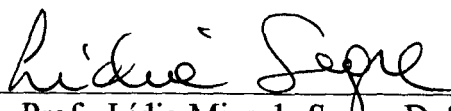


Trabalho Cooperativo: Análise de sua Utilização em Desenvolvimento de Software Baseado em Estudos de Casos

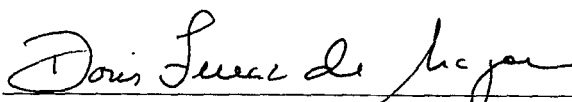
Rosa Maria Esteves Moreira da Costa

Tese submetida ao Corpo Docente da Coordenação dos Programas de Pós-Graduação em Engenharia da Universidade Federal do Rio de Janeiro como parte dos requisitos necessários para a obtenção do grau de Mestre em Ciências em Engenharia de Sistemas e Computação.

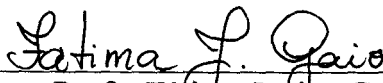
Aprovada por:



Prof. Lídia Micaela Segre, D. Sc.
(Presidente)



Profa. Dóris Ferraz Aragon, I.Docente



Profa. Fátima Janine Gaio, Ph.D.

COSTA, ROSA MARIA ESTEVES MOREIRA DA

Trabalho Cooperativo: Análise de sua Utilização em
Desenvolvimento de Software Baseado em Estudos de Casos
[Rio de Janeiro] 1994

VI, 115 p., 29,7 cm (COPPE/UFRJ, M.Sc, Engenharia de
Sistemas e Computação, 1994)

Tese - Universidade Federal do Rio de Janeiro, COPPE

1 - Trabalho Cooperativo 2 - Apoio Tecnológico à Cooperação

3 - Cooperação em Engenharia de Software

I. COPPE/UFRJ II Título (Série).

*A Esmeralda e Alfredo,
Ao Jade e a Luma,
meus pais e filhos adorados*

♥ Agradecimentos

Existem muitas pessoas que foram fundamentais na realização deste trabalho. É difícil nomeá-las todas, mas não posso furtar-me de citar aquelas que marcaram mais profundamente toda esta caminhada.

Agradeço:

Inicialmente às pessoas que gentilmente se submeteram as entrevistas e que sem as quais esta pesquisa não poderia ter sido realizada;

A CAPES pelo financiamento deste trabalho através da concessão de uma bolsa de estudos;

A meus queridos pais, Alfredo e Esmeralda, cuja ajuda e o incentivo foram, e são, inestimáveis e imprecindíveis;

Ao meu amor, Celso, pelo apoio e carinho;

A minha irmã Rita, irmão Zeca, cunhada Isabel, tia Carmem e vovó Dolores por toda a ajuda e "quebradas-de-galho";

A toda minha família pelo entusiasmo e força, em especial a Conchita e Cristina;

A professora Dóris Aragon que me aconselhou e incentivou nos momentos iniciais, pelas observações e gentil participação na banca examinadora;

A Ana Regina pelo carinho e atenção que me dedicou nos momentos difíceis;

A Neide Santos que me ajudou a dar o "pontapé inicial" neste trabalho, por todas as dicas e "bate-papos";

Ao Jano pelas conversas informais que elucidaram aspectos delicados;

As minhas orientadoras Fátima Gaio e Lídia Segre pelo apoio e interesse constantes;

Minhas amigas que estiveram e estão, mesmo a distância, sempre ao meu lado: Cleusa, Vera e Denise;

Os muitos amigos que contribuíram com a alegria, o incentivo e a esperança. Dentre eles lembro com especial carinho dos seguintes: Gilda, Odinaldo, John, Bebeto, Glória, Cristina, Sérgio, Fernanda, Gisela, Zé Francisco, Xexéo, Célia;

As secretárias e bibliotecária Cláudia, Ana Paula, Rose e Mercedes;

A Deus por ter me dado determinação e força de prosseguir sempre.

Resumo da Tese apresentada a Coordenação dos Programas de Pós-Graduação em Engenharia da Universidade Federal do Rio de Janeiro como parte dos requisitos necessários para a obtenção do grau de Mestre em Ciências (M.Sc)

Trabalho Cooperativo: Análise de sua utilização em Desenvolvimento de Software baseado em Estudo de Casos

Rosa Maria Esteves Moreira da Costa
Setembro 1994

Orientadora: Lídia Micaela Segre

Departamento: Engenharia de Sistemas e Computação

Esta tese caracteriza o Trabalho Cooperativo e particulariza a cooperação no desenvolvimento de produtos de software, levantando aspectos inerentes ao processo de desenvolvimento cooperativo de software.

Foram pesquisadas três empresas no intuito de verificar aspectos sociais e técnicos que surgem a partir de uma participação mais intensa dos usuários no ciclo-de-vida do desenvolvimento dos produtos.

Por fim, comparamos os três casos estudados buscando características comuns, que nos permitiram elucidar as variáveis mais importantes.

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

Cooperative Work: An Analysis of its Utilization in Software Development Based on Case Studies

Rosa Maria Esteves Moreira da Costa
September, 1994

Thesis Advisor: Lídia Micaela Segre

Department: Computer and System Engineering

This thesis characterizes the cooperative work and particularizes the cooperative aspects in the development of software products, also describes some features that arise from the cooperative development of software products.

We have visited three companies in different activities, interested in verifying the social and technical aspects due a great participation of the users in the life cycle of products.

Finally, we have compared the three cases studies in order to obtain the main aspects and specific problems of cooperation.

Sumário

Capítulo 1 Introdução	1
1.1 Motivação	1
1.2 Objetivos da tese	2
1.3 Divisão em capítulos	2
Capítulo 2 Trabalho Cooperativo	4
2.1 Definições	4
2.1.1 Grupo	4
2.1.2 Cooperação	5
2.2 O Computador alterando as relações de trabalho cooperativo	6
2.3 Organização e elementos sociais envolvidos na cooperação	8
2.3.1 Comunicação	8
2.3.2 Conflitos	9
2.3.3 Negociações	10
2.3.4 Liderança	11
2.3.5 Coordenação	12
2.4 Conclusões	14
Capítulo 3 O apoio computacional	15
3.1 Benefícios de ambientes computacionais apoiando a cooperação	15
3.2 Discutindo os sistemas de apoio ao trabalho cooperativo	16
3.2.1 Groupware	18
3.2.2 CSCW	24
3.3 Avaliação: discutindo vantagens e desvantagens	26
3.4 Áreas promissoras de adoção do trabalho cooperativo	29
3.4.1 Trabalho cooperativo em educação	30
3.4.2 Cooperação no escritório	32
3.4.3 Propostas de trabalho cooperativo na Engenharia de Software	33
3.4.3.1 Cooperação reintegrando ocupações fragmentadas	33
3.4.3.2 Importância do usuário	34
3.5 Conclusões	36

Capítulo 4 O Processo de Desenvolvimento de Software	37
4.1 Evolução	38
4.2 O Processo de desenvolvimento de software num enfoque tradicional	41
4.2.1 Desenvolvimento feito por equipes	41
4.2.2 Ciclos-de-vida e métodos	46
4.2.2.1 Modelos de ciclo-de-vida	46
4.2.2.2 Métodos e ferramentas	53
4.3 O Processo de desenvolvimento de software num enfoque de trabalho cooperativo	56
4.3.1 Conflitos e negociações no desenvolvimento de produtos de software	57
4.3.2 Novas abordagens de desenvolvimento de software	58
4.3.2.1 Projeto Participativo	60
4.3.2.2 JAD	62
4.3.2.3 SADT Aumentado	64
4.3.2.4 Teoria-W	69
4.3.2.5 Comparando os métodos	70
4.3.3 Apoio automatizado para metodologias cooperativas de desenvolvimento de software	76
4.4 Conclusões	78
Capítulo 5 Estudo de Casos	79
5.1 Objetivos	79
5.2 Metodologia adotada	80
5.3 Apresentação dos casos	81
5.3.1 Empresa petroquímica	81
5.3.2 Empresa atuante em estatística	87
5.3.3 Banco de desenvolvimento	93
5.3.4 Empresa que presta consultoria	97
5.4 Conclusões	99
Capítulo 6 Conclusões	104
Bibliografia	107

Lista de Figuras

- Figura 2.1 - Combinação de tempo x espaço do padrão de comunicações envolvido no trabalho cooperativo
- Figura 2.2 - Estrutura de relação entre os componentes e os processos de coordenação
- Figura 4.1 - Padrão de controle e comunicação numa equipe com programador chefe
- Figura 4.2 - Estrutura gerencial e estrutura de comunicação nas equipes democráticas
- Figura 4.3 - Estrutura gerencial e padrão de comunicação em equipes hierárquicas
- Figura 4.4 - Modelo Cascata do ciclo-de-vida baseado em Pressman
- Figura 4.5 - Modelo de Prototipagem Descartável
- Figura 4.6 - Modelo de Prototipagem Evolutiva
- Figura 4.7 - Modelo de Prototipagem Operacional baseado em Bresoff et al.
- Figura 4.8 - Modelo de Reutilização
- Figura 4.9 - Ciclo-de-Vida em Espiral
- Figura 4.10 - Comparação entre a abordagem tradicional de desenvolvimento de produtos de software e a abordagem cooperativa
- Figura 4.11 - Interpretações dos símbolos gráficos do SADT
- Figura 4.12 - Modelo das características do SADT
- Figura 5.1 - Organograma simplificado da empresa petroquímica
- Figura 5.2 - Relação entre as equipes envolvidas no desenvolvimento e uso da ferramenta
- Figura 5.3 - Organograma simplificado do Banco de Desenvolvimento

Lista de Tabelas

Tabela 4.1 - Comparação de algumas características entre os quatro métodos elaborada a partir de Carmel [CAWG 93], Boehn et al. [BORO 89] e Marca [MARC 91]

Tabela 5.1 - Empresas selecionadas e suas principais atividades

Tabela 5.2 - Tabela das características dos casos estudados

CAPÍTULO 1

Introdução

1.1 Motivação

Atualmente vivenciamos um considerável aumento das pesquisas sobre diversos aspectos da cooperação, seja ela aplicada à indústria, educação, áreas científicas e tecnológicas ou comércio. Uma das primeiras abordagens sobre cooperação foi explicitada em 1938 por Barnard, que colocou que cooperação é originada da necessidade de um indivíduo realizar tarefas para as quais, ele por variadas razões, sente necessidade de auxílio.

No contexto da cooperação, o apoio computacional é valiosa forma de intensificar as trocas de informações entre os membros de um grupo através de ferramentas de apoio a comunicação.

Por outro lado, verifica-se um aumento de interesse na aplicação de métodos de desenvolvimento de software como meio de alcançar um maior nível de qualidade dos produtos. Sob este aspecto, é interessante ressaltar que o processo de desenvolvimento de software vem evoluindo muito depressa. Os problemas aumentam de tamanho e complexidade, exigindo cada vez mais, um maior número de pessoas envolvidas no desenvolvimento dos projetos, sem que se observe um aumento de qualidade do software. O que se observa é que constantemente, os softwares são ineficientes, desenvolvidos artesanalmente, sem o uso de técnicas específicas, não satisfazendo as necessidades de seus usuários. Com isso, verifica-se um aumento das atividades de manutenção, muitas vezes dificultada pela falta de documentação adequada. Não houve capacidade do software acompanhar a rápida evolução do hardware, nem a difusão cada vez maior da informática nas diferentes áreas da sociedade.

A percepção de que a solução dos problemas associados ao desenvolvimento de software poderia se encontrar nos seus diferentes usuários, incentivou as pesquisas sobre métodos que pudessem apoiar a participação desses atores em todo o processo de projeto e desenvolvimento do produto. Tais métodos centram-se nos usuários, que fornecendo experiências e vivências, ajudam a modelar uma solução mais coerente com suas necessidades. Segundo Greenbaum et al. [GRKY 91] desenvolver software é difícil, não

pela complexidade dos problemas técnicos, mas por causa das interações sociais entre usuários e desenvolvedores.

Hoje, várias pesquisas estão em curso para levantamento de critérios e aspectos envolvidos na cooperação, integrando cientistas ligados à diferentes áreas do conhecimento, tal como: psicologia, sociologia, engenharia, informática, etc. Tais resultados apoiarão o desenvolvimento de ferramentas que incrementarão a comunicação, as decisões e o tráfego de informações intensificando e melhorando os processos cooperativos.

1.2 Objetivos da Tese

Apesar do tema "cooperação" ser antigo, sua aplicação em desenvolvimento de produtos de software é bastante recente. Poucas experiências têm sido relatadas sobre a utilização de métodos cooperativos no desenvolvimento de produtos de software.

Neste estudo estamos particularmente preocupados em caracterizar trabalho cooperativo enfocando os aspectos sociais envolvidos e identificar como o computador pode alterar as relações entre os participantes de uma equipe.

Devido ao grande florescimento de termos para designar os sistemas de apoio ao trabalho cooperativo que não guardam coerência entre si, buscamos na literatura várias definições e optamos por selecionar aquelas que mais congregavam a idéia de Groupware e de CSCW(Computer Supported Cooperative Work). Apresentamos algumas áreas em que a cooperação se mostra como uma estratégia bastante promissora para alcançar uma maior eficiência, dentre elas, a engenharia de software.

Em seguida, levantamos alguns métodos cooperativos de suporte ao desenvolvimento de produtos de software que apoiam e estimulam a participação do usuário. Finalmente, através do estudo de três casos pudemos constatar alguns aspectos considerados importantes e verificamos a existência de alguns elementos expostos nos capítulos 2, 3 e 4.

1.3 Divisão em Capítulos

Esta tese divide-se em seis capítulos, sendo o primeiro esta breve introdução.

No capítulo dois são apresentadas algumas definições básicas de cooperação, as vantagens do uso de computadores como ferramenta de apoio a cooperação e posteriormente, abordamos os aspectos sociais envolvidos num processo cooperativo.

No capítulo três são enfocados os benefícios do uso de ambientes computacionais apoiando a cooperação. Em seguida, são detalhados alguns sistemas de apoio ao trabalho cooperativo e algumas áreas em que a cooperação se mostra promissora.

No capítulo quatro descrevemos a evolução do desenvolvimento de software enfocando o desenvolvimento tradicional e expomos quatro métodos cooperativos de apoio ao desenvolvimento de software.

O capítulo cinco apresenta os resultados da pesquisa:

- ⇒ metodologia de trabalho empregada;
- ⇒ o perfil das empresas, a descrição dos casos;
- ⇒ a análise dos casos, confrontando-os ao que foi teoricamente elaborado nos dois primeiros capítulos;

Finalmente, no capítulo seis, são apresentadas as conclusões deste trabalho, no que se refere aos aspectos sociais e técnicos e são delineadas algumas áreas nas quais poderiam ser realizados estudos posteriores mais aprofundados. Logo após, encontra-se a lista bibliográfica.

CAPÍTULO 2

Trabalho Cooperativo

Neste capítulo a meta principal será a de definir cooperação, os elementos e aspectos sociais envolvidos, sua organização e como o poder da tecnologia computacional se aliou a esta corrente teórica impulsionando e facilitando procedimentos formais de trabalho em equipe.

2.1 Definições

2.1.1 Grupo

Dentre muitas definições de grupo, a que parece mais adequada aos propósitos deste trabalho é a de que: "são pessoas reunidas com um ou vários objetivos comuns" Minucucci [MINI 87]. É importante ressaltar a distinção entre os objetivos individuais e os objetivos do grupo pois, os objetivos de uma ação cooperativa não são simplesmente a soma de objetivos individuais. "A essência de um grupo transcende a experiência e as atividades de seus membros" Marca et al. [MABO 92].

Na formação dos grupos é preciso observar o fato de que as pessoas necessitam estar conscientes e dispostas a participar de uma equipe que busca alcançar um objetivo, criando um processo de interação entre os participantes que se influenciam reciprocamente. Minucucci [MINI 87] observa que dentro dos grupos deve prevalecer um ambiente de confiança mútua e de sinceridade entre os membros. As reações devem ser espontâneas e verdadeiras.

Austin et al. [AUST 90] assinalam que usualmente a razão de se estabelecer um grupo para desenvolver uma tarefa é trazer diferentes perspectivas e habilidades para a resolução de um problema. Logo, a participação de todos os membros do grupo é

desejável e mesmo indispensável para que os objetivos da ação cooperativa sejam alcançados.

2.1.2 Cooperação

Segundo Bannon et al. [BASC 91], o termo "Trabalho Cooperativo" possui uma longa história nas ciências sociais. Ele foi primeiramente empregado no século XIX por economistas como uma designação geral e neutra do trabalho envolvendo múltiplos atores. Eles colocam que em 1867, Marx definiu-o formalmente como sendo o trabalho planejado de indivíduos envolvidos num mesmo processo de produção ou em processos diferentes, mas interligados. Neste século o termo tem sido utilizado extensivamente com o mesmo sentido proposto inicialmente.

Entretanto, existem questionamentos quanto às nuances dos termos: trabalho coletivo, trabalho colaborativo, trabalho em grupo, trabalho em equipe, trabalho participativo, trabalho cooperativo. Eles possuem diferentes conotações, entretanto todos enfatizam o conceito de fusão dos membros de um conjunto. O termo "trabalho colaborativo" possui um ênfase especial na colaboração irrestrita, Bannon et al. [BASC 91] citam o exemplo "colaboração com o inimigo" para mostrar que a colaboração pode ser desenvolvida em situações onde nem sempre há o bom relacionamento das partes envolvidas. Mas, em geral podemos considerar todos esses termos como sinônimos de "Trabalho cooperativo".

Barnard [BARN 68] coloca que o conceito inicial de cooperação repousa na definição de organização como um "sistema que conscientemente coordena atividades ou forças de duas ou mais pessoas".

O trabalho cooperativo deve ser eficaz para atingir seus objetivos a partir da organização coletiva, e ao mesmo tempo, ser eficiente no sentido de satisfazer as idéias individuais. Logo, são requisitos indispensáveis a boa-vontade de colaborar e passar informações, a capacidade de comunicação, a capacidade de negociar e acatar propostas, já que a complexidade das relações crescerá proporcionalmente ao crescimento do número de participantes do grupo. Deve haver ainda uma estrutura democrática para que as pessoas possam cooperar sem se sentirem individualmente ameaçadas no seu poder. Com essas condições observadas, segundo Reder et al. [RESC 90], o trabalho cooperativo é desenvolvido através de uma série de interações entre os participantes do grupo. Os períodos de trabalho solitário que os participantes devotam para realizar suas partes da tarefa cooperativa são entremeados com reuniões de trocas de informações entre os participantes. O papel da comunicação é fundamental.

A comunicação poderá se realizar através de encontros face-a-face ou então ser efetuada por meios eletrônicos. Neste último caso, a comunicação é apoiada em ferramentas computacionais, promovendo a interação de pessoas afastadas fisicamente. Hoje, percebemos um aumento do interesse em apoiar as reuniões em equipamentos eletrônicos de comunicação, potencializando deste modo a troca de informações.

Os encontros face-a-face permitem a interação de pessoas reunidas num mesmo ambiente físico, com a troca de informações se dando através da comunicação oral e escrita. Esse tipo de reunião é bem conhecido, como também toda a problemática envolvida tal como: os conflitos são difíceis de serem negociados, pois todos falam ao mesmo tempo, freqüentemente os relatórios são confusos e de difícil manuseio, as discussões alongam-se por muito tempo, gerando grandes dificuldades gerenciais.

Em vista disso, um aspecto importante que deve ser considerado é o da coordenação. É interessante que os trabalhos sejam direcionados e estimulados, pois se as discussões sobre algum ponto específico se alongarem demasiadamente, o grupo pode perder o interesse inicial, fazendo com que o trabalho perca sua dinâmica.

A problemática que envolve as reuniões que usam o computador como meio de comunicação será discutida mais detalhadamente nos próximos capítulos.

Todos esses aspectos sociais envolvidos na cooperação são de extrema importância, pois é através do estudo das relações entre as pessoas e organizações envolvidas que o trabalho cooperativo poderá ser melhor estruturado e definido, gerando uma sólida base teórica que apoiará o aporte de aplicações computacionais.

2.2 O Computador Alterando as Relações de Trabalho Cooperativo

Um grande número de projetos vitais e importantes desenvolvidos atualmente são de tal envergadura que não são possíveis de serem abordados por uma só pessoa, havendo situações que são inerentemente grupais, tais como educação, administração, indústria, engenharia, comércio, etc. As equipes necessárias para desenvolver os projetos freqüentemente necessitam de uma formação multidisciplinar, visando a abordagem de vários aspectos contidos nesses trabalhos e viabilizando a concepção e o desenvolvimento dos mesmos.

