjunto de ligações. Além disso, um mapa mostra o documento no seu estado atual e todos os documentos conectados a ele.

#### **IV.3.2.6 - Co-Author [Hahn89]**

O sistema Co-Author foi desenvolvido como parte do MULTWORKS (MULTimedia Integrated WORKStation), do projeto de integração tecnológica ESPIRIT-II, suportado pela comissão das comunidades europeias sob o ESPIRIT.

O Co-Author é uma ferramenta que permite a colaboração assíncrona de múltiplos autores na produção de documentos hipermeio.

O sistema está baseado no princípio de que a co-autoria, num ambiente de hipermeio, pode ser vista como a criação, revisão contínua e manutenção de três bases de conhecimento distintas: Uma base de conhecimento de idéias (“IdeaKB”) que armazena o conteúdo conceitual dos documentos; uma base de documentos hipermeio (“HyperDocB”) que contém a caracterização formal dos esboços dos documentos; e uma base de objetos hipermeio (“HyperObjB”) que consiste de objetos hipermeio (pedaços de texto, figuras, imagens).

#### **Características:**

- O sistema provê um conjunto de operadores conceituais para criar/refinar os principais tópicos de um documento (Nós-idéias), num processo similar às atividades de “Brainstorm”. O relacionamento entre as idéias pode ser visualizado através de um mapa gráfico hierárquico;

- A especificação formal do documento é atingida a partir de operadores de sequenciamento e agrupamento, unindo logicamente o conteúdo das porções do documento hipermeio, de acordo com requisitos temáticos de estrutura e de mensagens;

- O esboço do documento é atingido a partir de ligações dos objetos idéias armazenados na base. Como resultado obtém-se um grafo da especificação da estrutura do documento, com seus tópicos arranjados numa ordem hierárquica e sequencial;

- O sistema provê um “contrato de tópicos” que permite dar responsabilidades, a um determinado autor, para a realização de um tópico, nos termos do texto, na elaboração do documento final;

- O sistema provê mecanismos que permite o gerenciamento de “anotações” de contribuições individuais de outros membros da equipe e de “configuração” de simples especificação / partes do texto para manter a coerência do documento. Estas tarefas são distinguidas por três operadores: (1) Operador de comentários, utilizado para criticar tópicos já existentes e para indicar alternativas ou falta de especificação; (2) Operador de manutenção, que executa testes de consistência, completeza e coerência das especificações; e (3) Operadores de alinhamento, que tratam as resoluções de conflitos, inconsistências e falta de especificações;

- As anotações ocorrem em todos os níveis do processo de autoria, ligando-se às estruturas criadas;

- O suporte ao trabalho de grupo se baseia nos procedimentos, através dos quais as contribuições individuais são combinadas para formar um documento composto pelo grupo. Os aspectos inconsistentes, incompletos ou alternativos são identificados e negociados pela troca de auxílios e contra-argumentos. Assim, o sistema enfatiza a idéia do mediador que visualiza os problemas do grupo como objetos de projeto ao invés de seguir um modelo puramente conversacional;

- O futuro protótipo do sistema será baseado numa arquitetura cliente-servidor, com clientes específicos que suportarão editores hipermeio e mapas gráficos para manipulação gráfica dos objetos. Além disso, estão sendo implementadas facilidades de conferência em tempo real para o gerenciamento de comunicações curtas do grupo.

#### IV.3.2.7 - Prep (Work in Preparation) [Neuw90]

O editor Prep é um ambiente multi-usuário projetado para suportar uma variedade de cooperações, em particular, a autoria e os relacionamentos através de comentários na comunicação escolar.

O editor se baseia em 3 aspectos principais: (1) Suporte às interações sociais entre co-autores e comentaristas; (2) Suporte para os aspectos cognitivos da co-autoria e; (3) suporte à praticidade, em ambos os tipos de interações.

##### **Características:**

- O Prep permite a definição de dois tipos de papéis sociais: Co-autor e Comentarista. O sistema estende o papel do comentarista além dos limites das anotações, permitindo uma intensa participação na elaboração e planejamento do documento, além de permitir o questionamento do esboço ou de uma seção do documento. Esta última tarefa é executada num relacionamento binário co-autor/comentarista;

- O Prep está preparado para aceitar entradas e emitir saídas numa variedade de formas, incluindo voz, papel e arquivos;

- No nível estrutural, o editor compartilha das características básicas de muitos sistemas de hipermeios;

- Como nos sistemas de hipermeios, o Prep permite a definição de trechos de textos (nós) que correspondem às idéias. Os nós podem conter textos, grades, árvores ou imagens;

- Durante a fase de planejamento, os autores podem escolher trabalhar com nós, como objetos livres, numa área de trabalho;

- O sistema permite a definição de ligações simples entre os nós, de maneira que uma rede de conceitos possa ser construída;

- O ponto central do editor é a sua interface, para a representação visual das informações, permitindo novas espécies de comunicação além de suportar os estilos existentes;

- A interface do editor é dividida em grades, denominadas de áreas de trabalho. Uma área de trabalho consiste de uma coluna da interface esparçadamente preenchida de nós. Cada coluna da grade é usada para armazenar diferentes conteúdos das áreas de trabalhos, como por exemplo artigos, planos de guias para confecção do documento etc;

- O Prep provê convenções para a comunicação sobre partes das áreas de trabalho, permitindo a definição de “rascunhos” do documento. Um rascunho define um espaço numa área de trabalho, definida por um autor, para que outras pessoas o acessem;

- Uma área de trabalho dispõe de construtores para a comunicação entre os co-autores e comentaristas, sobre um rascunho que eles compartilham. Os co-autores e comentaristas podem criar planos, se comunicar e comentar sobre os planos ou simplesmente anotar o conteúdo do documento;

- O editor provê áreas de trabalho para que os autores, durante a fase de planejamento, possam trabalhar com nós como objetos livres do documento;

- O editor Prep, futuramente, permitirá a incorporação de revisões para existirem como distintas versões do rascunho. Estas revisões, pelo mecanismo de versões, permitirão o trabalho nas áreas do rascunho sem a preocupação da perda do material mais antigo.

#### **IV.3.2.8 - GroupWriter [Malc91]**

O GroupWriter é um sistema cooperativo de edição de textos com características semelhantes aos processadores de textos convencionais. O sistema foi projetado para rodar em ambiente de microcomputadores interligados em rede, com uma conexão intermitente entre os usuários para troca de informações e para resoluções de conflitos.



## **Características:**

- O GroupWriter possui características de um sistema híbrido, pois permite que um mesmo documento seja visualizado diferentemente por diferentes grupos de pessoas. O sistema pode ser visto como um processador de texto convencional; como um sistema de controle de versões e anotações textuais ou voz; ou como um completo sistema de hipertextos;

- O sistema apresenta a interface compatível com o processador de texto MS-WORD, e os dados armazenados no formato compatível com os processadores que utilizam o padrão RTF. Além disso, apresenta uma alta qualidade tipográfica na visualização e na impressão;

- O sistema provê um mecanismo de controle de versões que permite o gerenciamento de múltiplas versões, tanto dos documentos quanto dos parágrafos que os compõe;

- O GroupWriter provê um mapa gráfico do documento para visualização do relacionamento entre as diferentes versões e o acesso a estas versões;

- A unidade básica de um documento GroupWriter é o parágrafo. Cada parágrafo contém uma única identificação e uma ligação para o parágrafo do qual deriva. Uma versão do documento é identificada por um nome e sua estrutura é armazenada como um lista de parágrafos com ligações para a versão do documento do qual foi derivado;

- Quando um documento é selecionado para edição, os parágrafos que contiverem versões alternativas ou anotações serão visualizados com linhas verticais dispostas, em sua margem esquerda que indicam a existência das versões e/ou anotações. Um menu associado a essas linhas está acessível para se efetuar operações sobre os parágrafos e anotações;

- O GroupWriter provê a operação “fundir” (“merge”) que permite a inclusão de sentenças escolhidas para serem incorporadas ao texto, após a comparação sentença-a-sentença entre um parágrafo e sua versão alternativa;

- O sistema provê dois tipos de anotações: Textuais e Voz. As anotações textuais podem ser vistas numa janela separada e podem ser incorporadas ao texto do documento. As anotações por voz podem ser ouvidas a partir da operação “play” existente no menu localizado nas linhas laterais.

#### IV.3.2.9 - SEPIA [Haak92]

O SEPIA é um sistema de hipertextos para autoria de hiperdocumentos com uma extensão para suportar a edição cooperativa em ambiente de computação distribuído, desenvolvido pela divisão de sistemas de hipermeios cooperativos do GMD Institute for Integrate Publication and Information Systems (IPSI).

##### Características:

- O hipertexto SEPIA está baseado em três objetos básicos: (1) Nós atomicos, são os objetos que possuem os textos do hiperdocumento; (2) Nós compostos, são usados para juntar informações relacionadas e manter as estruturas aninhadas hierarquicamente; (3) Ligações tipadas. Além disso, o sistema provê dois outros objetos: Nós anotação e as Ligações para estas anotações, que são especializações dos Nós compostos e das Ligações tipadas, respectivamente;

- Os nós compostos são usados para juntar informações necessárias à execução de uma tarefa ou trabalho em um tópico específico. Cada nó composto pode ser associado com um ou mais autores que necessitam compartilhar dados. Os nós compostos podem conter os mesmos objetos do hipertexto, permitindo a formação de múltiplas estruturas organizacionais;

- O sistema SEPIA suporta a criação de hiperdocumentos através de diferentes *Espaços de Atividade*. Cada Espaço de Atividade pode ser visto como um “folheador” (“browser”) específico de tarefas, que suporta uma parte do processo de criação do hiperdocumento;

- Os usuários criam um hiperdocumento, interagindo com quatro espaços de atividades dedicados a tarefas de geração de conteúdo, estruturação, planejamento, argumentação e edição do hiperdocumento final sob uma perspectiva retórica (Espaço de Conteúdo, Espaço de Planejamento, Espaço de Argumentação e Espaço Retórico, respectivamente). Cada um destes espaços possui objetos e operações específicos e servem como o nível mais alto na estrutura do hipertexto;

- A interface do SEPIA está baseada em duas decisões de projeto: (1) Uma janela “folheador” por espaço e por nó composto, usada para visualização e manipulação dos objetos compostos. O sistema suporta a visualização simultânea de diversos “folheadores” com diferentes nós compostos; (2) O estado dos objetos é sinalizado pelo uso de cores (p. ex. nós e ligações de mesmo tipo podem ser visualizados diferentemente, dependendo do estado interno);

- A manipulação e acesso dos dados do hiperdocumento e a visualização das informações são gerenciadas por um banco de dados multi-usuário, que provê o controle necessário para o compartilhamento dos documentos entre os diversos co-autores;

- O SEPIA provê três modos para cooperação: (1) Trabalho individual, que consiste no desenvolvimento de tarefas em nós não visitados por outros autores; (2) Modo fracamente acoplado, que consiste no desenvolvimento de tarefas em nós compostos que estão sendo acessados por outros autores; e (3) Modo fortemente acoplado, que consiste no compartilhamento do conteúdo de um mesmo nó composto;

- O controle de acesso que permite suportar os três modos de cooperação é efetuado através de bloqueio simples dos objetos e através de mudanças na coloração dos objetos;

- O modo fracamente acoplado exige dois requisitos no projeto do “folheador” do espaço de atividades: (1) Conscientização de grupo, onde os autores devem ser capazes de reconhecer a presença e ações de usuários concorrentes de um mesmo nó composto com intuito de determinar possíveis conflitos e possibilidades de cooperação mais próximas; e (2) Bloqueio de objetos atômicos como forma de evitar ações contra-producentes. Os objetos bloqueados são visualizados em diferentes colorações, associadas a diferentes atividades dos usuários;

- O modo fortemente acoplado exige mais três requisitos além daqueles descritos para o modo fracamente acoplado: (1) Visões compartilhadas que permitem que cada co-autor visualize um mesmo “folheador” em termos de tamanho e conteúdo; (2) Um dispositivo “telepointer” exibido nos diversos “folheadores” das janelas dos co-autores e controlado por um autor; e (3) Uma conexão por voz para comunicação entre os diversos co-autores.

### IV.3.3 - Resumo das Características dos Hipertextos para Edição Cooperativa

Semelhante à seção III.4, apresentamos um quadro comparativo (Tabela IV.3.3.1) dos sistemas hipertextos descritos em IV.3.2, com relação a algumas características consideradas importantes para os sistemas de hipertextos voltados para edição cooperativa. Tais características incluem: Os tipos de interação entre um grupo de usuários e um dado hiperdocumento; Os mecanismos implementados para controle de concorrência, controle de acesso e controle de versões; Os meios utilizados pelos sistemas de hipertextos para comunicação entre os usuários; Os meios utilizados para orientação dos usuários na navegação ao longo do hiperdocumento; A permissão para incorporação, ao longo do texto do documento, de anotações para revisão.

	<u>Tipos de Interação</u>	<u>Controle de concorrência</u>	<u>Controle de Acesso</u>	<u>Controle de Versões</u>	<u>Comunicação Externa</u>	<u>Navegação</u>	<u>Anotações</u>
<u>KWS</u>	Síncrona	Otimista / Gatilhos	Dono (Owner)	Sim	Não	Não	Sim
<u>Notecards</u>	Assíncrona	Controle de Área		Proposto	Correio Eletr.	Mapa Graf.	Proposto
<u>Neptune</u>	Assíncrona	Contextos	Ler / Gravar	Sim	Não	Lista / Mapa / Query	Sim
<u>Quilt</u>	Assíncrona	Gatilhos (Triggers)	Papéis	Sim	Correio Eletr.	Mapa / Tabela	Sim
<u>Internote</u>	Síncrona	Contextos (Webs)		Não	Não	Mapa Graf.	Sim
<u>Co-Author</u>	Assíncrona	Não	Dono (Owner)	Sim	Reunião Eletr. (proposto)	Mapa Gráf.	Sim
<u>Prep</u>	Assíncrona		Papéis	Proposto		Não	Sim
<u>GroupWriter</u>	Assíncrona			Sim	Não	Mapa Graf.	Sim
<u>SEPIA</u>	Síncrona	Transações / Bloqueio		Proposto	Conexão por Voz	Espaços de Atividades	Sim

Figura IV.3.3.1 - Características dos Hipertextos para Edição Cooperativa

## V - DUETO: UMA FERRAMENTA PARA EDIÇÃO COOPERATIVA DE TEXTOS TÉCNICOS

Este capítulo apresenta a especificação da ferramenta **DUETO** (**DU**plo Editor **T**écnicamente **O**rientado), iniciando-se com uma breve introdução, descrevendo o objetivo e as características gerais dessa ferramenta.

### V.1 - Introdução

A criação de um documento raramente pode ser visualizada como uma tarefa individual mas, em geral, envolve um grupo de pessoas onde o produto final é, quase sempre, resultado do esforço cooperativo dos participantes [Desl89].

O **DUETO** é uma ferramenta para edição cooperativa de textos desenvolvido para rodar em ambientes de trabalho cooperativo, com o objetivo de auxiliar os diversos grupos de usuários nas tarefas de criação conjunta de documentos (Co-Autoria).

O **DUETO** pode funcionar tanto em ambientes de redes locais, permitindo que os usuários interajam sincronamente em um mesmo documento, quanto em ambientes mono-usuários (“stand-alone”). No ambiente de redes, o editor se utiliza das facilidades do ambiente para prover sincronização das ações e permitir a troca de informações entre os participantes de um grupo de trabalho. No ambiente mono-usuário, o **DUETO** comporta-se como um editor convencional com algumas características a mais, tais como: inclusão de comentários, visualização da estrutura do documento, navegação a partir da estrutura do documento etc. Além disso, o editor não permite que o suporte à cooperação interfira no trabalho individual.

O editor utiliza-se da filosofia dos sistemas de hipertextos, apesar de não poder ser caracterizado como tal, para implementar algumas características importantes apresentadas por estes sistemas, com o intuito de dar maior liberdade aos co-autores no processo de criação de documentos.

O desenvolvimento de documentos baseado num modelo metódico e estruturado permite romper com a tendência do processamento sequencial, dando possibilidades para que os autores criem e acessem os documentos da maneira mais significativa para cada um deles.

A separação da estrutura do documento tem papel importante na autoria de documentos, principalmente em ambientes de trabalho cooperativo, pois permite a divisão das tarefas, no processo de desenvolvimento, entre os diversos co-autores.

A vantagem principal do editor é permitir a criação / edição simultânea de um documento, por diversos autores, coordenando suas ações e provendo meios para que os autores possam interagir entre si, durante a criação do documento.

Todo esse processo de coordenação e interação entre os colaboradores permite que a ferramenta inclua algum suporte relacionado aos aspectos sociais do trabalho cooperativo, em especial na definição de papéis sociais e na comunicação.

O desenvolvimento de ferramentas de edição cooperativa torna-se importante, apesar do pouco conhecimento, por parte dos desenvolvedores, sobre os aspectos sociais envolvidos em ambientes de edição cooperativa, pois ajudam no entendimento dos aspectos sociais do relacionamento e na definição de um conjunto de requisitos que possam suportar grupos de edição cooperativa.

## **V.2 - Especificação da Ferramenta**

A especificação do **DUETO** está descrita nas seguintes sub-seções: Requisitos Funcionais do **DUETO** e Modelo de Dados.

### **V.2.1 - Requisitos Funcionais do DUETO**

A partir da literatura pesquisada, foram selecionados os requisitos que consideramos que um editor de textos deve conter de forma que possa ser caracterizado como uma ferramenta cooperativa. Baseados nestes requisitos, foram descritas as características necessárias para o editor **DUETO** vir a ser utilizado em um ambiente de autoria cooperativa.

O **DUETO** está projetado para ser uma ferramenta que suporta a criação de documentos cooperativamente, tanto em ambiente de redes locais como em ambientes mono-usuário (“stand-alone”), neste último caso, funcionando como um editor de textos convencional.

## - **Cooperação Concorrente**

Hoje em dia, é comum, em diversos ambientes de trabalho e de pesquisa, haver diversas pessoas trabalhando juntas na criação de documentos. A cooperação concorrente envolve compartilhamento de informações entre os diversos grupos de usuários distribuídos, de forma a trocarem informações sobre as atividades que estão sendo desenvolvidas.

Mais recentemente, têm-se desenvolvido ferramentas que permitem aos usuários cooperarem entre si. Entre estas ferramentas, destacam-se os editores cooperativos de textos que permitem que diversas pessoas possam compartilhar textos na criação conjunta de documentos.

Um cenário típico da produção conjunta de um documento é um encontro inicial entre os co-autores, com intuito de produzirem um esboço inicial do documento e definirem responsabilidades sobre as partes dos documentos. Após isso, os participantes trabalham em paralelo, mas relativamente isolados, para produzirem as partes pelas quais são responsáveis. Finalmente estas partes são englobadas para constituírem o documento final [Newm91].

O **DUETO** é um sistema multi-usuário e, portanto, deve permitir a cooperação concorrente entre os diversos autores, no processo de autoria de documentos, independentemente de sua localização física.

Um documento pode ser acessado concorrentemente e seus autores podem editar diferentes componentes da estrutura do documento, simultaneamente. O sistema deve ser capaz de controlar o acesso dos participantes aos componentes do documento, mantendo as atualizações disponíveis no menor espaço de tempo possível.

## - **Modularização das Informações**

Modularidade é o resultado de projetar programas compostos por pequenos pedaços (módulos) de código, com os quais o software vai sendo construído.

A modularização provê um ambiente para integrar o processo de engenharia de documentos. O estágio de projeto produz um escopo modular onde os módulos ordenados do documento formam uma estrutura, tipicamente hierárquica. Convencionalmente, descreve-se a estrutura lógica dos

documentos como árvores, geralmente consistindo de capítulos, seções, sub-seções e outros componentes que arranjados dão sentido ao documento [Walk88b].

Um módulo num documento é um bloco de informações construído semanticamente, ao invés de ser um módulo físico arbitrário (p.ex. página ou tela). Por isso, os autores necessitam gerenciar os elementos significativos de um documento independentemente da especificação de sua aparência física.

O maior poder de um ambiente de desenvolvimento de software vem do seu conhecimento dos componentes dos sistemas de software. Análogamente, um ambiente de desenvolvimento de documentos deve estar baseado no conhecimento dos componentes e na maneira como estão conectados.

Num ambiente de autoria de documentos, o desenvolvimento modular dá oportunidade para a re-utilização de módulos, permitindo a criação de novos documentos baseados nos já produzidos (prototipagem). Com isso, novas organizações (protótipos) podem ser criadas e mantidas em paralelo.

Em ambientes de trabalho cooperativo, a modularização aumenta a cooperação entre os autores porque os módulos podem ser utilizados para decompor o documento, em partes independentes, e dividi-los entre os diversos autores.

O editor **DUETO** deve tratar o documento como um conjunto de segmentos (componentes) lógicos, independentes e interligados implicitamente para formar a árvore hierárquica do documento (estrutura do documento). Cada um destes módulos é constituído por um conjunto de informações que podem corresponder às diversas partes do documento ou à comentários anexos a estas partes. Estes módulos podem ser criados e/ou acessados por qualquer participante do documento.

#### - Acesso Não-Sequencial às Informações

Para [Smith87], o processo de autoria passa por diferentes atividades cognitivas, pois não é só traduzir idéias em palavras, mas identificar relações associativas entre as idéias, extraindo inferências, fazendo deduções, construindo grandes estruturas hierárquicas.



Alguns sistemas, tais como os sistemas de hipertextos, permitem organizar as idéias humanas, porque a sua estrutura semântica é baseada numa estrutura em rede que mapeia o pensamento humano. A criação desta rede de idéias faz com que estes sistemas dêem liberdade aos autores para “navegarem” ao longo dos documentos com o intuito de terem acesso ao conjunto de informações armazenado.

Alguns estudos têm revelado que apesar da leitura efetuada diretamente na tela do computador ser mais lenta que em papel, em sistemas que permitem o acesso aos textos de maneira não sequencial, este processo pode ser mais rápido pois torna a procura de texto relevante mais rápida e, conseqüentemente, pode-se extrair a informação relevante sem ter que ler muito.

A forma natural pela qual os usuários pesquisam/acessam informações, em editores convencionais, é efetuada através da pesquisa em uma parte do documento ou em todo o documento, por meio de palavras, frases ou trechos de textos, fornecidos pelos usuários.

Com isso, os sistemas cooperativos que auxiliam os usuários no processo de autoria de documentos podem ajudar os usuários a navegarem ao longo do conjunto de idéias gerado, permitindo, não só a pesquisa convencional, mas também a pesquisa ao longo das idéias geradas pelos participantes.

O editor **DUETO** deve auxiliar os participantes no acesso às informações, a partir dos componentes constituintes da estrutura dos documentos. O acesso aos componentes deve ser efetuado a partir de uma lista contendo o esboço do documento, apresentada em janelas específicas da interface, onde é visualizada a estrutura hierárquica do documento. Além disso, deve também permitir a pesquisa das informações pelos meios convencionais.

A liberdade de navegação é dada a todos, mas, com intuito de se evitar o acesso a informações consideradas não pertinentes a determinados usuários, devem existir mecanismos que possam auxiliar os usuários mais inexperientes na navegação ao longo do documento.

Essa liberdade na navegação pode causar problemas na orientação dos usuários ou problemas para encontrar a informação necessária, mesmo quando em pequenos textos. Em geral, os sistemas de hipertextos têm se preocupado com esses aspectos e, por isso, fornecem diversos mecanismos para orientar os usuários na navegação dos hiperdocumentos.

Tais mecanismos incluem mapas gráficos que exibem toda a estrutura dos documentos, listas para a visualização da estrutura dos hiperdocumentos e marcadores que são marcas inseridas nos nós dos documentos visitados.

É importante para os editores cooperativos, que tratam os documentos da mesma maneira como são tratados pelos sistemas de hipertexto (nós ou módulos de informações), que possam fornecer mecanismos que permitam aos diversos participantes tomarem conhecimento de sua localização, isto é, que ajudem os participantes a entender a sua localização corrente em relação à estrutura geral do documento e, também, permitir entender seus próprios movimentos.

Além da área para visualização da estrutura do documento, o editor **DUETO** deve exibir um mapa gráfico para apresentar toda a estrutura do documento de forma que o usuário possa se localizar na árvore.

Um outro mecanismo utilizado por alguns sistemas, em especial os sistemas de hipertextos, é a especificação de caminhos ou trilhas que possam ser utilizados como padrão de navegação para usuários que realizam uma pesquisa no documento.

O **DUETO** deve permitir a criação de caminhos (Trilhas), que representam seqüências pré-definidas de componentes dos documentos, para que possam ser acessados posteriormente, pelos leitores.

### - Adequação da Interface

Em geral, a interface corresponde aos recursos visuais e de manipulação oferecidos aos usuários de forma que os sistemas forneçam melhores condições na interação com os usuários.

A interface do editor **DUETO** deve estar baseada na metáfora **WYSIWIS** (What You See Is What I See), descrito em II.4.1, que permite a visualização do que cada participante está fazendo. A visualização pode ser efetuada na janela de edição, quando esta não possuir informações (por exemplo: um texto sendo editado). Neste caso, o texto é exibido na coloração cinza.

O editor também deve permitir a visualização das ações dos outros usuários em outras janelas da interface (interface multi-janelas), quando a janela de edição apresentar algum texto sendo

editado. Neste caso, o texto exibido nestas janelas deve ser visualizado na coloração normal, pois estas janelas não sofrem atualização, servindo apenas para a visualização instantânea (“snapshot”) do conteúdo dos componentes acessados por outros usuários.

Um aspecto importante na apresentação do texto em várias janelas é o número de janelas abertas simultaneamente na interface, que pode fazer com que o usuário perca o contexto geral das informações recuperadas.

Mesmo assim, o editor **DUETO** não deve limitar o número de janelas apresentadas e estas devem ser apresentadas umas sobrepostas a outras, de modo que o usuário não perca o contexto e tenha facilidades no relacionamento das informações.

Um problema com a apresentação de texto na tela de um computador refere-se à quantidade de informação que deve ser apresentada ao usuário. O ideal é não permitir a existência de janelas muito grandes para apresentação dos textos, para não atrapalhar o usuário na recuperação das informações. Quando as informações apresentadas nas janelas forem maior que o tamanho destas, o editor deve prover meios para que o conteúdo da informação possa ser rolado (“Scroll bars”), dando uma maior flexibilidade ao autor da informação. Quando houver mudanças nas dimensões da janela, o editor deve ser flexível o bastante para permitir que o texto possa ser reformatado (“Word-Wrap”) automaticamente.

O editor **DUETO**, portanto, não deve limitar o tamanho das informações apresentadas nas janelas, pois o usuário deve ter liberdade para criar a informação que necessitar. Para isso, o editor deve permitir que o texto possa rolar ao longo da janela. Além disso, as janelas podem ter as suas dimensões alteradas, portanto o editor deve efetuar a reformatação automática do texto para refletir estas modificações.

## **- Intercâmbio de documentos**

Dos editores cooperativos de textos e hipertextos para edição cooperativa pesquisados na literatura, a sua grande maioria não apresenta recursos que possibilitem melhorar a qualidade do produto final (apresentação impressa do documento).

Por isso, é de extrema importância que os editores possam exportar os documentos nos formatos dos editores que possuam bons recursos de editoração, para impressão do produto final dos documentos.

O **DUETO** deve prover meios que permitam exportar documentos para os formatos dos principais editores de textos comerciais.

### - Suporte aos Aspectos Sociais da Cooperação

O compartilhamento explícito de informações faz com que os autores necessitem coordenar suas ações. Eles devem coordenar suas atividades de maneira que o trabalho efetuado por um participante não impeça o progresso de outros [Fish88].

A coordenação das ações num ambiente cooperativo pode ser vista por uma perspectiva social, pois surge a necessidade de definição das tarefas para os diversos participantes, visto que eles estarão trabalhando em conjunto e não mais sozinhos. Por isso, cada participante num ambiente cooperativo desempenha um “papel social”, isto é, possui responsabilidades e funções próprias sobre os objetos de seu trabalho.

A maneira como os participantes compartilham as informações, coordenam suas atividades e supervisionam o trabalho de cada participante está diretamente relacionada à natureza do relacionamento interpessoal entre os participantes, o estágio do projeto e o papel social desempenhado por cada um [Fish88].

Em resposta aos problemas da coordenação, o **DUETO** deve permitir a definição de papéis sociais dos vários colaboradores em um documento. Os participantes devem possuir diferentes responsabilidades e autorizações de acesso que determinam as atividades a serem executadas no documento.

O editor suporta quatro (4) diferentes papéis sociais, a saber: Criador, Autor, Revisor e Leitor.