Uma possibilidade para melhorar o trabalho dessas equipes, aumentando seu potencial de comunicação e interação, é o uso de ambientes cooperativos baseados em computador. Esses ambientes podem conter ferramentas para análise, projeto, documentação, avaliação, tomada de decisão, ensino-aprendizagem, coordenação e gerência.

Um sistema computacional apoiando um ambiente cooperativo visa assistir o trabalho de grupos facilitando as tarefas de comunicação e coordenação, propiciando apoio para a interação entre os membros do grupo em variadas instâncias. Os componentes do grupo não precisam estar num mesmo ambiente geográfico, nem trabalhando necessariamente em tempo real. Em Ellis et al. [ELGR 91] encontramos uma tabela que apresenta as diferentes formas de combinação de tempo e espaço com suas respectivas formas de interação em trabalho cooperativo:

	Mesmo Tempo	Tempos Diferentes
Mesmo Lugar	Interação face-a-face	Interação Assíncrona
Lugares Diferentes	Interação Distribuída Sincronamente	Interação Distribuída Assincronamente

Figura 2.1: Combinação do tempo x espaço do padrão de comunicações envolvido no trabalho cooperativo [ELGR 91] (pp. 41)

As interações face-a-face se dão, em sua grande maioria, sem apoio de ferramentas. Já as reuniões no mesmo lugar e em tempos diferentes, precisam de alguma ferramenta que gerencie e armazene as trocas de informações entre os membros do grupo. Do mesmo modo que as reuniões em lugares diferentes e tempos diferentes.

As reuniões em lugares diferentes e mesmo tempo precisam do apoio de ferramentas que estabeleçam essas ligações em tempo real, fornecendo meios rápidos de coordenação e de resolução de conflitos.

Finalmente, em vista da transformação constante que experimenta a introdução da dinâmica do trabalho cooperativo, é preciso considerar alguns aspectos que devem nortear a introdução/adoção de novas tecnologias no ambiente de trabalho. Sob este ângulo, segundo Greenbaum et al. [GRKY 91], é importante que:

- ⇒ o computador favoreça o aperfeiçoamento de habilidades dos seus usuários;
- ⇒ os usuários participem de todo o processo de implantação e desenvolvimento dos projetos computacionais com os quais passarão a trabalhar;
- ⇒ a introdução de sistemas computacionais apoie o aumento da qualidade dos produtos e não somente a produtividade.

2.3 Organização e Elementos Sociais Envolvidos na Cooperação

No trabalho cooperativo, segundo classificação proposta por Barnard [Barn 68], existem aspectos sociais envolvidos tais como:

- ↳ a interação do indivíduo com o grupo - essa interação é consequência da cooperação. Quando o indivíduo aceita participar de um grupo, aceita seu contato com os outros participantes.
- ↳ a interação do grupo com o indivíduo - o grupo é visto como uma unidade interagindo com cada indivíduo. A comunicação será a base das interações, sendo o ponto fundamental para a continuidade dos trabalhos em equipe.

Segundo Jessup et al. [JEVA 93], a história sugere que para podermos compreender alguma nova tecnologia, devemos perceber claramente como as atividades eram desenvolvidas sem o apoio tecnológico. Logo, é importante abordar os aspectos sociais envolvidos na cooperação para que possamos perceber o valor do apoio tecnológico intermediando as interações de membros e equipes. A seguir detalharemos alguns desses aspectos.

2.3.1 Comunicação

A mais universal forma de cooperação humana e, talvez a mais complexa, é a fala. É um evento físico, enquanto requer de energia física convertida em ondas sonoras. É um ato biológico, necessitando de boca, dentes, língua, nariz, laringe, etc. É fenômeno social, na medida em que só pode ser aprendida através de contato social. A fala como atividade cooperativa cessa de existir se um desses fatores falha.

O processo de comunicação envolve a emissão e a interpretação de uma mensagem cifrada. Barreiras que impedem a interação podem surgir por vários motivos, tais como: falta de compreensão do código de mensagens, descuido no uso das palavras, falta de atenção, desconfianças, medo de passar informações, etc. Segundo Mathias et al. [MAFU 92], a comunicação perfeita ocorre quando um pensamento ou idéia é transmitido de forma que a figura mental percebida pelo receptor é exatamente igual à idéia que foi enviada pelo emissor.

A troca de informações pode criar opiniões antagônicas naturais geradas por diferenças culturais, sociais e psicológicas inerentes a cada indivíduo. A persistência dessas posições antagônicas gera conflitos que precisam ser analisados e negociados.

2.3.2 Conflitos

O conflito surge de várias fontes, podendo ser a expressão de uma disputa de poder, recursos ou posição social, divergência de idéias ou de estrutura de valores ou ainda, por problemas de comunicação. O conflito compreende relações psicológicas antagônicas e interação antagônica. "O conflito é um fenômeno social, podendo ser interpessoal, intergrupo ou intragrupo", Mathias et al. [MAFU 92].

Os conflitos podem ser causados por fatores de comunicação, estruturais ou de comportamento social. Os conflitos gerados por falhas na comunicação surgem a partir de mal-entendidos, dificuldades semânticas ou falta de informação. Os conflitos estruturais se referem à incompatibilidade entre as regras da empresa e as ações gerenciais visando a coordenação das atividades. Já os fatores pessoais desencadeados de conflitos são mais complexos, pois envolvem a personalidade de cada participante. Pessoas com características agressivas e dominadoras podem intensificar um conflito inicial, pois tentam tomar a frente de todos os problemas impondo seus pontos de vista. Entretanto, os conflitos não devem ser encarados unicamente como desencadeadores de desavenças e disputas. Eles podem ser o primeiro passo para a discussão de idéias ainda não amadurecidas, abrindo novas possibilidades, novos caminhos.

De acordo com Mathias et al. [MAFU 92], os conflitos podem ser analisados sob duas óticas: funcionais e não-funcionais. O conflito funcional auxilia os objetivos da organização, estimulando as discussões, aumentando o interesse, impedindo a estagnação. Já o conflito não-funcional é aquele que não se encaixa nesta definição e, portanto, não traz nenhuma vantagem para o crescimento das discussões. Os autores salientam a importância da análise formal e consciente das causas do conflito como forma valiosa de melhor entendê-lo, facilitando a escolha do método de negociação mais adequado à situação.

2.3.3 Negociações

"Negociação é uma forma cooperativa de resolver um conflito através da análise de uma gama de possibilidades" Mathias et al. [MAFU 92]. Os participantes tentam encontrar uma solução para os conflitos que satisfaça a maioria e que harmonize as relações dentro do grupo.

As diferenças de poder entre os participantes podem interferir no processo de negociação, não sendo produtivo que haja imposições de idéias ou ações. Logo, os participantes devem buscar o entendimento dessas diferenças e tentar alcançar uma solução que não agrida a união do grupo.

Existem alguns requisitos básicos que devem ser observados por aqueles que estão envolvidos numa negociação:

- O negociador que está com a palavra deve ser objetivo, sem usar artificios que confundam os outros participantes;
- É interessante ouvir sempre, com atenção, todas as argumentações e propostas;
- Manter a calma e não responder a possíveis agressões;
- Estar preparado para fazer concessões;
- Ser espontâneo e aberto a possíveis mudanças.

Todos esses aspectos contribuem para que a coesão do grupo se mantenha sempre homogênea.

Outro ponto que não deve ser esquecido é que as negociações necessitam alcançar um objetivo consensual. Logo, é importante atentar para o fato de que muitas vezes, durante longas negociações, são introduzidos objetivos intermediários que fazem com que os objetivos finais sejam pulverizados.

Carvalho et al. [CAFU 92] classificam a negociação em duas correntes: negociação competitiva e negociação cooperativa. Na negociação competitiva temos a busca do poder e de vantagens pessoais, portanto, não deveria, de fato ser enquadrada como uma legítima negociação. No enfoque do nosso trabalho a negociação cooperativa é a mais adequada, pois o objetivo é alcançar um consenso através da argumentação e aceitação de novas propostas.

Em Mathias et al. [MAFU 92] encontramos detalhados alguns métodos de resolução de conflitos, dentre os quais ressaltamos aqueles que poderiam ser empregados positivamente dentro de grupos de trabalho cooperativo:

- Método de um "terceiro participante": quando a negociação não é bem sucedida pode-se apelar para a intervenção de um participante que esteja fora do núcleo do conflito, o qual poderá intermediar as negociações ou atuar como juiz.
- Estabelecimento de um objetivo superior: consiste na determinação de um objetivo diferente daquele que gerou o conflito e que só pode ser alcançado através do trabalho cooperativo entre as partes que estão em conflito.
- ♦ Votação: é um procedimento democrático, embora não agrade à minoria derrotada, podendo deixar resquícios de conflito.
- Alteração de variáveis estruturais: um participante, geralmente um líder ou coordenador, altera a estrutura organizacional da equipe.

Todos esses métodos podem deixar sombras de conflitos, pois introduzem um elemento exterior ao núcleo do conflito inicial. O mais positivo seria que as partes envolvidas na questão conseguissem resolver seus problemas através da simples negociação, sem envolver novas pessoas, nem novos objetivos.

Carvalho et al. [CAFU 92] ressaltam que o negociador não deve perder a capacidade racional, se tornando parcial pelo envolvimento emocional a ponto de considerar os argumentos contrários infundados, pois atitudes assim são irrealistas e ineficazes. A solução deve vir de um acordo consensual, de forma que a decisão passe a ser, a partir daquele momento, de autoria de todos. Esse consenso é fundamental para que as decisões sejam colocadas em prática.

2.3.4 Liderança

Através do desenvolvimento dos projetos cooperativamente deverão surgir pessoas que possuam características de liderança. Essas lideranças podem ser decididas pelo grupo através de escolha, podem emergir naturalmente e serem aceitas conscientemente pelo grupo ou, serem fortalecidas através da evolução dos trabalhos.

O líder não pode ser visto como alguém hierarquicamente superior, ele deve ter sua importância fortalecida através da demonstração de sua capacidade de intervir

positivamente na finalização de alguma tarefa cooperativa, favorecendo a resolução de conflitos, intermediando negociações, aumentando a integração do grupo.

Por outro lado, as lideranças podem ser designadas pelas organizações, ou seja, um líder interno ou externo ao grupo é escolhido e imposto ao grupo pela estrutura da empresa. Esse tipo de imposição pode ser fonte geradora de conflitos, uma vez que tal líder poderá ser encarado como um elemento estranho, artificialmente integrado à equipe.

Em Carvalho et al. [CAFU 92] encontramos uma classificação para o exercício da liderança:

- democrática, onde o líder leva em consideração a opinião do grupo;
- autocrática, onde o líder decide tudo;
- "laissez-faire", onde o líder se omite, não decidindo nem orientando nada dentro do grupo. Seria, de fato, a não-liderança.

Minicucci [MINI 87] observa algumas atitudes que o líder deve ter:

a) Antes da reunião:

- levar os elementos do grupo a se conscientizarem dos objetivos do trabalho grupal;
- planejar o desenvolvimento da sessão ou das atividades da equipe.

b) Durante a reunião:

- incentivar a comunicação entre os integrantes;
- procurar manter a discussão dentro do tema e levar os elementos do grupo a caminharem em direção ao objetivo estabelecido.

2.3.5 Coordenação

Existe uma grande dificuldade de se definir, precisadamente, o que é coordenação. Malone et al. [MACR 90], em primeira instância, propõem uma definição encontrada em um dicionário, o "American Heritage Dictionary", onde "coordenação é o ato de trabalhar em conjunto harmoniosamente". No entanto, eles observam que tal definição seria muito abrangente e sugerem um estreitamento de limites propondo a definição do conceito de coordenação fundamentando em algumas questões: Como vários objetivos podem ser subdivididos em ações? Como as ações agrupadas podem produzir o objetivo? Como os recursos devem ser divididos entre os diferentes membros? Como as informações devem ser partilhadas?

Wexelblat [WEXE 92] coloca que coordenação é a habilidade de arranjar os encontros e outras interações entre os membros e também, de comandar atividades e alocar recursos de tal forma que as ações não gerem conflitos. A coordenação dos trabalhos se faz através da adoção de procedimentos administrativos e mediante reuniões periódicas entre os membros, produzindo relatórios onde ficam registradas as tomadas de decisão, as atividades e novas propostas geradas durante as reuniões. Ao final de algum tempo haverá uma grande quantidade de informação armazenada em papéis, dificultando seu manuseio, classificação, revisão e acesso. Os membros precisam, então, dispor de mecanismos de coordenação que harmonizem as atividades de modo que os objetivos possam ser alcançados mais rapidamente e com mais eficiência. Várias áreas do conhecimento contribuem na coordenação dos trabalhos: psicologia, administração, linguística, sociologia, direito, etc. Essa interdisciplinaridade contribui para que a definição de coordenação seja muito abrangente, dificultando sua compreensão.

Malone et al. [MACR 90] propõem uma estrutura onde relacionam componentes e processos de coordenação, conforme ilustrado na figura abaixo:

Componentes da Coordenação	Processo de Coordenação Associado
Objetivo	Identificar objetivos
Atividades	Relacionar os objetivos às atividades
Definir membros	Selecionar e distribuir as atividades entre os membros
Interdependências	Gerenciar as interdependências e sincronizar as atividades

Figura 2.2: Estrutura de relação entre os componentes e os processos de coordenação [MACR 90] (pp. 360)

As interdependências entre atividades são pontos comuns inseridos em duas ou mais ações diferentes. Através da identificação dos fatores de coordenação envolvidos na cooperação, a estrutura apresentada na figura 2.2 pode contribuir para a construção de ferramentas computacionais que coordenem e apoiem o trabalho cooperativo.

2.4 Conclusões

Neste capítulo procuramos conceituar cooperação e os fatores envolvidos em sua definição, através da bibliografia existente.

A organização dos grupos, a importância da comunicação, as combinações de "tempo versus espaço" em que as comunicações se estabelecem, os conflitos e suas formas de negociação, o surgimento de lideranças e a importância da coordenação são elementos fundamentais para viabilizar o trabalho cooperativo, como, também, para fornecer subsídios para embasar a construção de ferramentas automatizadas de apoio ao trabalho cooperativo.

Todos esses fatores influenciarão as futuras discussões sobre o desenvolvimento de produtos de software realizado por equipes que se apoiam na teoria cooperativa, assim como, o uso e desenvolvimento de ferramentas computacionais de apoio à atividades cooperativas, como veremos a seguir.

CAPÍTULO 3

O Apoio Computacional

Segundo Greenberg [GREE 91], inicialmente os computadores não foram adotados para assistirem pessoas trabalhando em conjunto. Estudos científicos sobre interação homem-máquina tenderam a focar o uso individual de computadores, assim como a maioria dos softwares encontrados hoje no mercado apoiam a interação individualizada usuário-sistema. Com a evolução dos estudos realizados sobre o trabalho em equipe, observou-se que a cooperação contribui para o aumento da qualidade do trabalho, da qualidade dos produtos gerados e da satisfação pessoal daqueles indivíduos envolvidos direta e indiretamente no desenvolvimento dos projetos.

Mais recentemente, com o aumento do interesse em sistemas multi-usuários, através dos chamados "Computer-Suported Cooperative Work" (CSCW) e "Groupware", começou-se a constatar a adequação dos computadores para apoiar positivamente trabalhos de grupo.

Neste capítulo abordaremos os benefícios do uso de ambientes computacionais apoiando a cooperação, as diversas definições de CSCW e Groupware, os vários tipos de sistemas de apoio ao trabalho cooperativo e discutiremos as vantagens e desvantagens da utilização desses sistemas. Enfocaremos também algumas das áreas em que a teoria cooperativa pode ser aplicada, tais como: educação, desenvolvimento de software e automação de escritórios.

3.1 Benefícios do uso de ambientes computacionais apoiando a cooperação

Como vimos no capítulo anterior a cooperação entre diferentes membros de um grupo exige interações constantes para troca de informações. Esses contatos podem se estabelecer mais rápida e eficientemente através de computadores, apoiados em ferramentas

computacionais o que de acordo com os vários autores consultados, entre eles, Grudin [GRUD 91], [GRUD 94] e Greenberg [GREE 91], traria contribuições positivas, tais como:

- o uso de ferramentas computacionais e metodologias formais ajuda a sistematizar as interações grupais, diminuindo a complexidade organizacional do grupo de trabalho;

- os espaços de atuação são partilhados entre os membros, estreitando laços de confiança e gerando uma integração mais sólida. Ressalta-se, ainda, a possibilidade de interações remotas entre membros de um grupo. De certo modo a fluidez de tais intercâmbios tornam a relação intra-grupo mais democrática;

- ♦ a formalidade propiciada por um ambiente computacional facilita a existência de uma linguagem comum, possibilitando desta maneira uma comunicação mais eficiente entre os membros do grupo;

- o armazenamento das informações não estruturadas trocadas durante o processo de comunicação entre os membros possibilita acessá-las e manipulá-las facilmente, de modo a organizá-las, com vista a seu posterior aproveitamento;

- ♦ a possibilidade de registrar o desenvolvimento dos trabalhos facilitando posterior consulta, o que poderá apoiar tomadas de decisão.

Grudin [GRUD 88] salienta que o apoio computacional para as atividades dos indivíduos dentro de grupos e no contexto das organizações irá, mudar de forma significativa a maneira das pessoas viverem. Mas o computador sozinho não seria o único responsável por essas mudanças. Apoiar o trabalho cooperativo em metodologias e utilizar ferramentas adequadas são requisitos importantes para a obtenção do sucesso desejado.

Essas ferramentas têm recebido diversas classificações e definições tornando praticamente impossível, na atualidade, a adoção de uma terminologia universal de consenso que possa ser compreendido e utilizado pelos vários pesquisadores da área. Na próxima seção nos fixaremos na diversidade de definições dos sistemas de apoio ao trabalho cooperativo e seus vários enfoques e abordagens.

3.2 Discutindo os Sistemas de Apoio ao Trabalho Cooperativo

As ferramentas e ambientes de apoio ao trabalho cooperativo vêm sendo desenvolvidas continuamente e percebe-se que as pesquisas tendem a enfatizar as questões tecnológicas envolvidas nesses projetos. Entretanto, é importante frisar que os aspectos

humanos envolvidos nas tarefas cooperativas são tão importantes quanto as tecnologias empregadas. Em Wilson [WILS 91] encontramos uma lista de aspectos humanos/sociais e tecnológicos considerados mais influentes:

Aspectos Humanos/Sociais:

- *Características individuais humanas*, tais como padrões de conversação e compromissos;
- ♦ *Aspectos organizacionais*, tais como a estrutura e cultura das organizações;
- *Questões de projeto de trabalho em grupo*, ou seja, o envolvimento do usuário no processo de projeto do trabalho, prototipagem e testes de usabilidade;
- *Aspectos de dinâmica do grupo*, como processos de tomada de decisão e processos cooperativos entre os participantes.

As preocupações tecnológicas de apoio ao trabalho cooperativo incluem:

- *Mecanismos de comunicação*, que permitem aos participantes situados em diferentes lugares estabelecerem contato através de som, imagem e textos;
- *Facilidades de partilhamento do espaço de trabalho*, que permitem as pessoas trabalharem em um mesmo espaço eletrônico ao mesmo tempo;
- ♦ *Facilidades de apoio às atividades do grupo*, que intensifiquem o processo de trabalho, facilitando a co-autoria e a geração de idéias.

Com o aumento de pesquisas e experimento na área de trabalho cooperativo vemos crescer polêmica em torno de suas vantagens, gerando uma serie de definições e abordagens, não existindo ainda um consenso entre os pesquisadores. Uma das indefinições que mais chama a atenção é a questão conceitual entre "**Groupware**" e "**Computer Supported Cooperative Work**" (CSCW).

Wilson [WILS 91] afirma que Groupware é o software de apoio ao trabalho cooperativo e CSCW trata das teorias envolvidas e influências de outras áreas, tais como interação homem-máquina, aspectos psicológicos, sociológicos e organizacionais. Já Nakauchi et al. [NAAN 92] colocam que CSCW são sistemas que dão suporte ao trabalho cooperativo de um grupo ou de uma organização.

Segundo Ellis et al. [ELGR 91] e Ishii [ISHI 92], Groupware é o software que apoia e mantém o trabalho de grupo. Enquanto que Jonhson-Lenz et al. [JOJO 91] definem Groupware como o software juntamente com o sistema social envolvido. Já

Rodden [RODD 92] e "The Mocca Group" [THEM 92] colocam que o software mais o sistema social é a definição de CSCW.

Mais recentemente, os autores dos artigos publicados no livro editado por Jessup e Valacich [JEVA 93] se referem aos sistemas de apoio ao trabalho cooperativo como "Group Support Systems" (GSS), ou seja, "Sistemas de Apoio a Grupos". Mas percebe-se que GSS engloba as discussões sobre os aspectos sociais envolvidos. Logo, para esses autores GSS engloba o software e o sistema social.

Grudin [GRUD 91] interpreta tais diferenças como um indicador de enfoques entre "tecnologistas" e "cientistas sociais e comportamentais". Os desenvolvedores de pequenos projetos ultrapassam os obstáculos técnicos sem perceberem as influências sociais e organizacionais que governam o sucesso dos Groupwares. Em contraste, com os desenvolvedores de grandes projetos que resolvem suas questões pesquisando os fatores não-técnicos que afetam o uso do sistema. O desenvolvimento de projetos que apoiem a cooperação e que considerem os impactos da adoção das novas tecnologias sobre os grupos e sobre as organizações requer uma expansão do nosso conhecimento. Por isso, um elemento fundamental de CSCW é o estudo do comportamento do indivíduo e do grupo nas organizações e nos ambientes de trabalho.

Até a atualidade, os sistemas de apoio ao trabalho cooperativo ainda não foram suficientemente testados e avaliados para que os resultados possam ser generalizados. Nunamaker et al. [NUDV 93] enfatizam que é perigoso generalizar conclusões obtidas através do estudo com grupos que não utilizam apoio computacional, extendendo-os para grupos que valem-se deste apoio tecnológico e vice-versa. Por outro lado, as tecnologias envolvidas não cessam de serem aperfeiçoadas e dificilmente haverá um modelo geral que possa servir em todas as situações.

No presente trabalho consideramos **Groupware** como o software que apoia e mantém o trabalho de grupo e **CSCW** como a área científica que estuda as relações sociais e os Groupwares.

3.2.1 Groupware

Segundo Ellis et al. [ELGR 91] uma das primeiras referências ao termo Groupware foi elaborada por Johnson-Lenz et al. [JOJO 82] que enfocava o sistema baseado no computador, combinado ao processo social das relações dentro do grupo. A finalidade do Groupware é apoiar a comunicação e a colaboração dos grupos, coordenando suas atividades através de ambientes partilhados. Os autores enfatizam que essa definição

não especifica que os usuários devam trabalhar em tempo real. O sucesso seria alcançado através do equilíbrio entre os processos sociais e a estrutura tecnológica, sendo que as noções de "objetivo comum", "ambientes partilhados", "cooperação" e todos os aspectos envolvidos são de vital importância na construção e implementação de Groupwares.

As metodologias para projetar groupware ainda foram pouco exploradas, e várias abordagens estruturais são possíveis. Johnson-Lenz et al. [JOJO 91] apresentam algumas das abordagens existentes:

Groupware mecanicista - É baseado numa teoria social que considera que a interação humana é passível de ser modelada como numa máquina. O groupware deve prover mecanismos bem definidos e determinísticos de interação. O grupo é forçado a seguir um caminho pré-definido. Como exemplo temos os sistemas para autoria compartilhada e os sistemas de gerência de projeto. As inflexibilidades de tal estrutura tendem a reduzir a capacidade criativa do grupo e gerar resistências.

Groupware Contextual - É oposto ao mecanicista. Baseia-se na teoria social que considera que os sistemas humanos se auto-organizam naturalmente, através de interações não-determinísticas. Este groupware promove um espaço aberto para as pessoas se encontrarem e se auto-organizarem, onde a coerência social emergirá das interações entre os indivíduos, contrapondo-se ao modelo mecanicista, que tenta produzir a coerência. Propicia ao usuário mais liberdade de escolha e autonomia individual. Os autores ressaltam que essa abordagem de projeto é não-mecanicista apenas a nível de interação e coerência social, pois a nível de organização de conhecimentos e de interface com o usuário, a abordagem continua sendo mecanicista.

Groupware Centrado no Objetivo - Representa uma ponte entre as duas teorias anteriores. A interação humana é vista como um processo dinâmico e criativo em constante mutação. Em lugar de medir a eficácia de um sistema em termos puramente mecanicistas, como pela produtividade e eficiência, esta abordagem usa medidas como a capacidade de ser criativo e dinâmico. Seria o mais vantajoso dos três, pois acompanha a evolução das pessoas e do grupo, além de permitir que os membros do grupo adicionem novos modelos rapidamente.

Castro et al. [CADR 91] salientam que os primeiros groupwares apoiavam a comunicação dos participantes via computador, através da interação assíncrona, permitindo que cada usuário trabalhasse em horários de sua escolha. A necessidade de ferramentas que possibilitassem o trabalho cooperativo simultâneo estimulou o surgimento de produtos que suportassem a interação entre os participantes em tempo-real.

Existem vários tipos de Groupware na área de suporte ao trabalho cooperativo. Apresentaremos a seguir, uma taxonomia baseada em domínios de aplicações:

☰ **Sistemas de mensagem**

É o exemplo mais comum de groupware. Segundo Ellis et. al. [ELGR 91] ele apoia trocas de mensagens textuais assíncronas entre grupos de usuários. Alguns exemplos deste tipo são os chamados Correios Eletrônicos e BBS (bulletin board systems).

Os sistemas de correio eletrônico não exigem que o receptor esteja presente quando a mensagem chega. As mensagens são armazenadas e podem ser consultadas posteriormente ao seu envio. "Os sistemas de correio eletrônico provocaram uma mudança nas atitudes e comportamento das pessoas no ambiente organizacional" Silva et al. [SILF 92]. O correio eletrônico tem servido como instrumento para que as distâncias geradas pela hierarquia sejam estreitadas. Um funcionário subalterno pode manter contatos com alguém hierarquicamente superior sem que sejam observadas as formalidades tradicionais, favorecendo o intercâmbio de informações. Por outro lado, permite que mensagens circulem em níveis hierárquicos, muitas vezes, incompatíveis com sua importância. Silva et al. [SILF 92] assinalam que neste tipo de sistema, seria vantajoso estabelecer algum tipo de controle no fluxo de mensagens entre níveis hierárquicos diferentes. Se bem que tal sugestão seja conflitante com a democratização da comunicação, tida como uma das principais vantagens deste sistema.

É interessante ressaltar as idéias de Allen [ALLE 90] sobre o correio eletrônico não poder ser enquadrado como Groupware se serve apenas para trocas de mensagens. Segundo Grudin [GRU2 94], os correios eletrônicos devem apoiar o trabalho de um grupo, distribuindo informações entre seus diferentes membros.

Os BBS são sistemas de mensagem que funcionam através de listas de tópicos de interesses, ou seja, a pessoa se inscreve em uma ou mais listas específicas. Desse modo, ela receberá todas as mensagens relativas ao tema de seu interesse e não receberá as mensagens que circularão em outras listas, evitando o recebimento de mensagens inúteis.

☐ Editores Multiusuários

Os componentes do grupo podem compor e editar documentos cooperativamente. Existem editores que funcionam assincronamente e outros em tempo real. Este tipo é objeto de uma série de pesquisas em andamento em algumas universidades brasileiras, conforme exemplificado a seguir:

- De acordo com Castro et al. [CADR 91] o Sistema de Suporte a trabalho cooperativo (SISTRAC) que faz parte do projeto A_HAND, em andamento na UNICAMP-SP, visa motivar e permitir a utilização do computador em trabalhos que necessitam de cooperação entre membros de uma equipe. O SISTRAC possibilita o uso em trabalho cooperativo, dos mesmos programas aos quais os usuários estão habituados no seu trabalho individual. Editores de texto ou qualquer programa interativo não gráfico pode ser compartilhado. A transição do trabalho individual para trabalho de grupo é facilitada pelo compartilhamento de programas monousuários, enquanto atividades de grupo podem ser apoiadas por aplicações projetadas para tal fim.

- Outro projeto em desenvolvimento é o MULEC, que se propõe a ser um editor de textos cooperativo e concorrente. Este projeto está sendo desenvolvido na COPPE Sistemas/UFRJ por Tornaghi et al. [TOSO 92]. O sistema visa apoiar atividades educacionais, possibilitando que alunos editem textos cooperativamente, assim como desenvolvam brincadeiras que ajudarão na construção do conhecimento, incentivando a criatividade e a troca de informações.

- ♦ Em Camargo [CAMA 92] encontramos a descrição do FLECHA que é uma das ferramentas cooperativas da Estação de Desenvolvimento de Software TABA da COPPE Sistemas/UFRJ. FLECHA é um editor para representação gráfica do Modelo de Objetos do Sistema de Gerência de Objetos da Estação TABA. Ele permite a utilização de anotações, a troca de mensagens, facilita a interação entre colaboradores e o compartilhamento de informações entre eles. Este editor ainda fornece suporte ao trabalho simultâneo, permitindo que diferentes pessoas trabalhem em diferentes partes do diagrama no mesmo instante, utilizando versões mais atuais de cada parte, e tomando conhecimento instantâneo das alterações realizadas pelos outros usuários.

☐ Sistemas de suporte a grupos de decisão (SSGD)

Os sistemas de suporte a decisões têm por objetivo apoiar grupos de trabalho na discussão exploratória de assuntos não estruturados, agilizando o processo decisório e dando condições de melhoria da qualidade final. Silva et al. [SILF 92].

Podemos ter sistemas que apoiem a estruturação de decisões - ferramentas para votação e elaboração de ranking - e ferramentas para a análise de tópicos e idéias geradas em grupo. Esses sistemas oferecem facilidades onde decisões precisam ser tomadas rapidamente com o apoio de todo o grupo, alcançando um aumento da produtividade. Alguns SSGD são implementados como sala de reunião eletrônica contendo várias estações de trabalho ligadas em rede e equipamentos de audio/vídeo.

☰ Sistemas de Suporte a Reuniões

Os participantes podem estar na mesma sala ou em lugares distintos. O computador é o meio básico de comunicação, estando ou não as pessoas separadas fisicamente. Existem vários modelos de implementação:

♦ Reunião em tempo real:

Os usuários podem ficar juntos em uma sala de reunião eletrônica ou fisicamente afastados, interagindo sincronamente através dos terminais ou estações de trabalho. No caso de reuniões face-a-face, as salas podem possuir um telão onde as informações gerais são mostradas a todos os participantes, ou então um quadro-branco onde ficam registradas as diferentes opiniões e decisões. A multimídia pode ser explorada de forma a aproveitar todas as vantagens da introdução de som e imagem, intensificando o nível de participação.

♦ Teleconferência:

São reuniões apoiadas em sistemas de comunicação eletrônica que podem ou não serem estabelecidas em tempo real. Yamamoto [YAMA 91] classifica as teleconferências em:

- ⇒ Pública - qualquer pessoa pode ingressar nela durante a sua realização, mesmo sem ter sido convidada;
- ⇒ Acesso Restrito - só um gerente pode autorizar o ingresso de um membro, é o tipo mais usado;
- ⇒ Privada - somente os participantes convidados tomam conhecimento de sua existência.

Segundo Batista et al. [BASF 92] numa teleconferência existem n participantes, entretanto o diálogo no espaço compartilhado ocorre sempre entre pares. Geralmente um gerente coordena as reuniões e o diálogo se desenrola entre ele e o participante que detém o controle do espaço compartilhado. O gerente é o responsável pelo controle dos canais de conversação e qualquer outro participante que deseje dialogar deve informar o gerente. Os participantes possuem liberdade para conversar paralelamente a qualquer hora através do correio-eletrônico conjugado que a teleconferência oferece. Os equipamentos envolvidos, incluem, além do computador, vídeo, projetores e equipamento de audio.

♦ Conferência "desktop":

Pretende combinar as vantagens da teleconferência e as da conferência em tempo real. Este método usa as estações de trabalho como interface e também processa aplicações partilhadas pelos participantes, permitindo múltiplas janelas por estação de trabalho. A multimídia pode ser explorada neste tipo de conferência.

☰ Sistemas de Calendário

A proposta principal dos Sistemas de Calendário é otimizar o gerenciamento do tempo, possibilitando o partilhamento de agendas, facilitando a marcação de encontros e reuniões, o gerenciamento de recursos como sala de reunião ou de aula. Cada membro do grupo pode acessar as agendas dos outros participantes. Segundo Grudin [GRUD 94] esse sistema permanece em um nível baixo de adoção, devido a falta de um sistema eficiente de marcação automática de reuniões e a necessidade da intervenção de uma secretária.

Datanews [DATA 92] apresenta algumas sugestões para o sucesso na implementação de groupwares:

- ⇒ definir um objetivo específico com metas concretas a serem atingidas após a implementação do sistema;
- ⇒ selecionar cuidadosamente a equipe de trabalho, não se fixando em grupos que já desenvolvam suas tarefas sob uma mesma seção ou departamento;
- ⇒ dependendo do sistema escolhido e do número de pessoas envolvido, nomear editores que coordenem as entradas irrelevantes ou incorretas;

⇒ examinar os processos de grupos existentes envolvendo usuários, profissionais de sistemas de informação, especialistas em mudanças organizacionais e gerentes comerciais;

⇒ fornecer programa de treinamento para familiarizar os participantes com a nova tecnologia;

⇒ considerar o groupware como um meio mais rápido de integrar novos funcionários em suas atividades profissionais;

⇒ saber discernir o momento de trocar uma comunicação eletrônica por uma face-a-face e vice-versa;

⇒ avaliar continuamente se a organização conseguiu incrementar qualitativamente seus objetivos específicos.

3.2.2 CSCW

Em nosso estudo consideramos "Computer-Suported Cooperative Work" (CSCW) como a área científica que motiva e valida projetos de Groupware. Através do CSCW é possível estudar como as pessoas trabalham em equipe, como o computador e as tecnologias afins afetam o comportamento do grupo. Segundo Wilson [WILS 91] existem duas grandes áreas de interesse dentro dos estudos de CSCW: o apoio ao trabalho de pessoas dentro de um grupo e a tecnologia que deve ser desenvolvida para tal propósito.

Bannon et al. [BASC 91] colocam que CSCW deve pesquisar os aspectos listados abaixo e que são o cerne dos seus principais questionamentos:

- Articulação do trabalho cooperativo;
- Partilhamento do espaço de informações;
- Adaptação da tecnologia às organizações e vice-versa.

Como os esforços cooperativos envolvem grande número de tarefas secundárias de controle entre os vários indivíduos, surgem questões: que membro fará o que? Como? Onde? Quando? Que membro está mais capacitado para determinada tarefa? O trabalho articulado seria então o controle e a coordenação de tarefas, pessoas e recursos tecnológicos.

As observações feitas por Nunamaker et al. [NUDV 93] e Bannon et al. [BASC 91] nos estimulam a perceber a importância da tecnologia computacional apoiando a cooperação, pois os sistemas computacionais devem permitir que as informações geradas

durante processos cooperativos possam ser acessadas facilmente pelos membros da equipe e até mesmo por pessoas externas. Os problemas que se colocam referem-se ao controle de acesso a essas informações, o acesso à identidade daqueles que geraram alguma informação e responsabilidade de decisões. Logo, os sistemas cooperativos devem favorecer a interrelação entre membros facilitando a comunicação, o armazenamento de informações e todo o controle necessário na coordenação das atividades cooperativas.

A introdução de mudanças tecnológicas numa organização pode acarretar alterações estruturais na organização do pessoal e do processo de trabalho. É necessário, então, que o projeto de sistemas cooperativos leve em consideração a compreensão dos processos envolvidos nas interações entre membros, a tecnologia e a própria organização.

Um ponto muito importante e que precisa ser mais aprofundado é o do medo da perda do poder dentro da hierarquia das organizações. Pois com a introdução de novas tecnologias ocorrem mudanças organizacionais e gerenciais, com novas distribuições de cargos e posições gerando resistências e inseguranças. Essas mudanças podem acarretar um aumento de competição entre os membros da equipe. Isso contrasta com o apregoado aumento de democracia no ambiente de trabalho causado pela introdução de sistemas cooperativos. Logo, deve-se buscar sistemas que expandam as opções dos trabalhadores, de tal forma que eles possam desenvolver estratégias de trabalho que permitam uma participação consciente e democrática.

Greenberg [GREE 91] ressalta a interdisciplinaridade de CSCW, devendo reunir pesquisadores de várias especialidades, como: ciência da computação, ciência cognitiva, psicologia, sociologia, linguística, antropologia, gerência, etc, cada um contribuindo com diferentes perspectivas e metodologias de aquisição de conhecimento dos grupos e sugerindo como o trabalho de grupo pode ser apoiado.

Existem ainda uma vertente de discussões sobre o avanço europeu em CSCW e a sofisticação técnica dos americanos. Segundo Grudin [GRUD 92], a escandinávia já estudava aspectos socio-técnicos na década de 60 e que somente nos anos 80 é que os EUA começaram a abordar tal assunto. Mais ainda, até hoje os EUA tendem a focar seus avanços nas áreas técnicas - através de sistemas de apoio ao trabalho cooperativo - enquanto que a Europa centra suas pesquisas nos impactos organizacionais da tecnologia.

Pode-se afirmar que os congressos de CSCW realizados a partir de 1986, nos EUA e Europa, contribuem para que essas diferenças comecem a ser atenuadas, através da integração de pesquisadores dos dois continentes.

3.3 Avaliação: Discutindo vantagens e desvantagens

Como vivenciamos um crescente interesse na área de trabalho cooperativo com apoio computacional é importante abordar, também, alguns pontos negativos ou pouco eficientes, conforme observado por alguns autores.

O trabalho cooperativo apoiado em groupwares não implica somente em vantagens e lucros como vem sendo amplamente divulgado, existem alguns aspectos que só são percebidos a longo prazo. Esses pontos começam a ser discutidos principalmente na Europa e nos EUA onde técnicas e ferramentas de apoio a trabalho cooperativo vem sendo utilizadas há mais tempo. A escolha da ferramenta de Groupware apropriada para cada caso envolve aspectos organizacionais mais complexos do que a simples interligação remota dos membros do grupo.

Em Datanews [DATA 92] são expostos alguns desses aspectos:

- **Excesso de informações compartilhadas** - Segundo gerentes e consultores existe um interesse muito grande no compartilhamento de informações e o objetivo do trabalho muitas vezes se perde. Para tentar minimizar esse problema já existem empresas que designam editores especializados que gerenciam as informações eliminando entradas irrelevantes, imprecisas ou enganosas, diminuindo assim a quantidade de dados inúteis.

- **Reuniões eletrônicas substituem as reuniões tradicionais** - É necessário ter consciência de que, em certos casos, um contato face-a-face agilizará as discussões. Por outro lado, a comunicação eletrônica elimina alguns conflitos que poderiam surgir no contato pessoal, pois formaliza as discussões, minimizando o potencial de envolvimento emocional. O fundamental é ter capacidade de julgar e perceber quando interações complementares se fazem necessárias.

- **Maior participação da equipe aumenta a qualidade e a produtividade** - Segundo alguns especialistas, há indícios da existência de um fenômeno chamado "vadiagem social". Quanto mais cresce a participação de alguns componentes, outros se sentem acomodados e não se interessam em participar. Isso pode ser causado pela falta de capacidade de coordenar os esforços do grupo como um todo e pelo enfraquecimento da auto-estima de alguns participantes. Para amenizar esse problema é interessante que os papéis de cada participante estejam bem claros e que os coordenadores dos projetos estejam atentos para intervir, promovendo a integração do grupo.

• **Disputa de poder** - Como já foi ressaltado em diversas oportunidades, os sistemas de apoio ao trabalho cooperativo promovem a interação de pessoas que estão afastadas fisicamente. Mas podemos questionar: Até que ponto as informações são verdadeiramente compartilhadas? No mundo competitivo em que vivemos uma informação estratégica pode gerar poder. Compartilhar tal informação pode resultar na perda de alguma vantagem pessoal, e/ou de poder. Esses temores podem ser dissipados através da conscientização da equipe sobre a importância da sua participação para que o objetivo do trabalho seja plenamente alcançado. O que por outro lado, pode esbarrar na cultura organizacional, como por exemplo, a avaliação de desempenho a nível individual.

• **Falta de infra-estrutura para instalação de novas tecnologias** - Nem todas as empresas interessadas em instalar Groupware, e manter um grupo coeso trabalhando, têm a infra-estrutura tecnológica e social necessárias para o desenvolvimento dos trabalhos de forma produtiva. A grande parte dos groupwares precisa de pelo menos uma rede local. Se extrapolarmos para outros espaços geográficos seria necessário um sistema de mensagens à distância. Se a empresa possui vários groupwares, os gerentes e diretores precisarão consultar uma grande quantidade de sistemas de mensagem.

Quando uma empresa implanta uma ligação através de rede visa, primordialmente, o incremento e otimização das tomadas de decisão, gerando aumento da produtividade. Entretanto, segundo Carvalho et al. [CAFU 92], estudos recentes realizados na Universidade de Carnegie Mellon, com pequenos grupos de trabalho, levaram à resultados opostos. Constatou-se a divergência de opiniões em função direta com o aumento da participação.