Todo documento possui um Criador que é o responsável pela sua criação e iniciação. O Criador é um autor que possui responsabilidades de definir quem são os colaboradores e que papéis terão no acesso ao documento. Além dessa função, o Criador é o responsável por criar as diferentes

configurações do documento que representam a evolução do documento, por definir os caminhos (Trilhas) a serem seguidos pelos diferentes Leitores, e mais, pode fazer qualquer tipo de comentário e emitir mensagens aos outros participantes do documento.

O Autor ou Co-Autor é o participante direto do editor na execução das atividades de autoria dos documentos. Os Autores possuem permissão para criação dos componentes da estrutura dos documentos e são responsáveis pela edição dos textos que compõem o corpo dos componentes; além disso, podem criar contextos para modificação de um conjunto de componentes, podem anexar qualquer comentário e emitir mensagens aos demais participantes.

O Revisor é um participante que possui permissão para leitura do documento, com intuito de anexar comentários ou sugestões aos componentes do documento.

O Leitor, como o Revisor, somente tem acesso à leitura do documento. O Leitor pode “navegar” ao longo da estrutura do documento, a partir de caminhos pré-definidos pelo Criador, ou pode optar por seguir uma trilha própria. Os Leitores têm permissão apenas para o envio de mensagens diretas aos demais participantes do documento.

## **- Controle de Concorrência**

Em sistemas de groupware, torna-se necessário a implementação de mecanismos que permitam controlar as ações efetuadas concorrentemente, como forma de ajudar na resolução de conflitos que, geralmente, ocorrem entre os participantes.

Por ser um editor cooperativo que suporta a edição concorrente, o **DUETO** deve implementar um modelo de controle de concorrência que permita prevenir situações conflitantes. O modelo a ser implementado pelo editor é o bloqueio simples (lock) dos componentes da estrutura do documento, durante a edição e modificação do componente por um determinado Autor. Um Autor tem garantido o bloqueio durante a edição do componente, permitindo aos outros participantes apenas o acesso para leitura do componente.

O editor deve informar aos outros participantes que um determinado componente encontra-se bloqueado através de uma marca inserida no início do nome do componente, visualizada nas janelas que contêm a lista com a estrutura do documento.

Além disso, o editor também deve implementar o conceito de contextos, como outro modelo para controle de concorrência. Um contexto é formado por um sub-conjunto de componentes da estrutura do documento. A finalidade da criação de contextos é garantir, ao Autor, a propagação das alterações para os diversos componentes do contexto em resposta às modificações efetuadas em um componente. Com isso, o acesso aos componentes do contexto é bloqueado a outros Autores, garantindo ao Criador do contexto a continuidade das ações.

O editor deve suportar múltiplos contextos, onde cada contexto é formado por um conjunto de componentes distintos em relação aos outros contextos.

### - Suporte à Comunicação entre os Participantes de um Documento

O compartilhamento de informações em um ambiente de autoria cooperativa de documentos (Co-Autoria) deve ir além da edição conjunta de textos. A comunicação entre os participantes para troca de informações é de vital importância no processo de edição cooperativa.

A comunicação pode resultar num melhor relacionamento entre os participantes, pois permite que estes possam trocar informações sobre as tarefas que estão sendo desenvolvidas e emitir sugestões sobre o trabalho desenvolvido.

O editor **DUETO** deve prover facilidades para que os usuários possam se comunicar através de mensagens, durante o processo de edição conjunta. Além disso, o editor deve suportar a inclusão e recuperação de comentários/anotações nos componentes da estrutura dos documentos de modo que Revisores, e mesmo Autores, possam fazer sugestões sobre o conteúdo dos componentes.

Segundo Neuwirth [Neuw90], muitas sugestões podem não ser facilmente descritas, por isso alguns Autores preferem reescrever pedaços de texto ao invés de comentá-los. Para tal problema, o **DUETO** deve permitir que os comentários/sugestões, feitos pelos Revisores, possam ser transferidos para o corpo do texto, evitando que o autor tenha que escrevê-lo novamente.

A seguir descrevemos as facilidades providas pelo editor para troca de informações entre os participantes.

## **MENSAGENS DIRETAS:**

São mensagens dirigidas a um participante ou a um grupo de participantes. O sistema não armazena este tipo de mensagem, servindo apenas para troca rápida de informações durante uma sessão de edição. As mensagens são enviadas imediatamente após sua transmissão.

## **COMENTÁRIOS:**

São sugestões efetuadas por Revisores e anexadas ao componente da estrutura para o qual foi dirigida. O **DUETO** suporta dois (2) tipos de comentários, a saber:

### **PÚBLICOS:**

São comentários que podem ser lidos por qualquer participante e somente são retirados pelo Criador do documento.

### **PRIVADOS:**

São comentários visíveis apenas pelo participante para o qual o comentário foi direcionado. Este tipo de comentário somente pode ser retirado pelo receptor e o seu conteúdo pode ser transferido para o interior do componente.

## **- Controle de Versões**

Em ambientes cooperativos, onde a ênfase reside no compartilhamento de informações, surgem diversas situações de conflitos pois podem ocorrer algumas mudanças nos objetos sem que determinados participantes tomem conhecimento.

Como descrito em II.4.3, em ambientes cooperativos para autoria de documentos o armazenamento e manipulação de diferentes versões dos objetos torna-se imprescindível pois permite que os participantes possam acessar diferentes estágios na evolução dos documentos. Além disso, a comparação entre a versão atual dos objetos e as anteriores permite que os participantes, que

por algum motivo estiveram ausentes, tomem conhecimento das mudanças que ocorreram nesses objetos.

O **DUETO** deve prover meios para o armazenamento de múltiplas versões dos componentes da estrutura dos documentos e, conseqüentemente, permitir o acesso a versões anteriores dos componentes. O suporte para o gerenciamento dos documentos também deve permitir o relacionamento entre as versões dos componentes e, caso necessário, retornar a versões anteriores dos componentes se as mudanças subsequentes foram inadequadas. As diferenças entre as versões podem ser visualizadas por diferentes colorações.

Além disso, o sistema deve permitir que o Criador do documento possa armazenar diferentes configurações do documento, representando estados do documento. As configurações são formadas pelas versões mais atuais dos componentes do documento, no momento da criação.

#### **- Gerenciamento Eficiente no Armazenamento e Recuperação de Documentos.**

A base para qualquer sistema de gerenciamento de documento é um eficiente suporte no armazenamento e recuperação dos documentos. O sistema deve prover um controle centralizado para suportar o gerenciamento de toda a informação necessária à reprodução do documento e, baseado num modelo de grupos, ser capaz de prover meios para gerenciar o acesso concorrente.

O **DUETO** deve possuir o suporte de um sistema gerenciador de banco de dados que permita armazenar/recuperar todas as informações, gerenciando todos os aspectos relacionados com o compartilhamento das informações e o controle de acesso, de forma a manter a integridade das informações manipuladas pelos participantes.

#### **- Histórico de Acesso ao Documento**

Em um ambiente cooperativo para edição de documentos, torna-se importante que os participantes tomem conhecimento das ações efetuadas por cada participante durante uma sessão e, além disso, também possam tomar conhecimento das ações efetuadas nos documentos, pois nem sempre todos os usuários estão concorrentemente no desenvolvimento dos documentos.

Para isso, torna-se necessário que os sistemas que suportam o trabalho cooperativo forneçam meios para informar todas as ações que estão ocorrendo ou ocorreram nos objetos.



O **DUETO** deve prover meios para armazenar as informações sobre as principais ações efetuadas no documento, visando manter o usuário consciente do que ocorreu com o documento durante a sua ausência. Qualquer alteração (p.ex. um componente foi apagado da estrutura do documento) que tenha ocorrido em sessões anteriores em que o usuário não estava presente deve ser informada pelo editor.

O editor também deve permitir que cada usuário visualize um histórico (lista de visitas) das suas ações sobre o documento, isto é, que componentes da estrutura do documento foram acessados e quanto tempo levou no acesso. Além disso, é importante manter durante uma sessão de co-autoria, um histórico de todos os usuários que estão acessando um documento e que tarefas estão efetuando.

### V.2.2 - Modelo de Dados

O modelo de dados do editor **DUETO** pode ser visualizado na figura V.2.2.1. Para apresentação do modelo foi adotado o Modelo de Entidades e Relacionamentos (MER), como proposto por [Chen86] e com capacidades estendidas [Setz86] [Nava89].

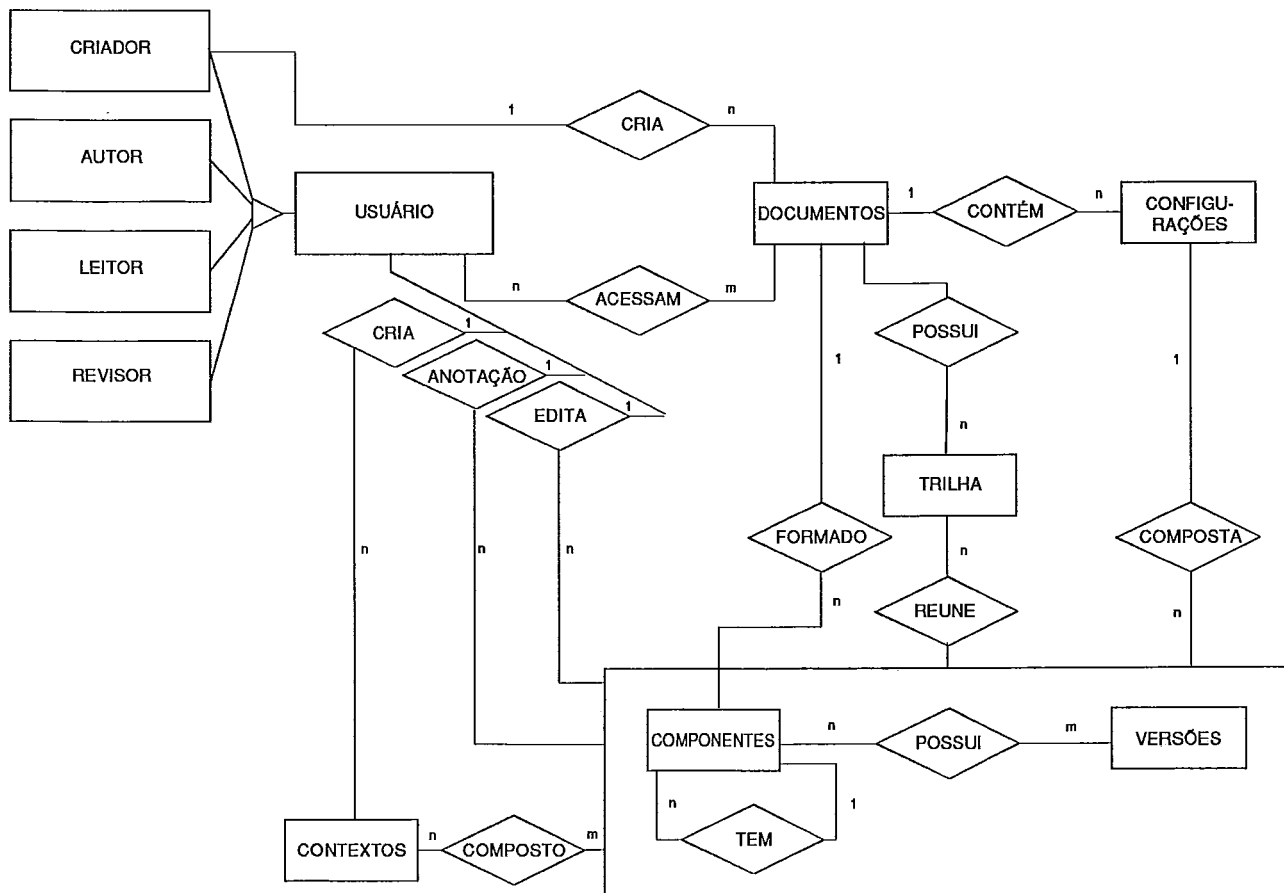


Figura V.2.2.1 - Modelo de Dados do DUETO

A seguir, descrevemos sucintamente o conjunto de entidades que ocorrem no modelo de dados (MER) do editor **DUETO**:

### **Documentos:**

Todo documento possui um usuário que é o Criador do documento. Um documento pode ser acessado por vários outros usuários dependendo do tipo de acesso permitido. Um documento é formado por componentes que compõem a estrutura hierárquica do documento. Um documento possui várias configurações que por sua vez são compostas de versões de componentes.

### **Usuário:**

Um usuário é uma pessoa autorizada a acessar um documento. Um usuário pode ser especializado em várias sub-classes, dependendo do nível de acesso: um usuário pode ser o Criador de um documento, que possui todos os direitos sobre o mesmo; ou o Autor de um documento, que possui direitos de edição/modificação do texto do documento; ou o Leitor de um documento, que possui o acesso apenas para leitura do documento; ou ainda o Revisor de um documento, que possui acesso para leitura e inserção de comentários no documento.

### **Componentes:**

O conjunto de componentes compõem a árvore do documento. Um componente pode ser constituído hierarquicamente por vários componentes. Um componente possui diversas versões, correspondendo aos diferentes estágios de seu desenvolvimento.

### **Versões:**

O conjunto de versões corresponde às diferentes representações dos componentes durante o ciclo de desenvolvimento do documento.

## **Configurações:**

O conjunto de configurações representa os diferentes estágios de desenvolvimento de um documento. Cada configuração corresponde a um estado do documento, formado por diferentes versões dos componentes.

## **Trilhas:**

As trilhas representam sequências pré-estabelecidas de componentes de um documento. São criados pelos criadores dos documentos com intuito de propor caminhos a serem percorridos pelos leitores.

## **Contextos:**

Representam um sub-conjunto de componentes dos documentos, criados por um autor, para impedir o acesso, aos demais participantes, enquanto os componentes do contexto estão sendo modificados pelo próprio autor.

A seguir, descrevem-se os relacionamentos entre as entidades componentes do modelo de dados.

No **DUETO**, um documento é criado por um usuário, denominado Criador, que não possui restrições no acesso ao documento. Além disso, outros usuários podem ter acesso ao documento, onde suas ações são proporcionais à permissão de acesso que o usuário possui no documento.

Cada documento é formado por diversos componentes que, por sua vez, são compostos por outros componentes, formando a estrutura hierárquica do documento. Cada componente possui diferentes estados de desenvolvimento representados por suas versões.

Um componente possui comentários/anotações anexas, efetuado por um usuário autorizado a fazê-los.

Um documento possui diferentes configurações que são compostas por diferentes versões dos componentes.

Um usuário possui contextos que são formados por um conjunto de versões de componentes, distintos em cada contexto.

Uma trilha reúne uma sequência de componentes de um documento e é utilizada pelos usuários para navegarem ao longo da estrutura do documento.

### **V.3 - Público-Alvo**

O editor **DUETO** está, inicialmente, projetado para grupos de pessoas envolvidas diretamente com a produção conjunta de documentos técnicos. Nestes grupos estão incluídos o pessoal do meio acadêmico como professores, estudantes de graduação, mestrado e doutorado e, também, pesquisadores das diversas áreas do conhecimento.

As tarefas desenvolvidas no processo de criação conjunta de documentos podem ser vistas como uma mistura de planejamento, geração de idéias, tomadas de decisões e resolução de conflitos cognitivos [Oslo92]. Devido à natureza destas atividades, acreditamos que um editor cooperativo possa diminuir os problemas que ocorrem no desenvolvimento conjunto quando são utilizadas ferramentas convencionais, pois permitem coordenar as atividades dos participantes, coletar e organizar suas idéias e permitem também o compartilhamento e edição dos objetos correntes de seus trabalhos.

A utilização, primeiramente no meio acadêmico, permitirá que sejam verificados os vários aspectos envolvidos no desenvolvimento conjunto de textos e servirá também como teste para avaliar-se o quanto tecnologia poderá afetar o trabalho dos participantes.

## VI - IMPLEMENTAÇÃO DO EDITOR DUETO

A implementação do editor **DUETO**, apresentada neste capítulo, é formada por: descrição das características implementadas, descrição da interface do **DUETO**, recursos computacionais utilizados, arquitetura do **DUETO**, armazenamento e exemplo de utilização.

### VI.1 - Características do **DUETO**

As características apresentadas nesta seção são baseadas nos requisitos funcionais (descritos em V.2.1), que um editor deve possuir para ser considerado uma ferramenta cooperativa para autoria de documentos.

A implementação destas características deu origem ao editor **DUETO**, um protótipo de uma ferramenta cooperativa destinada a autoria de documentos.

A seguir, descrevemos as características implementadas na primeira versão do Editor **DUETO**.

#### - Quanto aos documentos

Os documentos criados no editor **DUETO** são vistos como conjuntos de componentes (módulos) agrupados logicamente para formar a estrutura do documento. Os componentes da estrutura do documento podem ser compartilhados pelos diversos participantes, para edição e visualização dos textos.

O editor provê outro construtor denominado de Anotações que são trechos de textos que podem ser anexados aos componentes e servem para os Autores e/ou Revisores fazerem comentários sobre o conteúdo dos componentes.

#### - Quanto à funcionalidade

No **DUETO** não existem modos explicitamente diferenciados para leitura ou para edição. Estes modos estão implicitamente relacionados ao papel social desempenhado por cada participante num determinado documento e ao controle das ações concorrentes.

Um participante pode, dependendo do seu papel social, criar/editar componentes e anotações, definir trilhas, enviar mensagens aos demais participantes, criar contextos, etc.

A estrutura modular dos documentos permite que os participantes possam “navegar” ao longo da estrutura dos documentos e trilhas. Somente uma trilha pode estar ativa durante a navegação do documento.

No **DUETO** os contextos e componentes somente podem ser destruídos pelos seus criadores e as trilhas somente pelo Criador do documento. Por questões de controle de concorrência, um componente só pode ser apagado se ele e seus inferiores hierárquicos não estiverem sendo acessados, para edição, por outros participantes.

#### **- Quanto à segurança**

O controle de segurança implementado para acesso ao editor se restringe à segurança provida pela rede Novell [Nove90], onde são definidos grupos de usuários com acesso restrito a determinados diretórios da rede.

O controle de acesso aos documentos está implementado através da definição de 4 tipos de papéis sociais (Criador, Autor, Revisor e Leitor). As informações pertinentes (p.ex. nome e senha) para o editor efetuar o controle de acesso, são obtidas das informações armazenadas no servidor, pelo sistema da rede.

O Criador de um documento é o seu responsável e possui as funções de criar/editar componentes e anotações, definir os demais participantes do documento, criar configurações, destruir componentes, anotações e trilhas e enviar mensagens. O Autor pode criar/editar componentes e anotações, criar contextos, seguir trilhas e enviar mensagens aos demais participantes. O Revisor somente pode acessar os componentes para leitura, anexar comentários para revisão, seguir trilhas e enviar mensagens. O Leitor só tem acesso aos componentes e comentários públicos para leitura e pode seguir trilhas.

### - Quanto à concorrência

O editor **DUETO** implementa para controle das ações concorrentes, o mecanismo de bloqueio simples dos componentes dos documentos. Com isso, o editor garante que um documento bloqueado durante a edição não seja alterado por outros participantes do documento. O acesso por outros participantes, aos componentes bloqueados, somente é permitido para leitura.

O editor também implementa o conceito de contextos como um mecanismo de controle de concorrência permitindo, assim, que um conjunto de componentes possam ser bloqueados por um Autor para efetuar modificações que podem ser propagadas aos outros componentes.

### - Quanto à navegação

O editor permite que os participantes possam navegar ao longo do documento através de sua estrutura, apresentada na área de escopo (descrita em VI.2) da interface. A lista dos componentes é exibida nesta área, de maneira que os usuários possam visualizar a hierarquia do documento. Deste modo, os usuários podem se orientar ao longo da navegação.

Para esta versão, nenhum outro mecanismo que possibilite orientar os participantes durante a navegação foi implementado.

### - Quanto à interface

A interface do editor **DUETO**, descrita em VI.2, está baseada na interface provida pelo ambiente **WINDOWS**. A interface está projetada de modo a permitir que as ações efetuadas concorrentemente, sejam refletidas nas telas dos demais participantes no menor espaço de tempo possível. Com isso, o editor garante que cada participante tome consciência das ações dos demais usuários.

O editor informa aos demais participantes que um componente está bloqueado, através de marcas inseridas na frente do nome do componente exibido na área de escopo do editor. E o texto do componente bloqueado é visualizado nas janelas de edição dos demais participantes na coloração cinza.

O editor não limita o número de janelas apresentadas na interface do usuário e estas podem ter as suas dimensões alteradas de acordo com as preferências de cada participante. Para esta versão não foi implementado nenhum mecanismo que permita reformatar automaticamente o texto do componente, apresentado nas janelas, quando estas tiverem suas dimensões modificadas. O conteúdo do componente pode ser maior do que é mostrado na janela do usuário pois o editor provê meios para que a informação seja rolada (“Scroll Bars”).

#### **- Quanto à comunicação**

O editor DUETO provê meios para os usuários se comunicarem através de mensagens diretas e comentários que podem ser públicos ou privados.

As mensagens diretas não são retidas pelo editor servindo apenas para troca de informações e os comentários são anexados aos componentes a que foram destinados, para serem recuperados posteriormente.

#### **- Quanto ao intercâmbio de documentos**

Para esta versão o DUETO não provê meios para exportar os documentos para formatos dos principais editores de textos, permitindo somente que os documentos sejam exportados no formato ASCII.

#### **- Quanto ao armazenamento**

Para a primeira versão do DUETO, foi implementado um sistema de recuperação e armazenamento baseado no sistema de arquivos convencionais, com todo o controle efetuado pelas estações (módulo gerenciador de arquivos - ver VI.4), e com um suporte de rotinas específicas [Nove89] para o gerenciamento do acesso concorrente.

#### **- Quanto ao controle de versões**

O DUETO cria automaticamente as versões dos componentes dos documentos mas, para esta versão, não foi implementado nenhum mecanismo que permita a manipulação das diferentes versões



dos componentes.

O editor também permite que as diferentes versões dos componentes sejam armazenadas para formar as configurações dos documentos. O DUETO também não provê nenhum mecanismo para a manipulação das configurações criadas.

#### - Quanto ao histórico de acesso

Para esta versão, o DUETO apenas provê um histórico das ações que estão sendo efetuadas concorrentemente, permitindo assim que cada usuário tome conhecimento das ações dos demais.

## VI.2 - Interface do DUETO

A implementação da interface do DUETO (figura VI.2.1) teve como foco a representação visual para facilitar a manipulação dos objetos e simplificar sua aprendizagem. A interface é composta por um conjunto de objetos (janelas, botões e cardápios) e está baseada na interface do ambiente MS-WINDOWS [Micr87].

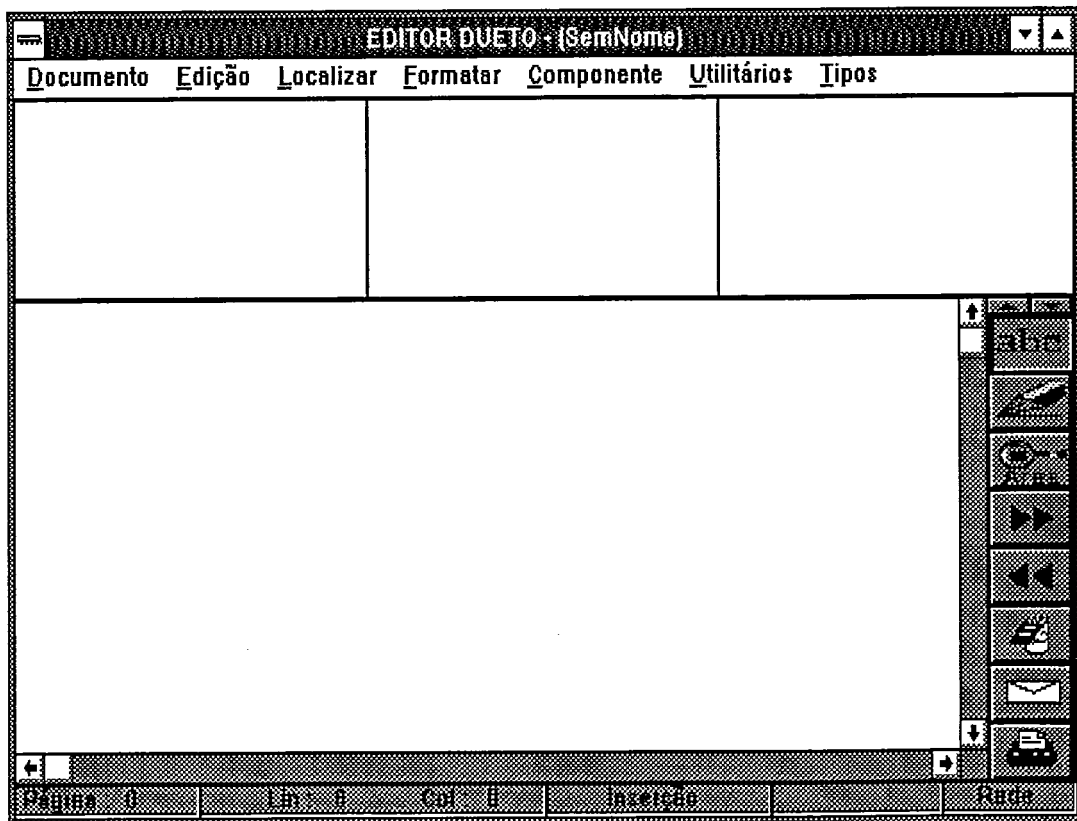


Figura VI.2.1 - Tela Principal do Editor DUETO

A seguir, descreve-se cada um dos objetos disponíveis na interface.

## **Janelas**

O editor **DUETO** apresenta três áreas distintas na sua interface: área de edição, área de escopo e área de informações.

A área de edição é composta por uma janela que permite a execução de todas as tarefas relacionadas com a edição e apresentação de textos dos componentes do documento.

A área de escopo é constituída por um conjunto de três janelas que permitem a visualização e seleção dos componentes da estrutura do documento. A seleção de um ítem em uma determinada janela faz com que sejam apresentados na janela seguinte, todos os componentes hierarquicamente inferiores ao item selecionado. Caso seja necessário, os componentes exibidos nas janelas são movidos à esquerda para apresentação dos níveis mais inferiores, ou à direita quando retornamos aos níveis superiores.

A área de informações é formada por uma janela destinada a exibir informações úteis para os usuários sobre a edição dos componentes, tais como: o número da página do documento que está sendo visualizada; a posição (linha e coluna) do cursor; o modo de edição (inserção/edição); informação sobre o modo dos caracteres maiúsculo/minúsculo (tecla Caps Lock) e informações sobre o modo de execução do editor (rede ou mono-usuário).

A maioria das funções do editor exhibe janelas específicas para execução do diálogo com os usuários.

## **Botões**

O **DUETO** possui um conjunto de doze (12) botões, em sua maioria associados às opções do cardápio para execução das funções do editor.

A área da interface, onde se encontram os botões, não possui espaço suficiente para apresentação de todos os botões. Em virtude disto, o editor exhibe dois botões situados na parte superior desta área que possuem a função de efetuar o rolamento dos botões para apresentação dos botões não-visíveis.

A seguir, descrevem-se os botões que não apresentam associação com as opções do cardápio:

### Zoom de área

Este botão permite alterar o tamanho da área de edição do **DUETO**. Dependendo do tamanho da área o botão tem um figura diferente para representá-la. A figura VI.2.2 mostra um exemplo deste botão.

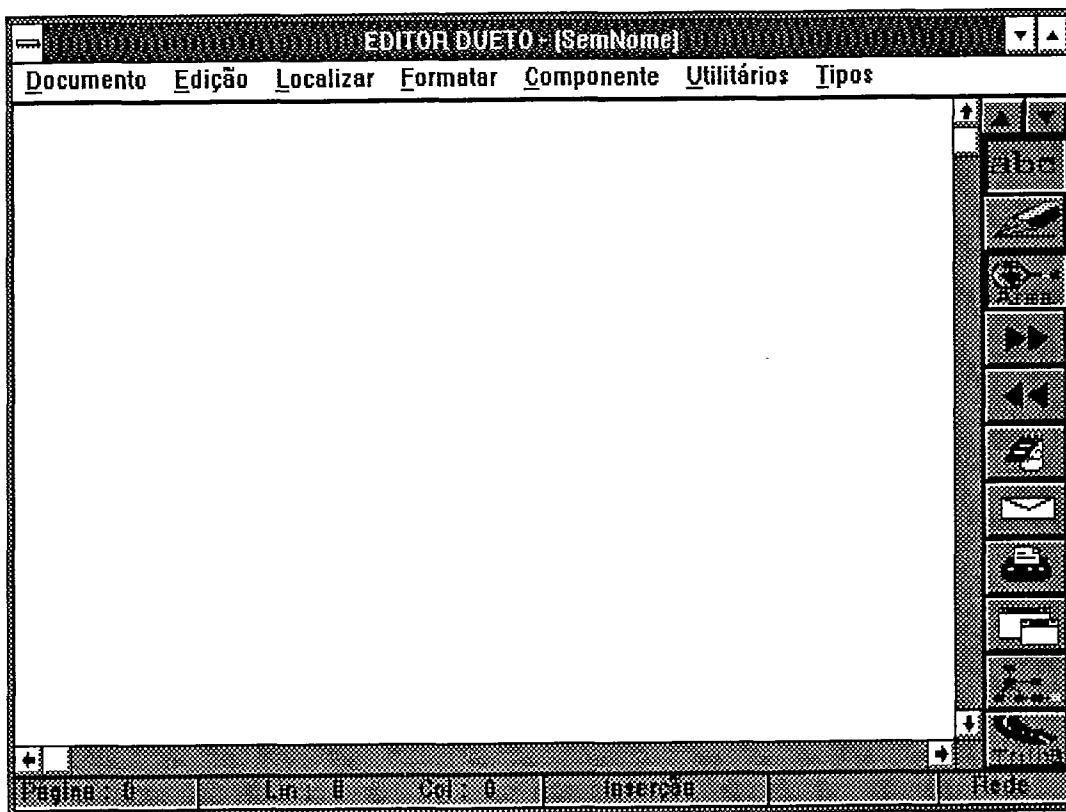


Figura VI.2.2 - Alteração de tamanho da área de edição.