Outro aspecto que deve ser devidamente avaliado são os softwares que permitem que uma só pessoa monopolize a rede, não podendo haver interrupções por parte de outros membros da equipe, provocando atrasos nas comunicações e nas decisões.

Segundo Cockburn et al. [COTH 91] existem indícios de que os Sistemas de Groupware ignoram as necessidades individuais de cada participante, o que causaria rejeição. Grudin [GRUD 88] identificou dois fatores que contribuem para a rejeição dos usuários:

- **Trabalho adicional:** para os trabalhadores que se encontram em cargos mais baixos, a introdução dos Groupwares representa um adicional na carga de tarefas, muitas vezes, não trazendo nenhuma melhoria nas condições de trabalho, além de gerar inseguranças com respeito à substituição do trabalho humano pela máquina.

- Mudança nos métodos de trabalho: os trabalhadores são forçados a aprender novas técnicas impostas pelos novos sistemas, gerando alterações em suas rotinas de trabalho.

Alguns estudos de caso evidenciam que, se as ferramentas embutem imposições de mudanças nos métodos de trabalho existentes, essas ferramentas tendem a ser rejeitadas, Grudin [GRUD 88]. Uma forma de amenizar essa rejeição é promover um trabalho de motivação e conscientização sobre a importância da introdução das novas ferramentas, fornecendo também, treinamento específico.

O grupo participante do workshop do CSTB [CSTB 90] ressalta que existe pouca interação entre os pesquisadores no sentido de se tentar levantar as principais dificuldades para que ferramentas apropriadas possam ser desenvolvidas. Enfatiza também, que o distanciamento entre as Universidades e o sistema produtivo contribui para que as pesquisas acadêmicas relevantes não sejam incorporadas pela sociedade.

No Brasil começam a aparecer os resultados das primeiras pesquisas realizadas na área da cooperação, para o levantamento de aspectos sociais envolvidos nas interações entre os membros de uma equipe. Duarte et al. [DUFL 92] realizaram um estudo de caso na PUC-RJ, onde através da observação de sessões de desenvolvimento de projetos de software realizadas por grupos de voluntários, procuraram identificar os aspectos cooperativos envolvidos, para poder guiar posteriormente, o desenvolvimento de tecnologia computacional que apoie essa atividade. A técnica de observação do trabalho desenvolvido pelos grupos, juntamente com a técnica de análise desenvolvida compõem uma metodologia proposta para o levantamento dos aspectos cooperativos.

A técnica de observação (abordagem observacional) busca identificar as informações relevantes para o contexto da atividade. A técnica de análise adotada parte da percepção da atividade de desenvolvimento cooperativo de projetos de software através de dois enfoques: o ponto de vista de um projetista de software e o ponto de vista de um especialista em interação. O projetista teria preocupação em identificar situações características de projeto de software e o especialista em interação identificaria e classificaria situações intrínsecas ao processo de comunicação entre os componentes do grupo.

Através da identificação dos aspectos relacionados ao projeto cooperativo de software e ao processo de interação, foram obtidos os requisitos preliminares para um sistema de apoio ao projeto cooperativo em software, apresentados, sucintamente, a seguir:

- o sistema não deve basear-se em modelos que definam, a priori, os papéis a serem cumpridos pelos membros da equipe;

- o sistema deve apoiar ações, visando o planejamento das atividades da equipe;
- o sistema deve ser passível de expansão, de forma a poder adotar diferentes metodologias de desenvolvimento de software;
- o sistema deve apoiar o desenvolvimento incremental do projeto;
- ♦ o sistema deve fornecer um conjunto de formas de comunicação;
- o sistema deve ser capaz de identificar com quem cada membro está interagindo.

Logo, mostra-se fundamental pesquisar sobre o comportamento de grupos e de indivíduos e natureza dos locais de trabalho e das organizações, pois fornecer suporte computadorizado para cooperação é difícil e requer um amplo entendimento da forma como organizações e grupos funcionam, Rocha [ROCH 92].

Segundo Kling [KLIN 91], é essencial que os pesquisadores, profissionais e gerentes compreendam as reais oportunidades sociais e limitações da nova família de tecnologias. A computação sozinha raramente é capaz de efetuar transformações nas organizações. As novas tecnologias expandem os limites do possível e os sistemas computacionais de trabalho cooperativo reduzem as complexidades do trabalho. Entretanto, o estudo do processo de trabalho e da organização das pessoas trabalhando em equipe não é suficiente para a compreensão do uso e dos impactos gerados por sistemas computacionais de trabalho cooperativo. Existem diversas variáveis envolvidas, tais como: cultura da empresa, formas de organização do trabalho, democratização das relações de trabalho, circulação de informações, etc. Os estudos devem ser aprofundados de modo a examinar a integração desses sistemas em processos de trabalho mais ricos e complexos tecnologicamente.

Ishii [ISHI 92] nota que a idéia chave para as próximas gerações de tecnologia de apoio à colaboração é a junção entre indivíduo e grupo. Ferramentas de apoio a grupos devem ser combinadas a ferramentas de trabalho individual.

3.4 Áreas Promissoras de Adoção do Trabalho Cooperativo

Existem algumas áreas em que a teoria cooperativa é aplicada como forma de obter maior eficiência, eficácia, qualidade e rapidez dos trabalhos.

Na Educação é introduzida com o objetivo de diminuir a competição e incrementar a ajuda mútua entre os alunos. Na parte administrativa das organizações intensifica a comunicação e as trocas de informações, colaborando para agilizar as tomadas de decisão. Já na área de projetos e desenvolvimento de produtos de software fomenta a participação do usuário nos projetos e possibilita o reagrupamento de ocupações computacionais fragmentadas.

3.4.1 Trabalho Cooperativo em Educação

Com a constante difusão do interesse em trabalho cooperativo, percebeu-se sua adequação à área educacional. De acordo com Manning et al. [MALU 91] e Hassard [HASS 91], o trabalho cooperativo serve como alternativa ao processo tradicional, que cria competição entre os alunos, enquanto que a cooperação estimula a interação e a ajuda mútua.

Manning et al. [MALU 91] observam que está havendo o reconhecimento crescente de que a escola tradicional estimula um comportamento competitivo individual. Em contrapartida, há evidências de que o aprendizado cooperativo contribui positivamente para o sucesso acadêmico, o desenvolvimento de habilidades de convivência social e da auto-estima. O trabalho cooperativo estimula os estudantes a trabalharem em equipe, possuindo um objetivo comum. Em vez de haver competição, há ajuda mútua.

Segundo os autores citados acima, a forma de aprendizado cooperativo que tem se mostrado mais eficaz é a do modelo cujos trabalhos são individuais, mas os objetivos são grupais. Os estudantes devem se comunicar sem ambiguidade, resolver conflitos e se apoiar mutuamente. Stahl [STAH 91] considera que na era da informação os estudantes não necessitam dominar um conteúdo, mas dominar o processo de aprendizagem. E nesse aspecto, o aprendizado cooperativo mostra-se de grande valia. Dentro desse quadro, o professor deixa de centralizar o poder e o saber para ser o coordenador e o incentivador. O aprendizado do grupo deve se sobrepor ao aprendizado individual. Por outro lado, com a introdução do computador, novos problemas surgem. Isto é, com a utilização cada vez maior de computadores de pequeno porte nas escolas e em casa, é necessário estabelecer formas de trabalho em grupo, que viabilizem que os resultados finais sejam iguais ou melhores do que os obtidos no desenvolvimento de trabalho cooperativo sem apoio computacional.

Norman [NORM 91] observa que, como em trabalho cooperativo estão envolvidos fatores sociais, culturais e psicológicos, esses modelos deveriam ser definidos por uma equipe multidisciplinar, constituída de especialistas em computação, cientistas

sociais e representantes da comunidade que vai utilizar essas ferramentas, pois esses sistemas devem preencher as necessidades do grupo como um todo.

Existem várias ferramentas para apoio de trabalhos cooperativos na área educacional, entre elas destacamos as seguintes: hipertextos, correio eletrônico, editores cooperativos de textos, sala de aula eletrônica.

Hipertextos

Hipertexto é geralmente definido como uma forma não linear de construir situações. Os sistemas hipertexto são apropriados a aplicações que exijam o armazenamento de informações para consultas de forma não-sequencial, Faria [FARI 91]. A operação das informações nesses sistemas pode ser análoga ao funcionamento da mente humana, facilitando sua utilização pelos estudantes.

Stahl [STAH 91] lembra que as pessoas podem interagir enquanto criam e lêem documentos de hipertexto. Em trabalho cooperativo com apoio de hipertextos uma pessoa cria nós e ligações que podem ser complementadas por outra pessoa, estimulando a troca de idéias e experiências, a argumentação e a organização. Temos que ressaltar que as pessoas terão flexibilidade para exercer o controle sobre os seus ritmos de aprendizado.

O hipertexto realizado cooperativamente, vai demandar uma equipe de autores, um verdadeiro trabalho coletivo. Constitui oportunidades de integração do saber e de transpor obstáculos da comunicação escrita tradicional.

Correio Eletrônico

Esse tipo de comunicação pode funcionar como incentivo a pesquisas a novas fontes de informações. A troca de experiências pode ser proveitosa evitando que erros cometidos anteriormente por outras pessoas ou grupos, sejam repetidos.

O correio eletrônico é amplamente usado por estudantes, professores, instituições de pesquisa, possuindo diversas listas de tópicos de interesse sobre os mais variados assuntos, inclusive para crianças.

Editores Cooperativos de Textos

Quando a edição é assíncrona, os co-autores trabalham num mesmo documento em tempos diferentes, podendo passar anotações e/ou mensagens entre si. Na edição síncrona, o resultado do trabalho de um co-autor é imediatamente visível para os outros, incluindo as mensagens e anotações. A edição cooperativa de textos geralmente permite a leitura de qualquer segmento, mas um só "escritor" por segmento. Essa ferramenta facilita a criação de textos e incentiva a comunicação escrita.

Salas de Aula Eletrônicas

Cada aluno tem uma estação de trabalho pessoal, interligada em rede com os outros componentes do grupo. Um arranjo comum é arrumar em formato de "U" com uma grande tela a frente, de onde o facilitador/professor guia o encontro. Será ele que escolherá a ferramenta a ser usada. Existem vários modelos de tarefas: os participantes podem trabalhar em diferentes aspectos de um problema simultaneamente. De uma maneira ideal, os sistemas devem permitir oportunidades iguais de participação. Há indícios de que as salas de aula eletrônicas contribuem para o desenvolvimento da rapidez de decisão e criatividade.

As perspectivas da utilização de hipertextos, editores cooperativos, sala de aula eletrônica, correios eletrônicos, entre outros, são cada dia mais promissoras. É preciso desenvolver modelos educativos que aproveitem as novas tecnologias desenvolvidas para alcançar uma ampla adoção desses recursos, intensificando o aprendizado. Se bem que o alto custo financeiro dos equipamentos e do software específico, ainda é um grande obstáculo para a ampla implementação de tais sistemas na grande maioria das escolas brasileiras.

3.4.2 Cooperação no Escritório

A difusão da informática é crescente nos processos administrativos das empresas e organizações, ou seja, nos escritórios.

O termo "Automação de Escritório" tem sido amplamente utilizado com várias abordagens e definições. Em Mariano [MARI 92] encontramos uma definição proposta por Hirschhein [HIRS 85] que aborda o termo de forma bem resumida, congregando as idéias gerais. Tal autor coloca que Automação de Escritório refere-se à uso integrado das tecnologias de computação, comunicação, produtos de escritório e conhecimentos de ciências sociais para dar suporte a várias atividades e funções dentro do escritório ou no ambiente de escritório. Hoje o termo mais atual é "Sistema de Informação de Escritórios".

Mariano [MARI 92] detalha três níveis de automação dentro do escritório: o primeiro e que deu origem ao termo Automação de Escritório, consistia na automação de tarefas administrativas consideradas mais simples, como utilização de planilhas eletrônicas e processadores de texto; o segundo consistia em Sistemas de Informação Gerencial (MIS), que tinham o objetivo de fornecer apoio em processamento de dados para a média gerência da empresa; e por fim os Sistemas de Apoio à Decisão, que são desenvolvidos para dar

apoio informacional aos executivos e alta gerência. Como esses três níveis precisavam estar integrados possibilitando um melhor fluxo de informações dentro da organização, foi desenvolvido o Sistema de Informação de Escritórios.

Para que as organizações possam utilizar os Sistemas de Informação de Escritórios são necessárias tecnologias eletrônicas e redes de comunicação. Essas tecnologias permitem às pessoas processarem documentos e informações mais rapidamente e com maior eficiência e eficácia.

Segundo Schlichter [SCHL 92] o processamento de documentos é frequentemente uma atividade de um grupo de pessoas. As comunicações para trocas de informações podem ser baseadas em sistemas como Correio-Eletrônico, Conferência Eletrônica, Agenda Eletrônica e Editores Cooperativos de Textos. Existem ferramentas tais como COLAB, NOTECARDS e QUILT para áreas específicas da cooperação dentro de organizações. A COLAB se fixa na intermediação de reuniões face-a-face; NOTECARDS fornecem funções de estruturação e recuperação de informações para pessoas ou grupos de trabalho; QUILT fornece uma estrutura para escrita cooperativa, comunicação entre co-autores e coordenação de atividades e informações compartilhadas.

Como podemos constatar em Mariano [MARI 92], que algumas das empresas brasileiras já possuem iniciativas de automação de escritórios e de trocas de informações entre as diversas unidades de uma mesma organização, pois o nível de integração das informações é fator preponderante para o desenvolvimento de sistemas estratégicos.

3.4.3 Propostas de Trabalho Cooperativo na Engenharia de Software

3.4.3.1 Cooperação reintegrando ocupações fragmentadas

Com o rápido desenvolvimento dos computadores, segundo Tierney [TIER 91], as ocupações computacionais começaram se fragmentando em especialismos, surgindo novas ocupações, através do particionamento do processo de trabalho. Não só os cargos, como também as tarefas de desenvolvimento de sistemas foram fragmentadas em: especificação funcional, especificação de projeto, desenvolvimento de programas, implementação, testes, etc, onde cada fase seria trabalhada separadamente e denomina-se "ciclo-de-vida".

Na década de 70 emerge o Analista de Sistemas. A princípio as fronteiras entre a Análise de Sistemas e a Programação foram difíceis de serem reconhecidas e

delimitadas. No entanto, nessa década a divisão implementada era a de que programadores ocupavam-se mais da elaboração de desenvolvimento dos programas, sua implementação e testes. Enquanto os analistas estariam mais ligados aos usuários e a gerência, ocupando-se das fases de análise e projeto. Isso pode ser observado pelo tempo gasto com reuniões com os usuários: a maioria dos programadores (74%) gasta menos de 10% de seu tempo com usuários enquanto que somente 19% dos analistas gastam esse mesmo tempo, Tierney [TIER 91].

Em particular, na década de 70 o software era reconhecido como o fator limitador em alcançar a produtividade, a eficiência e a confiabilidade dos sistemas. Entre os problemas levantados estavam: o tamanho dos sistemas, a dificuldade de medir o progresso dos programadores, a gerência do grande número de funcionários nos departamentos em expansão e a documentação inadequada dos sistemas.

O Trabalho Cooperativo seria um caminho para a reintegração dessas ocupações fragmentadas. Trabalhadores especialistas e usuários podem ser reagrupados, e dessa forma, reunirem conhecimentos complementares em torno de um objetivo comum. Em sua pesquisa, Carneiro [CARN 93] pode constatar, em empresas brasileiras, alguns indicadores desse reagrupamento de ocupações profissionais e de uma participação cada vez maior do usuário nas equipes desenvolvedoras de projetos. Ultimamente observa-se que a função de Analista de Sistemas vem sendo dividida em um profissional ligado diretamente às definições do projeto e outro mais voltado para a identificação de novas aplicações estratégicas da tecnologia de informação nos negócios: o Analista de Negócios, confirmando o constatado por Friedman et al. [FRCO 93], já na década de 80, em países desenvolvidos, o surgimento do analista/programador, que seria a categoria de analistas projetistas. Isso representa uma reversão da tendência a fragmentação prevalescente na década de 80.

3.4.3.2 Importância do usuário

Dentro das características de trabalho cooperativo o usuário assume um papel central. Sua importância é constantemente frisada e muito discutida, mas o usuário ainda não parece ocupar seu lugar dentro das equipes desenvolvedoras. Ainda não foi assimilada a importância do papel desempenhado pelo usuário no processo de desenvolvimento de projetos.

Há uma série de resistências à participação direta e extensiva ao longo do processo de desenvolvimento de aplicativos. Geralmente sua participação se restringe a definição de requisitos e, posteriormente, a fase de validação, quando os erros detectados e suas alterações contribuem para um significativo aumento dos custos.

Muitas vezes o distanciamento entre usuários e equipe desenvolvedora é imposta pela própria estrutura organizacional das empresas, onde não pode haver saltos hierárquicos para que se estabeleçam conversas, muitas vezes informais, sobre o andamento dos trabalhos. As empresas se organizam rigidamente e fazem com que os contatos só ocorram através de reuniões formais, onde o fluxo de informações pode ser melhor controlado.

Walton [WALT 88] assinala que as novas tecnologias de informação possuem potencial de alterar radicalmente os indivíduos, as instituições e a sociedade, logo dentro deste contexto, é indispensável maior atenção aos métodos de desenvolvimento e implementação dessas tecnologias emergentes em termos de hardware, software e dos sistemas de gestão, permitindo que o usuário participasse ativamente do projeto à implantação das inovações tecnológicas. Todos os grupos interessados e afetados pelos resultados do projeto devem intervir no processo de desenvolvimento, tornando-o mais democrático e gerando um produto de maior qualidade.

Munford [MUNF 88] acentua que a participação do usuário é o meio idôneo para projetar sistemas técnicos. Os diretores e especialistas das empresas devem estar convencidos de que a satisfação das necessidades humanas têm tanta importância no projeto como a satisfação das necessidades técnicas e comerciais. Ela cita Leavitt, segundo o qual a tarefa de projetar seria o resultado da fusão de quatro variáveis: tecnologia, estrutura de tarefas, necessidades humanas e necessidades organizacionais.

E quem seriam esses usuários que deveriam participar do processo ? São aqueles que operam os aplicativos ? Ou seriam aqueles que trabalharão com os relatórios finais, mas não operam os computadores ? Essas fronteiras muitas vezes tornam-se confusas, mas percebemos uma tendência, na literatura existente, a considerar como usuário aquelas pessoas que acessarão os relatórios finais. Neste trabalho são considerados usuários todas as pessoas que terão contato com o sistema e serão afetadas por ele, seja através dos computadores ou através do uso das informações dos relatórios e resultados.

Segundo Bannon [BANN 91] os usuários não são simplesmente objetos passivos, eles são agentes ativos. Em seu trabalho encontramos narrativas sobre a angústia de alguns usuários que gostariam de compreender mais profundamente seus trabalhos e não somente agir mecanicamente.

Hoje podemos observar um crescente interesse na área da interação homem-máquina, com o envolvimento de pesquisadores de várias disciplinas como psicólogos, ergonomistas, engenheiros, etc. Bannon [BANN 91] coloca algumas sugestões de mudança no campo da interação homem-máquina que devem ajudar a diminuir as distâncias entre

teoria, experimento, projeto de sistemas e o atual estado do trabalho. Essas propostas procuram contribuir para ajudar a tornar as relações mais amigáveis, integrando o usuário na equipe desenvolvedora:

- Deve ser dada mais atenção ao processo de projeto, os usuários devem participar de todas as etapas do desenvolvimento, para perceberem a natureza interativa e as mudanças de concepção.
- A maioria dos estudos sobre interação homem-máquina focalizam as relações do indivíduo trabalhando com sistemas computacionais. É interessante focalizar a mudança de enfoque para trabalho cooperativo, avaliando sua importância.
- Não generalizar resultados obtidos em experiências de laboratório.
- Procurar ter sempre um psicólogo na equipe de projeto que poderá avaliar e conduzir os estudos dessas relações.

3.5 Conclusões

Neste capítulo enfocou-se a importância dos ambientes computacionais apoiando a cooperação. Discutiu-se o escopo dos sistemas de apoio ao trabalho cooperativo, enfatizando distintas definições e propostas.

Através dos pontos levantados pôde-se notar que o estudo dos Groupwares e as pesquisas associadas em CSCW representam uma tendência atual. Mesmo as características colocadas como negativas são abordadas como desafios que estimulam o aperfeiçoamento das metodologias, ferramentas e das discussões de aspectos culturais, sociais, psicológicos e políticos envolvidos nos processos interativos.

Observou-se que vários autores ressaltam a importância de que não só a tecnologia seja privilegiada e que, cada vez mais, cientistas sociais sejam envolvidos no desenvolvimento de sistemas de apoio à cooperação.

Por fim abordamos algumas áreas em que a cooperação tem proporcionado profundas e vantajosas modificações. Na área educacional substituindo a competição entre os estudantes pela ajuda mútua. Nos escritórios interligando pessoas e equipes, agilizando procedimentos. Já na área de projetos e desenvolvimento de produtos de software como meio de reagrupar funções muito fragmentadas, assim como, favorecendo a participação dos usuários nas diferentes etapas do desenvolvimento como forma de satisfazê-los e aumentar a qualidade do produto gerado.

CAPÍTULO 4

O Processo de Desenvolvimento de Software

Nos primórdios da computação os usuários eram os próprios programadores. Com o crescimento do hardware as aplicações se expandiram da área científica para a área administrativa e comercial. Começa então a separação entre usuários finais e desenvolvedores, e posteriormente, o desenvolvimento dos produtos começou a ser feito por equipes. As primeiras equipes de desenvolvimento eram pequenas e geralmente as tarefas dividiam-se em análise e programação. Com o aumento do tamanho e complexidade dos problemas as equipes foram crescendo, tornando-se uma atividade coordenada por um grupo de pessoas, onde a participação dos usuários ocupa um espaço crescente.

Os métodos tradicionais de desenvolvimento de produtos de software não eram apropriados para apoiar o trabalho dessas equipes, e os países escandinavos saíram na frente no desenvolvimento de métodos cooperativos específicos e com variadas pesquisas e experiências na área de trabalho cooperativo.

Neste capítulo apresentaremos uma retrospectiva da evolução do desenvolvimento de software desde a prática "ad doc" até a programação em larga-escala. Levantamos algumas questões e chegamos às equipes organizadas segundo um modelo.

Em seguida discutimos brevemente os modelos de ciclo-de-vida, os métodos mais difundidos e suas ferramentas associadas, abordando suas deficiências em apoiar o trabalho de equipes.

Apresentamos alguns métodos cooperativos desenvolvidos mais recentemente que valorizam a participação dos usuários em todas as etapas do processo de desenvolvimento de produtos de software. Comparamos esses métodos e apresentamos o resultado numa tabela, de modo a facilitar sua visualização.

Por fim, abordamos algumas questões relativas a falta de pesquisas acadêmicas e ressaltamos os pontos que deveriam ser mais explorados para a obtenção de ferramentas confiáveis para o apoio ao trabalho cooperativo.

4.1 Evolução

O processo de desenvolvimento de produtos de software vem sofrendo alterações na sua estrutura de produção através do aprimoramento de técnicas e do constante avanço tecnológico do hardware e das redes de comunicação.

Os primeiros softwares emergiram da prática "ad doc" onde a intuição e o talento de amadores predominavam. As trocas de informações sobre as experiências no desenvolvimento não eram comuns devido à falta de equipamentos e tecnologia capazes, e também, à falta de reconhecimento por parte dos desenvolvedores da importância do fluxo de informações como agilizador de processos. Assim, essa falta de intermediação acarretava que o mesmo problema era resolvido várias vezes de formas diferentes.

Segundo Lewis et al. [LEOM 90] e Shaw [SHAW 90] nos anos 60 os desenvolvedores de programas e sistemas eram os próprios usuários. Os programas eram pequenos e realizavam uma tarefa de cada vez em linguagens difíceis de serem manipuladas. As técnicas gerenciais eram inexistentes e o controle sobre as tarefas eram bem informais.

No fim dos anos 60 percebeu-se a importância do formalismo lógico e da organização dos dados para que a tarefa de desenvolvimento, alteração e consulta fosse facilitada, os algoritmos e estruturas de dados começam, então, a receber maior ênfase. Nessa fase predomina a programação em pequena escala.

Os anos 70 trazem linguagens mais simples, com dados organizados de forma mais rígida, mas ainda em estruturas simples e pequenas. Os programas são executados uma vez e terminam. O esforço do desenvolvedor permanece individual e pouco gerenciado. As instituições percebem que aplicações computacionais podem agilizar processos administrativos, e com isso, as entradas e saídas dos programas ganham importância, mas a documentação continua fraca. Os programas e sistemas possuem "donos", que são pessoas que desenvolvem e utilizam os aplicativos, e nenhuma outra pessoa pode ou é capaz de operá-los. Há um crescente desenvolvimento de produtos de software sistematizando rotinas e procedimentos, servindo de rampa de acesso para o desenvolvimento em larga escala, a partir de 1975. Os sistemas tornam-se maiores e mais complexos exigindo um maior número de pessoas envolvidas no desenvolvimento dos projetos. Com isso, era preciso o apoio de métodos de análise e projeto para coordenar os

trabalhos das equipes desenvolvedoras. Os programas executam indefinidamente e as ferramentas integradas começam a disseminarem-se.

A década de 80 é marcada pelas ferramentas CASE e ferramentas correspondentes a cada fase do ciclo-de-vida.

Para os anos 90 há fortes indícios que a engenharia de software será dominada por sistemas inteligentes, que, em combinação com estações de trabalho e técnicas de automação para desenvolvimento, serão largamente usadas na indústria de software. As tecnologias de Orientação à Objetos sairão dos laboratórios para a prática juntamente com as arquiteturas paralelas.

O surgimento e o crescimento natural desse conjunto de técnicas de desenvolvimento, controle e manutenção de sistemas de software forçou o surgimento de uma área de pesquisa denominada Engenharia de Software.

Segundo Shaw [SHAW 90], em 1968 o termo Engenharia de Software nomeou uma conferência que discutia problemas da produção de software, sendo que as discussões basearam-se no progresso das técnicas de desenvolvimento da fase artesanal até a fase do desenvolvimento em larga escala. Em 1976 Boehm [BOEH 76] propôs a definição do termo como: "aplicação prática do conhecimento científico no projeto de construção de programas computacionais e a documentação associada que é necessária para desenvolver, operar e manter-los". Hoje, na prática, o termo é mais frequentemente usado para referir-se aos modelos de ciclo-de-vida, métodos de desenvolvimento, técnicas para estimativa de custos, documentação, ferramentas de controle de configuração, técnicas para controle de qualidade e outras técnicas que estabelecem padrões para as atividades de produção de software.

Entretanto, não podemos deixar que o termo "Engenharia" se restrinja às técnicas científicas envolvidas no processo prático de desenvolvimento superestimando a eficiência técnica, pois o software é a ponte de um sistema que une pessoas e o hardware. Os aspectos técnicos coexistem com aspectos sociais, organizacionais, econômicos, políticos, psicológicos, comportamentais, legais e gerenciais. Entre estes, um é constantemente ressaltado e discutido como sendo o que influencia diretamente todos os outros: a gerência.

Alguns desses aspectos não técnicos que envolvem as equipes desenvolvedoras foram amplamente discutidos na 13ª Conferência de Engenharia de Software, em 1991 nos EUA, em painel que foi mediado por Marc Kellner [KELL 91]. Abaixo abordaremos algumas questões que podem influenciar negativamente os grupos desenvolvedores com algumas propostas para amenizar o possível quadro desfavorável.

♦ **Gerência Ineficiente**

A falta de gerência efetiva pode causar uma enorme gama de problemas no desenvolvimento dos projetos: fraco planejamento, desmotivação da equipe, equipe inoperante por falta de visão das habilidades individuais, execução ineficiente devida a organização inadequada, dificuldades técnicas devido a falta de direcionamento de atividades ou falta de habilidades específicas.

Algumas ações sugeridas como forma de amenizar esses problemas são: definição de duas linhas de trabalho sendo uma técnica e outra gerencial, treinamento em habilidades gerenciais e técnicas, supervisão por gerentes mais experientes. Essa visão é reforçada por DeMarco [DEMA 91] e Tully [TULL 91], que colocam que os gerentes se preocupam demais com aspectos técnicos e de menos com aspectos sociais.

Curtis [CURT 91] chama a atenção para o fato de que as questões humanas são geralmente consideradas como o lado mais "soft" da Ciência da Computação. Mas que os programas continuam a ser escritos por pessoas e não por máquinas, sistemas especialistas ou geradores automáticos de programas. A tecnologia não pode ser vista com único meio de aumentar a qualidade e a produtividade. Logo, o desenvolvimento de técnicas é tão importante quanto o desenvolvimento de técnicas gerenciais.

Mas para a aplicação dessas idéias é fundamental o apoio estratégico da chefia das organizações. "Frequentemente, o maior desafio que um gerente enfrenta não é o de perceber a necessidade de mudanças ou de definir a mudança que é necessária, mas o de fazer com que a estrutura da empresa incorpore a mudança", Sivieri [SIVI 90]. A função de gerente de projetos de software exige não apenas o comando eficiente da equipe, mas também, a visão estrutural do funcionamento da empresa. Logo é importante que a integração gerente com superiores, e gerente com subordinados seja perfeita para que a absorção da mudança se faça realmente.

♦ **Falta de Cooperação entre os Componentes da Equipe**

O desenvolvimento de produtos de software é historicamente uma atividade que estimula o caráter individual da programação em pequena escala. Mas, esse modelo se aplicado a grandes projetos, contribue negativamente para a qualidade e os custos.

Segundo Schlumberger [SCHU 91], culturalmente os indivíduos preferem ser reconhecidos por si só, e não como parte integrante de um grupo. Novamente vemos

perpetuados os velhos paradigmas de que através da atividade individual se alcança maior autonomia, gerando poder sobre as informações e aplicações.

As opções para a reversão deste quadro são: o desenvolvimento de uma cultura organizacional que incentive os esforços grupais, melhor circulação de informações entre os diferentes membros, coordenação efetiva de todas as etapas, adoção de técnicas de programação "egoless".

♦ Diferenças de Desempenho

Dentro das equipes desenvolvedoras pode haver grandes diferenças individuais referentes a experiência, habilidades e conhecimento. Essas diferenças refletem-se diretamente no desempenho do grupo e podem contribuir para o surgimento de lideranças paralelas e competitivas, fazendo com que a produtividade decresça. Hanata [HANA 89] coloca que o desenvolvimento de software envolve conflito de idéias e que é importante que a pluralidade de idéias dos vários participantes envolvidos num processo de desenvolvimento de software sejam harmonizadas.

Esses aspectos podem ser amenizados através do treinamento e de uma gerência que estimule o desenvolvimento profissional, investimentos em ferramentas e métodos que aumentem a produtividade e a ajuda mútua dos componentes do grupo.

4.2 O Processo de Desenvolvimento de Software num Enfoque Tradicional

4.2.1 Desenvolvimento Feito por Equipes

Como foi abordado anteriormente o desenvolvimento de software foi se modificando através dos tempos e passou de trabalho individual para trabalho em equipe. A organização desses grupos é tarefa delicada e que deve ser bem estudada para que os objetivos sejam efetivamente alcançados.

Na organização dos grupos o número de membros é difícil de ser estabelecido. Grupos pequenos são mais fáceis de serem gerenciados, a comunicação se estabelece facilmente e os trabalhos desenvolvem-se mais harmoniosamente. Por outro

lado, alguns problemas são grandes e complexos demais para serem desenvolvidos por equipes pequenas, necessitando um maior número de participantes. Ghezzi et al. [GHJM 91] colocam que o número de pessoas de uma equipe desenvolvedora de software é influenciado pelas características do software a ser desenvolvido. Esse número deve-se situar entre 3 e 8 pessoas, dependendo da tarefa. Mas esse número não deve ser determinado aleatoriamente, e sim, após ser feita uma estimativa do esforço de desenvolvimento. Para isso, existem modelos específicos que determinam quantas pessoas-mês ou pessoas-ano são necessárias para o desenvolvimento dos trabalhos. A organização gerencial do grupo também é importante na realização das tarefas e influenciará diretamente no custo e na qualidade do produto desenvolvido.

"Não existe modelo de ciclo-de-vida ideal para todos os projetos, assim como não há organização de equipes apropriadas para todos os problemas", Ghezzi et al. [GHJM 91].

A seguir serão expostos alguns tipos de organização de equipes desenvolvedoras de produtos de software enfatizando seus pontos negativos e positivos.

TIPOS DE EQUIPES

As equipes podem ser organizadas seguindo vários modelos propostos por diversos autores. A seguir apresentaremos uma classificação proposta por Hanata [HANA 89].

- 1) Equipes com programadores chefe
- 2) Equipes democráticas
- 3) Equipes hierárquicas
- 4) Equipes especialistas

Essas equipes são bastante abordadas por vários autores, o que não significa que foram plenamente utilizadas. Na prática é difícil a implantação de um modelo de organização de equipes rígido. É mais fácil adaptar os pontos positivos de cada uma delas aos modelos pretendidos pela organização. A seguir veremos mais detalhadamente cada uma delas:

1) Equipes com Programador Chefe

Neste modelo de organização os vários membros da equipe são ligados a um programador chefe que é responsável pelo controle das tarefas e desempenho dos trabalhos. Ele fica responsável pelo detalhamento do projeto e pela escolha das técnicas a serem empregadas. Geralmente o chefe é ligado a um gerente superior que é responsável pelos aspectos administrativos do projeto. Logo, o programador chefe é responsável pelo controle das atividades técnicas e decisões. Com a expansão da informática para várias áreas, como comércio, indústria e educação, se faz necessária a intervenção de especialistas técnicos dessas áreas para ajudarem com informações específicas para o desenvolvimento dos projetos. O Programador Chefe é o responsável pela possível inclusão temporária desses especialistas na equipe inicial. Mais recentemente, o Programador Chefe é substituído pelo Analista Chefe. O bibliotecário é responsável pela organização e guarda das documentações, do próprio software e das decisões da equipe.

A figura abaixo representa graficamente os padrões de controle e comunicação deste modelo:

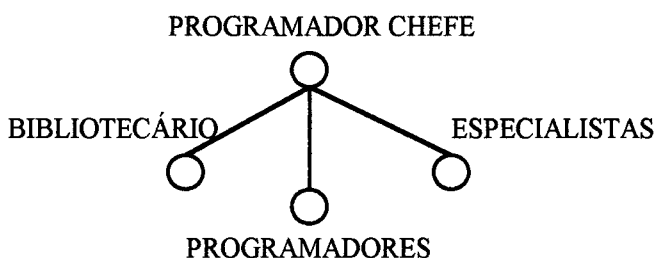


Figura 4.1: Padrão de controle e comunicação numa equipe com programador chefe [GHJM 91]

Esse modelo tem como pontos negativos a falta de democracia e de autonomia dos membros, pois todas as decisões devem ser tomadas com o conhecimento do chefe. O sucesso deste modelo depende da habilidade pessoal do chefe em amenizar esses pontos negativos e também do tamanho do problema a ser resolvido.

2) Equipes Democráticas

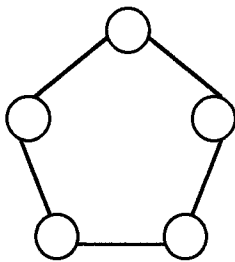
Nas equipes com organização democrática as decisões são tomadas por consenso e todo o trabalho é considerado como de todo o grupo. Cada membro revisa o

trabalho realizado pelo outro. O grupo como um todo é responsável pelo trabalho de cada um. Ghezzi et. al [GHJM 91] enfatizam que esse tipo de organização colabora para o aumento de satisfação dos trabalhadores o que por sua vez, contribue para o aumento da qualidade do produto final.

A descentralização do controle é favorável para o desenvolvimento de projetos longos. Como a comunicação intra-grupo é mais estimulada, o grupo é capaz de sustentar trabalhos mais demorados e complicados, pois pode encontrar melhores soluções que um indivíduo sozinho. Por outro lado, como aspecto negativo, é ressaltado que esse modelo organizacional não é vantajoso para grupos muito grandes onde a comunicação entre os membros reduzira a produtividade individual.

Abaixo temos as figuras que representam a estrutura gerencial e os padrões de comunicação desse tipo de equipe:

ESTRUTURA GERENCIAL



PADRÃO DE COMUNICAÇÃO

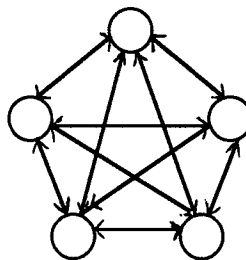


Figura 4.2: Estrutura gerencial e padrão de comunicação em equipes democráticas [GHJM 91]

3) Equipes Hierárquicas

A organização das equipes de forma hierárquica, busca combinar os benefícios dos modelos de equipe com Programador Chefe e equipes Democráticas, tentando evitar suas desvantagens.

A seguinte estrutura organizacional e de comunicação adotada, está representada na figura abaixo:

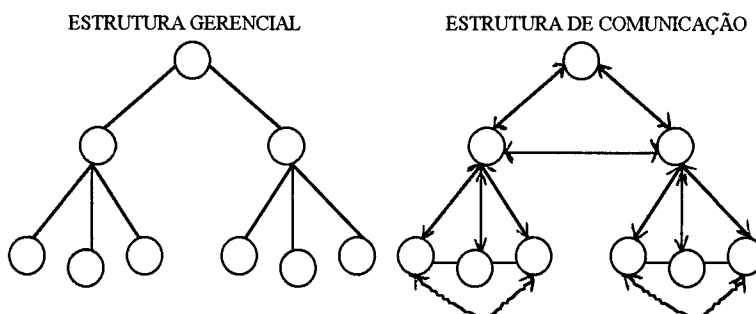


Figura 4.3: Estrutura gerencial e estrutura de comunicação nas equipes hierárquicas [GHJM 91]

O grupo possui um líder que é responsável pela coordenação do grupo, decide estratégias de projeto e gerencia trabalhos. No nível mais abaixo se situam os sublíderes, que assistem o líder intermediando sua comunicação com a equipe desenvolvedora e controlam a análise e o desenvolvimento dos projetos. No terceiro nível encontram-se os programadores, que ficarão responsáveis pela codificação, teste e documentação dos softwares.

A comunicação é descentralizada. Cada grupo se comunica com o seu superior imediato, trazendo vantagens encontradas nos pequenos grupos de trabalho.

Segundo Hanata [HANA 89] esse modelo adapta-se a projetos de grande escala e incentiva o aumento da capacidade de integração das pessoas, pois as diferenças profissionais tenderão a serem minimizadas através de treinamento.

Ghezzi et. al [GHJM 91] colocam que o segundo nível poderia ser ocupado por profissionais experientes (nível senior) e que o terceiro nível por membros menos experientes (juniors). Essa distribuição favoreceria os juniors a aprenderem com a experiência dos seniors, que desenvolveriam as tarefas mais complexas.

4) Equipes Especialistas

Essas equipes são agrupadas dinamicamente dependendo do tipo de problema a ser resolvido. São equipes onde os participantes fornecem conhecimento específico para a resolução dos problemas. Elas podem entrar na composição de outras equipes, adaptando-se às estruturas organizacionais e de comunicação vigentes no grupo.

4.2.2 Ciclos-de-vida e Métodos

No desenvolvimento dos projetos são empregados variados métodos e ferramentas que são baseados em modelos de ciclo-de-vida. Através do bom uso dessas técnicas é que os produtos de software poderão ter sua qualidade e produtividade garantida. Não deixando de reforçar a idéia de que a capacidade gerencial, dos desenvolvedores e dos usuários envolvidos exercerá influência direta nesses dois aspectos.

4.2.2.1 Modelos de Ciclo-de-vida

O ciclo-de-vida de um produto de software consiste de várias fases transpostas para se chegar ao produto final desejado. Segundo Pressman [PRES 92] essas fases podem ser resumidas em três: definição do sistema, desenvolvimento e manutenção. Elas são encontradas em todo o desenvolvimento de sistemas independente de área de aplicação, tamanho de projeto ou complexidade do sistema.

Entretanto "não existe um modelo único que seja adequado à produção e manutenção de qualquer software. A escolha de um modelo de ciclo-de-vida depende basicamente do propósito do ciclo-de-vida e das características do software a ser produzido" Chitman [CHIT 91]. Logo, as características do modelo de ciclo-de-vida devem adequar-se às características do produto a ser desenvolvido.

Travassos [TRAV 93] enfatiza que a adoção dos modelos de ciclo-de-vida aliados a utilização de métodos de desenvolvimento associados a cada uma de suas etapas, visa aumentar a produtividade, melhorar a qualidade dos produtos, obedecer o orçamento previsto e cumprir prazos de desenvolvimento.

A seguir apresentaremos brevemente alguns modelos de ciclo-de-vida.

♦ Modelo Cascata

Neste modelo de ciclo-de-vida cada etapa só começa quando a anterior termina, sendo que a saída de uma fase é a entrada para a fase seguinte. Segundo Chitman [CHIT 91], o uso deste modelo encoraja a definição dos requisitos antes do projeto e construção dos sistema, encoraja o planejamento de como os componentes irão interagir antes de construí-los, capacita o gerente do projeto a caminhar progressivamente e descobrir possíveis erros antecipadamente.