### Ir para Frente

Este botão permite que os leitores possam mover-se para frente (níveis hierárquicos mais inferiores) na trilha.

### Ir para Trás

Este botão permite que os leitores possam retornar (níveis hierárquicos superiores) na trilha.

### Linha

Este botão permite o desenho de linhas para criação de tabelas, diagramas etc.

## Cardápios

O editor possui uma barra de cardápio que permite a seleção e execução das diversas funções implementadas pelo editor.

A seguir, descrevem-se todas as opções do cardápio apresentadas pelo editor:

### Opções do Cardápio:

Opção: **Documento**

Opções Subordinadas:

**C**riar

**A**brir

**F**echar

**S**alvar

**I**mprimir

**E**xportar

**T**erminar

A figura VI.2.3 mostra o cardápio “pull-down” da opção Documento.

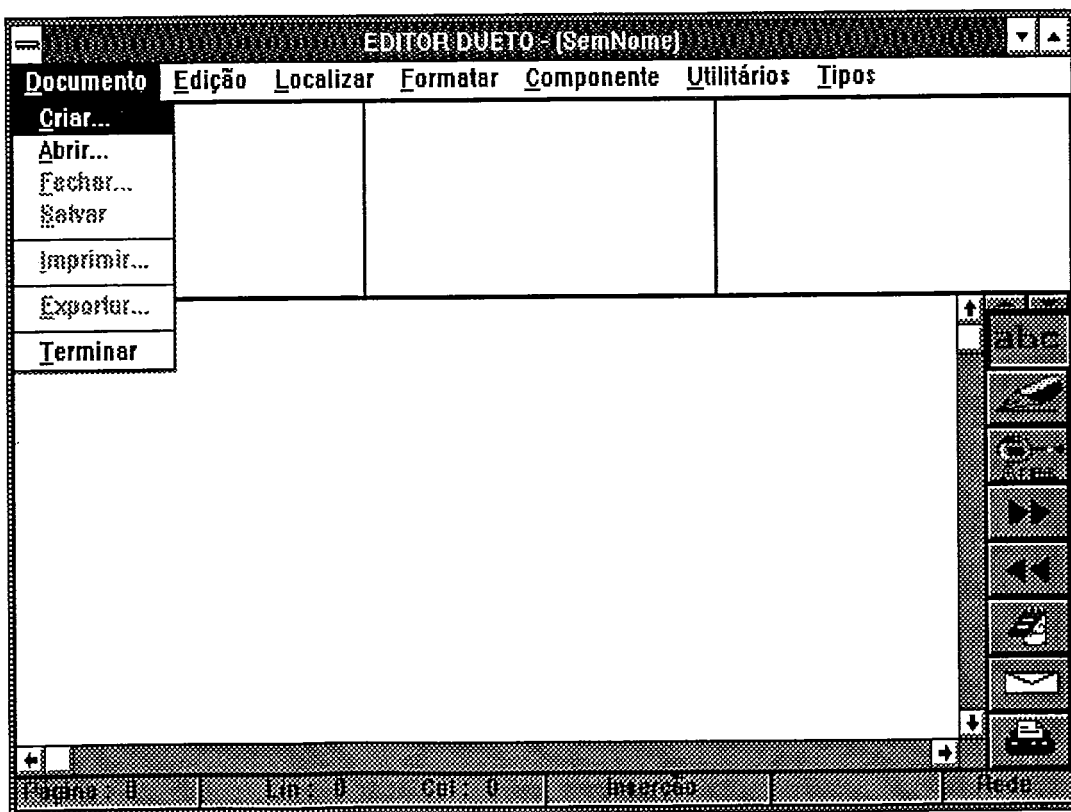


Figura VI.2.3 - Cardápio da opção Documento

Descrição:

### Documento/Criar

Permite que o usuário crie um novo documento no **DUETO**. A criação de um documento associa ao usuário, automaticamente, o papel de criador do documento. O documento criado é armazenado no diretório onde localizam-se todos os arquivos do editor.

A criação de um documento faz com que o editor apresente uma tela que exhibe os campos nome do documento e criador do documento (figura VI.2.4). O usuário necessita digitar apenas o nome a ser dado ao documento. Em ambiente de rede o nome do criador é recuperado, automaticamente, pelo **DUETO** na sua inicialização. Em ambiente mono-usuário o editor exhibe, na inicialização, uma tela para que o usuário se identifique no **DUETO**.

Um documento criado no **DUETO** permanecerá indisponível aos demais usuários até que seja liberado pelo seu Criador através da função **Documento/Salvar** (descrita a seguir). Isto permite que o criador possa definir os direitos de acesso (papéis) dos demais autores e, caso queira, propor um esboço inicial para o documento.

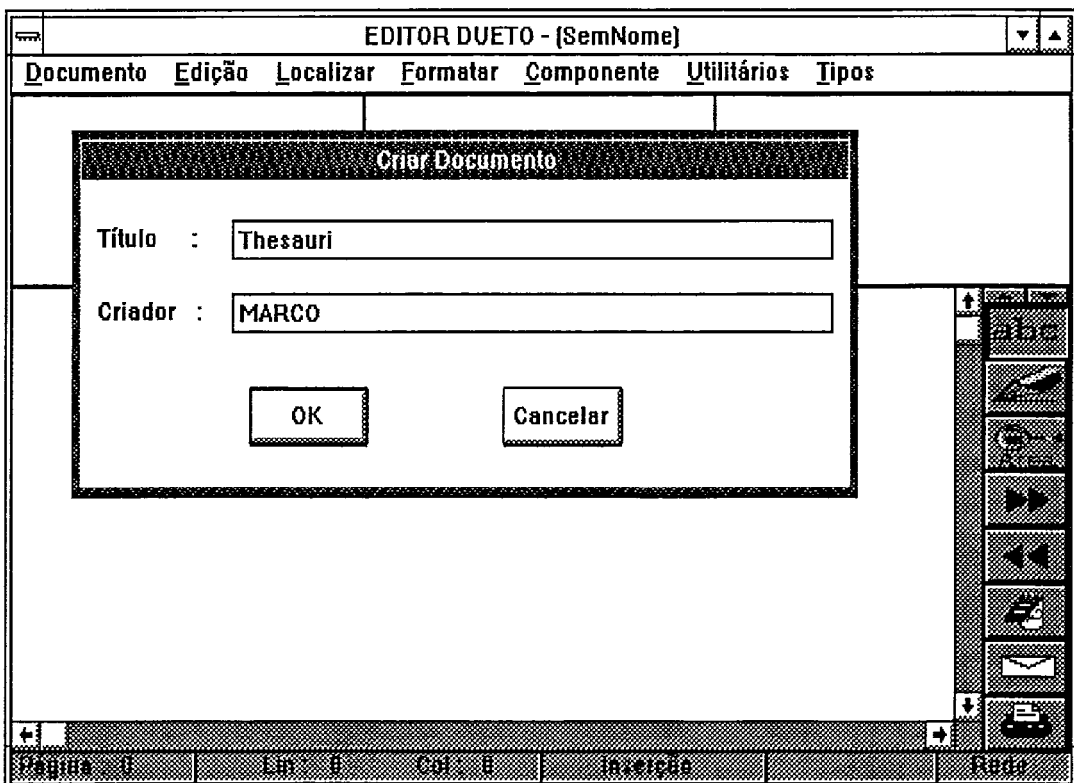


Figura VI.2.4 - Tela para criar documento.

## Documento/Abrir

Permite que os usuários autorizados a utilizar um determinado documento possam torná-lo disponível para uso. A escolha desta opção faz com que o editor exiba uma tela com todos os documento que o usuário está autorizado a acessar, juntamente com o seu papel, definido para cada documento.

A seleção de um documento faz com que o editor exiba informações sobre o documento que está sendo selecionado, tais como: Título, Configuração, Data de Criação, Data de Atualização e o Número de Usuários que estão, naquele momento, acessando o documento. Além disso, os componentes do primeiro nível (Nível 0) da estrutura do documento são exibidos na primeira janela da área de escopo do editor.

As figuras VI.2.5, VI.2.6 e VI.2.7 mostram a abertura de um documento **DUETO**.

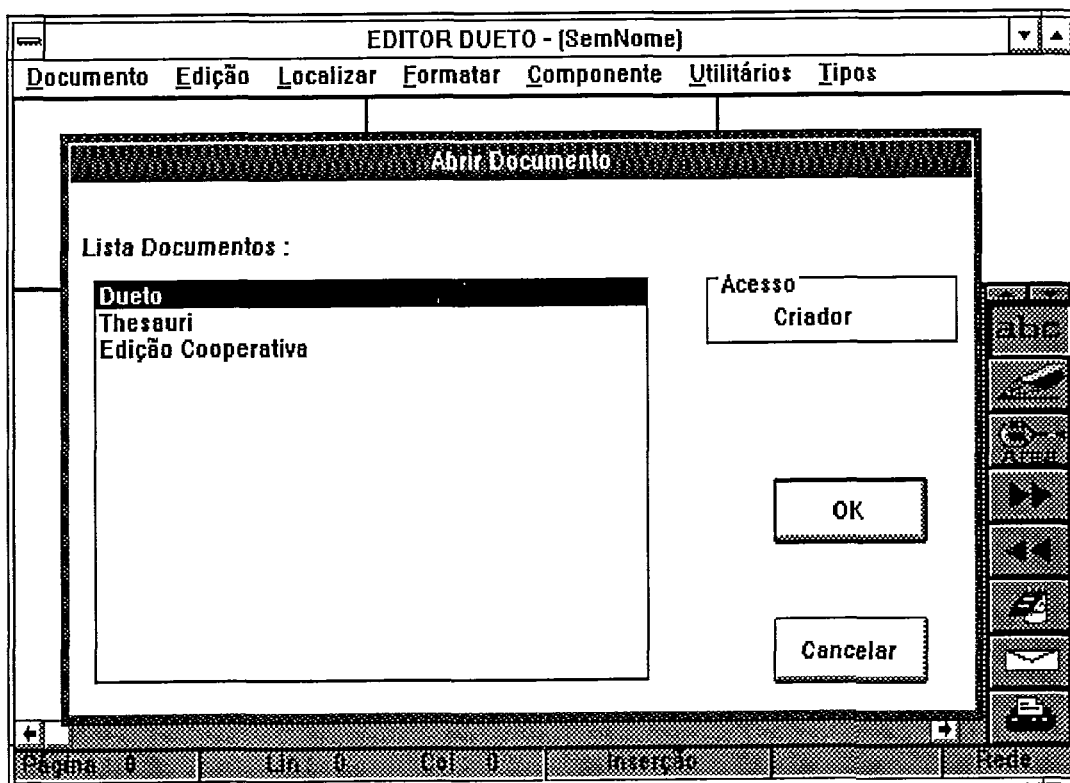


Figura VI.2.5 - Tela de abertura de um documento.

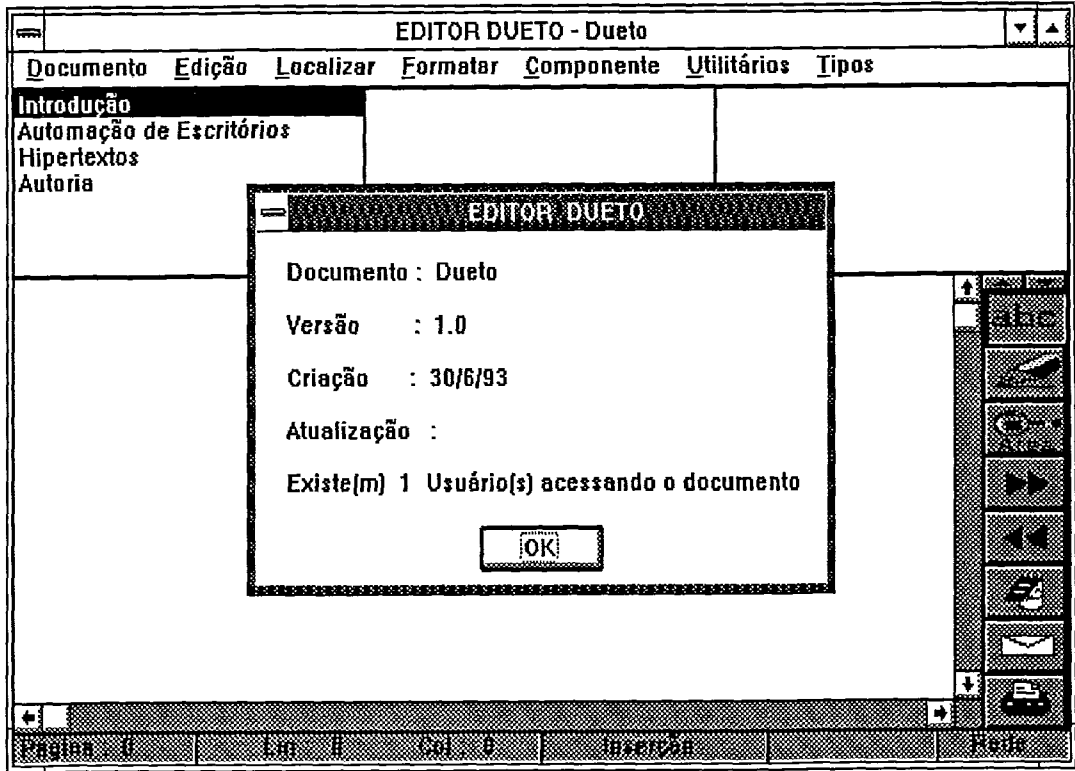


Figura VI.2.6 - Tela com informações do documento.

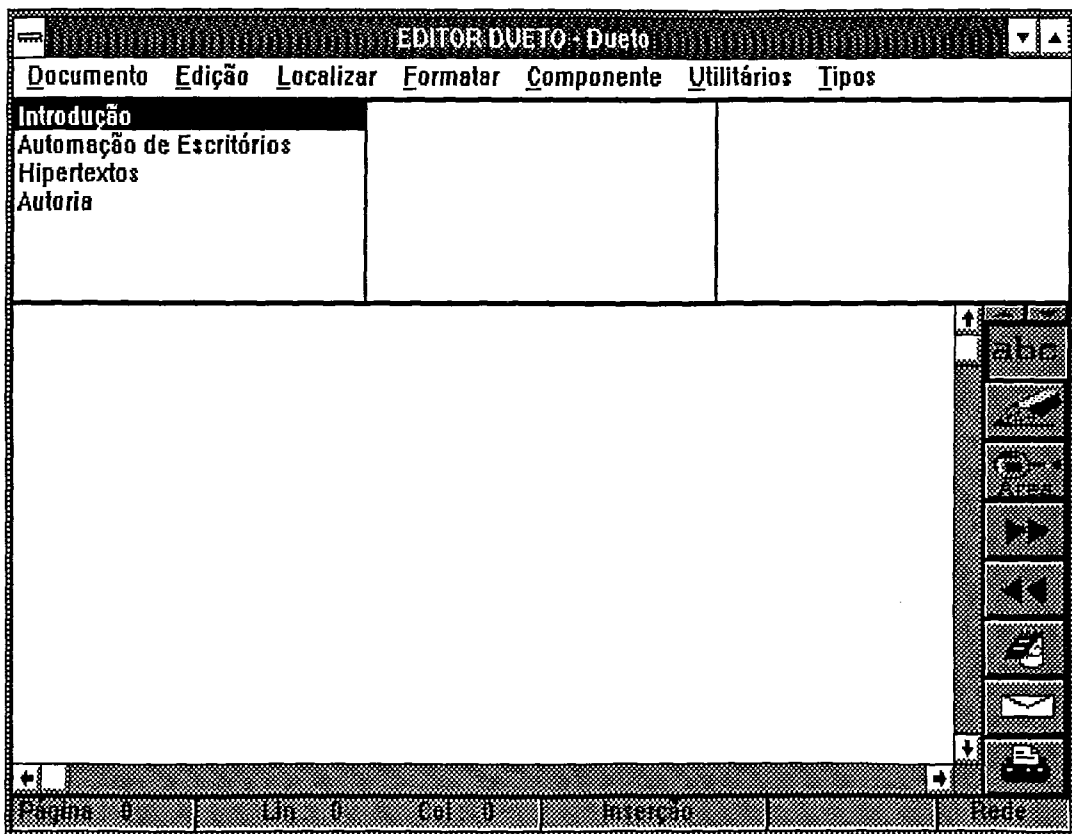


Figura VI.2.7 - Documento aberto no DUETO.

## Documento/Fechar

Dependendo do contexto onde se esta inserido, esta opção possui duas funções: fechar um documento ativo no **DUETO** e finalizar uma sessão de edição de um componente do documento. Esta opção faz com que o editor abandone o modo em que se encontra.

No fechamento de um documento, dependendo do papel que o usuário possui em relação ao documento, esta opção faz com que o editor apresente uma tela pedindo a confirmação para atualização na base, das possíveis alterações que ocorreram enquanto o documento esteve ativo. Nesta tela existe um botão que, ao término de uma sessão de edição, pode ser selecionado para criar uma nova versão do componente editado, se tiver ocorrido alguma alteração no componente.

A figura VI.2.8 ilustra esta opção.

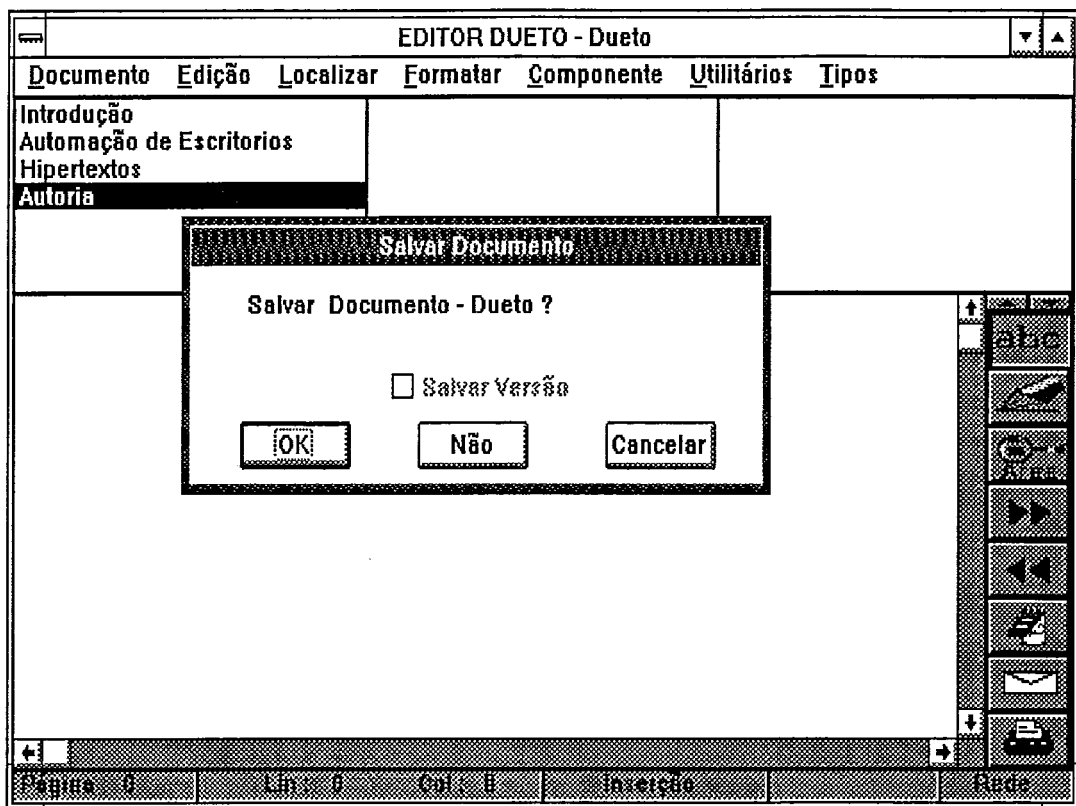


Figura VI.2.8 - Tela da opção Documento/Fechar.

## Documento/Salvar

Esta opção tem função semelhante a anterior, mas o editor não abandona o modo em que se encontra. Esta opção permite além de salvar as alterações efetuadas, torná-las disponíveis aos outros usuários, que estão acessando o documento. Nesta opção não existe a possibilidade de se criar uma nova configuração para o documento nem uma nova versão para o componente.



## Documento/Imprimir

Permite imprimir o documento completo ou componentes da estrutura do documento. Esta opção tem função semelhante aos editores de textos convencionais.

## Documento/Exportar

Permite que um usuário exporte um documento **DUETO** para um arquivo ASCII, de maneira que possa, se desejar, imprimi-lo a partir de um editor convencional que lhe ofereça maiores recursos de formatação. Esta função exibe uma janela de diálogo para o usuário informar o nome do arquivo ASCII a ser criado.

A figura VI.2.9 mostra esta função.

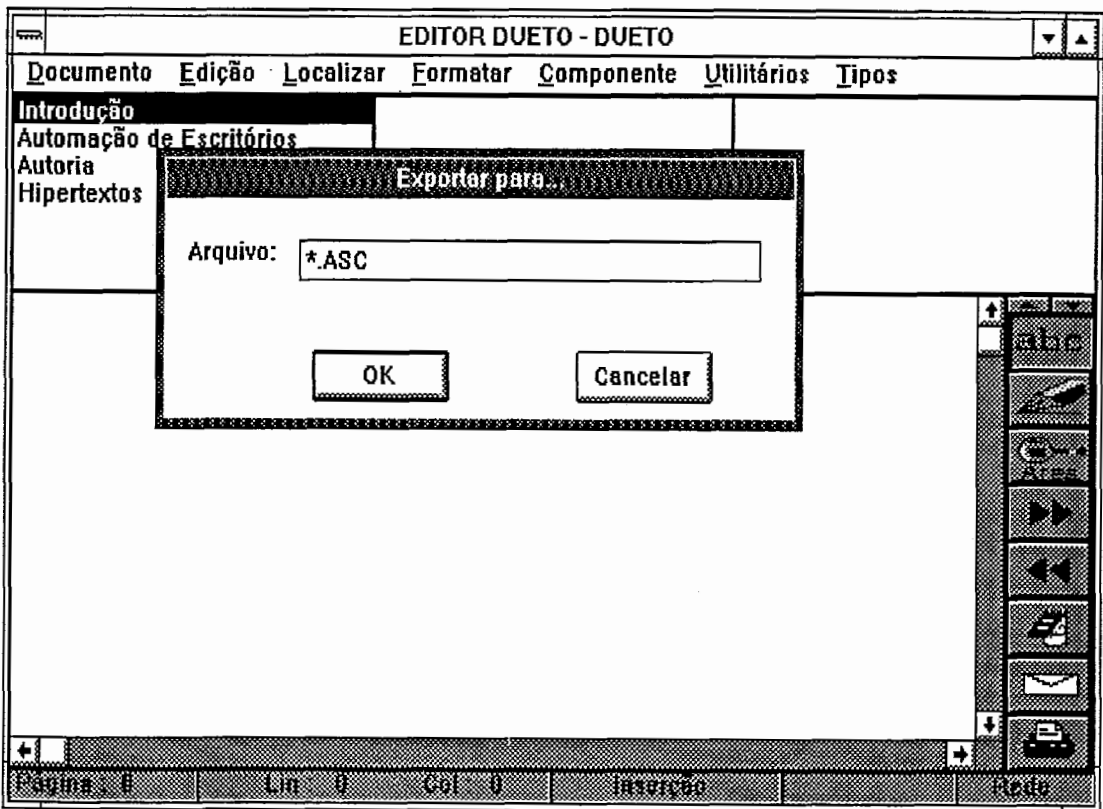


Figura VI.2.9 - Tela da opção Documento/Exportar

## Documento/Terminar

Permite que um usuário saia do **DUETO**. Caso tenha ocorrido alguma alteração no documento, esta função procede da mesma forma que a opção **Fechar**.

Opção: **Edição**

Opções Subordinadas:

**Desfazer**

**Cortar**

**Copiar**

**Repor**

A figura VI.2.10 mostra o cardápio “pull-down” da opção Edição.

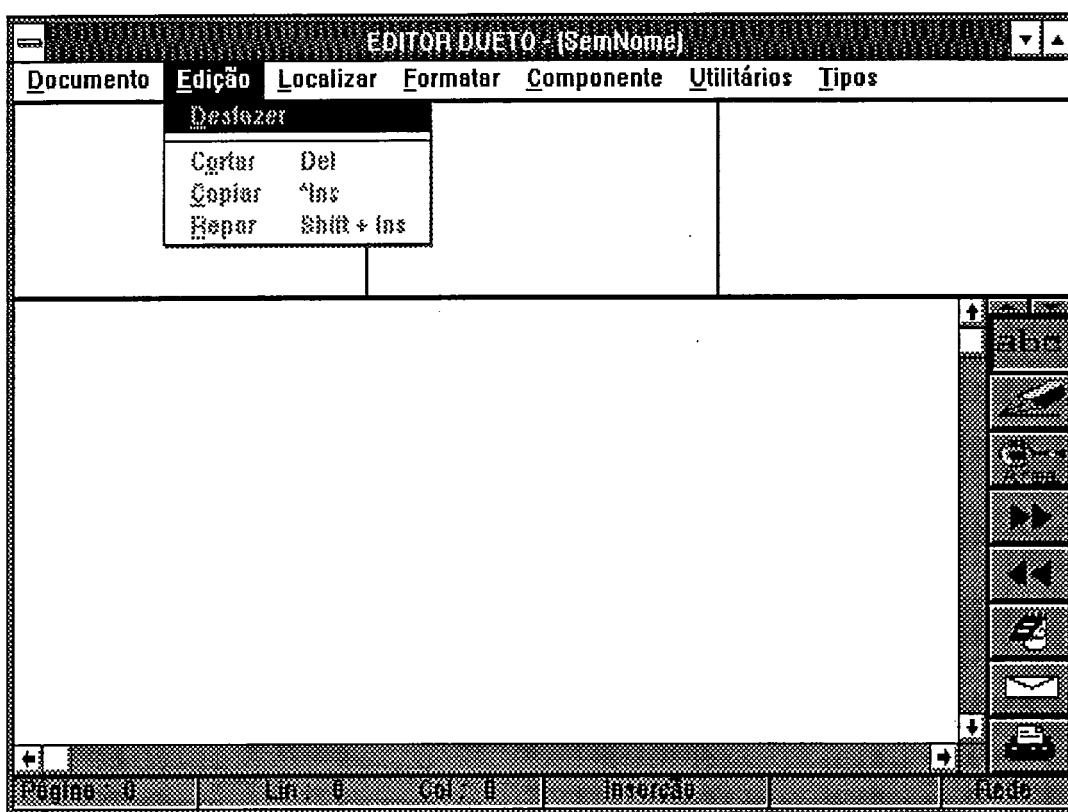


Figura VI.2.10 - Cardápio da opção Edição.

Descrição:

### **Edição/Desfazer**

Permite desfazer a última ação efetuada pelo usuário durante a edição de um componente do documento.

### **Edição/Cortar**

Permite mover o texto corrente, marcado em negrito, para a área de descarte. O texto movido para a área de descarte é apagado do texto que está sendo editado.

### **Edição/Copiar**

Esta opção é semelhante à opção Cortar, só que o texto movido para a área de descarte não é apagado do texto.

### **Edição/Repor**

Permite a inserção de textos da área descarte, para o interior do texto do componente que esta sendo editado. O texto inserido é visualizado em negrito no momento da inserção.

Opção: **Localizar**

Opções Subordinadas:

**Procurar**

**Repetir Última Procura**

**Trocar**

A figura VI.2.11 mostra o cardápio “pull-down” da opção Localizar.

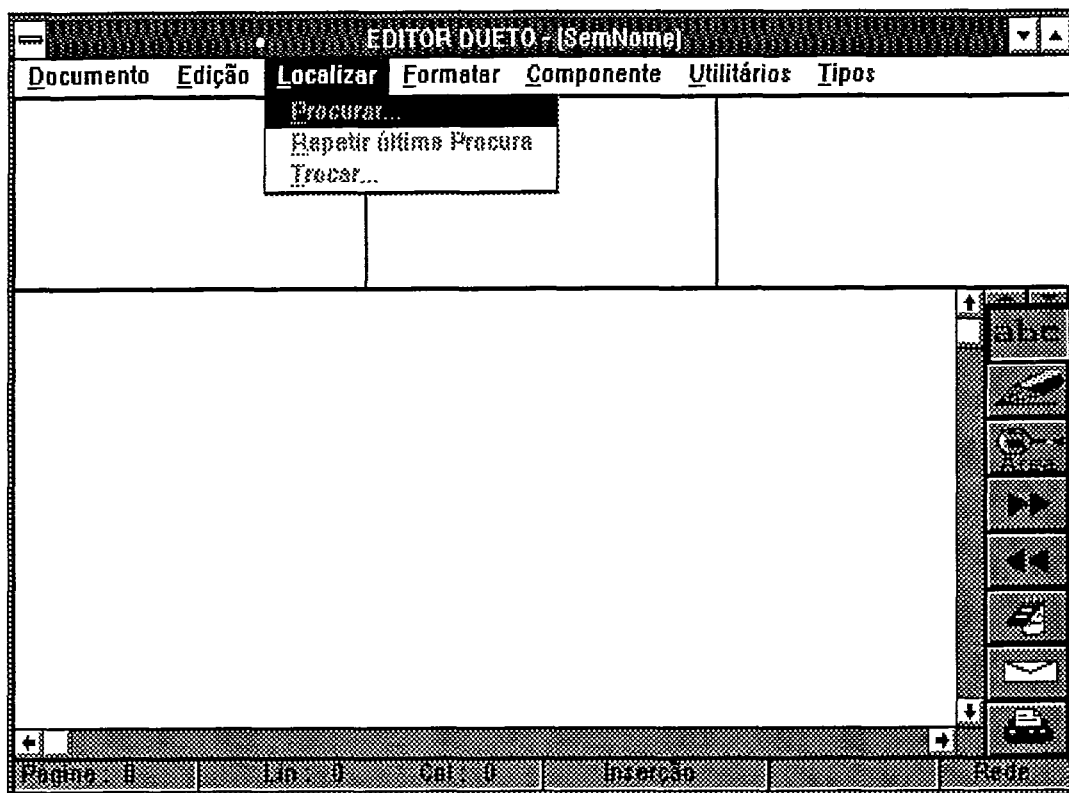


Figura VI.2.11 - Cardápio da opção Localizar.

Descrição:

### Localizar/Procurar

Permite procurar, ao longo do texto dos componentes, um determinado texto selecionado. Esta opção faz com que o editor apresente uma janela de diálogo com o texto selecionado pela posição do cursor ou, se o usuário desejar, permite digitar o texto a ser pesquisado. A caixa fica ativa até que todo o texto seja pesquisado ou até ser desativada pelo usuário. A caixa possui um botão que permite procurar a próxima palavra no texto. Cada palavra encontrada no texto é exibida em negrito.

A figura VI.2.12 mostra um exemplo desta opção.

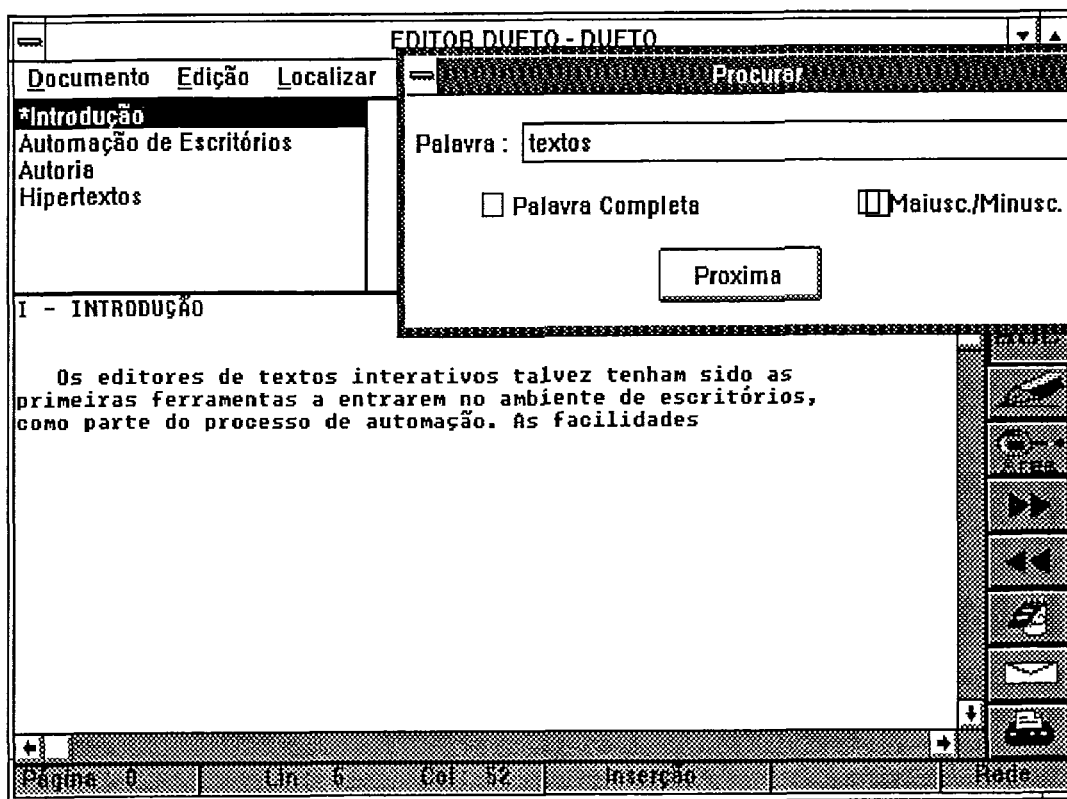


Figura VI.2.12 - Tela da opção Localizar/Procurar

### Localizar/Repetir Última Procura

Esta opção funciona do mesmo modo que a opção Procurar, só que o texto a ser pesquisado é o último selecionado pela opção Procurar.

### Localizar/Trocar

Esta opção permite fazer a pesquisa de um texto selecionado do mesmo modo que na opção Procurar, trocando-o por outro texto selecionado pelo usuário. O usuário tem a opção de ir trocando o texto a cada palavra encontrada ou trocá-las todas de uma só vez. Semelhantemente a opção Procura, a janela de diálogo fica ativa até o fim da pesquisa ou até ser desativada pelo usuário.

A figura VI.2.13 mostra um exemplo desta opção.

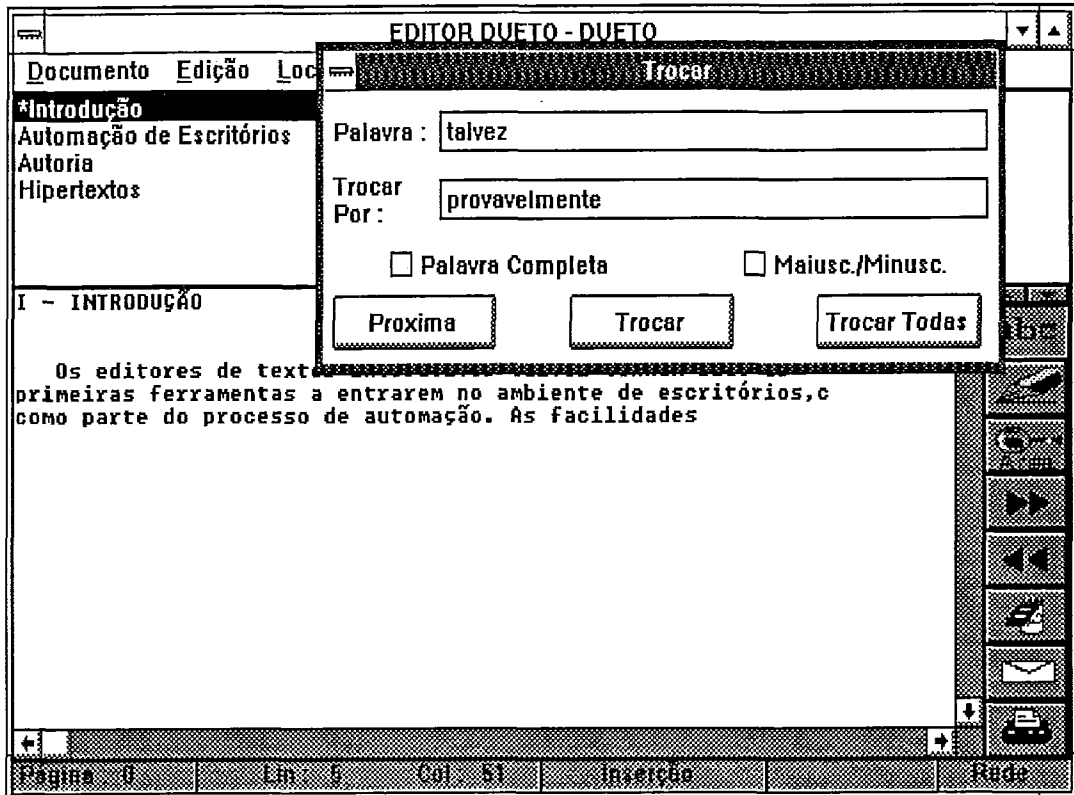


Figura VI.2.13 - Tela da opção Localizar/Trocar

Opção: **Formatar**

Opções Subordinadas:

**Página**

**Tabulação**

A figura VI.2.14 mostra o cardápio “pull-down” da opção Formatar.

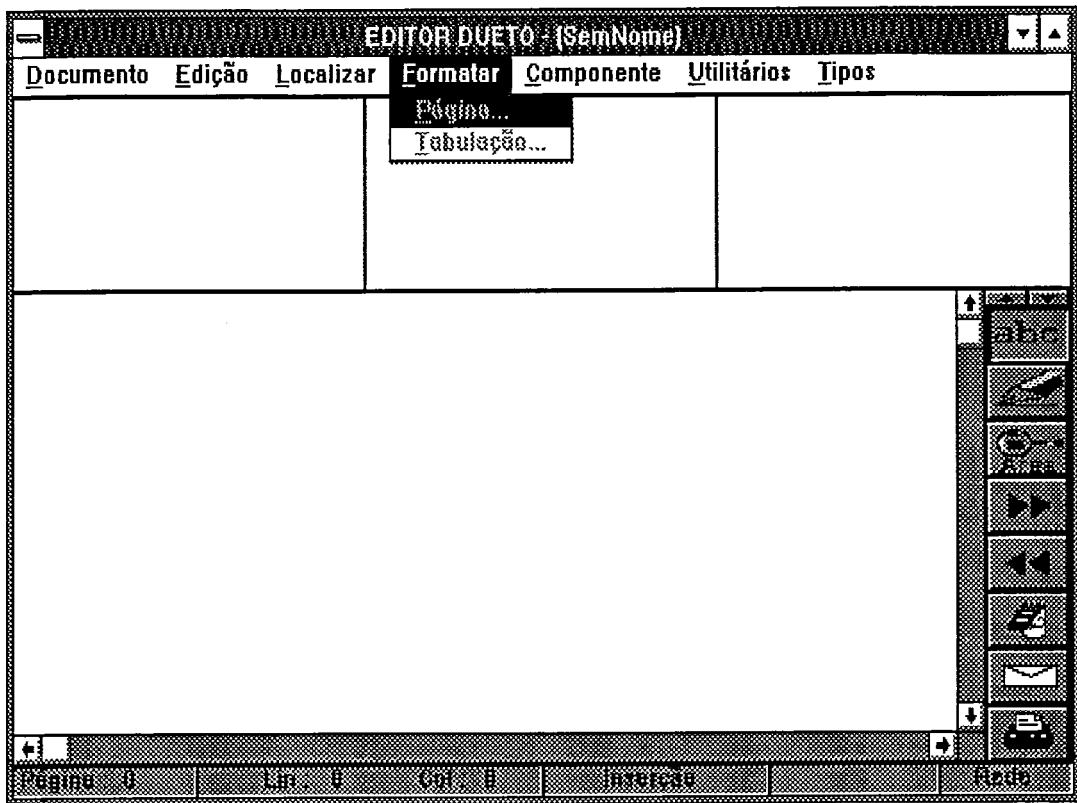


Figura VI.2.14 - Cardápio da opção Formatar.

Descrição:

### Formatar/Página

Permite que os usuários definam a formatação das páginas dos documentos através do controle de elementos, como: dimensão das margens, número de linhas por página e o espaçamento entre as linhas.

### Formatar/Tabulação

Permite alterar o número de espaços a serem inseridos na tabulação.

Opção: **Componente**

Opções Subordinadas:

**Criar**

**Editar**

**Alterar**

**Apagar**

**Mover**

**Anotar**    **Incluir**

**Recuperar**

**Conteúdo**

A figura VI.2.15 mostra o cardápio “pull-down” da opção Componente.

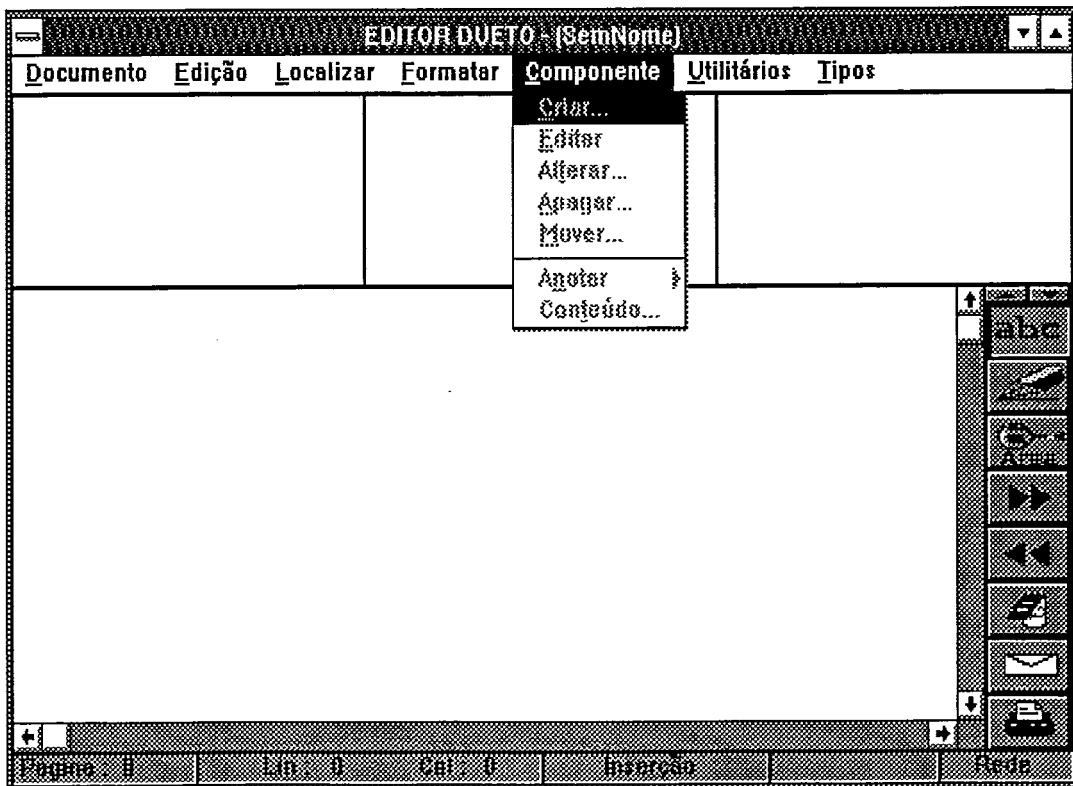


Figura VI.2.15 - Cardápio da opção Componente.



Descrição:

## **Componente/Criar**

Permite que os usuários criem componentes que formam a estrutura do documento. Um componente criado pode ser inserido no nível hierárquico da janela ativa da área de escopo no momento da criação, ou pode ser inserido no nível imediatamente inferior ao nível ativo, formando a sub-árvore do componente corrente na janela. Para o **DUETO**, a janela ativa é a que exibe o nível mais inferior da hierarquia dentre os níveis que estão sendo visualizados.

Dentro do nível hierárquico, o componente é sempre inserido imediatamente após o componente ativo exibido na janela. O componente criado passa a ser o componente corrente e, no caso de ser constituinte de uma sub-árvore, a janela que exibirá o componente passa a ser a janela ativa.

Ao ativar esta opção, o editor apresenta uma janela de diálogo que exibe o nome do componente corrente, onde o usuário deve digitar o nome do componente a ser criado e o nível hierárquico (Irmão ou Filho) em relação ao componente corrente.

As figuras VI.2.16 , VI.2.17, VI.2.18 e VI.2.19 mostram duas sequências de criação de um componente, uma delas mostra um componente sendo criado no mesmo nível (Irmão) do componente corrente e a outra mostra a criação de um componente de nível inferior (Filho) do componente corrente. Além disso, na segunda sequência mostra a movimentação à esquerda da estrutura, para apresentação do novo componente.

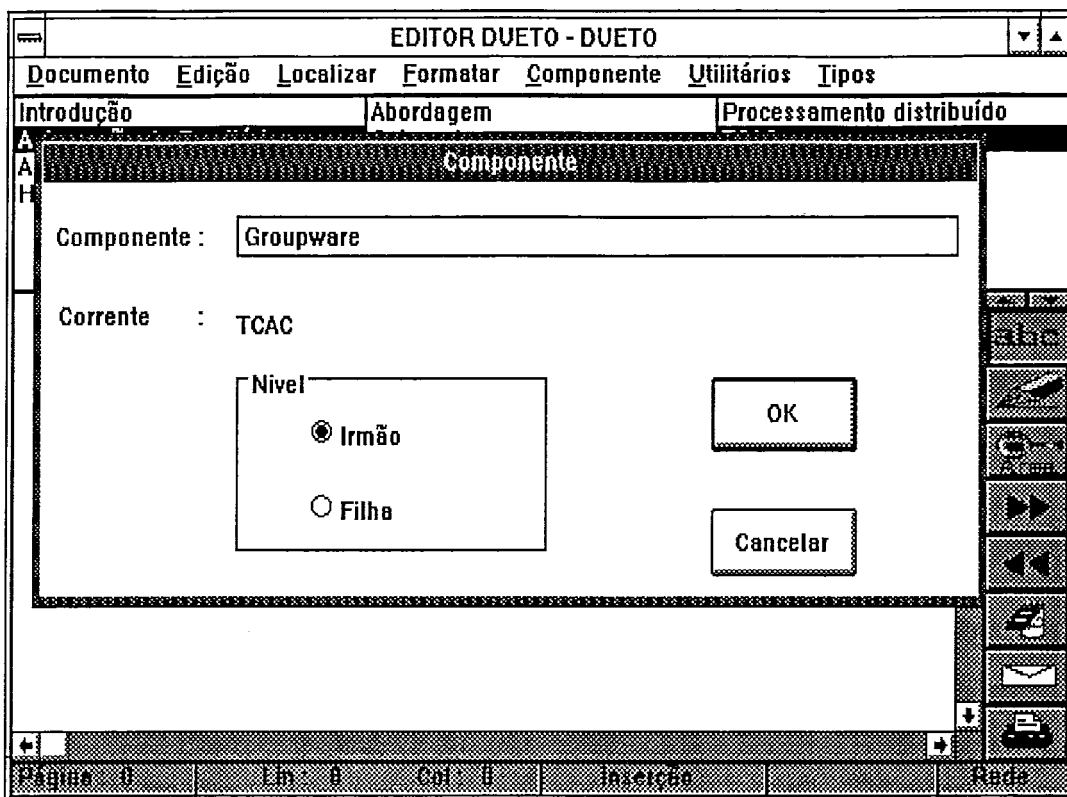


Figura VI.2.16 - Tela de criação de um componente(Irmão)

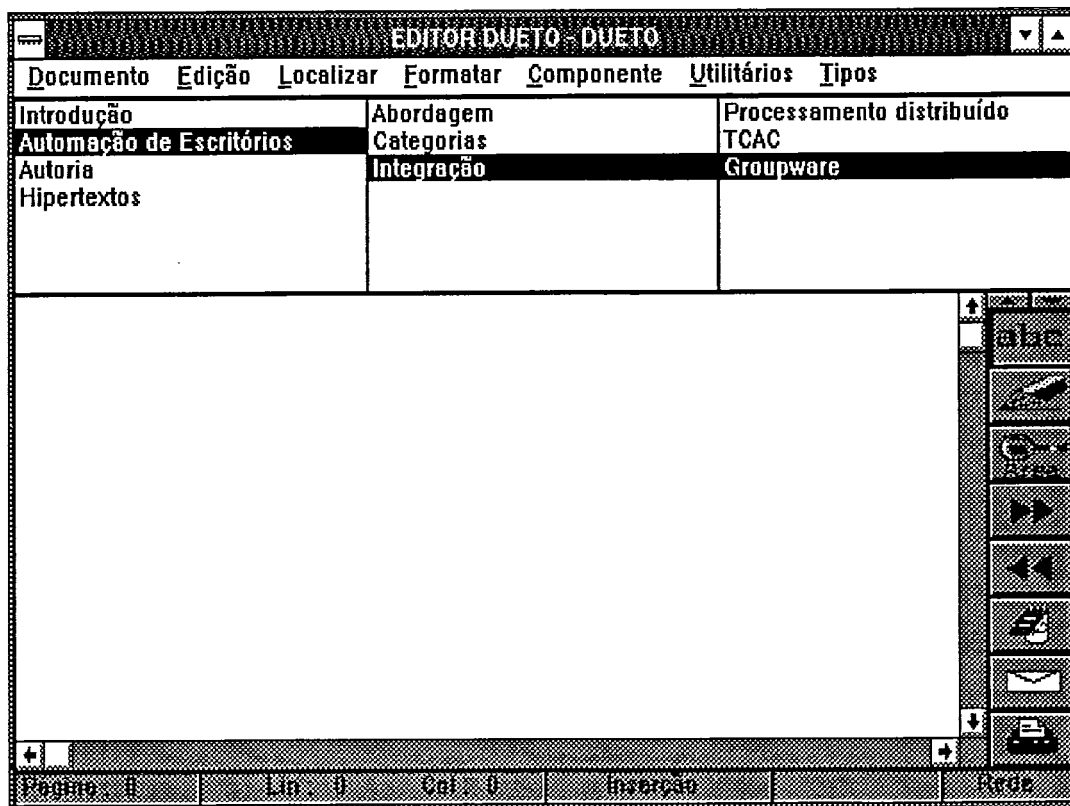


Figura VI.2.17 - Visualização da estrutura com o componente criado

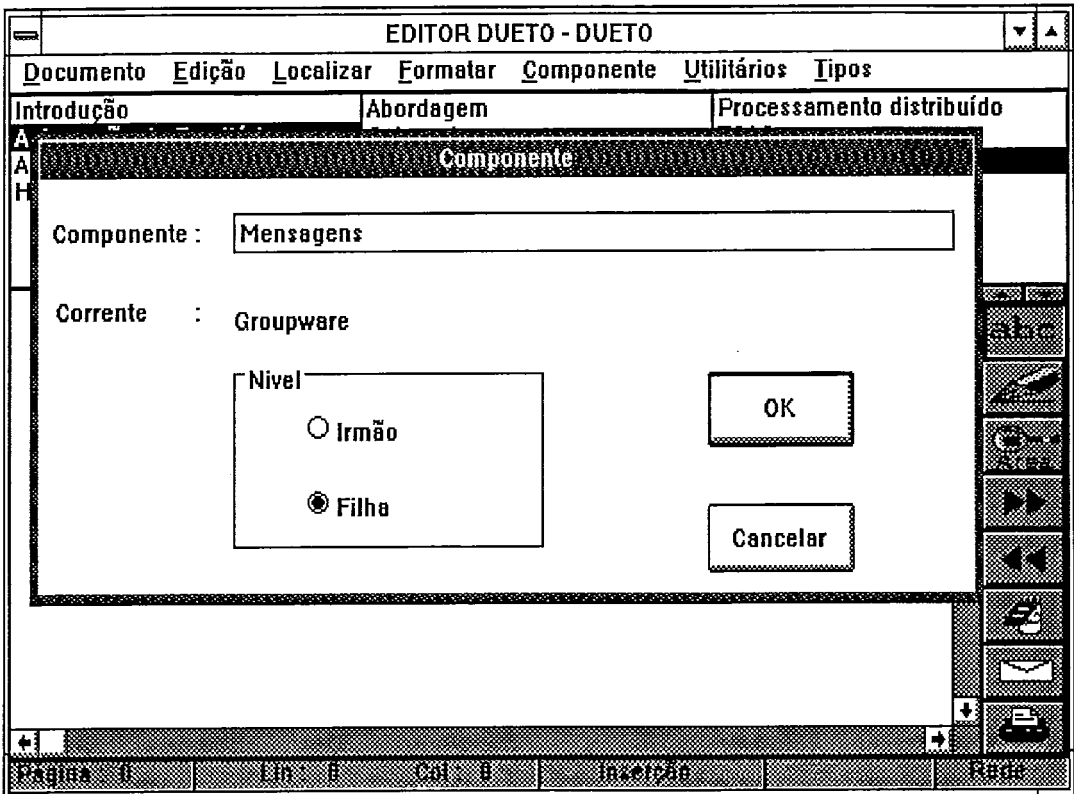


Figura VI.2.18 - Criação de componente de nível hierárquico inferior (filho)

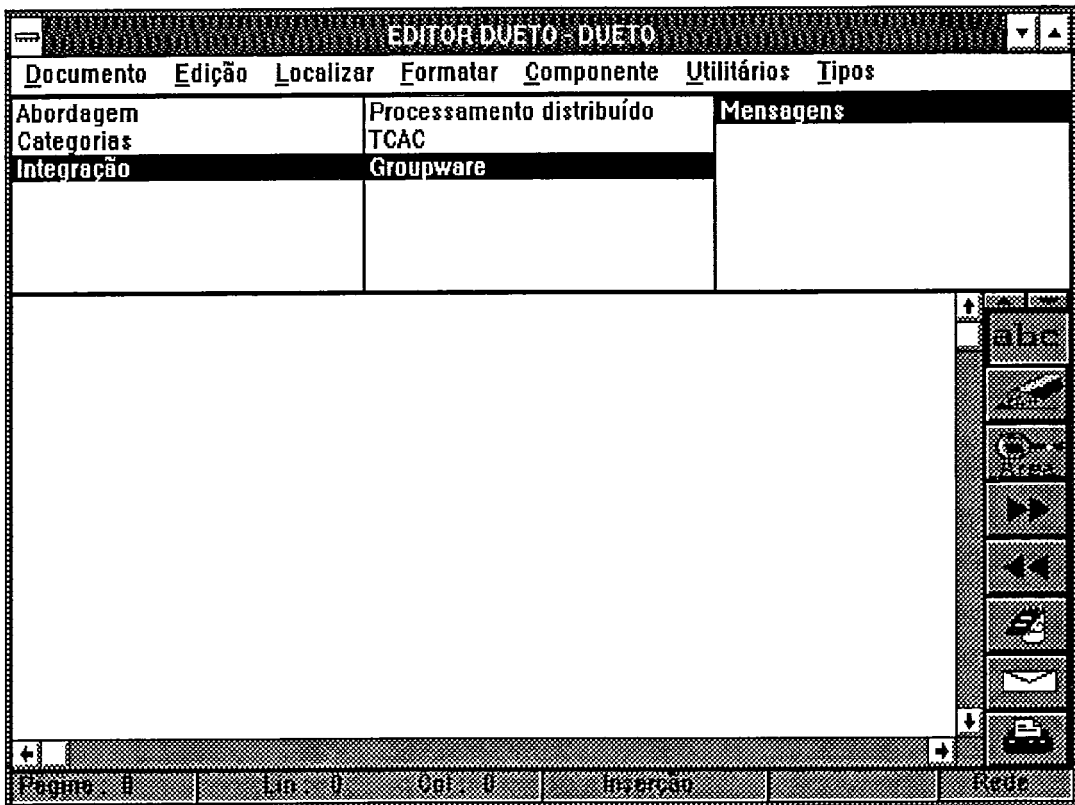


Figura VI.2.19 - Visualização da estrutura com componente criado (movimentação à esquerda)

## Componente/Editar

Permite que os usuários autorizados (Criador e Co-Autor) editem ou modifiquem os componentes da estrutura do documento. A escolha dessa opção ativa a área de edição para que o usuário possa editar texto do componente corrente. Caso já exista algum texto associado ao componente, a última versão será recuperada e inserida na área de edição.