A figura abaixo descreve um modelo cascata simples:

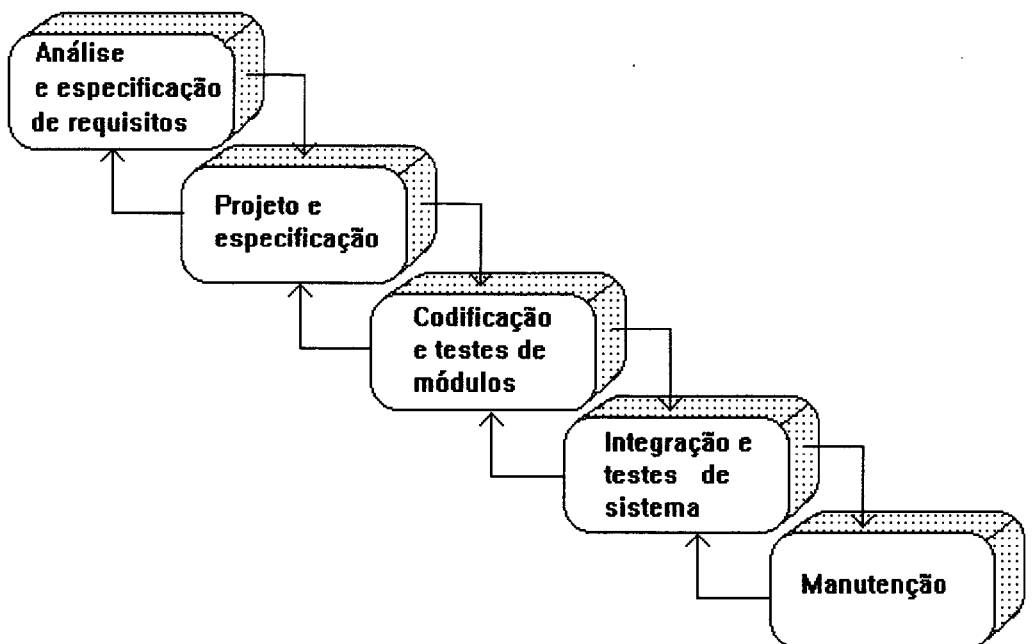


Figura 4.4: Modelo Cascata de ciclo-de-vida baseado em Pressman [PRES 92]

Segundo Bersoff et al. [BEDA 91], as metodologias de engenharia de software estão entrando numa nova fase de evolução. Tradicionalmente o desenvolvimento de produtos de software segue o Modelo em Cascata. Recentemente a indústria de software vem se deparando com problemas que o Modelo Cascata não é capaz de abordar. São problemas complexos onde os requisitos não são totalmente conhecidos no início do processo de desenvolvimento ou estão fluindo constantemente. Para tentar resolver essa situação, a indústria de software vem experimentando novos modelos: Prototipagem Evolutiva, Prototipagem Operacional, Prototipagem Descartável, Reutilização de Software e Síntese Automática de Software.

♦ Modelos de Prototipagem

Prototipagem deve ser útil em desenvolvimento de produtos de software se é capaz de reduzir custos ou riscos associados com seu projeto e implementação. O uso desta abordagem é vantajoso, também, na exploração de questões técnicas. Como os usuários, frequentemente, acham mais fácil manipular um protótipo do que ler especificações, prototipagem facilita a compreensão do problema por pessoas não ligadas à área de informática. De acordo com Bersoff et al. [BEDA 91], a prototipagem se popularizou porque os produtos de software não satisfaziam plenamente as necessidades dos usuários.

⇒ Prototipagem Descartável

Os produtos de software desenvolvidos através da prototipagem descartável usam requisitos considerados válidos e que servem de incentivo para que surjam novos requisitos, ganhar experiência ou descartar aspectos que fujam dos objetivos pretendidos. Com isso espera-se reduzir os riscos de não satisfazer as necessidades dos usuários, reduzir os custos de desenvolvimento, pois há menos modificações durante o desenvolvimento e aumentar a probabilidade de sucesso do projeto devido a maior troca de informações entre desenvolvedores e usuários.

Seus problemas associados situam-se no nível gerencial: dificuldade de considerar o produto pronto e controlar a qualidade dos produtos gerados.

A Prototipagem Descartável atua, principalmente, no início do ciclo-de-vida, como pode ser observado na seguinte representação:

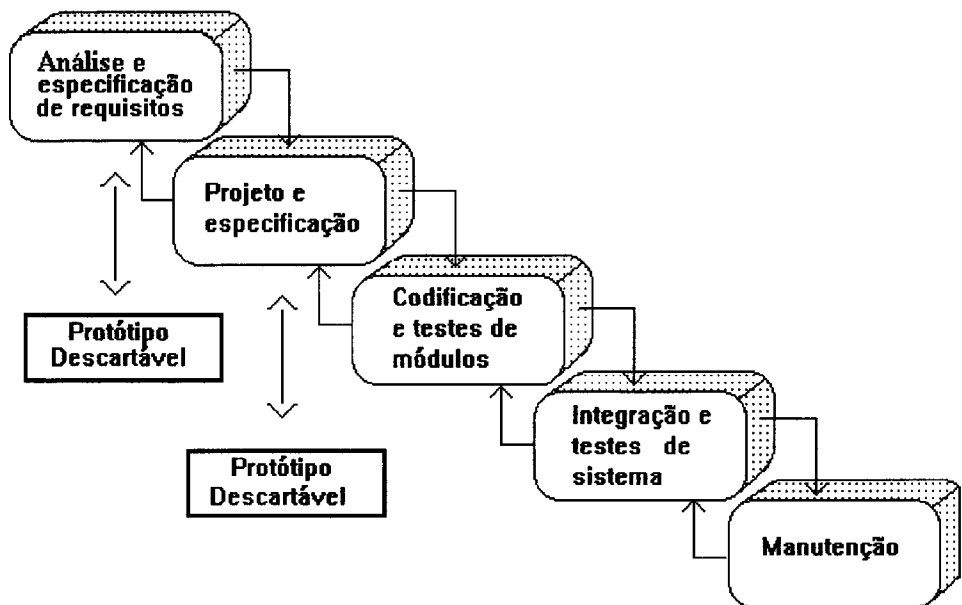


Figura 4.5: Modelo de Prototipagem Descartável [BEDA 91]

⇒ Prototipagem Evolutiva

Prototipagem Evolutiva é um programa de qualidade usado para validar requisitos, ganhar experiência e estimular o afloramento de novos requisitos ou validar um projeto. O protótipo é modificado e desdobrado repetidamente enquanto novas informações são obtidas, reduzindo os riscos da construção de um produto que não satisfaça as necessidades dos usuários e aumentando as probabilidades de sucesso devido a maior comunicação entre usuários e desenvolvedores.

A diferença com a Prototipagem Descartável, é que na Descartável, somente as partes do sistema que não estão bem compreendidas são construídas, já na Prototipagem Evolutiva as partes do sistema que estão bem entendidas são as primeiras a serem construídas, depois o resto pode continuar a ser desenvolvido sobre uma base bem sólida.

Entre os problemas associados destacam-se: desconhecimento sobre os níveis de qualidade e documentação dos protótipos e incorporação de modificações sem que os riscos e o tamanho do software sejam aumentados.

Seu modelo possui a representação abaixo especificada:

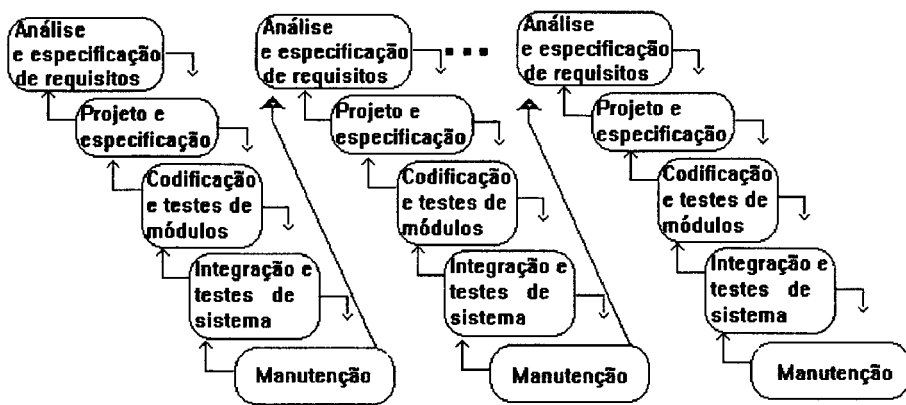


Figura 4.6: Modelo de Prototipagem Evolutiva [BEDA 91]

⇔ Prototipagem Operacional

A Prototipagem Operacional combina o que há de melhor das Prototipagens Evolutiva e Descartável.

Uma base estável é construída usando-se os princípios de engenharia de software, incorporando-se somente as características do software que são bem conhecidas, compreendidas e aceitas, seguindo os primeiros passos da Prototipagem Evolutiva. A versão resultante é submetida aos usuários com o apoio de um prototipador, que será responsável pelas possíveis modificações no sistema. Após as modificações serem realizadas é necessário que os usuários validem-nas. Em seguida, as modificações provenientes dos vários usuários são combinadas gerando um novo desenvolvimento, mais amplo. Essa versão é novamente submetida ao controle dos diferentes usuários e repete-se todo o processo.

Seus problemas associados: encontrar um prototipador capaz de compreender a aplicação e trabalhar em harmonia com os usuários.

Abaixo o descrevemos graficamente:

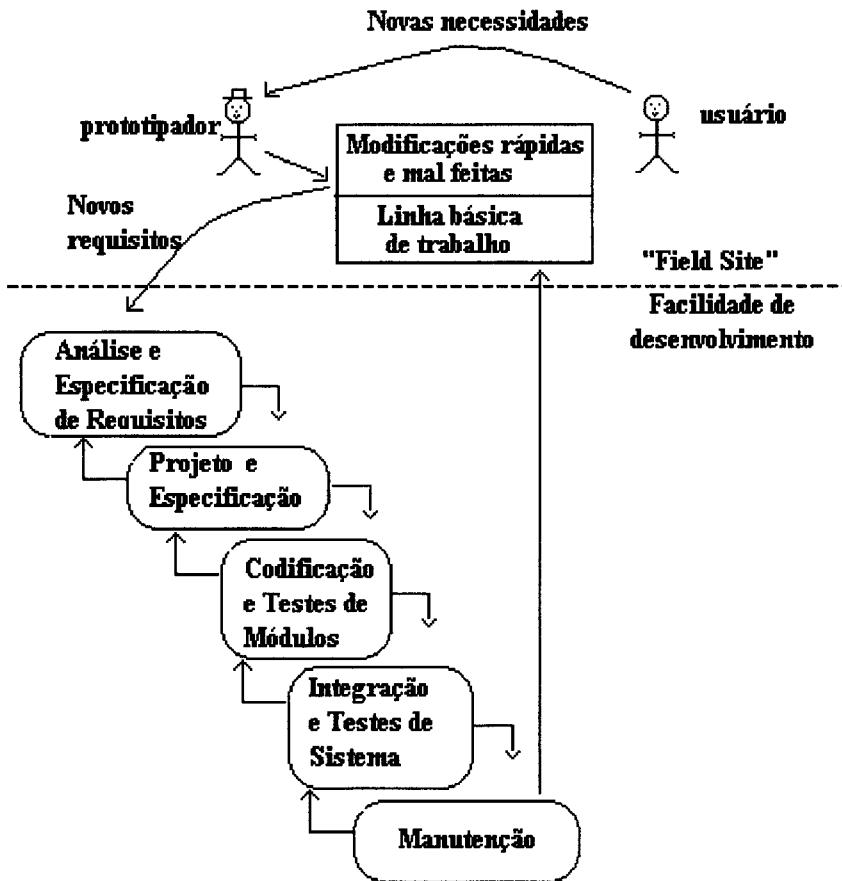


Figura 4.7: Modelo de Prototipagem Operacional baseado em Bersoff et al. [BEDA 91]

♦ Síntese Automática de Software

Síntese Automática de Software significa automatizar o desenvolvimento dos produtos de software, transformar os requisitos ou especificações de projeto de alto nível em código operacional com o apoio de uma ferramenta, que podem ser utilizadas em diferentes níveis do desenvolvimento. As situações em que a Síntese Automática de Software é utilizada são:

⇒ os usuários e/ou desenvolvedores especificam os requisitos e uma ferramenta é usada para transformar os requisitos em software;

⇒ os usuários e/ou desenvolvedores especificam os requisitos e uma ferramenta é usada para gerar automaticamente um projeto de alto nível. Em seguida, o desenvolvimento é concluído através dos meios convencionais.

Seus maiores problemas associados são essencialmente técnicos: Que linguagem deve ser usada ? Os usuários devem ser capazes de desenvolver as especificações?

♦ Modelo Baseado em Reutilização de Software

O enfoque baseado em reutilização de software busca reduzir o custo de desenvolvimento incorporando projetos e codificações anteriores em novos produtos de software. Com isto, obtem-se softwares mais confiáveis, visto que, os componentes já foram avaliados anteriormente. Bersoff et al. [BEDA 91] colocam que podem ser reutilizados códigos testados previamente, projetos aprovados anteriormente, especificações de requisitos ou planos de teste e procedimentos já utilizados. Esse conhecimento prévio do componente não impede que surjam problemas associados à prática de reutilização: falta de conhecimento sobre a potencialidade de reutilização de um componente, falta de catalogação, formas de acesso e como usar esses componentes na construção de sistemas complexos. Existem, também, problemas de controle de entrada e saída dos componentes do repositório. Abaixo temos sua representação gráfica:

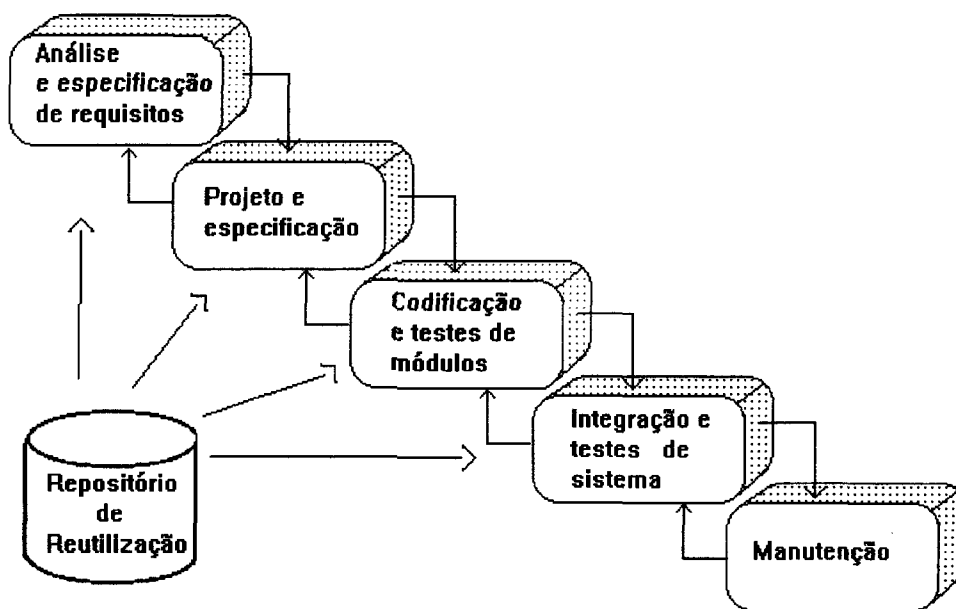


Figura 4.8: Modelo de Reutilização [BEDA 91]

♦ Modelo Espiral

O Modelo Espiral foi desenvolvido baseado em experiências no desenvolvimento de aplicações que utilizavam, principalmente, o Modelo Cascata, e também, visando diminuir os problemas e acrescentar vantagens aos modelos de ciclo-de-vida. A principal característica do Modelo Espiral é ser cíclico e não linear.

Segundo Boehn [BOEH 91] cada ciclo da espiral é constituída de quatro estágios que começam com a identificação dos objetivos da parte do produto que será elaborada, das alternativas para a implementação dessa parte e dos problemas impostos na aplicação dessas alternativas. O próximo passo é avaliar as alternativas relativas aos objetivos e aos problemas eliminando as fontes de risco. Depois, formula-se estratégias para resolver as fontes de risco. A próxima etapa é desenvolver um protótipo que servirá de base para a futura evolução do produto seguindo a abordagem básica do Modelo Cascata modificado para incorporar o desenvolvimento incremental segundo a figura abaixo:

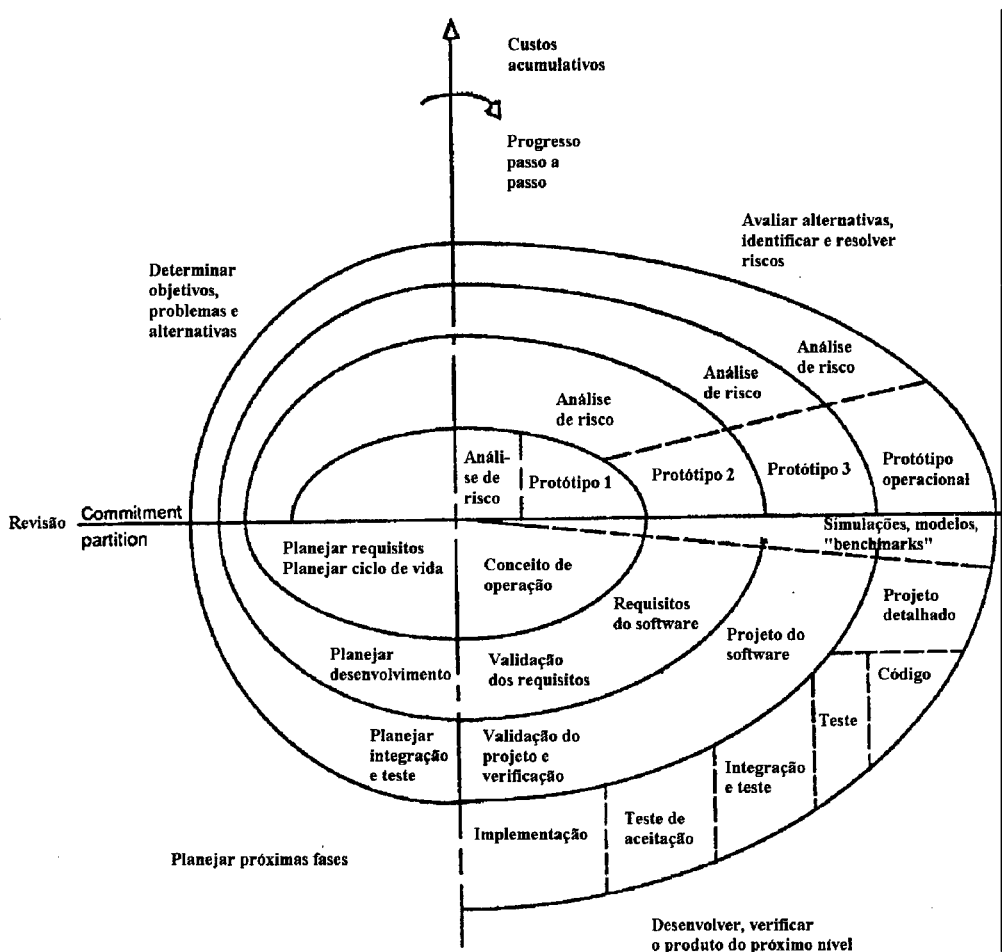


Figura 4.9: Ciclo-de-vida em Espiral [BOEH 91]

Na figura percebe-se que cada nível de especificação do software é seguido por uma fase de validação e da preparação de planos para o ciclo seguinte. Cada ciclo é completado por uma revisão envolvendo as principais pessoas interessadas no produto, assegurando o pleno acordo para o prosseguimento na fase seguinte.

A principal vantagem do Modelo Espiral, segundo Boehm [BOEH 91] é sua gama de opções abarcar as melhores características dos modelos de ciclo-de-vida existentes, evitando suas maiores dificuldades.

É interessante que os engenheiros de software e os gerentes conheçam esse modelo como forma de ajudar a controlar custos, melhorar o apoio aos usuários e aumentar a qualidade dos produtos gerados.

Os modelos acima expostos são a base dos métodos e ferramentas de desenvolvimento de software.

4.3.2.2 Métodos e Ferramentas

Um breve exame na literatura nos mostra a existência de vários métodos e ferramentas que apoiam e aprimoram o processo de desenvolvimento de software. Entretanto, várias questões relativas à produção de produtos de software ainda persistem. Uma dessas questões é a que se refere à não existência de um método universal que possa ser usado satisfatoriamente em todas as situações. Ao contrário, existem métodos mais adequados para certos tipos de problemas com características específicas, logo, é necessário a escolha daquele que melhor se adapte a cada caso particular, levando em consideração a cultura da organização, os conhecimentos prévios e as preferências da equipe desenvolvedora.