O componente é bloqueado durante a edição, só sendo permitido o acesso por outros usuários, apenas para visualização. Se o componente escolhido para edição já estiver sendo bloqueado por outro usuário, a opção não se completará, uma mensagem será enviada ao usuário, informando sobre o bloqueio, e este apenas terá acesso para visualização do texto que está sendo editado. O texto de um componente que está bloqueado será visualizado por outros usuários, de modo diferente do usual (modo cinza).

As figura VI.2.20 mostra a opção de edição do componente corrente com texto associado.

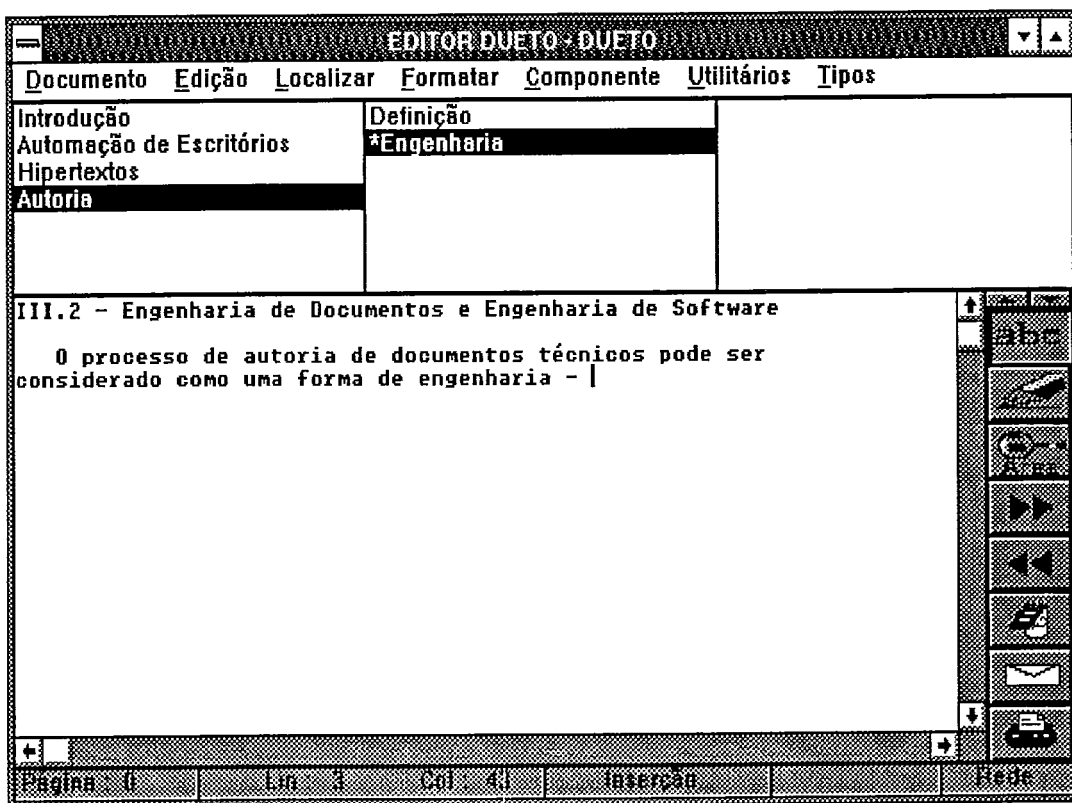


Figura VI.2.20 - Opção Componente/Editar

## Componente/Alterar

Permite que o usuário possa alterar o nome de um componente da estrutura do documento. Esta opção somente será executada se o componente que será alterado não estiver sendo acessado (editado) por outro usuário. Os componentes serão bloqueados durante a alteração.

## Componente/Apagar

Permite que o Criador do documento possa apagar, da estrutura do documento, um componente e os seus componentes-filhos que formam a sub-árvore da qual é o raiz. Semelhante à opção **Alterar**, esta só será executada se o componente corrente e os componentes-filhos não estiverem sendo acessados por outros usuários. Antes da opção ser executada, o **DUETO** envia uma mensagem para o Criador, informando se a operação de bloqueio foi bem sucedida e pedindo autorização para prosseguimento da operação. Juntamente com o envio da mensagem ao Criador, o **DUETO** envia mensagens aos demais usuários que estejam, no momento, acessando o documento, para informá-los que o componente será apagado.

As figuras VI.2.21 e VI.2.22 mostram um exemplo desta opção.

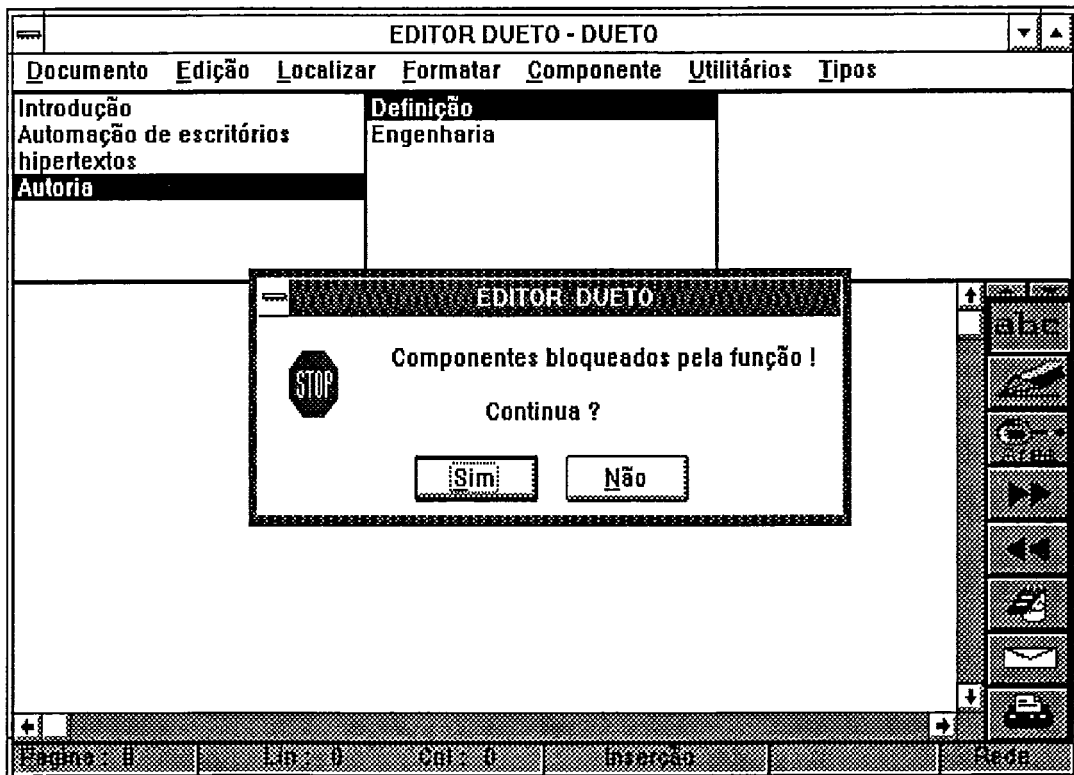


Figura VI.2.21 - Tela 1 da opção Componente/Apagar

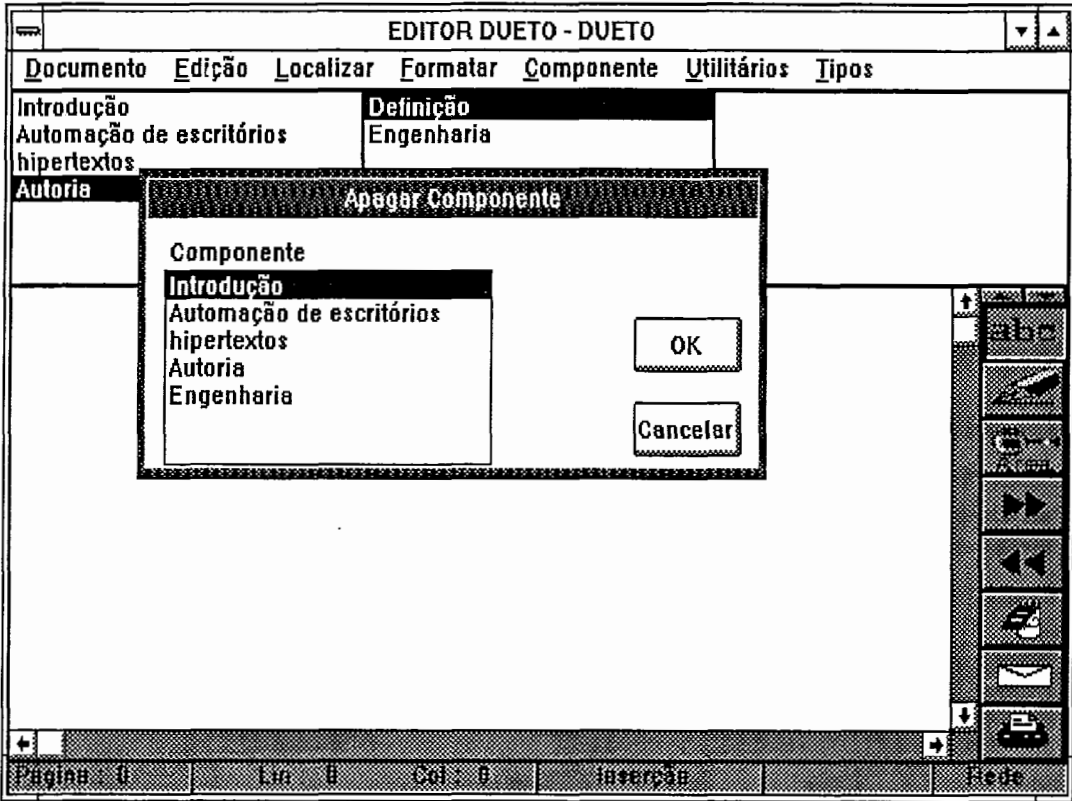


Figura VI.2.22 - Tela 2 da opção Componente/Apagar

### Componente/Mover

Permite que os usuários possam mover um componente e os seus componentes-filho para qualquer outro nível hierárquico da estrutura, ou para uma outra posição dentro do mesmo nível.

Semelhante à opção **Apagar**, esta também só é executada se o componente corrente e os componentes-filho não estiverem sendo acessados por outros usuários. Os componentes também serão bloqueados durante a alteração.

Dentro de um ambiente de autoria de documentos, essa função torna-se bastante importante pois permite que os usuários possam organizar e reorganizar o documento, através da movimentação dos componentes entre os diversos níveis, visando obter a melhor estrutura formal do corpo do documento.

Para execução desta função o editor exibe uma lista contendo todos os componentes da estrutura do documento, à exceção do componente a ser movido, para a seleção do novo componente pai. O primeiro elemento da lista é denominado de RAIZ. O usuário também deverá informar a posição que o componente deverá ser inserido no novo nível.

### Componente/Anotações/Incluir

Permite a inclusão de comentários ou anotações que são anexados ao componente ativo do documento. A escolha desta opção permite que o editor exiba uma janela para que o usuário preencha as informações referentes à anotação, tais como: título da anotação, tipo da anotação, destinatário e o texto da anotação.

O nome do destinatário somente é obrigatório quando o comentário for do tipo privado. Para comentários do tipo público esta informação é desconsiderada pelo editor.

A figura VI.2.23 mostra um exemplo desta opção.

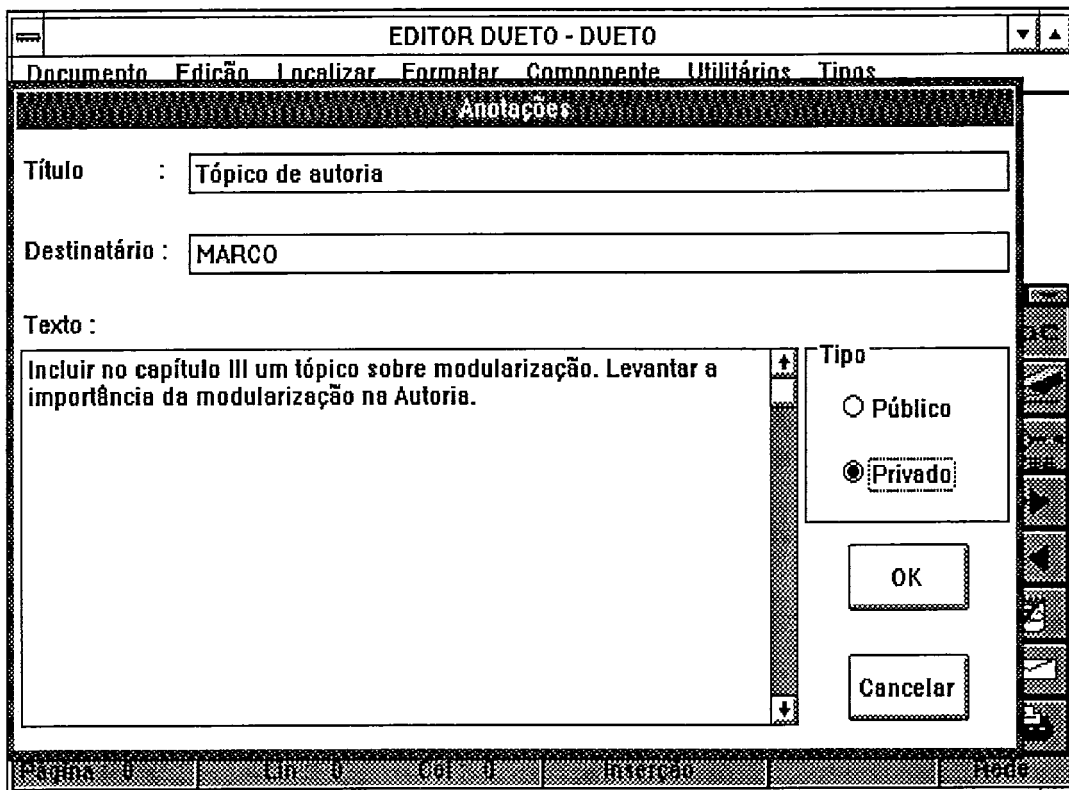


Figura VI.2.23 - Tela de inclusão de anotação

### Componente/Anotações/Recuperar

Permite recuperar todos os comentários ou anotações anexados a um componente do documento. Dentre os comentários do tipo privado, somente são recuperados aqueles dirigidos ao usuário.

O editor exibe uma janela contendo a lista de todos os comentários e, para visualização, basta o usuário selecioná-los, através do mouse. Efetuada a seleção, as informações referentes aos comentários, bem

como o texto, são exibidos na janela. Na parte inferior da janela há um botão (Copiar) que permite, quando acionado, copiar o texto da anotação para a área de descarte. Caso o usuário desejar, o texto pode ser inserido, no modo edição, no corpo do componente.

As figuras VI.2.24 mostram a opção de recuperação de um comentário.

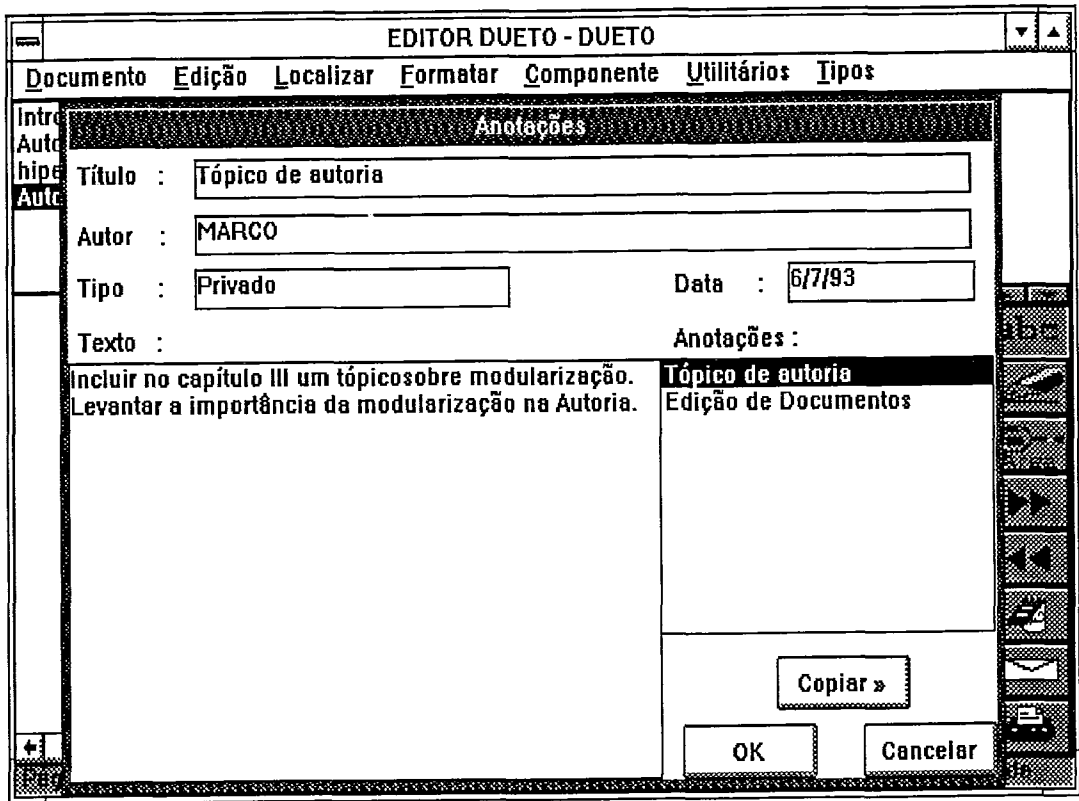


Figura VI.2.24 - Tela para recuperação das anotações ligadas a um componente

### Componente/Conteúdo

Permite a visualização simultânea de vários componentes do documento em outras janelas, sem desativar a área de edição. Para a visualização de um determinado componente, basta que o usuário informe o nome do componente na janela de diálogo exibida pelo editor.

Esta opção tem função importante em ambientes de co-autoria de documentos, não somente por permitir o acesso a várias partes do documento, mas também, por permitir que os usuários tomem conhecimento das ações de outros usuários que estão acessando estes componentes.

A figura VI.2.25 mostra um exemplo desta opção.



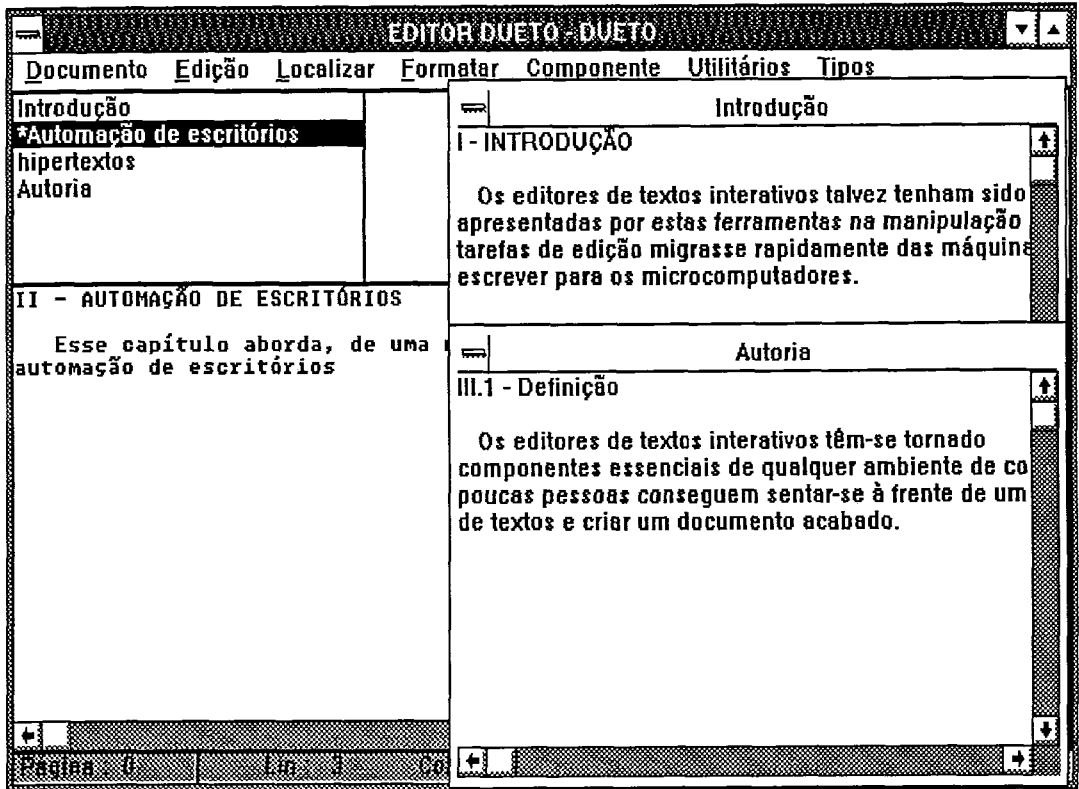


Figura VI.2.25 - Opção Componente/Conteúdo

Opção: **Utilitários**

Opções Subordinadas:

**Autores**

**Listar**

**Mensagens**

**Criar**

**Trilha**

**Destruir**

**Ativar**

**Desativar**

**Contextos Criar**

**Liberar**

**Configuração**

A figura VI.2.26 mostra o cardápio “pull-down” da opção Utilitários.

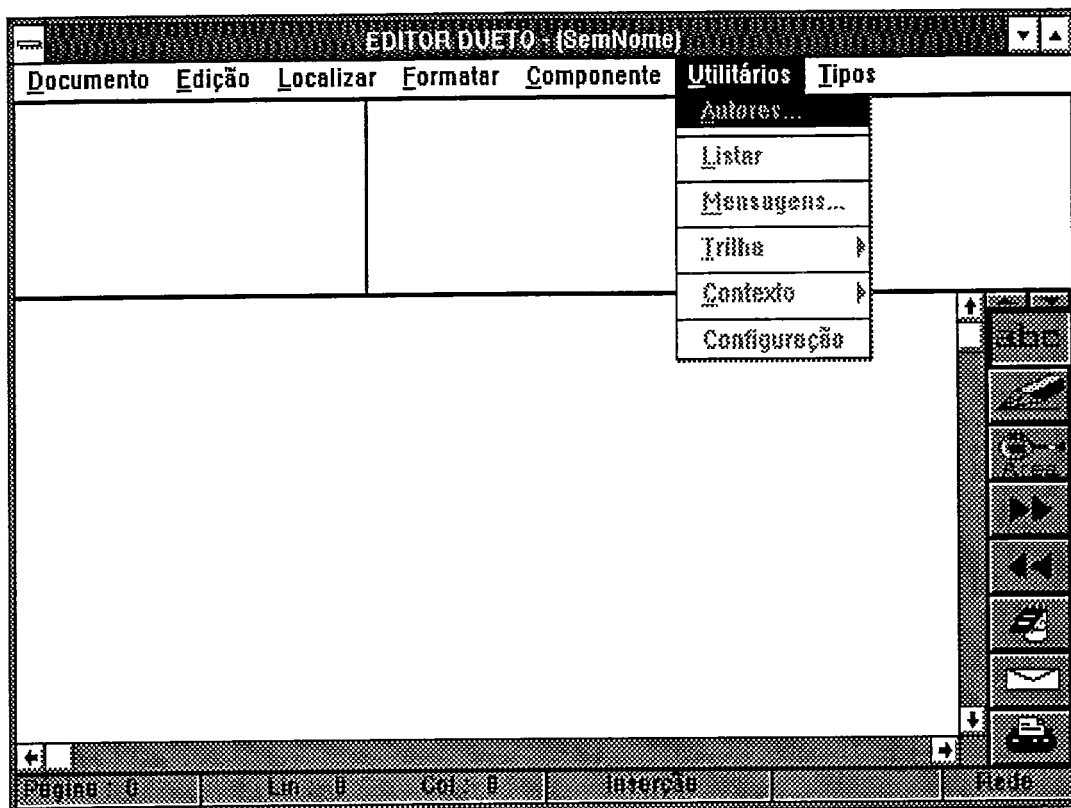


Figura VI.2.26 - Cardápio da opção Utilitários.

Descrição:

### Utilitários/Autores

Permite definir os participantes de um documento juntamente com o papel social que cada um desempenhará no documento.

A inclusão de um participante é, somente, efetuada pelo criador do documento. Esta opção somente está ativa na janela do criador do documento e, portanto, nas janelas dos demais participantes esta opção estará desativada. A escolha desta opção permite que o editor apresente um janela para que o gerente inclua o nome dos participantes em uma lista de participantes. Os nomes devem ser os mesmos que os utilizados no ambiente da rede. Juntamente com o nome, o criador deve selecionar o papel do participante no documento através da seleção de um dos três botões de papéis sociais definidos pelo **DUETO**.

Os participantes também podem ser retirados da lista, bastando, para isto, que o criador selecione um nome na lista e acione o botão **Retirar** para que o participante seja retirado da lista.

A figura VI.2.27 mostra um exemplo desta opção.

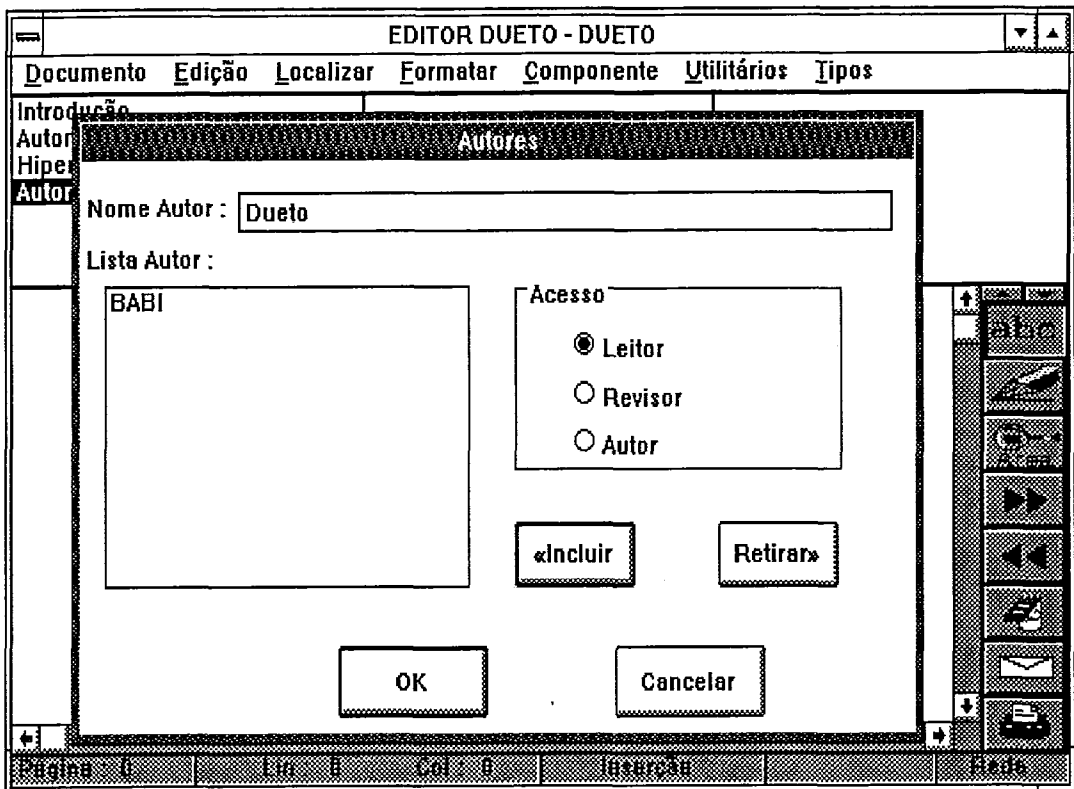


Figura VI.2.27 - Opção Utilitários/Autores

## Utilitários/Listar

Mostra uma lista contendo informações sobre a localização dos componentes na rede, nome dos participantes e os componentes do documento que estão sendo acessados naquele momento.

Essa opção pode ser visualizada na figura VI.2.28

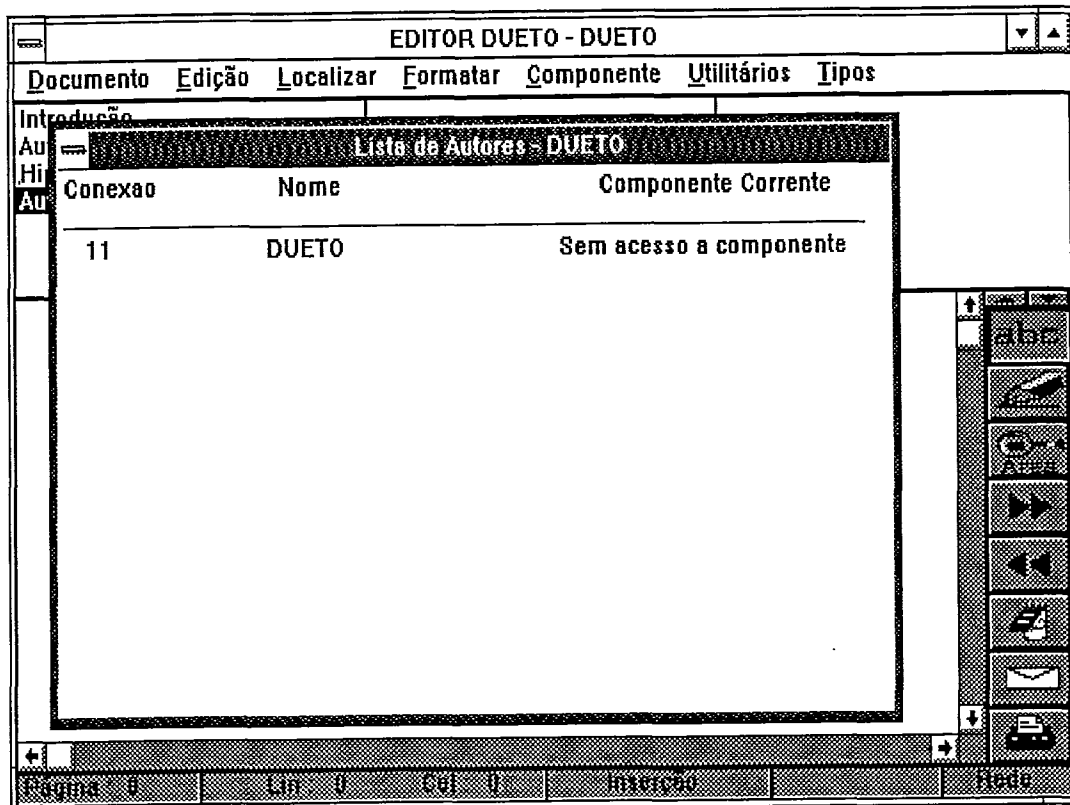


Figura VI.2.28 - Opção Utilitários/listar (Participantes acessando o documento)

## Utilitários/Mensagens

Permite o envio de mensagens diretas aos demais participantes de um documento para troca de informações. Este tipo de mensagem não fica armazenada no sistema e são enviadas aos destinatários logo após a liberação pelo autor da mensagem. Os usuários que as recebem tem suas ações congeladas até que liberem a tela contendo a mensagem.

Para envio da mensagem, o editor exhibe uma tela em que o usuário deve editar a mensagem a ser enviada, escolher, depois, o nome do destinatário a partir de uma lista com os nomes dos participantes que estão acessando o documento. Esta tela contém, também, a opção **Todos** que permite que o editor desconsidere a lista e envie a mensagem para todos os participantes.

A figura VI.2.29 mostra a tela para envio de uma mensagem.

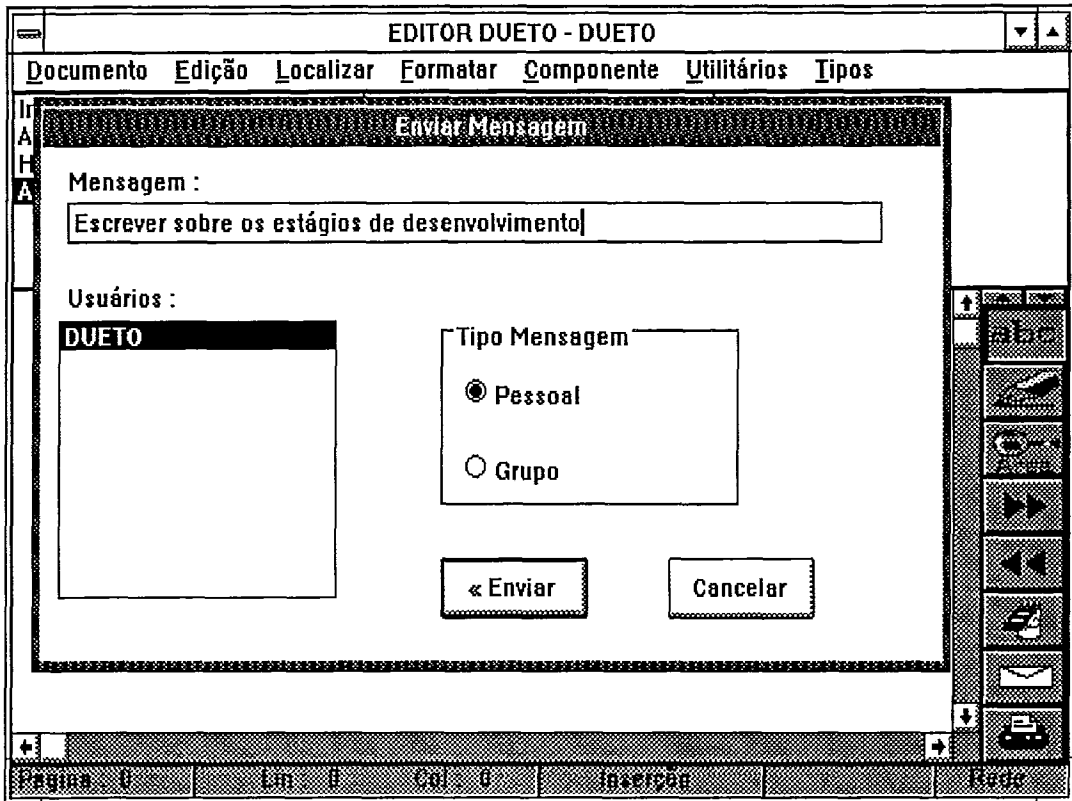


Figura VI.2. 29 - Opção Utilitários/Mensagem (Envio de Mensagem)

### Utilitários/Trilha/Criar

Permite a criação de caminhos (conjunto de componentes) a serem seguidos pelos leitores no acesso a um documento. Esta opção exibe uma tela onde o usuário deve informar o nome da trilha a ser criada e escolher, a partir de uma lista, os componentes que irão fazer parte da trilha. O editor só seleciona os componentes que possuem textos associados. Os componentes selecionados são visualizados numa lista lateral. A primeira trilha criada é considerada, pelo **DUETO**, como a trilha “default” do editor, sendo exibida para todos os leitores quando acessam o documento.

A figura VI.2.30 mostra um exemplo desta opção.

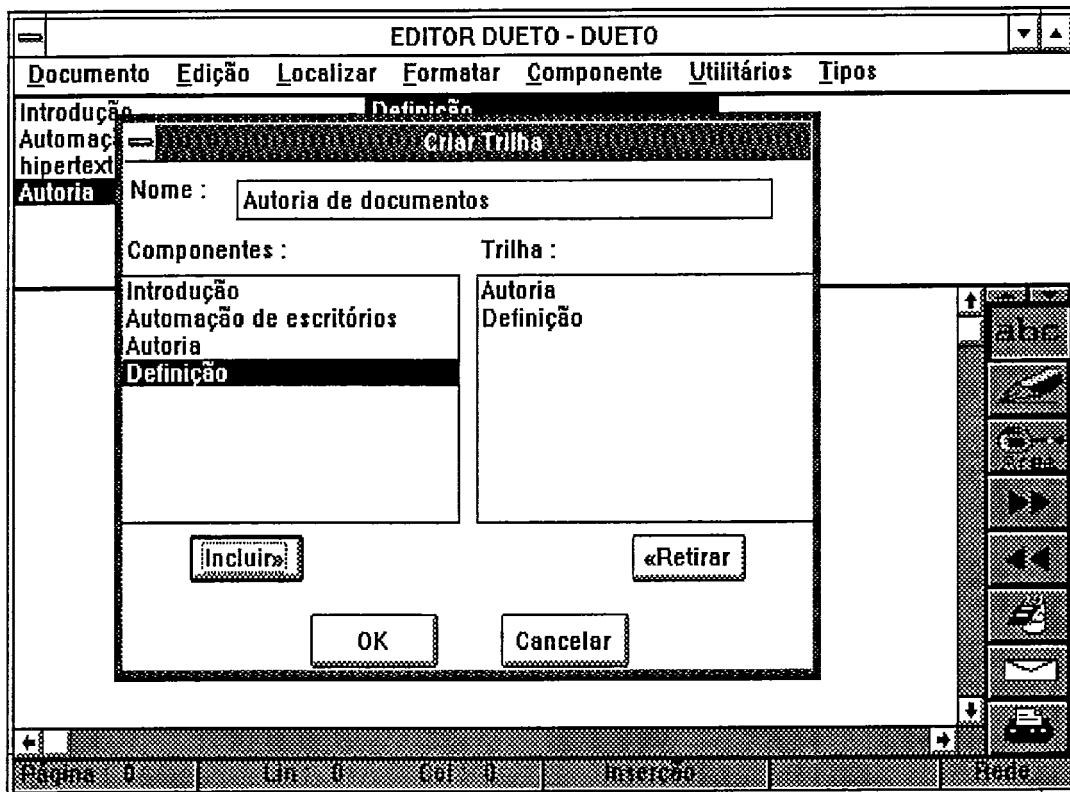


Figura VI.2.30 - Tela da opção Utilitários/Trilha/Criar

### Utilitários/Trilha/Destruir

Permite que o criador destrua (elimine) uma trilha de um documento. Essa opção exibe uma lista de trilhas, onde o criador deve selecionar qual será destruída. A trilha selecionada deverá estar desativada para ser destruída.

### Utilitários/Trilha/Ativar

Permite que um leitor possa ativar uma trilha diferente da trilha "default", a partir de uma lista contendo todas as trilhas disponíveis para o documento. A trilha escolhida é movida para a primeira janela da área de escopo.

### Utilitários/Trilha/Desativar

Permite que um leitor possa desativar a trilha corrente, a partir da qual está acessando o documento. O editor exibe, novamente, a lista contendo todas as trilhas onde o usuário deve selecionar uma nova trilha. Caso o leitor não selecione uma nova trilha, a trilha "default" retornará como a corrente do documento.

### Utilitários/Contextos/Criar

Permite que um autor possa criar um contexto, a partir de uma lista de componentes do documento, para efetuar modificações no conjunto de nós que forma o contexto. Semelhante à opção Trilha/Criar, somente os componentes que possuem texto associado e estão liberados no instante da criação é que são visualizados na lista.

A figura VI.2.31 exibe um exemplo desta opção.

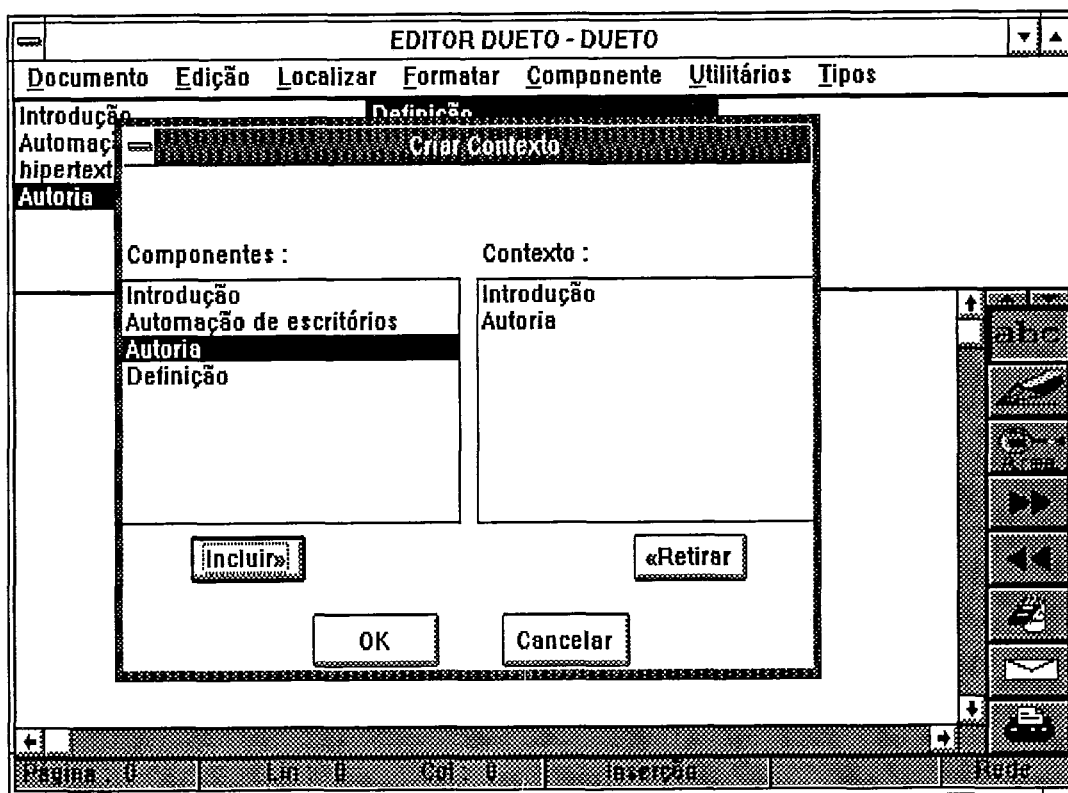


Figura VI.2.31 - Tela da opção Utilitários/Contexto/Criar

### **Utilitários/Contextos/Liberar**

Permite que um autor possa liberar todos os componentes que formam o contexto corrente.

### **Utilitários/Contextos/Exibir**

Permite que um autor possa exibir o nome e conexão dos participantes que estão mantendo contextos.

### **Utilitários/Configuração**

Permite a criação de uma nova configuração do documento contendo as últimas versões dos componentes da estrutura do documento. Essa opção somente é efetuada pelo criador do documento e para sua execução todos os componentes devem estar liberados. Para isso, o sistema bloqueia o acesso ao documento até o final da operação. Se algum participante do documento estiver acessando qualquer componente, o editor cancela a operação e informa, ao criador, que não pode completar até que os componentes estejam liberados.

Opção: **Tipos**

Opções Subordinadas:

**Fontes**

**Tamanho**

**Alinhamento**

**Estilos**



A figura VI.2.32 mostra o cardápio “pull-down” da opção Tipos.

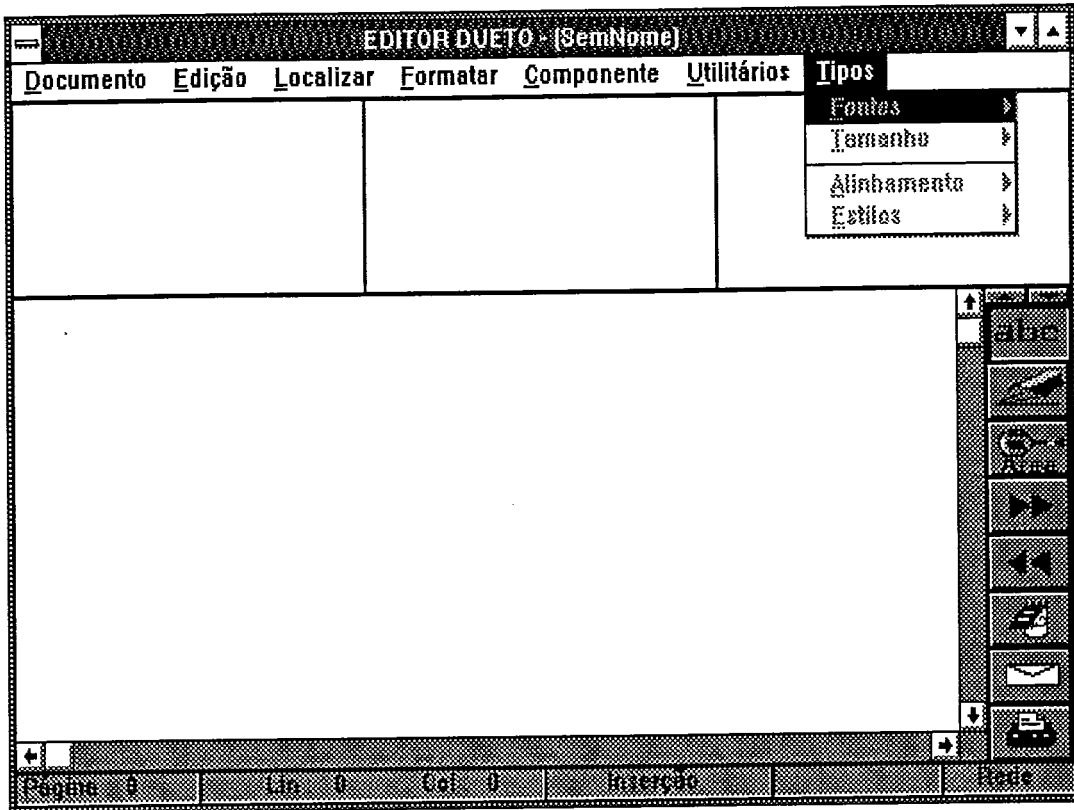


Figura VI.2.32 - Cardápio da opção Tipos.

Descrição:

### Tipos/Fontes

Permite seleccionar um fonte para apresentação do texto. Os tipos de fontes existentes são os fontes disponíveis no ambiente windows.

### Tipos/Tamanho

Permite seleccionar diferentes tamanhos de caracteres relacionados ao fonte corrente.

## **Tipos/Alinhamento**

Permite seleccionar o alinhamento dos parágrafos. O alinhamento de parágrafos pode ser efetuado das seguintes maneiras: margem esquerda, margem direita, centralizado ou alinhamento entre as margens.

## **Tipos/Estilos**

Permite seleccionar diferentes estilos de carácter, tais como: normal, negrito, sublinhado ou itálico.

## **VI.3 - Recursos Computacionais Utilizados**

### **- Recursos de Hardware**

Para o desenvolvimento do **DUETO** utilizou-se a seguinte plataforma:

- . Um micro-computador PC 386 com 25 MHz;
- . Monitor SVGA;
- . 4 Mb de memória RAM.

A utilização do editor está limitada a equipamentos que permitam a utilização do sistema **Windows 3.0**. Por questões de desempenho, aconselha-se a utilização em equipamentos que possuam características, no mínimo, semelhantes às descritas acima.

## - Recursos de Software

O **DUETO** foi desenvolvido para rodar sob o **WINDOWS 3.0**, em ambiente de rede (rede Novell V. 3.11) [Nove90]. Para sua implementação foi utilizada a linguagem C (BorlandC 2.0) [Ezze91], que possui ferramentas que permitem o desenvolvimento de aplicações para o ambiente Windows. Além disso, utilizou-se uma biblioteca de funções Netware - C [Nove89], desenvolvida pela Novell para o ambiente de rede, e implementa vários serviços que permitem o acesso ao sistema NetWare, através de chamadas às API's (Application Programming Interface) do NetWare.

## VI.4 - Arquitetura do DUETO

A arquitetura do editor **DUETO** é apresentada conforme a figura VI.4.1.

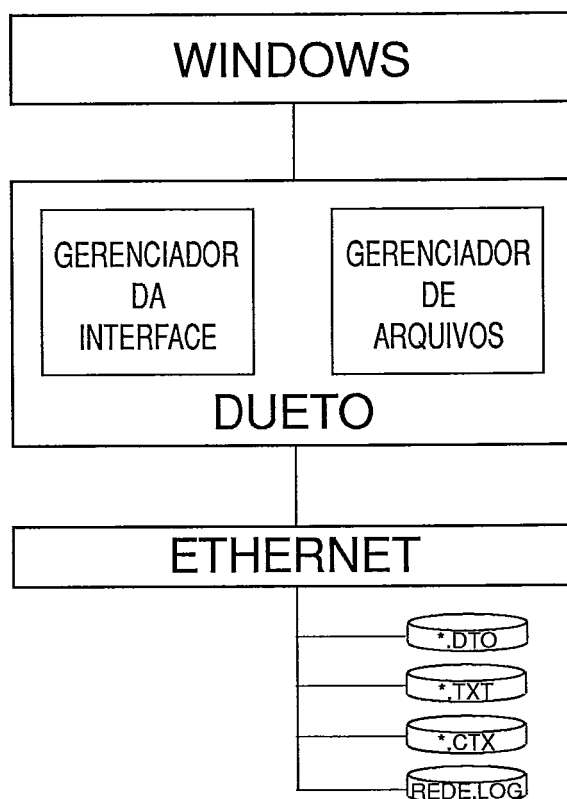


Figura VI.4.1 - Arquitetura do DUETO

O editor **DUETO** é um software multi-usuário que suporta grupos de participantes trabalhando sincronamente, em um ambiente de rede local, onde cada estação encontra-se conectada via Ethernet [Nove90] para permitir a comunicação entre as diversas estações e o acesso aos documentos armazenados na estação servidora.

A arquitetura do **DUETO** é composta, em linhas gerais, por dois módulos: o **módulo Gerenciador da Interface** e o **módulo Gerenciador de Arquivos**. O Gerenciador da Interface coordena as atividades de interação com os usuários e o Gerenciador de Arquivos é o responsável pela coordenação das atividades concorrentes entre os usuários, pelo envio de mensagens, pelo acesso e compartilhamento dos documentos e anotações.

O módulo Gerenciador da Interface é constituído por dois outros módulos: o **módulo Editor** e o **módulo Estrutura**. O Editor é responsável pelas funções de edição e formatação de textos dos componentes dos documentos, editados na janela da área de edição. Além disso, é responsável pela apresentação dos textos nas demais janelas da interface. Este módulo comunica-se com o Gerenciador de Arquivos de modo que possa refletir, nos textos apresentados nas janelas da interface, as ações efetuadas concorrentemente pelos participantes e, também, os direitos de acesso de cada participante.

O módulo Estrutura é responsável pelo controle e apresentação da estrutura dos documentos exibidas nas janelas da área de escopo da interface. Este módulo se comunica com o Gerenciador de Arquivos para controlar e exibir qualquer modificação efetuada na estrutura dos documento e exibir informações sobre os componentes que estão sendo editados.

No módulo Gerenciador de Arquivos encontram-se as chamadas às funções do sistema operacional da rede, para implementar os mecanismos que permitem a troca de mensagens entre os participantes e o acesso concorrente aos arquivos do editor possibilitando o compartilhamento de informações.

## **VI.5 - Armazenamento**

Para a implementação do editor **DUETO**, utilizou-se um projeto de arquivos convencionais para armazenamento e recuperação das informações. O controle de concorrência está baseado num serviço oferecido pelo sistema da rede, que controla o acesso concorrente aos arquivos [Nove89].

O controle de acesso as informações, implementado pelo **DUETO** impõem que cada objeto armazenado possua um atributo de bloqueio (“lock”).

O projeto resultou em arquivos com as seguintes extensões:

- Arquivos com extensão **DTO**: São arquivos que armazenam as principais informações, sobre os documentos, criados pelo editor. As chaves de acesso a essas informações são criadas internamente pelo **DUETO**. A seguir, descrevemos o conjunto de informações contidos nesses arquivos.

- Documentos : Armazenam informações sobre um determinado documento, incluindo: Título do documento, Data de Criação, Data da última alteração e Número da configuração atual.

- Participantes: Contêm informações sobre os diversos participantes com o papel social (direitos de acesso) definido para o documento, a saber: Nome do participante, Documento e papel social.

- Componentes : Armazenam informações sobre a estrutura dos documentos. Estas informações constituem uma lista que representa a árvore hierárquica dos documentos. São elas: Título do componente, Documento, nível hierárquico e versão atual.

- Configurações : Armazenam informações sobre as várias configurações dos documentos. Estas informações constituem, também, uma lista que representa a árvore de configuração dos documentos e inclui dados sobre: Número da configuração, Componente, Versão do componente e Data de criação.

- Trilhas : Armazenam informações sobre os caminhos pré-definidos (Trilhas) para a navegação ao longo do documento. Semelhantemente aos componentes e configurações, estas informações também constituem uma lista com dados que incluem: Componente, Versão do componente, Nome do Leitor, Nome da trilha e Documento.

- Anotações : Contêm anotações efetuadas pelos participantes nos componentes dos documentos. Tais informações incluem: Título da anotação, Tipo, Destinatário, Autor, Data de Criação, Documento, Componente e Texto.

Arquivos com extensão **TXT**: São arquivos que armazenam os informações e textos das diferentes versões dos componentes dos documentos. Os nomes dos arquivos são formados por uma sigla, criada internamente pelo editor, correspondente ao documento.

Arquivos com extensão **CTX**: São arquivos temporários que armazenam informações sobre contextos criados em um documentos os usuários que criaram esses contextos. Estas informações constituem uma lista com dados sobre os componentes que formam os contextos. O nome dos arquivos segue o padrão adotado pelos arquivos com extensão **TXT**.

Arquivo **“REDE.LOG”**: É o arquivo que permite manter informações sobre as atividades dos usuários, isto é, permite coordenar as atividades de todos os usuários durante as sessões de autoria de documentos. Estas informações incluem dados sobre o documento, sobre o componente que está sendo acessado e a localização dos participantes, para envio de mensagens.

## **VI.6 - Exemplo de Utilização**

Nesta seção, apresentamos um exemplo de utilização do **DUETO**. Devido ao excesso de telas e, conseqüentemente, ao número excessivo de figuras, a apresentação das figuras ficará limitada aos aspectos mais relevantes do desenvolvimento. Para o exemplo utilizaremos um trabalho onde será demonstrado o seu desenvolvimento com dois participantes. Cada participante desempenha um papel diferente no documento: um criador e um co-autor.

A partir desta proposta, o criador do documento inicia o desenvolvimento propondo os tópicos a serem desenvolvidos, sem se importar com a estrutura do documento. Durante o desenvolvimento dos tópicos inicia-se um processo de interação entres os co-autores, através de comentários e troca de mensagens, no sentido de se chegar a um consenso do que deve conter cada um dos tópicos. Além disso, existe, também, o processo de estruturação do documento para se chegar a sua forma final.

A seguir, mostraremos esse processo de desenvolvimento.

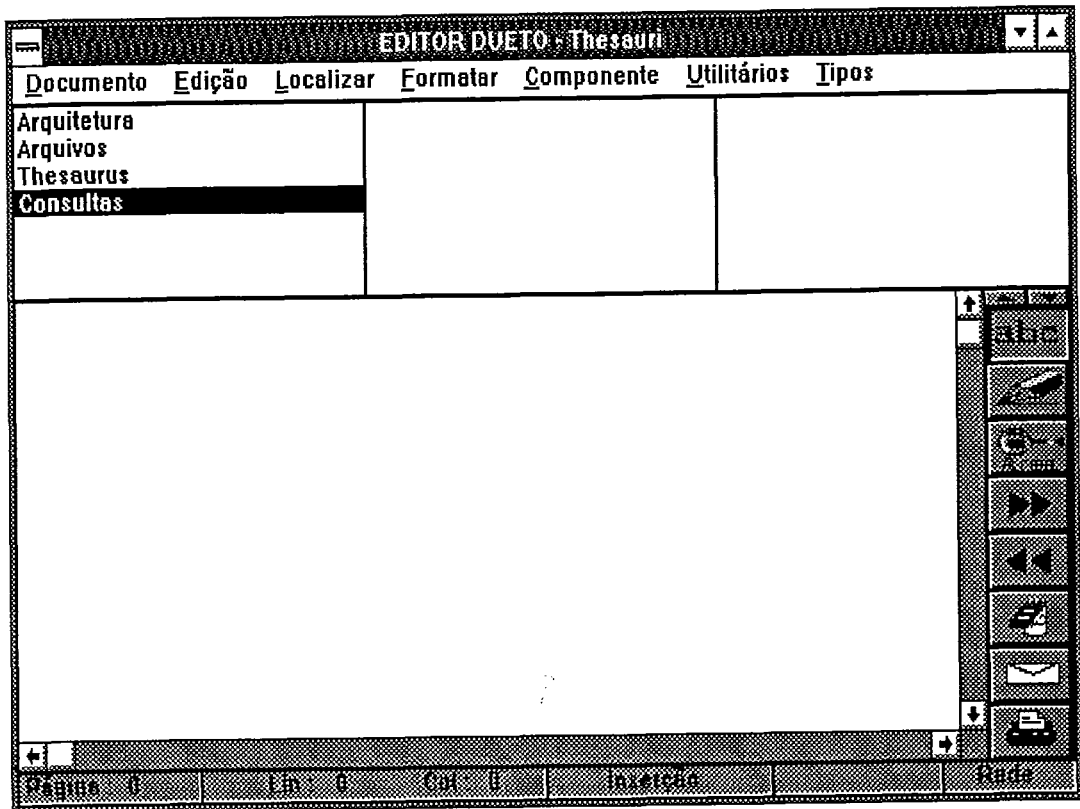


Figura VI.6.1 - Criação de um documento com os primeiros componentes propostos

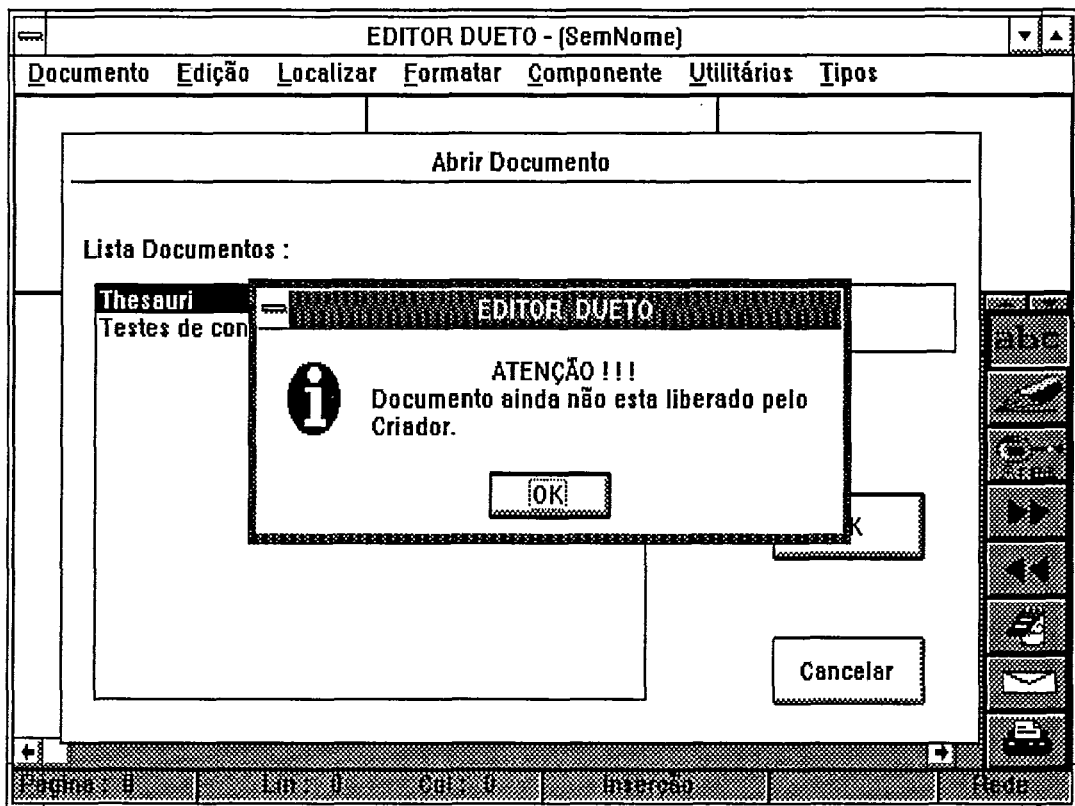


Figura VI.6.2 - Usuário já definido pelo criador tentando acessar um documento não liberado.