Crispim [CRIS 91] ressalta que o uso de métodos facilita a comunicação entre desenvolvedores e usuários, auxilia no entendimento e na decomposição do produto, torna mais fácil a definição dos limites do problema e provê recursos para elaborar uma solução viável e satisfatória.

Os métodos fornecem técnicas, instrumentos e ferramentas que viabilizam a realização das atividades definidas no modelo do ciclo-de-vida. As técnicas são conjuntos de princípios para a execução de uma tarefa específica do processo de desenvolvimento de software. Em Rocha et al. [ROAB 87] encontramos a seguinte classificação:

- ◆ Técnicas construtivas: guiam o processo de construção de software, oferecendo meios para que ele possa ser construído de forma disciplinada;

- Técnicas normativas: estabelecem normas e atributos de qualidade que guiam a construção de um software;

- Técnicas gerenciais: oferecem meios para planejar e controlar o processo de desenvolvimento de um software.

Os instrumentos tornam possível a utilização de um método. Rocha et al. [ROAB 87] classifica-os em:

- ★ Instrumentos para geração: dão apoio às técnicas construtivas;

- ★ Instrumentos para avaliação da qualidade: dão apoio às técnicas normativas e tornam possível o uso de métodos para avaliação da qualidade;

- ★ Instrumentos para apoio à gerência: viabilizam a utilização de técnicas gerenciais, tornando possível o planejamento e o controle do processo de desenvolvimento de um produto.

Para que esses instrumentos e técnicas sejam utilizados de forma produtiva é interessante o uso de ferramentas de apoio ao método. Na seleção de ferramentas deve-se levar em consideração a adequação ao método, a experiência prévia e a preferência da equipe, a relação custo-benefício em caso de compra e o suporte dos desenvolvedores.

As ferramentas Case (Computer Aided Software Engineering) fornecem um ambiente de software para ajudar o desenvolvimento de produtos. Elas minimizam os obstáculos comumente encontrados durante o desenvolvimento dos produtos e aumentam o empenho no uso de metodologias. Geralmente as ferramentas CASE apoiam uma metodologia específica e segundo Pressman [PRES 92], são mais importantes que o hardware para alcançar um aumento de qualidade e de produtividade.

Os benefícios que o uso de ferramentas CASE nos aportam são:

- ⇒ sistemas com maior qualidade;

- ⇒ menor quantidade de manutenção;

- ⇒ melhor documentação;

- ⇒ comunicação mais clara com o usuário;

- ⇒ aceleração do ciclo de desenvolvimento;

- ⇒ aumento da satisfação do usuário.

Dentre a gama de métodos existentes destacamos os seguintes, por serem mais utilizados e conhecidos:

- Essential System Analysis (Análise Essencial de Sistemas), McMenamin e Palmer [MCPA 91];
- Jackson System Development (Método de Jackson), Cameron [CAME 83];
- Object-Oriented System Analysis (Análise e Projeto Orientado a Objetos), Shlaer [SHME 88];
- Structured Analysis and Design Technique - SADT (Análise Estruturada e Técnica de Projeto) Ross [ROSS 85];
- Structures System Analysis (Análise Estruturada de Sistemas), Demarco [DEMA 78];
- Structured Design (Projeto Estruturado), Stevens [STEV 88].

Esses métodos detalham as atividades, especificando regras de construção de diagramas, linguagens, conjunto de objetos, cenários, etc. Entretanto, numa análise mais detalhada, não percebemos preocupações com a organização do trabalho humano ou com problemas de relacionamento dentro da equipe e com relação ao usuário. Essa questão leva a equipes não democráticas, onde a participação de cada um é restrita pela dificuldade do método e das ferramentas em apoiar o trabalho de equipes. Logo, é interessante focar os esforços que vêm sendo empreendidos no sentido de fornecer apoio técnico e teórico ao trabalho dessas equipes, integrando tarefas, facilitando a comunicação, buscando a satisfação da equipe e de usuários, gerando produtos confiáveis e alcançando a verdadeira cooperação.

4.3 O Processo de desenvolvimento de software num enfoque de Trabalho Cooperativo

Ultimamente as pesquisas na área de Engenharia de Software buscam encontrar metodologias de desenvolvimento de produtos de software que valorizem e apoiem a participação dos diferentes usuários.

As atividades dos membros de uma equipe necessitam ter o suporte de equipamentos e ferramentas que facilitem a interação do grupo através de técnicas colaborativas, tais como: negociações e compromissos. Todos os esforços devem se concentrar para que a interação atinja todos os níveis hierárquicos e que através dela advenha a cooperação.

Essa idéia é reforçada pelos engenheiros de software que participaram do workshop "Computer Science and Technology Board" [CSTB 90]. Segundo foi colocado, o desenvolvimento de sistemas é um exercício de colaboração e se faz necessário maximizar sua eficiência através de metodologias adequadas, assim como, o apoio de ferramentas apropriadas contribuiria para um aumento da produtividade e da qualidade dos produtos.

Mais recentemente, têm sido desenvolvidas algumas teorias que possuem um ponto de vista menos centrado em técnicas e mais focado nos fatores humanos inerentes aos processos de trabalho. Algumas delas são provenientes das países escandinavos onde a pesquisa na área da cooperação é mais antiga e mais profunda. Outras surgiram nos EUA, onde os aspectos técnicos permanecem muito valorizados e explorados, mas já possuem algumas influências escandinavas.

A interação dos participantes de um grupo que visa alcançar um objetivo comum é uma relação rica em conteúdos e idéias, mas difícil de ser vivenciada. Como já foi discutido anteriormente, essa interação gera conflitos que necessitam ser solucionados. Nesta seção discutiremos os conflitos e negociações, particularizando-os para o caso da Engenharia de Software. Apresentaremos algumas abordagens que estimulam uma maior democracia no ambiente de trabalho e quatro métodos de desenvolvimento de produtos de software que apoiam a participação ativa do usuário. Em seguida comparamos esses métodos e apresentamos os resultados num quadro comparativo. Finalmente discute-se o apoio automatizado ao desenvolvimento cooperativo de software, destacando os pontos que devem receber maior atenção de pesquisadores.

4.3.1 Conflitos e negociações no desenvolvimento de produtos de software

O processo de desenvolvimento cooperativo de projetos é um processo político que envolve conflitos em quase todos os seus passos, onde, os conflitos entre a equipe desenvolvedora versus usuários e os conflitos internos da equipe são distintos:

⇒ os conflitos da equipe com o usuário possuem mais a forma de conflito funcional, pois o objetivo do usuário é obter um produto final que preencha todas as suas expectativas. Já o objetivo da equipe é desenvolver esse produto da forma mais eficiente possível, para que não haja necessidade de muitas modificações depois de sua implementação.

⇒ os conflitos desencadeados dentro da equipe, podem tomar conotações de conflitos não-funcionais, onde um membro buscaria mais poder, ou haveria a formação de subgrupos cada um tentando impor suas idéias e formas de trabalho, destruindo dessa forma a coesão necessária para que os trabalhos ocorram de forma cooperativa.

É interessante acentuar a importância da participação das pessoas que serão realmente afetadas pelo projeto, e não a participação de representantes dos usuários, se bem que existam casos em que o elevado número de usuários não permite a participação de todos. Nesses casos, deve-se tentar escolher membros que realmente estejam bem integrados com seus companheiros e que saibam representá-los.

As negociações para resolução de conflitos dentro de equipes de desenvolvimento de produtos de software podem se dar seguindo as técnicas propostas no Capítulo 2, ou seja: inclusão de um "terceiro participante", estabelecimento de um objetivo superior, votação, alteração das variáveis estruturais do grupo. Mas como também já foi colocado, o mais positivo é resolver os conflitos dentro da própria equipe, sem envolver novas pessoas ou objetivos.

As negociações entre a equipe desenvolvedora e os usuários devem permitir um maior contato entre as partes envolvidas a fim de que uns possam melhor compreender os argumentos dos outros, já que as áreas de domínio de conhecimento são, geralmente, distintas. Algumas vezes o conflito surge a partir da não compreensão de algum ponto específico do projeto em desenvolvimento. Esse problema pode ser amenizado, segundo é colocado por Greenbaum et al. [GRKY 91], com o total envolvimento do usuário integrando-o em todas as partes do desenvolvimento do projeto. A mudança do enfoque do

processo de desenvolvimento do projeto, de uma atividade de descrição, para ações cooperativas com o uso de protótipos, também ajudará a reforçar a coesão do grupo. É importante que o usuário não seja visto como um objeto de estudo, mas sim, como um agente ativo dentro de todo o processo de desenvolvimento.

4.3.2 Novas abordagens de desenvolvimento de software

Com o enfoque tradicional de desenvolvimento de sistemas sendo objeto de críticas, alguns pesquisadores buscaram opções que solucionassem a chamada "crise do software", ou seja, que o software satisfizesse, plenamente, seus usuários.

Segundo Munford [MUNF 88] a repercussão das idéias democráticas na filosofia dos sistemas pode ser considerável, pois em lugar de relacionar a prática do projeto com uma série de objetivos predominantemente técnicos, valoriza a participação das pessoas e grupos diretamente afetados pela implantação do software. A autora salienta que o projeto cooperativo de sistemas exige um sistema de valores humanistas: uma metodologia que facilite o projeto de sistemas humanos satisfatórios e uma estrutura democrática que permita uma participação eficaz, sendo esses os meios que conduzem ao objetivo da cooperação. Ela baseou seus estudos em casos vivenciados na prática a partir de 1972, onde os grupos interessados no resultado do projeto deveriam intervir no processo. De suas idéias e experiências, surgiram três modelos de participação:

⇒ Projeto Consultivo: o usuário atua apenas como consultor do grupo de desenvolvimento;

⇒ Projeto Representativo: são escolhidos "representantes" dos diversos grupos de usuários;

⇒ Projeto Consensual: o nível de participação dos usuários amplifica-se e alcança um nível mais profundo de participação, com o envolvimento de todos os usuários no processo de desenvolvimento do projeto. Os usuários não participam diretamente no projeto, pois continua a existir o papel do representante. A diferença com o projeto representativo situa-se no nível de interação entre o representante e o grupo representado. Neste caso o representante não tem poder para decidir sozinho e deve consultar continuamente seus colegas.

"Apesar de todas as dificuldades, o usuário deve pretender, e tem o direito de dispor de um sistema que permita a quem o utilize, usá-lo com propriedade,

compreendendo e controlando sua estratégia de elaboração e também, o significado das operações executadas", Mussio [MUSS 87].

Segundo Moysés [MOLI 93] embora muita atenção esteja começando a ser dada às necessidades do usuário na literatura, a maioria tenta propor soluções de como melhor "integrar o usuário" no processo de desenvolvimento dos sistemas. O enfoque escandinavo se distingue claramente desses outros enfoques ao considerar que "diversos grupos chamados usuários podem aprender, participar, cooperar ativamente com os projetistas dos sistemas". O interesse aqui não é "encaixar" usuários em um processo já existente de desenvolvimento, mas de criar novas maneiras de trabalho conjunto. Ao contrário do enfoque tradicional a participação não significa simplesmente entrevistar o interessado no início e fazê-lo verificar e aprovar o projeto, mas sim, envolvê-lo ativamente no processo criativo de todo o desenvolvimento do produto de software. Dessa integração advém uma troca de informações e experiências: a equipe desenvolvedora aprende sobre a área de aplicação e os usuários podem aprender sobre a área de desenvolvimento. Esse processo de trocas de experiências é chamado "aprendizado mútuo" e é fortemente defendido pela escola escandinava, King [KING 91].

Greenbaum et al. [GRKY 91] apresentam de forma bem simplificada uma figura, onde contrastam aspectos do enfoque tradicional de desenvolvimento de projetos com o enfoque cooperativo que é apresentado por eles no livro "Design at Work":

Abordagem Tradicional Foco em:	Abordagem Cooperativa Foco em:
<ul style="list-style-type: none"> • problemas • fluxo de informações • tarefas • habilidades formais • regras • indivíduo • regras baseadas em procedimentos 	<ul style="list-style-type: none"> • situações e desdobramentos • relações sociais • conhecimento • habilidades inerentes • competência mútua • interação grupal • trabalho baseado na experiência

Figura 4.10: Comparação entre a abordagem tradicional de desenvolvimento de produtos de software e a abordagem cooperativa [GRKY 91].

A seguir veremos mais detalhadamente algumas dessas novas propostas: Projeto Participativo [CAWG 93], JAD [CAWG 93], Teoria-W [BORO 89] e SADT Aumentado [MARC 91], estabelecendo algumas comparações, de forma a facilitar a visualização de seus aspectos comuns e distinções.

4.3.2.1 Projeto Participativo

Projeto Participativo (PP) é classificado por Carmel et al. [CAWG 93] como sendo um projeto de consenso, onde os usuários são continuamente envolvidos no processo de desenvolvimento dos projetos.

Dois temas centrais governam a implementação dos princípios de PP, são eles:

1º) Aprendizado mútuo - usuários e projetistas aprenderão uns com os outros, aspectos de suas práticas de trabalho e possibilidades técnicas. Os usuários serão treinados e familiarizados com o computador e os projetistas conhecerão mais profundamente detalhes sobre a área de trabalho dos usuários.

2º) Projetar fazendo - A modelagem e os testes são baseados na experimentação prática. A criatividade de todos gera novas situações e soluções inovadoras.

O PP não é baseado em altas tecnologias de comunicação, mas sim, em meios e instrumentos já familiares para a maioria das pessoas, tais como, quadro-negro e painéis de recados.

Algumas técnicas aplicadas em PP que foram descritas por Carmel et al. [CAWG 93] serão expostas a seguir:

1) Formulação de Especificações e Modelos

Os usuários são estimulados a participarem de todo o processo de desenvolvimento e os projetistas são incentivados a experimentarem a prática de trabalho dos usuários envolvidos no projeto.

Os usuários visitam organizações que possuem tecnologias mais avançadas para que eles possam vislumbrar novas alternativas que se encaixem no processo de modernização de suas ocupações profissionais, e os projetistas participam do dia-a-dia dos usuários para que as especificações e os modelos possam se aproximar ao máximo da realidade, para conhecer e entender as suas funções e seus postos de trabalho.

2) Prototipagem Cooperativa

A prototipagem cooperativa visa estabelecer um processo de desenvolvimento onde usuários e equipe desenvolvedora participam ativamente e criativamente baseados em suas diferentes qualificações. Bodker et al. [BOGR 91] colocam que a proposta de Prototipagem Cooperativa deve envolver o usuário mais do que a prototipagem tradicional que se apoia, principalmente, na perspectiva dos projetistas e engenheiros, e os usuários têm contato com o produto quando as principais idéias já foram exploradas e possíveis alterações requerem muita negociação, ficando as modificações sujeitas à boa-vontade da equipe desenvolvedora.

No enfoque cooperativo os processos de análise e projeto são rapidamente substituídos pelo desenvolvimento de um protótipo que dará aos participantes uma visão concreta de suas principais idéias, requerendo que as habilidades dos usuários sejam confrontadas com as novas possibilidades técnicas. O ideal é que a prototipagem cooperativa seja desenvolvida por um pequeno grupo de projetistas e usuários com acesso a ferramentas computacionais flexíveis para que o desenvolvimento e as modificações do protótipo sejam rápidas. Cada sessão necessita ser preparada de acordo com o estágio atual do processo de projeto. Em alguns casos o projetista necessita intervir nas sessões de prototipagem quando percebe que as discussões estão fugindo do ponto central do produto.

O usuário, por sua vez, deve ter algum tempo para usar o produto computacional na prática do seu dia-a-dia de trabalho. Quando o usuário propõe modificações, elas são amplamente discutidas e avaliadas, sendo que em alguns casos as modificações são introduzidas pelo próprio usuário, já familiarizado com ferramentas que facilitam a prática do desenvolvimento contínuo, tais como: Hypercard e ORACLE.

Bodker et al. [BOG2 91] assinalam que prototipagem é um processo de aprendizagem. As informações geradas a partir da manipulação do protótipo devem ser difundidas entre os funcionários e gerentes que não tenham participado diretamente do processo de prototipagem. Em geral os protótipos são valiosos instrumentos de educação e treinamento de futuros usuários, e similarmente os participantes de um processo de prototipagem estão aptos a agirem como professores.

Um aspecto que deve ser considerado é que nem todos os softwares são interessantes de serem prototipados. Pressman [PRES 92] coloca que as aplicações mais vantajosas são aquelas que criam imagens dinâmicas, interagem fortemente com o usuário ou necessitam de algoritmos que devem ser desenvolvidos evolutivamente. Segundo observou Carneiro [CARN 93], o uso de prototipação tem facilitado a comunicação entre usuários e a equipe desenvolvedora, principalmente durante as fases iniciais de um projeto.

A implantação de PP se depara com vários obstáculos tais como: a relutância dos executivos e profissionais ligados aos projetos de aumentar a participação do usuário e da introdução desses novos métodos e técnicas.

Como em PP os usuários e os analistas desenvolvem novos papéis, diferentes daqueles assumidos no desenvolvimento tradicional, também surgem dificuldades que a princípio parecem simples, mas que possuem importância fundamental no desenvolvimento dos trabalhos. Bjerknes [BJER 93] cita algumas soluções encontradas no desenvolvimento do projeto Florence (1983-1987), na Noruega. Ele enfatiza que algumas vezes os usuários não dedicam tempo suficiente para o desenvolvimento do projeto. A fim de evitar esse desequilíbrio é interessante especificar um contrato onde constará quantas horas por semana serão dedicadas ao projeto. Na resolução dos conflitos, o autor propõe um grupo responsável em intermediar e guiar a resolução dos mesmos. E mais uma vez ressalta a importância em se ouvir os usuários.

Projeto Participativo é frequentemente nomeado como a "abordagem escandinava" para o desenvolvimento de sistemas. Nos EUA esta abordagem está começando a receber alguma atenção. Mas ao mesmo tempo surgem questionamentos sobre os contextos socio-políticos empregados na escandinávia e como eles poderiam ser empregados nos EUA. Um aspecto é apontado como principal barreira para sua adoção: a falta de métodos ou modelos. Como consequência os projetistas baseiam-se somente na experiência própria, tornando PP uma teoria muito idealista.

4.3.2.2 JAD (Joint Application Design)

JAD é considerado por Carmel et al. [CAWG 93] como uma metodologia com um alto nível de envolvimento dos usuários e como a solução para a obtenção de produtos mais confiáveis e com maior qualidade. Esses aspectos contribuem para que nos EUA essa metodologia seja a mais difundida dentre aquelas que envolvem ativamente o usuário. Foi originada na IBM nos anos 70 e somente alguns anos mais tarde recebeu atenção da indústria.

JAD não possui grande base teórica, o que é considerado como principal causa da comunidade acadêmica não se interessar por essa metodologia. A metodologia das reuniões foi influenciada pela dinâmica de grupo e pelo estudo de como os trabalhos se desenvolvem dentro do grupo. Os encontros são estruturados, disciplinados e o

desenvolvimento de sistemas é baseado em um modelo de ciclo-de-vida. Com isso tem-se um aumento da qualidade, redução de custos e redução do tempo de desenvolvimento.

Segundo Carmel et al. [CAWG 93], JAD pode ser visto como uma metodologia e como uma técnica. É uma metodologia porque quando aplicada no desenvolvimento de sistemas baseados em ciclo-de-vida forma o centro em volta do qual todas as atividades se desenvolvem, e é uma técnica, por possuir uma estrutura capaz de conduzir uma reunião com o envolvimento dos usuários.

Existem algumas estruturas básicas que são ressaltadas como fundamentais para que o JAD se desenvolva plenamente:

- Coordenação - um gerente ou um líder coordena as reuniões, (há evidências que esse líder é a chave do sucesso da metodologia);
- ♦ Estruturação - a reunião deve possuir um plano de ação, que guie as discussões;
- Documentação - um dos participantes deve ser responsável pela documentação das reuniões. Essas anotações devem ser mantidas rigorosamente atualizadas.
- Dinâmica de grupo - Devem existir alguns protocolos de conversação, resolução de conflitos, mas que não impeçam o fluxo de idéias.

Existem algumas definições de papéis dos membros participantes dos encontros:

⇒ Usuário: são as pessoas diretamente afetadas pelos produtos em desenvolvimento. Nos EUA se incluem os operadores bem como os gerentes;

⇒ Responsável Executivo: é aquele que define os propósitos iniciais do projeto e os direcionamentos principais;

⇒ Líder/coordenador: guia as discussões fazendo interrupções quando necessário. É uma pessoa treinada para tal, com conhecimentos em dinâmica de grupo e metodologias de desenvolvimento de produtos de software. É o responsável pela agenda, documentação e encaminhamento de discussões;

⇒ Escriba: é o responsável pelas anotações de decisões e resultados obtidos durante as reuniões;

⇒ Equipe de projeto: é o pessoal técnico envolvido: analistas, gerentes de projeto e especialistas.