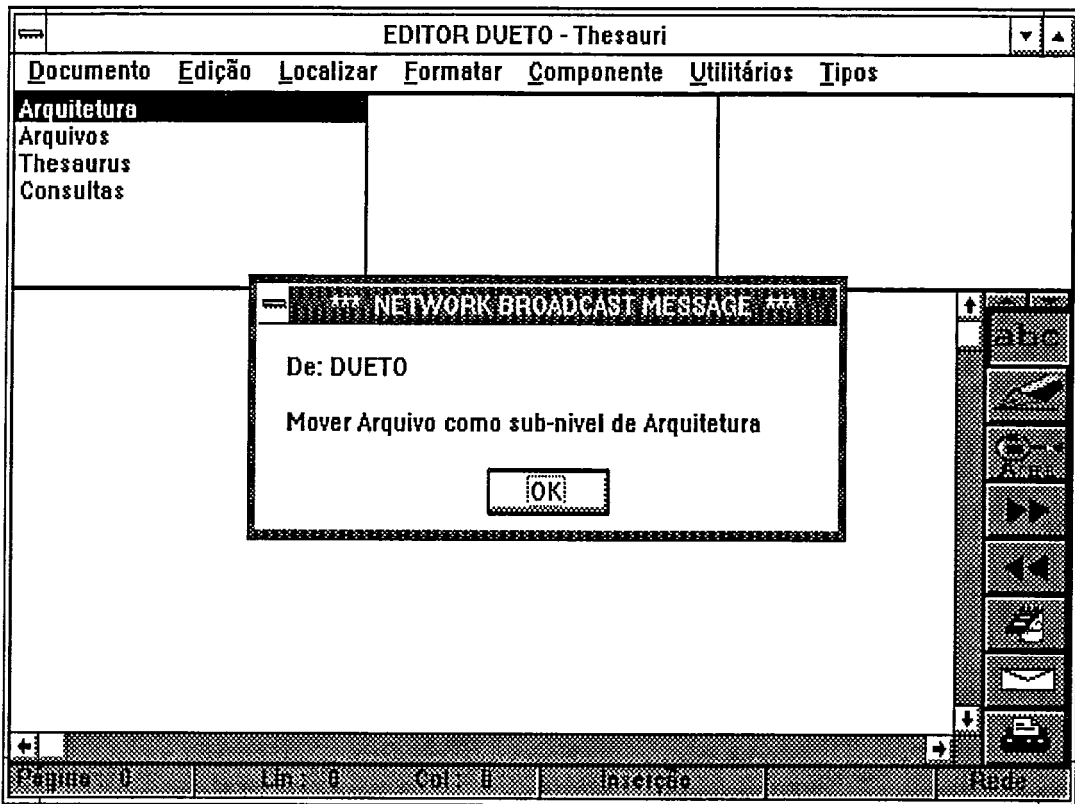


Figura VI.6.3 - Recepção de mensagem com informação sobre a estrutura do documento

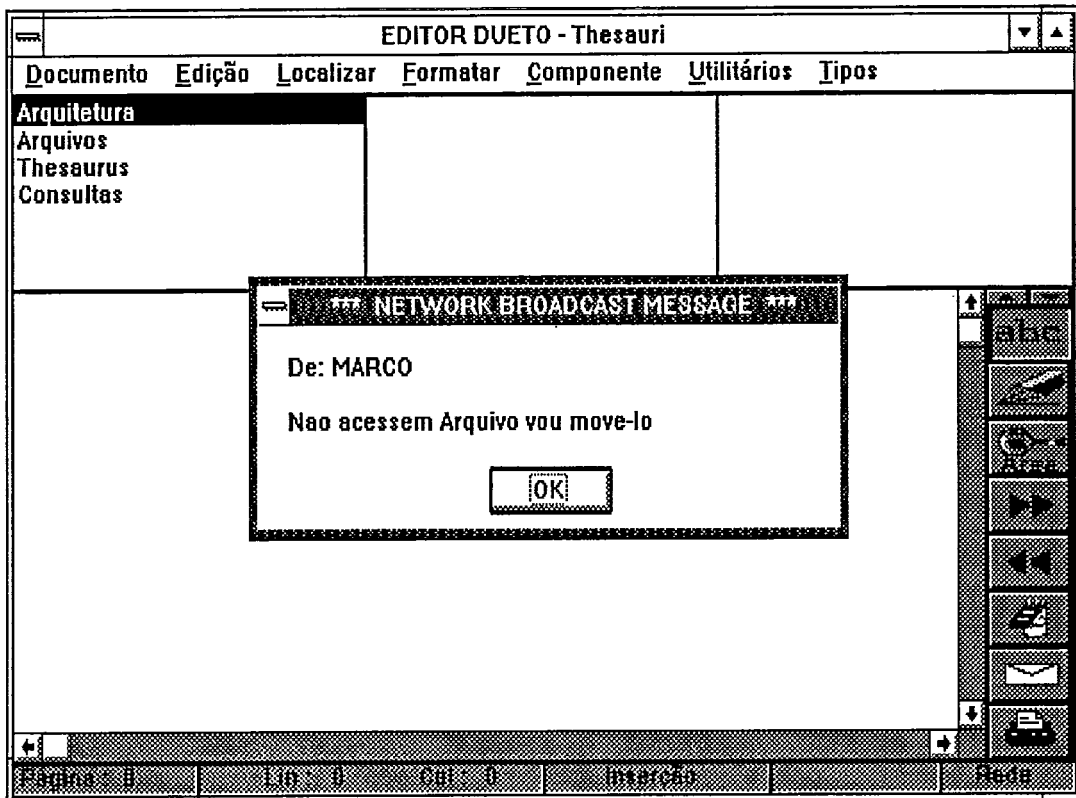


Figura VI.6.4 - Recepção de mensagem com aviso de movimentação de componente



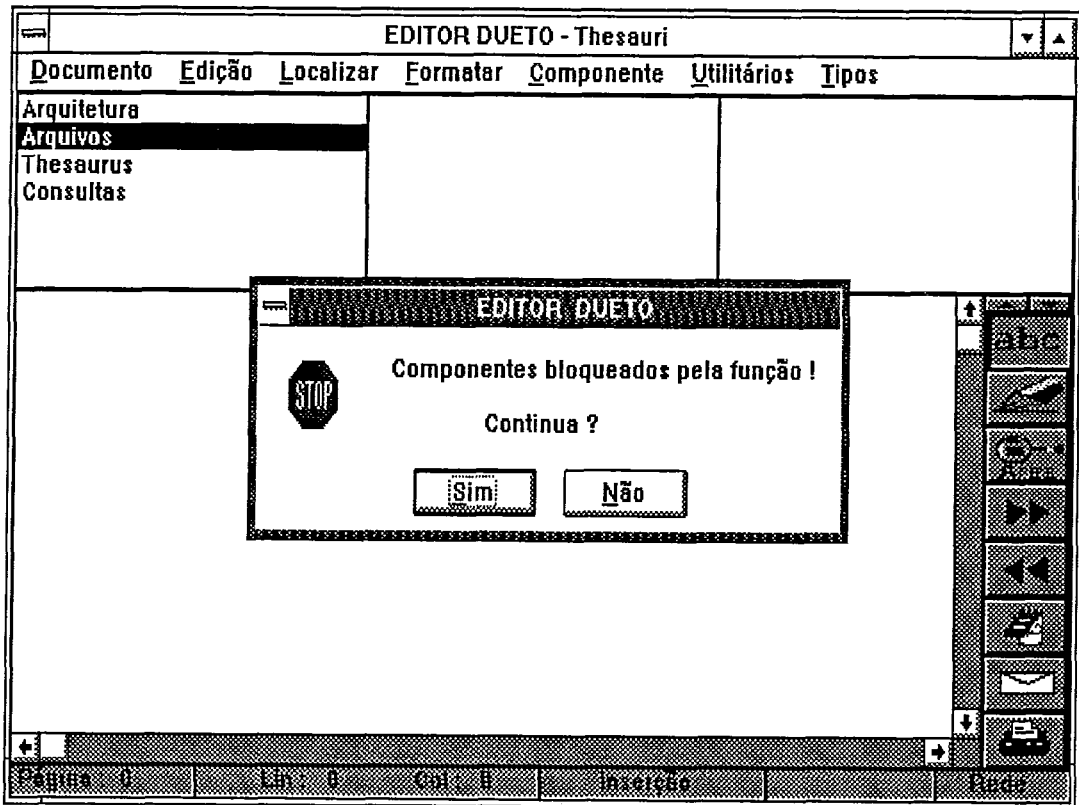


Figura VI.6.5 - Tela com informação de bloqueio do componente a ser movido

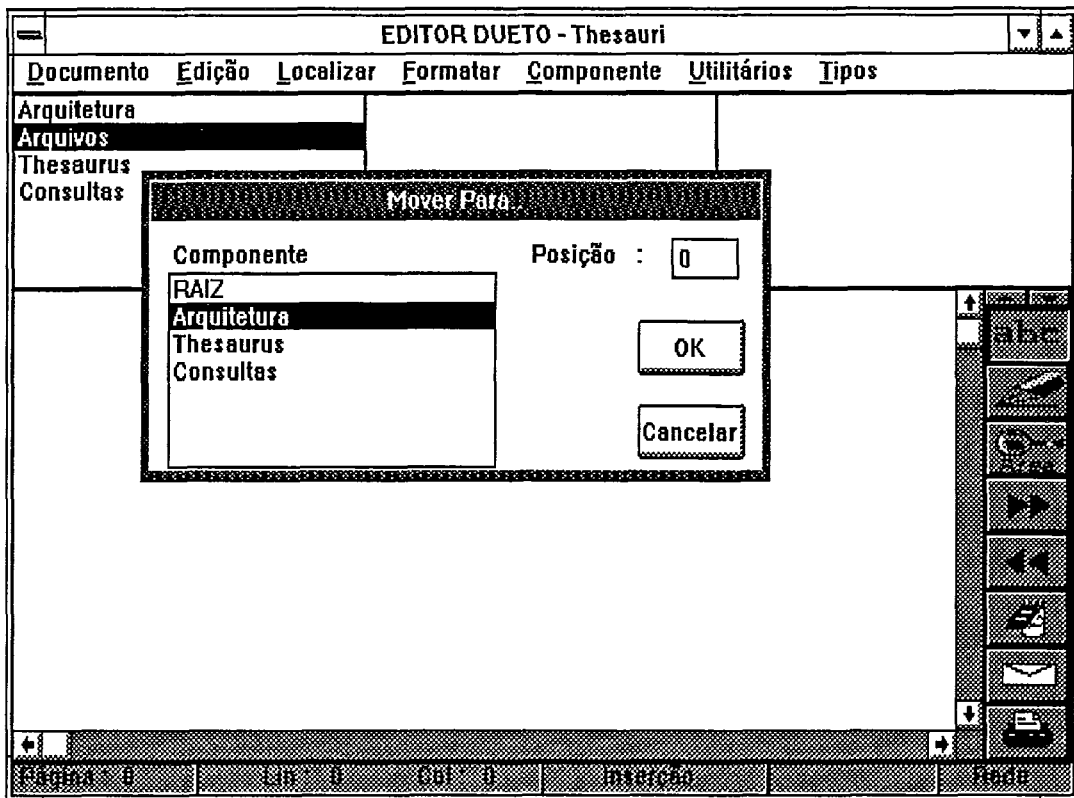


Figura VI.6.6 - Tela para escolha de um novo componente raiz

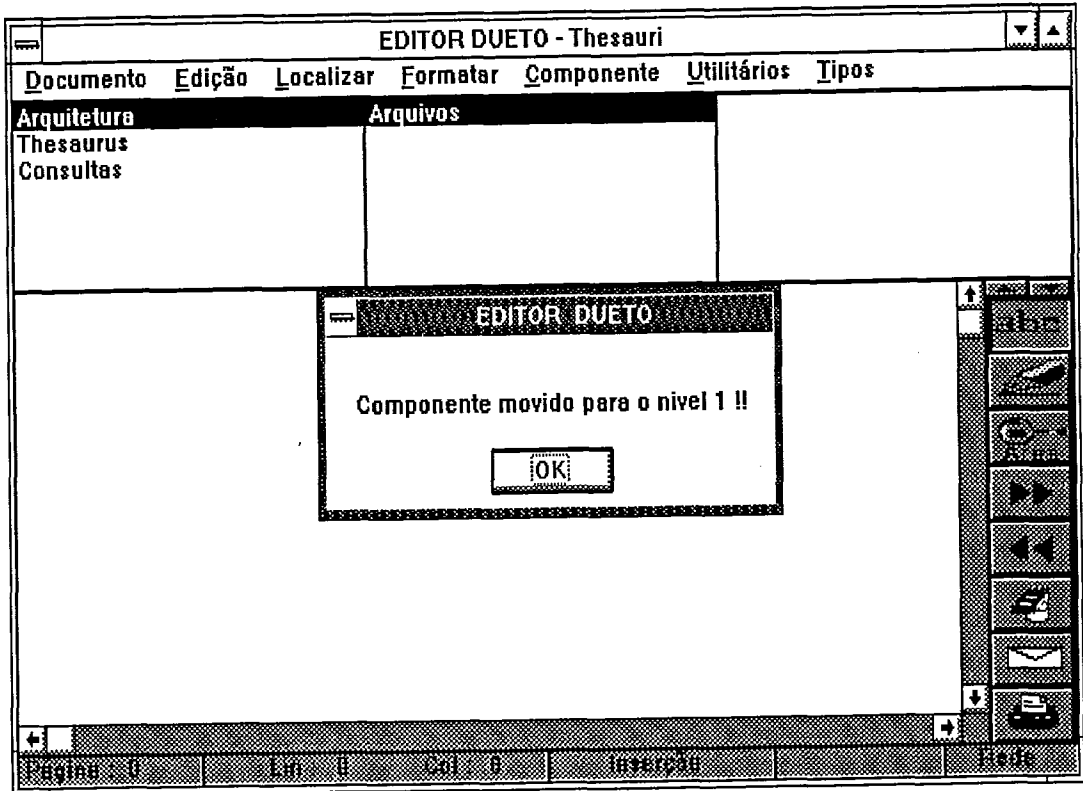


Figura VI.6.7 - Tela com informação da movimentação do componente

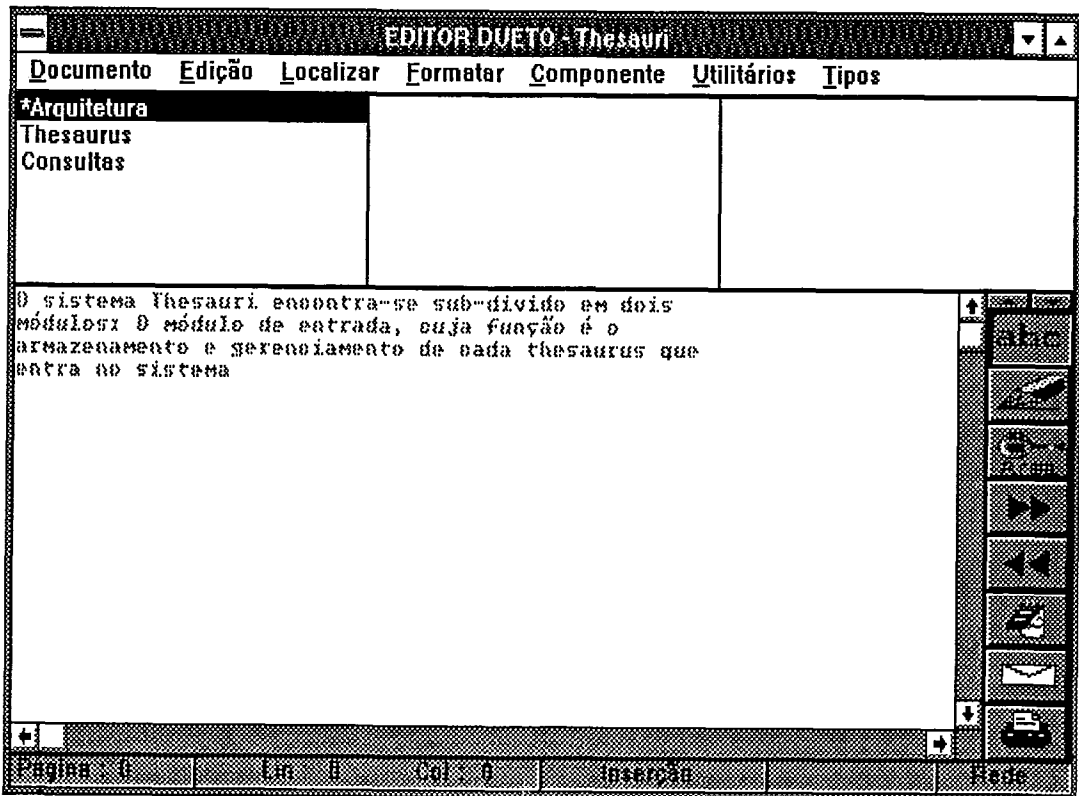


Figura VI.6.8 - Acesso a um componente bloqueado por outro usuário(opção Componente/Editar)

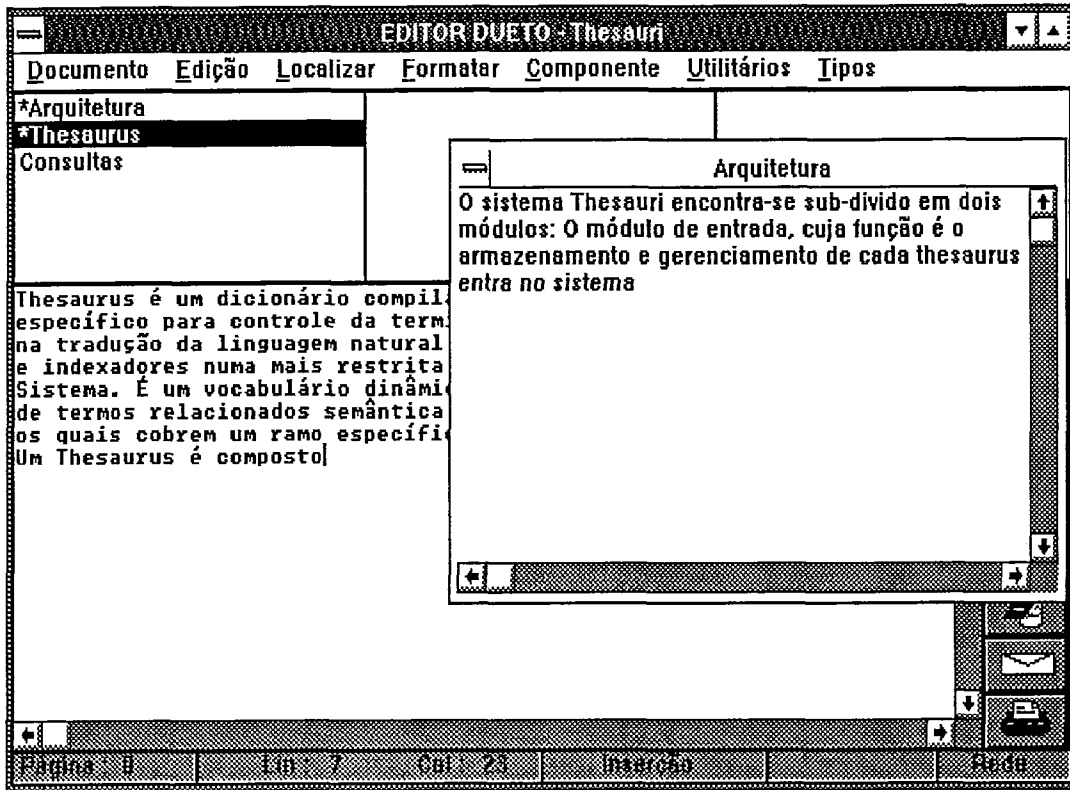
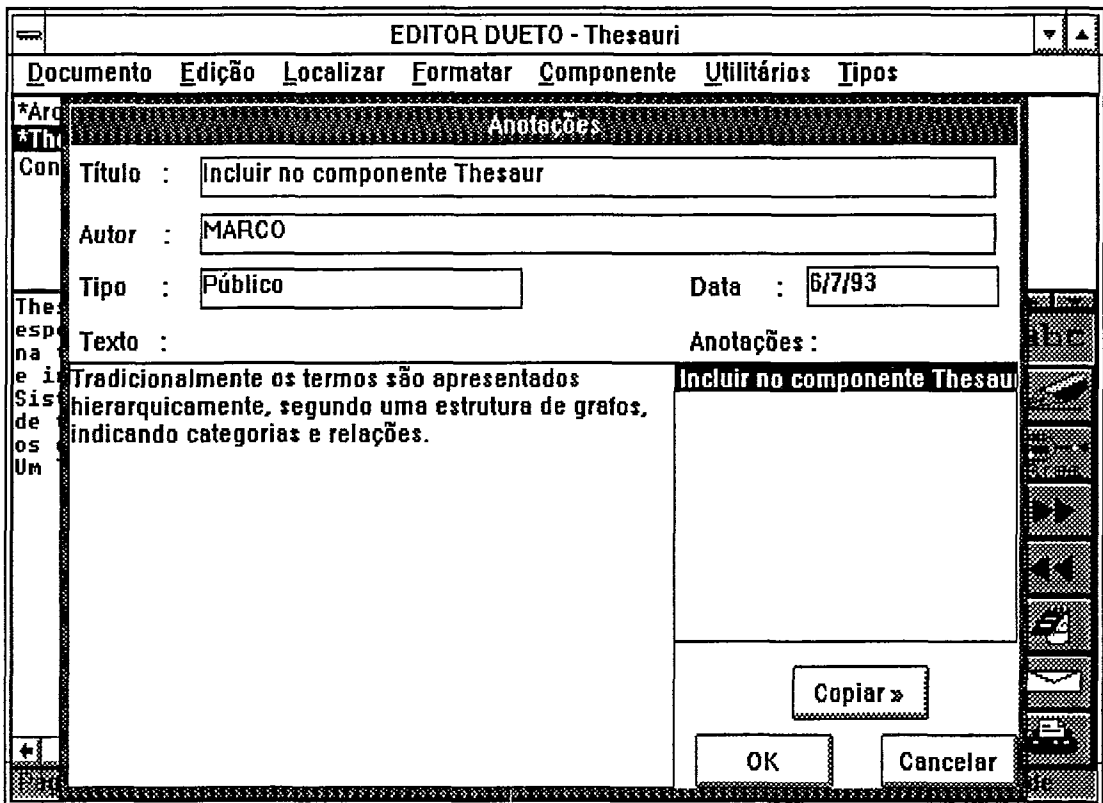


Figura VI.6.9 - Visualização de um componente editado por outro usuário(opção Conteúdo)



VI.6.10 - Inserção de comentário em um componente

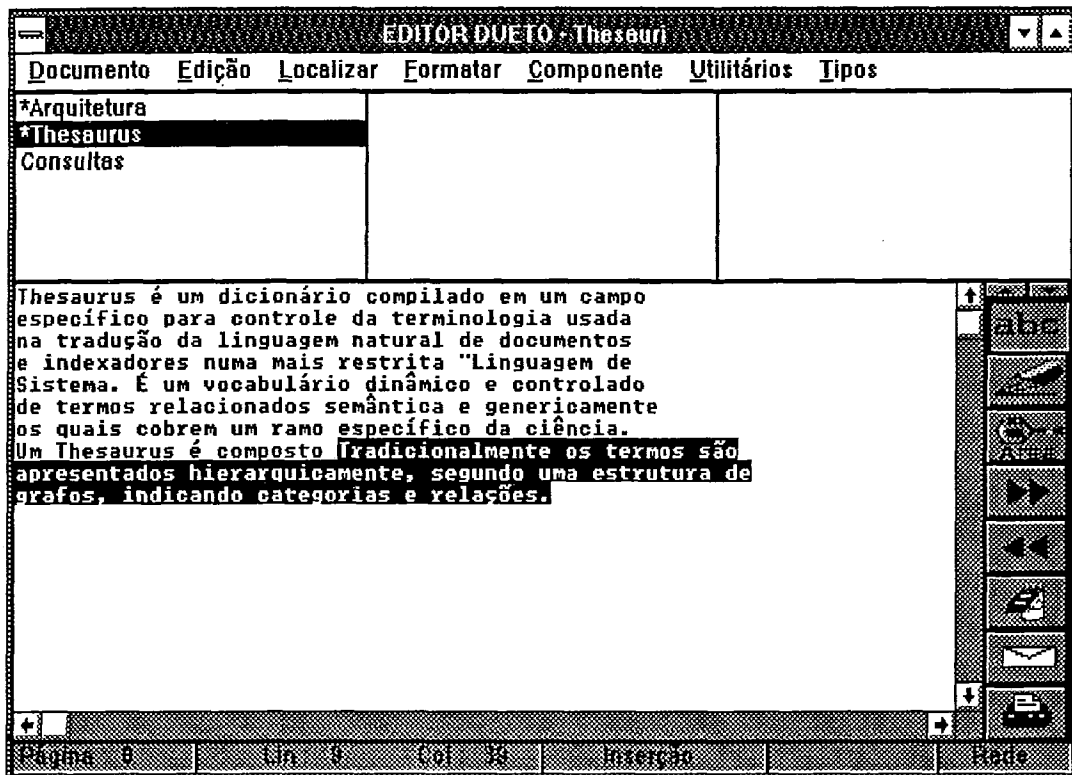


Figura VI.6.11 - Inclusão de comentário no texto do componente

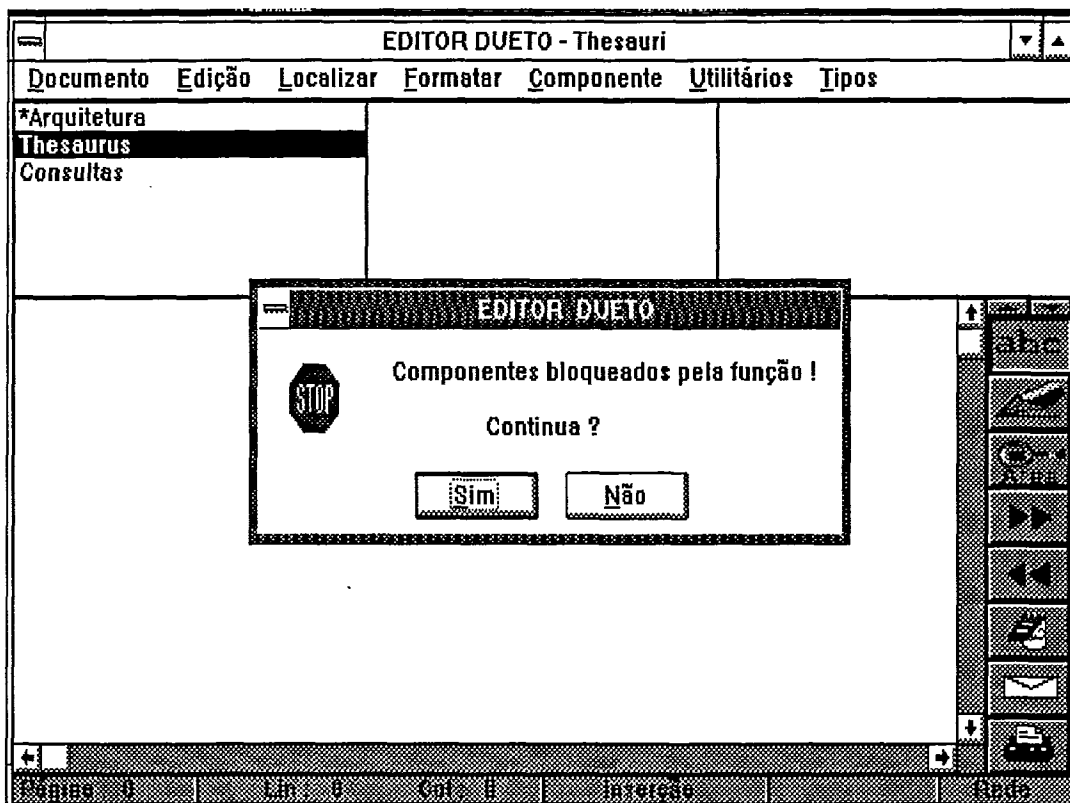


Figura VI.6.12 - Tela para confirmação de continuidade da função Configuração

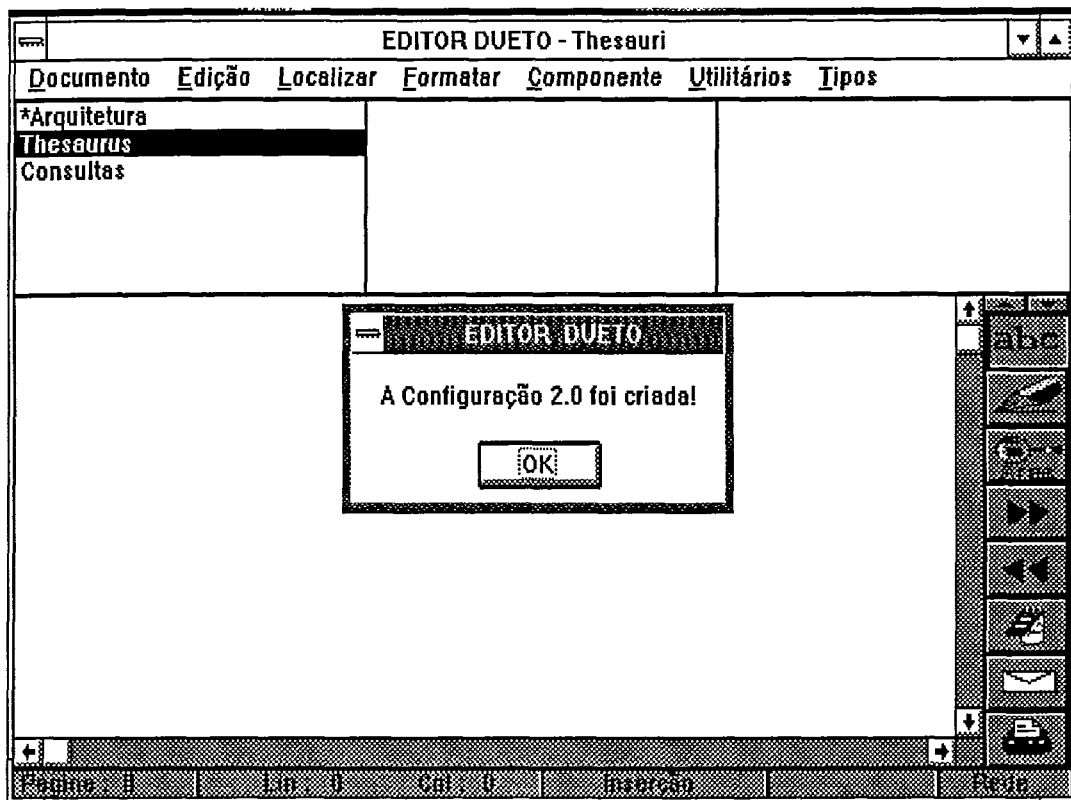


Figura VI.6.13 - Informação da criação de uma nova configuração do documento

## VII - CONCLUSÕES

O trabalho apresentado nessa dissertação reuniu diferentes aspectos relacionados aos ambientes de edição cooperativa de documentos, permitindo, assim, propor e implementar um protótipo de um editor cooperativo de documentos que levasse em conta todas as características destes ambientes.

Apesar de o **DUETO** não ser um sistema de hipertexto, mostramos que muitas das características implementadas são baseadas nestes sistemas, pois achamos que os sistemas de hipertextos são ferramentas importantes no processo de estruturação do conhecimento.

Com base nos estudos realizados, chegamos à proposta de um projeto que possui a maioria das características básicas de um sistema de edição cooperativa em ambientes de co-autoria de documentos. A implementação destas características resultou em algumas vantagens, tais como:

- Facilidades na estruturação dos documentos da maneira mais adequada ao grupo de co-autores;
- Flexibilidade na navegação ao longo do texto, desobrigando os participantes a seguirem numa única direção;
- Suporte à comunicação entre os diversos participantes através de mensagens, comentários e anotações nos componentes da estrutura dos documentos, permitindo uma maior interação entre os participantes;
- Suporte à definição de “papéis” sociais, com intuito de se especificar as funções e responsabilidades dos participantes de um documento, permitindo, assim, a redução dos problemas relacionados à coordenação das atividades no ambiente de co-autoria;
- Facilidades na visualização simultânea de vários componentes da estrutura do documento (Interface multi-janelas);

- Facilidades na visualização das ações efetuadas por outros participantes com possibilidades de interagirem entre si através da troca de mensagens direta;
- facilidades para a criação de versões dos componentes e para a criação de diferentes configurações dos documentos;

Quanto ao **DUETO**, tentamos propor uma ferramenta que fosse flexível para permitir a modularização das idéias dos diversos colaboradores, que podem ser estruturadas através de contribuições individuais na forma de comentários críticos e anotações, com intuito de se chegar a uma estrutura coerente do documento. Além disso, o suporte à cooperação provido pelo editor não é intrusivo, isto é, não interfere no trabalho individual quando usado em ambiente mono-usuário, fazendo com que o **DUETO** possa funcionar como um editor de textos convencional.

Mesmo assim, achamos importante a utilização do **DUETO** em grupos reais, pois existe uma série de questões a serem verificadas para que possamos obter respostas para os aspectos sociais que estão envolvidos nos ambientes de trabalho cooperativo. O uso dessa ferramenta possibilitará adequar as funções oferecidas e acrescentar novas funções que possam ser necessárias nesses ambientes. Em função dos resultados obtidos, o escopo de abrangência do **DUETO** poderá ser ampliado para as diversas áreas envolvidas com a produção conjunta de documentos.

Há muito a pesquisar nesta área, mas acreditamos, com esse trabalho, ter contribuído com a pesquisa sobre o tema. Algumas questões continuam em aberto, e podem ser objeto de trabalhos futuros, tais como:

- Criação de um sistema eficiente, que permita armazenar e gerenciar os documentos, eliminando assim o módulo de controle de arquivos das estações;
- Criação de um mecanismo eficiente na recuperação dos documentos após falhas do sistema;
- Criação de mecanismos que possibilitem o acesso e manipulação de múltiplas versões paralelas dos documentos e que permitam também a comparação entre as diferentes versões de modo a trilhar a história do documento;
- Criação de mecanismos (p. ex. Mapa gráfico) que possibilitem orientar os leitores quanto a sua localização na estrutura dos documentos, quando o acesso é efetuado através de trilhas pré-definidas.

Quanto a implementação do **DUETO**, tê-lo desenvolvido em uma linguagem convencional exigiu tempo e esforço grande. Poderíamos ter obtido resultados mais rápidos se a linguagem escolhida para o desenvolvimento fosse baseada no paradigma da orientação a objetos. O desenvolvimento de aplicações para o ambiente Windows também não foi tarefa fácil, apesar das diversas vantagens oferecidas por este ambiente. Cabe ainda ressaltar que a utilização da biblioteca de funções desenvolvida pela Novell foi extremamente valiosa, não apenas para os processos de comunicação e de controle de acesso, mas também para permitir a integração do **DUETO** ao próprio ambiente de redes.



## BIBLIOGRAFIA

- [Ahu88] - Ahuja, S. R.; Ensor, J. R. & Horn, D. N. The Rapport Multimedia Conferencing System. ACM SIGOIS Bulletin 2(3):1-8p. March 1988.
- [Aksc87] - Akscyn, R.; McCracken, D. & Yoder, E. KMS: A Distributed Hypermedia System for Managing Knowledge in Organizations. Hypertext'87 Papers. 1-20p. Chapel Hill, N.C.. November 1987.
- [Baum88] - Baumann, P. & Köhler, D. Archiving Versions and Configurations in a Database System for System Engineering Environments. Proceedings of the International Workshop on Software Version and Configuration Control. Grassau, West Germany. 313-325p., January 27-29, 1988.
- [Bern88] - Bernstein, M. The Bookmark and the Compass: Orientation Tools for Hypertext Users. ACM SIGOIS Bulletin. 2(4):34-45p., October 1988.
- [Beve88] - Bever, M. & Ruland, D. Aggregation and Generalization Hierarchies in Office Automation. Proceedings Conference on Office Information Systems. 250-264p. March 23-25, 1988. Palo Alto California.
- [Bieb92] - Bieber, M. P. & Kimbrough, S. O. On Generalizing the Concept of Hypertext. Manage. Inf. Syst. Quarterly. 16(1):77-93p., March 1992.
- [Bush88] - Bush, V. As We May Thing. in GREIF, I. Computer -supported Cooperative Work: A book of readings. Ed. Morgan Kaufmann Publishers, San Mateo, California, 1988.
- [Cama92] - Camargo, C. S. P. FLECHA : Um Editor Gráfico Cooperativo para o Modelo de Objetos do GEOTABA. Tese de Mestrado. COPPE-UFRJ. 86p. 1992.
- [Catl89] - Tomothy, C.; Bush, P. & Yankelovich, N. InterNote: Extending a Hypermedia Framework to Support Annotative Collaboration. Hypertext'89 Proceedings. 365-379p. Pittsburgh. Pennsylvania. November 1989.
- [Chen76] - Chen, P. P. The Entity-Relationship Model : Toward a unified View of Data. ACM transactions on Database Systems. 1(1). March 1976.

- [Chri86] - Christodoulakis, M. et alli. Multimedia Document Presentation, Information Extraction and Formation in MINOS: A Model and a System. ACM TOIS 4(4):345-383p., October 1986.
- [Conk88] - Conkling, J. Hypertext: An Introduction and Surveys. in Greif, I. Computer-supported Cooperative Work: A book of readings. Ed. Morgan Kaufmann Publishers, San Mateo, California, 1988.
- [Crof87] - Croft, W. B. & Stemple, D. W. Supporting Office Document Architectures with Constrained Types. Proceedings of ACM SIGMOD 1987, Annual Conference San Francisco. May 27-29, 1987.
- [Davi86] - Davison, J. W. & Zdonik, S. B. A Visual Interface for a Database with Version Management. ACM TOIS. 3(3):226-256p., July 1986.
- [Deli87] - Delisle, N. M. & Schwartz, M. D. Context - A Partitioning Concept for Hypertext. ACM TOIS 5(2):168-186p., April 1987.
- [Deli89] - Delisle, N. M. & Schwartz, M. D. Collaborative Writing with Hypertext. IEEE Transactions on Professional Communication. 32(3):183-188p. September 1989.
- [Dero89] - Deroose, S. J. Expanding the Notion of Links. Hipertext'89 Proceedings. 61-81p. Pittsburgh. Pennsylvania. November 1989.
- [Dour92] - Dourish, P. & Bellotti, V. Awareness and Coordination in Shared Workspaces. Conference on Computer-Suported Cooperative Work. Proceedings CSCW'92. Toronto, Canada. November 1992. ACM. 107 -114p. 1992.
- [Eglo90] - Eglowstein, H.; Wszola, S. & Thompson, T. Word Processor: That Build Character. Byte 15(9):132-152 p., September 1990.
- [Elli88] - Ellis, C. A. & Nutt, G. J. Office Information Systems and Computer Science. Computer-supported Cooperative Work: A book of readings. Ed. Morgan Kaufmann Publishers, San Mateo, California, 1988.
- [Elli91] - Ellis, C. A. et alli. GROUPWARE - Some Issues and Experiences. Communications of the ACM 34(1):39:58p., January 1991.
- [Eng88] - Engelbart, D. & Lehtman, H. Working Together. Byte 13(13):245-252 p. December 1988.

- [Eng88a] - Engelbart, D. A Conceptual Framework for the Augmentation of Man's Intellect. in Greif, I. Computer-supported Cooperative Work: A book of readings. Ed. Morgan Kaufmann Publishers, San Mateo, California, 1988.
- [Evel88] - Eveland, J. D. & Bikson, T. K. Work Group Structures and Computer Support: A Field Experiment. ACM TOIS. 6(4):354-379p., October 1988.
- [Ezze91] - Ezzell, Ben. Borland C++ 2.0 Programming. Addison Wesley. 1<sup>a</sup> ed. 563p. 1991.
- [Fish88] - Fish, R. et alli QUILT: A Collaborative Tool for Cooperative Writing. Conference on Office Information Systems. Palo Alto, California. March 2-3, 1988. 30-37p.
- [Flor88] - Flores, F. et alli. Computer Systems and Design of Organizational Interaction. ACM TOIS 6(2):153-172p., April 1988.
- [Foun90] - Fountain, A. M. et alli MICROCOSM: An Open Model for Hypermedia with Dynamic Link. Hypertext Concepts systems and applications. Proceedings of the European Conference on Hypertext. Versailles. November 27-30, 1990. 298-311p.
- [Furu82] - Furuta, R.; Scofield, J. & Shaw, A. Document Formatting Systems. ACM Comp. Surv. 14(3):417-472p., September 1982.
- [Gess90] - Gessner, R. Building a Hypertext System- Hypertext for Every Programmer's Toolbox. Dr. Dobb's Journal. June 1990.
- [Gree90] - Greenberg, S. Sharing Views and Interactions with Single-User Applications. Conference on Office Information Systems. ACM SIGOIS Bulletin. 11(2,3):227-237p., April 25-27, 1990. Cambridge, Massachusetts, 1990.
- [Grei86] - Greif, I.; Seliger, R. & Weihl, W. Atomic Data Abstraction in a Distributed Collaborative Editing System. 13 th. Annual Symposium on principles of programming languages. 160-172p., St. Petersburg, ACM, January 1986.
- [Grei88] - Greif, I. & Sarin, S. Data sharing in Group Work. in Greif, I. Computer-supported Cooperative Work: A book of readings. Ed. Morgan Kaufmann Publishers, San Mateo, California, 1988.
- [Grei92] - Greif, I.; Seliger, R. & Weihl, W. A Case Study of CES: A Distributed Collaborative Editing System Implemented in Argus. IEEE Transactions on Software Engineering. 18(9):827-839p., September 1992.

- [Grun91] - Grundin, J. CSCW : Introduction. Communications of the ACM 34(12):36:51p., December 1991.
- [Haak92] - Haake, J. M. & Wilson, B. Supporting Collaborative Writing of Hyperdocuments in SEPIA. Conference on Computer-Suported Cooperative Work. Proceedings CSCW'92. Toronto, Canada. November 1992. ACM. 138-146p. 1992.
- [Hala87] - Halasz, F. G. Reflections on Notecards: Seven Issues for the Next Generation of Hypermedia systems. Hipertext'87 Papers. 345-365p. Chapel Hill, N.C.. November 1987.
- [Hala88] - Halasz, F. G. Reflections on Notecards: Seven Issues for the Next Generation of Hypermedia systems. Communications of the ACM. 31(7): 837- 852p., July 1988.
- [Halo90] - Halonen, D. Shared Hardware: A Novel Technology for Computer Support of Face to Face Meetings. Conference on Office Information Systems. ACM SIGOIS Bulletin. 11(2,3):227-237p., April 25-27,1990. Cambridge, Massachusetts, 1990.
- [Hahn89] - Hahn, U. et alli. Co-AUTHOR - A Hypermedia Group Authoring Environment. Proceedings of the EC- CSCW'89 - First European Conference on Computer Supported Co-operative Work. Gatwick, London, UK. September 13-15, 1989.
- [Hirs86] - Hirschheim, R. A. Understanding the Office: A Social-Analytic Perspective. ACM TOIS. 4(4):331-344p., October 1986.
- [Hora84] - Horak, W. & Kronet, G. An Object - Oriented Office Document Architeture Model for Processing and Interchange of Documents. Proceedings of second ACM - SIGOA Conference on Office Information Systems. june 25-27, 1984.
- [Iris89] - Irish, P. M. & Trigg, R. H. Supporting Collaboration in Hypermedia: Issues and Experiences. Journal of American Society for Information Science. 40(3):192-199p., May 1989.
- [Ishi91] - Ishi, H. & Miyake, N. Toward an Open Shared Wokspace: Computer and Video Fusion Aproach of Teamworkstation. Communications of the ACM 34(12):36:51p., December 1991.
- [Jame85] - James, G. Document Databases. Ed. Van Nostrand Reinhold Company Inc. 184p. 1985.

- [Katz86] - Katz, R. H. et ali. Version Modelling Concepts for Computer-aided Design Databases. Proc. of ACM SIGMOD Int'l Conference on Management of Data. Washington, DC May 28-30, 1986. 132-143p. 1986.
- [Kenn88] - Kennington, R. W.; Seeley, D. A. & Morales, J. R. A Collaborative Document Editing System. Australian Computer Conference. 1988' Information Technology Towards 2000 Proceedings. Sydney, NSW, Australia, Sept 21-23, 1988. 89-109p.
- [Knis90] - Knister, M. J. & Prakash, A. DistEdit: A Distributed Toolkit for Supporting Multiple Group Editors. Conference on Computer-Suported Cooperative Work. Proceedings CSCW'90. Los Angeles, CA. October 1990. New York : ACM. 343-355p. 1990.
- [Krae88] - Kraemer, K. L. & King, J. L. Computer - Based Systems for Cooperative Work and Group Decision Making. ACM Comp. Surv. 14(3)., September 1982.
- [Lage86] - Lages, N. A. & Nogueira, J. M. S. Introdução aos Sistemas Distribuídos. Ed. Unicamp, 229p., SP, São Paulo, 1986.
- [Lee88] - Lee, J. & Malone, T. W. How can Groups Communicate when they use Different Languages? Translating between Partially Shared Types Hierarchies. ACM SIGOIS Bulletin 2(3): 22-29p., March 1988.
- [Lewi88] - Lewis, B. T. & Hodges, J. D. Shared Books: Collaborative Publication Management for an Office Information System. ACM SIGOIS Bulletin 2(3):197-204p., March 1988.
- [Malc91] - Malcolm, N. & Gaines, B. R. A Minimalist Approach to the Development of a Word Processor Supporting Group Writing Activities. in ACM SIGOIS Bulletin 12(2,3):147-152p. Conference on Organizational Computing Systems. Atlanta, Georgia. November 5-8, 1991.
- [Matt90] - Mattos, N. M. et ali. An Approach to Integrated Office Document Processing and Management. Conference on Office Information Systems. ACM SIGOIS Bulletin. 11(2,3):118-122p., April 25-27, 1990. Cambridge, Massachusetts, 1990.
- [Melo88] - Melo, R. N. Banco de Dados Não Convencionais: A Tecnologia do BD e suas novas Áreas de Aplicação. VI Escola de Computação, Campinas, S.P. 1988.
- [Mend92] - Mendelson, E. Seven Windows Word Processors -What You'll See Is What You'll Want. PC Magazine. 4(11):113-176p., February 25, 1992.

- [Mend92b] - Mendes, M. E. X. O Paradigma de Hipertexto e o Processo de Ensino/Aprendizagem: Uma Relação Promissora. Tese de Mestrado. COPPE-UFRJ. 170p. 1992.
- [Mend90] - Mendelson, E. Word Processors: The Best and the Brighest. PC Magazine. *9*(21):107-192p., December 1990.
- [Meyr82a] - Meyrowitz, N. & Van Dam, A. Interactive Editing Systems: Part I. ACM Comp. Surv. *14*(3):1-354p., September 1982.
- [Meyr82b] - Meyrowitz, N. & Van Dam, A. Interactive Editing Systems: Part II. ACM Comp. Surv. *14*(3):355-415p., September 1982.
- [Micr87] - Microsoft Corporation Microsoft Windows User's Guide. Washington, USA. 1987.
- [Muhl90] - Muhlhauser, M. Software Engineering in Distributed Systems - Approaches and Issues. Ninth Annual International Phoenix Conference on Computers and Communications. 408-418p. Scottsdale, AZ, USA. 21-23 March, 1990.
- [Naff88] - Naffah, N. Office Automation Systems: An Overview. Proceedings of first International Informatics Congress of Rio de Janeiro (SUCESU). 166-176p., Brazil, 22-26 Agosto. 1988.
- [Nava89] - Navathe, S. B. & Elmasri, R. Fundamentals of Database Systems. The Benjamin/Cummings Publishing Company, Inc. 1989.
- [Neuw90] - Neuwirth, C. M. et alli Issues in the Design of Computer Support for Co-authoring and Commenting. CSCW'90-Proceedings of the Third Conference on Computer Supported Cooperative Work. Los Angeles, California. October 1990. 183-195p.
- [Newm91] - Newman-Wolfe, R. E. & Pelimuhandiram, H. K. MACE: A Fine Grained Concurrent Editor. in ACM SIGOIS Bulletin *12*(2,3):240-254p. Conference on Organizational Computing Systems. Atlanta, Georgia. November 5-8, 1991.
- [Niel90] - Nielsen, J. Trough Hipertext. Comm. of the ACM. *33*(3):297-310p., March 1990.
- [Niel90b] - Nielsen, J. Hypertext and Hypermedia. Academic Press, Inc. 263 p. 1990.
- [Nier89] - Niertrasz, O. M. & Tschritzis, D. C. Integrated Office Systems. in Kim, W. & Lochovsky, F. H. **object - oriented concepts, databases and applications**. New York, ACM. Reading, Adison-Wesley. 1989.

- [Nove89] - Novell Inc. Netware C Interface-DOS. Professional Development Series. Vol. I/II Austin, Texas. June 1989. 1989.
- [Nove90] - Novell Inc. Netware Utilities Reference. Provo, Utah. June 1990. 1990.
- [Ohku90] - Ohkubo, M. & Ishii, H. Design and Implementation of a Shared Workspace by Integrating Individual Workspaces. Conference on Office Information Systems. ACM SIGOIS Bulletin. *11*(2,3):227-237p., April 25-27, 1990. Cambridge, Massachusetts, 1990.
- [Olso92] - Olson, J. S. et alii. How a Group-Editor Changes the Character of a Design Meeting as well as its Outcome. Conference on Computer-Suported Cooperative Work. Proceedings CSCW'92. Toronto, Canada. November 1992. ACM. 91- 98p. 1992.
- [Parsa89] - Parsaye, K. et alii. Intelligent Databases - Objectet-Oriented, Deductive Hypermedia Technologies. John Wiley & Sons , Inc. 477p. 1989.
- [Perl89] - Perlman, G. Asynchronous Design/Evaluation Methods for Hypertext Technology Development. Hipertext'89 Proceedings. 61-81p. Pittsburgh. Pennsylvania. November 1989.
- [Pime89] - Pimentel, M.G.C. Sistemas hipertextos: Discussões e uma Proposta. Anais do XXII Congresso Nacional de Informática - SUCESU. São Paulo, 128-134p., 1989.
- [Pint90] - Pintado, X. & Tschritzis, D. SATELLITE: A Visualization and Navigation Tool for Hypermedia. Conference on Office Information Systems. ACM SIGOIS Bulletin. *11*(2,3):227-237p. April 25-27, 1990. Cambridge, Massachusetts, 1990.
- [Prev90] - Prevelakis, V. Versioning Issues for Hypertext Systems. Object Manegement - Gestion d'Objects. Centre Universitaire d'Informatique. 89-105p. Edited by D. Tschritzis. July 1990.
- [Rada89] - Rada, R. Writting and Reading Hypertext: An Overview. Journal of American Society for Information Science. *40*(3):164-171p. May 1989.
- [Radab89] - Rada, R. et alii. Collaborative Writting of Text and Hypertext. Hypermedia. *1*(2):93-110p., 1989.
- [Rose89] - Rose, C. Client-Server Computing Sophisticated Lan. LAN Technology. 50-56p., December 1989.

- [Salt89] - Salton, G. Automatic Text Processing: The Transformation, Analysis, and Retrieval of Information by Computer. Addison-Wesley Publishing Company. 530p. 1989.
- [Scac89] - Scacchi, W. On the Power of Domain-Specific Hypertext Environments. Journal of American of Society for Information Science. 40(3):164-171p., May 1989.
- [Sei86] - Relatório da Comissão Especial de Automação de Escritórios. SEI - Secretaria Especial de Informática. Ministério das Minas e Energia. 1986.
- [Setz86] - Setzer, V. W. Projeto Lógico e Projeto físico de Banco de Dados. V Escola de Computação. Belo Horizonte. 289p. 1986.
- [Shen92] - Shen, H. & Dewan, P. Access Control for Collaborative Environments. Conference on Computer-Suported Cooperative Work. Proceedings CSCW'92. Toronto, Canada. November 1992. ACM. 51-58p. 1992.
- [Ship89] - Shipman, F. M.; Chaney, R. J. & Gory, G. A. Distributed Hypertext for Collaborative Research: The Virtual Notebook System. Hipertext'89 Proceedings. 129-135p., Pittsburgh. Pennsylvania. November 1989.
- [Smit87] - Smith, J. B.; Weiss, S. F. & Ferguson, G. J. A Hipertext Writing Environment and its Cognitive Basis. Hipertext'87 Papers. 195-214p., Chapel Hill, N.C.. November 1987.
- [Sobi91] - Sobiesiak, R. & Mylopoulos, J. A Conceptual Modelling Approach to Authoring-in-the-Large for Hypertext Documents. in ACM SIGOIS Bulletin 12(2,3):225-239p. Conference on Organizational Coputing Systems. Atlanta, Georgia. November 5-8, 1991.
- [Stef88] - Stefik, M. et alli Beyond the Chalkboard: Computer Support for Collaboration and Problem Solving Meeting. in Greif, I. Computer-supported Cooperative Work: A book of readings. Ed. Morgan Kaufmann Publishers, San Mateo, California, 1988.
- [Stre89] - Streitz, N. A. et alli. From Ideas and Arguments to Hyperdocuments: Travelling through Activity Spaces. Hipertext'89 Proceedings. 343-363p. Pittsburgh. Pennsylvania. November 1989.
- [Tane89] - Tanenbaum, A. S. Computer Network. 2ªed. Prentice-hall International Editions. 658p. 1989.



- [Trig86] - Trigg, R. H. & Weiser, M. TEXTNET: A Network - Based Approach to text Handling. ACM TOIS 4(1):1-23p. January 1986. 1986.
- [Tsic90] - Tschritzis, D. & Prevelakis, V. Perspectives on Hypertext Structures Systems. Object Manegement - Gestion d'Objects. Centre Universitaire d'Informatique. 89-105p., Edited by D. Tschritzis. july 1990.
- [Wai89] - Wai, G. Take your Pick. Byte, 14(7):215-225p. July 1989. 1989.
- [Walk88a] - Walker, J. H. Supporting Document Develepment with Concordia. Computer. 48-59p. January 1988.
- [Walk88b] - Walker, J. H. The Role of Modularity in Document Authoring Sysytems. Proc. of the ACM Conference on Document Processing Systems. Santa Fé, New México. December 5-9, 1988. 117-124p. 1988.
- [Woel86] - Woelk, D. et alli An Object-Oriented Approach to Multimedia Databases. Proc. of ACM SIGMOD Int'l Conference on Management of Data. Washington, DC May 28-30, 1986. 132-143p., 1986.
- [Yank88] - Yankelovich, N. ; Meyrowitz, S. & Drucker, S. Intermedia: The Concept and Construction of a Seamless Information Environment. IEEE Computer 21(1):81-96p., January 1988.
- [Youn90] - Young, L. D. Linking Considered Harmful. Hypertext Concepts Systems and Applications. Proceedings of the European Conference on Hypertext. Versailles. November 27-30, 1990. 298-311p.