A conduta nos encontros de JAD é diferente para cada nível de desenvolvimento do ciclo-de-vida. Nos primeiros encontros as questões são mais complexas e amplas: definir objetivos, decompor o problema principal em pequenos problemas, definir fronteiras, é o que hoje é comumente denominado "Análise de Negócio". Nessas reuniões os participantes definem a organização das tarefas e a alocação de pessoal juntamente com os prazos. Existem painéis nos quais são mantidas listas e textos. Os coordenadores estimulam os participantes a andarem pela sala a fim de complementarem esses textos e listas com suas contribuições pessoais.

Quando os encontros chegam na fase de projeto, os aspectos levantados na fase inicial são aprofundados. Os encontros passam a ter maior duração, (por volta de 3 a 5 dias), enquanto que nas fases iniciais esses encontros são mais breves, por volta de 1 a 2 dias.

Hoje algumas partes desses encontros podem ser apoiadas em ferramentas CASE e alguns groupware, mas há resistências quanto a entrada dessas novas tecnologias como forma de manter as reuniões o mais simples possível.

Como acontece em PP, a introdução de JAD provoca resistências gerenciais, administrativas, e até mesmo, alguns usuários tendem a não aceitá-la facilmente.

4.3.2.3 SADT Aumentado

Originalmente SADT (Structured Analysis and Design Technique) foi criado para descrever sistemas e suas operações. Segundo Marca [MARC 91], SADT é uma linguagem e um processo para análise de sistemas. A linguagem de caixas e setas facilita o trabalho dos analistas no desenho de diagramas de atividades, informações e componentes físicos de um sistema. O SADT facilita a especificação de requisitos dos usuários, assim como, a revisão e validação dos documentos gerados nessa fase.

Marca [MARC 91] ressalta que o uso de SADT na modelagem de aspectos cooperativos é totalmente plausível, pois SADT possui mecanismos descritivos que podem ser facilmente compreendidos, já que utiliza uma linguagem gráfica. Mas os métodos tradicionais de desenvolvimento de produtos de software não podem, simplesmente, serem aplicados no desenvolvimento cooperativo. Eles precisam ser modificados de acordo com os aspectos sociais envolvidos.

No SADT tradicional existem três aspectos metodológicos que não são adequados para a sua aplicação em grupos de trabalho:

1º) Entrevista com o Usuário - no método tradicional o usuário descreve as tarefas que ele desenvolve em seu ambiente de trabalho. Para um melhor entendimento das atividades dos usuários é mais interessante para a equipe desenvolvedora ir ao local de trabalho do usuário e participar do desenvolvimento de suas tarefas, possibilitando à equipe uma compreensão mais global da gama de atividades e das interações envolvidas.

2º) Entrevistas Individuais - tradicionalmente as técnicas de entrevistas enfatizam o questionamento individual. O mais proveitoso seria entrevistar toda a equipe de usuários ao mesmo tempo, a fim de se perceber a divisão de tarefas e o próprio desenvolvimento dos trabalhos dentro da equipe. O trabalho é descrito a partir de uma visão individual, mas recebe também, as perspectivas de todo o grupo.

3º) Limitação dos Requisitos Iniciais - os requisitos costumam ser especificados em documentos que são escritos e revistos usando-se o ciclo autor/leitor do SADT. Essas especificações são úteis para delinear as necessidades iniciais do sistema pretendido, mas acabam se tornando definitivas sem dar oportunidades ao usuário de propor eventuais modificações.

O mais apropriado, nesse caso, é o uso de um protótipo simples que possa ser utilizado pelos futuros usuários definitivos. Com isso obtém-se a validação dos requisitos iniciais e a correção de requisitos inválidos.

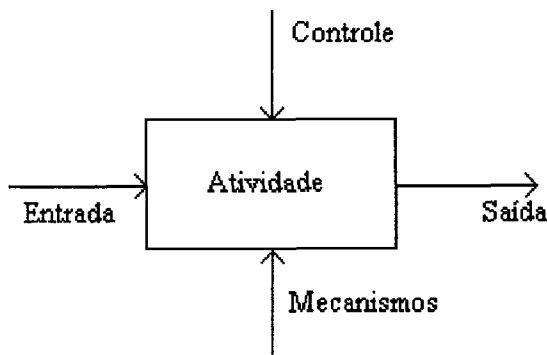
O SADT foi aumentado baseado no paradigma proposto por Winograd e Flores [WIFL 86], que sugere que o trabalho se desenvolva através de conversações em que são estabelecidos requisitos e compromissos. Através da realização dos compromissos assumidos e do surgimento de novos requisitos são criados contextos e o trabalho evolui, alcançando os objetivos estabelecidos pelos participantes do grupo.

Na perspectiva do trabalho ser desenvolvido através de conversação, os símbolos gráficos do SADT recebem outras interpretações, conforme o quadro abaixo:

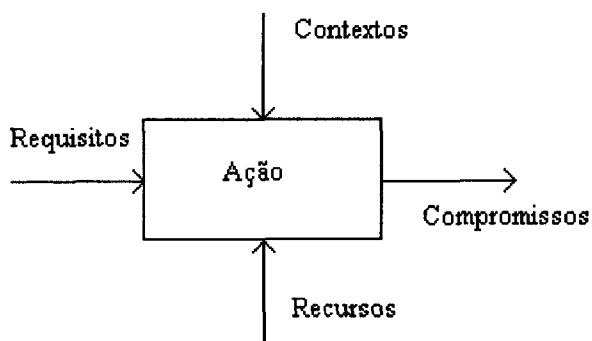
Símbolos Gráficos do SADT	Trabalho Individual	Trabalho em Equipe
Caixas	Tarefas	Ação
Flechas de entrada e saída	Uso ou geração de informações	Requisitos e compromissos
Flechas de controle	Guia de linhas e regras	Contextos e conversação
Flechas de mecanismos	Recursos humanos	Indivíduos, grupos e organizações

Figura 4.11: Interpretações dos símbolos gráficos do SADT.

Logo, o gráfico tradicional:



transforma-se em:



Essa comparação demonstra claramente a mudança do enfoque do trabalho individual baseado em tarefas e informações, para o enfoque do trabalho cooperativo onde o trabalho é baseado na conversação e ação.

Essas duas visões podem ser resumidas e comparadas a seguir:

Modelo de características do SADT	Trabalho Individual	Trabalho em Equipe
Conteúdo	Tarefas e Informação	Manipulação dos contextos através de conversação e ações
Propostas	Perceber como as entradas geram saídas	Definir o que as pessoas falam e fazem nos contextos
Precisão	Nível de detalhamento das tarefas e informações	Nível de detalhamento das conversações
Fronteiras	Interfaces do sistema	Interfaces organizacionais

Figura 4.12: Modelo das características do SADT [MARC 91].

Um modelo baseado nessas características poderá apoiar pessoas trabalhando cooperativamente através de conversações e ações. Uma perspectiva de linguagem-ação focaliza as interações sociais do espaço de trabalho e considera a informação no contexto dessas interações.

Esse modelo de SADT Aumentado foi desenvolvido com o propósito de especificar os requisitos necessários para o desenvolvimento de um sistema de apoio ao trabalho cooperativo. Segundo Marca [MARC 91] o Projeto CONTRACT começou a partir da seguinte questão: "Como especificações devem ser definidas para que o software resultante seja capaz de apoiar o trabalho cooperativo possuindo trocas rápidas de informações e complexas interações humanas?". Esse projeto vislumbrava um método cooperativo para o desenvolvimento de um software capaz de apoiar o trabalho cooperativo.

SADT Aumentado possui uma arquitetura que engloba os principais elementos da cooperação e os divide em três grupos, facilitando sua identificação e separação:

1) Modelo de Cooperação - Descreve os contextos para o grupo e as conversações que o grupo realiza quando uma determinada situação se apresenta. Para

desenvolver esse modelo o analista deve permanecer com os usuários observando seus trabalhos, de modo a poder entender e definir o contexto desses trabalhos e os resultados provenientes das conversações.

2) Modelo de Conversação - Define o conteúdo de cada conversação relevante no desenvolvimento dos trabalhos. Esse modelo é criado usando-se as flechas dos contextos criados no modelo de cooperação, que serão detalhadas em tipos de conversação.

3) Modelo de Negociação - Define um método para manter atualizadas as informações de cada conversação que é negociada. Essas conversações se desenvolvem entre duas pessoas. Nesses encontros ocorrerá o levantamento de requisitos, compromissos, rejeições, aceitações, negociações, que apoiarão o desenvolvimento dos conteúdos definidos no modelo de conversação.

A metodologia SADT Aumentada foi experimentada por um grupo de analistas envolvidos no Projeto CONTRACT que apontaram algumas vantagens:

- ⇒ levantar os requisitos fica muito mais fácil e simples quando o trabalho dos usuários é vivenciado diretamente pelos projetistas;
- ⇒ as sessões de autoria cooperativa entre usuários e projetistas diminuem o tempo de validação de requisitos, pois os processos são alterados e as alterações confirmadas ou não, imediatamente, pelos usuários;
- ⇒ poder ser empregada com métodos de desenvolvimento orientados à objetos, pois possui princípios encontrados na programação orientada à objetos, onde o tipo de conversação é o objeto e o protocolo é o método para esse objeto.

Já os usuários, no desenvolvimento do mesmo projeto, apontaram como pontos positivos:

- ⇒ facilitar a rápida distinção e avaliação dos requisitos e propostas que serão positivas para o desenvolvimento dos trabalhos;
- ⇒ proporcionar rápida determinação dos custos de uma renegociação de compromissos;
- ⇒ guardar as decisões tomadas numa determinada reunião.

Essas vantagens apontadas tanto por analistas como por usuários no desenvolvimento do Projeto Contract indicam que o SADT Aumentado é um método cooperativo que valoriza as relações sociais envolvidas no trabalho.

4.3.2.4 Teoria-W

Os problemas de gerência de um projeto são inúmeros e envolvem relações humanas, aspectos econômicos e tecnológicos. Boehm et al. [BORO 89] assinalam que a dificuldade principal é satisfazer as expectativas de todas as diferentes pessoas envolvidas no desenvolvimento dos projetos de produtos de software, ou seja: usuários, equipe desenvolvedora, gerência, equipe de manutenção, chefia das organizações, etc.

A Teoria-W foi desenvolvida por Boehm et al. [BORO 89] e se propõe a evitar os problemas mais comuns encontrados no desenvolvimento de produtos de software e tem como princípio fundamental: "Tornar vencedoras todas as partes envolvidas no projeto", ou seja, satisfazer todas as pessoas envolvidas no projeto. Nesta teoria, o gerente é caracterizado como o negociador, como aquele que intermediará os conflitos de forma a encontrar soluções que satisfaçam à todos.

A Teoria-W apoia-se em dois princípios básicos:

- Planejar o "vôo" e "voar" de acordo com o plano;
- Identificar e gerenciar os riscos.

Os planos são importantes na Teoria-W, pois refletirão um compromisso mútuo estabelecido pelas várias partes envolvidas. Esses planos são organizados através de questões simples e objetivas:

- 1) Objetivos (**Por que** desenvolver tal atividade ?);
- 2) Produtos (**O que** será desenvolvido ?);
- 3) Responsabilidades (**Quem** é o responsável por cada resultado ?) e (**Onde** estão eles organizacionalmente lotados ?);
- 4) Abordagem (**Como** cada resultado será alcançado?);
- 5) Recursos (**Quanto** (financeiramente) será necessário para alcançar os resultados ?).

Segundo Boehm et al [BORO 89] a Teoria-W faz uma expansão de quatro abordagens de negociação que foram apresentadas por Fisher et al. [FIUR 81] em 1981:

- 1) Separar as pessoas dos projetos;
- 2) Focalizar os interesses, não as posições;
- 3) Inventar opções para que os ganhos sejam mútuos;
- 4) Insistir no uso de critérios objetivos.

Dessas quatro abordagens derivam-se os princípios que se seguem:

1) Compreender como as pessoas querem vencer, o que implica em:

- ⇒ Identificar as pessoas chave;
- ⇒ Se projetar em outras situações de vitória;
- ⇒ Aproximar-se dos usuários através de entrevistas, protótipos, análises operacionais, compreensão da experiência prévia que eles possuem com a automação.

2) Estabelecer expectativas razoáveis, o que implica em:

- ⇒ Manter os usuários bem próximos para poder identificar e resolver expectativas irreais;
- ⇒ Mostrar às pessoas os pontos de vista dos outros usuários;
- ⇒ Incentivar a procura de critérios de solução;
- ⇒ Relacionar as expectativas com a experiência anterior.

3) Associar as tarefas das pessoas com suas condições de vencedores, o que implica em:

- ⇒ Quebrar as funções, atividades e fases em partes e configurar combinações de subopções em pacotes vencedores para cada participante;
- ⇒ Procurar assegurar a qualidade dos produtos.

Boehm et al. [BORO 89] frisam que a aplicação desses princípios contribuiria fortemente para o sucesso do desenvolvimento dos projetos de produtos de software.

4.3.2.5 Comparando os Métodos

Os métodos tradicionais de desenvolvimento de software propiciam pouco envolvimento dos usuários ao longo de todo o processo. Os métodos participativos empregam variadas formas organizacionais criando uma estrutura de apoio à participação dos usuários. A forma básica envolve grupos compostos por usuários de diferentes níveis hierárquicos e funções, que são continuamente envolvidos no processo de desenvolvimento dos projetos. Clemente et al. [CLBE 93] observam que não são métodos particulares ou técnicas que são decisivas, mas um forte foco político na participação, comunicação e aprendizado.

A abordagem dos quatro métodos acima expostos, centra-se na ativa participação dos usuários, sendo que, a forma de envolvimento difere consideravelmente.

As reuniões que seguem o método de Projeto Participativo permitem uma ampla interação dos membros do grupo e estimulam os projetistas a vivenciarem o trabalho dos usuários. O uso de protótipos facilita a definição dos requisitos e objetivos do sistema, assim como, ajuda os participantes a vislumbrarem como o computador pode apoiar suas práticas de trabalho. Entre os quatro métodos apresentados é o que possui mais abordagens sociais, acentuando o contexto social do ambiente de trabalho e proporcionando aos trabalhadores controle sobre suas tarefas. Miller [MILL 93] observa que, a principal razão para a emergência de abordagens como Projeto Participativo, na escandinávia, é a orientação social democrática do movimento trabalhista, segundo a qual, as pessoas devem possuir direito de influenciar diretamente nos aspectos que concernem a eles em seus postos de trabalho.

Em JAD encontramos a organização das reuniões mais estruturada, com um coordenador responsável pela dinâmica do grupo. O aprendizado entre os participantes se dá através da troca de experiências relatadas nas reuniões, não havendo indicação de vivenciá-las na prática. Os membros possuem liberdade de expor e discutir suas idéias, mas de uma forma disciplinada e apoiada em algum ciclo-de-vida e num plano de ações. Tem como objetivo central, melhorar a qualidade dos sistemas, diminuindo seus custos e tempo de desenvolvimento do ciclo-de-vida. Esse método não pressupõe o uso de protótipos. O software é "visualizado" graficamente em quadros brancos, através de desenhos e gráficos, símbolos e quadros magnéticos.

SADT Aumentado visa facilitar o levantamento e a especificação do software juntamente com os usuários, fornecendo toda a riqueza e complexidade das interações sociais no trabalho. As reuniões são estruturadas com base em propostas e compromissos, num processo interativo apoiado numa linguagem gráfica. Os analistas são os responsáveis pelas reuniões, onde as entrevistas são realizadas com todo o grupo, pois uma sessão de entrevistas em grupo fornece a oportunidade de usuários e analistas compreenderem o trabalho que está sendo desenvolvido e os progressos empreendidos. A prototipagem é utilizada como forma de validar requisitos, descobrir informações adicionais e suas relações. Os analistas e projetistas são estimulados a vivenciarem o trabalho dos usuários como forma de compreender as nuances das interações entre as pessoas.

Já a Teoria-W visa desenvolver software que satisfaça plenamente todas as pessoas afetadas por ele: usuários, gerentes, equipe de manutenção, equipe de desenvolvimento, etc. Possui seu foco principal na gerencia e nas negociações das partes

envolvidas no projeto, sendo o gerente o responsável pela negociação. O método propõe um roteiro de princípios que deve ser seguido como forma de garantir a qualidade do produto e a satisfação dos envolvidos. Aborda a prototipagem como forma de aproximar as partes envolvidas. É o método mais estruturado entre os quatro apresentados, possuindo planos de ação para cada etapa, sendo o gerente a pessoa responsável pela aplicação desses planos. Possui ainda diferentes estruturas de reuniões, cada uma relacionada com a etapa que está sendo desenvolvida. As informações e experiências são transmitidas oralmente, não propondo vivência de trabalhos.

Como forma de melhor visualizar esses resultados apresentamos as seguintes tabelas, que por causa da quantidade de informações agrupadas, teve que ser dividida em três páginas:

Tabela 4.1 : Comparação de algumas características entre os quatro métodos elaborada a partir de Carmel et al. [CAWG 93], Boehm et al. [BORO 89] e Marca [MARC 91]

	Projeto Participativo	JAD	SADT Aumentado	Teoria-w
Objetivo	Desenvolvimento participativo de software, melhorar o trabalho das pessoas	Melhorar os sistemas, incluindo a visão do usuário	Descrever sistemas e suas operações, melhorar o trabalho das pessoas	Melhorar a gerência do processo de desenvolvimento de software
Crítérios	Qualitativos: aprendizado e educação mútua, resolução de conflitos, democracia	Quantitativos: índices de desempenho, economia de tempo	Qualitativos: facilita a especificação de requisitos e a revisão e validação de documentos	Qualitativos: cumprir o planejamento definido, identificar e gerenciar os riscos
Base	Relações de trabalho, aprendizado mútuo, prototipagem	Dinâmica de grupo, engenharia de software	Aprendizado mútuo, prototipagem, compromisso	Compromissos, planejamento, gerência de projeto

	Projeto Participativo	JAD	SADT Aumentado	Teoria-w
Temas	Democracia do ambiente de trabalho, democracia industrial, contexto social, humanização	Trabalho em equipe, aceleração de projeto	Participação do usuário, conversação, ação, recursos humanos, negociação, responsabilidade	Satisfação de todos os envolvidos no projeto, planejamento estruturado
Dinâmica das reuniões	Livre e criativa	Rígida, com definição de papéis, protocolos de conversação e ciclo-de-vida	Estruturada com base em estabelecimento de requisitos e compromissos	Estruturada com base em questões e roteiros
Responsável pela coordenação	Usuários e analistas	Especialista no método	Analistas	Gerente
Estimula aprendizado mútuo ?	Sim, com vivências	Sim, mas somente através de trocas de experiências durante as reuniões	Sim, com vivências	Não
Apoio de protótipo	Sim	Não	Sim	Sim

	Projeto Participativo	JAD	SADT Aumentado	Teoria-w
Perspectivas sobre os usuários	<ul style="list-style-type: none"> - Trabalhadores operacionais são vistos como usuários; - Participação indiscriminada do usuário nas reuniões; - Usuários são vistos como fonte primária de conhecimento; - Participação no desenvolvimento de software; - Representativo. 	<ul style="list-style-type: none"> - Trabalhadores operacionais e gerentes são considerados usuários; - Seleção de usuário baseada no critério de competência; - Usuários são vistos com uma fonte de conhecimento; - Representativo. 	<ul style="list-style-type: none"> - Trabalhadores operacionais e gerentes são considerados usuários; - Usuários ligados a área de domínio da aplicação; - Consultivos. 	<ul style="list-style-type: none"> - Todos os afetados diretamente pelo software são usuários: compradores, equipe desenvolvedora, equipe mantenedora, gerentes, usuários finais, etc; - Representativos.

4.3.3 Apoio Automatizado para Metodologias Cooperativas de Desenvolvimento de Software

Apesar das pesquisas na área de métodos e ferramentas de apoio à desenvolvimento cooperativo de software estarem se difundindo e dos pesquisadores estarem mais conscientes da importância dos aspectos sociais, os desafios ligados à aspectos técnicos corporativos ainda persistem. Cada organização precisa de uma ferramenta que se adapte à sua estrutura econômica, organizacional e metodológica. Em Greenberg [GRE2 91] encontramos uma grande lista de referências sobre trabalho cooperativo, onde podemos constatar a preocupação dos autores com a participação dos usuários no desenvolvimento de software, e também, a falta de relatos sobre ferramentas de apoio à essa atividade.

Em Normam et al. [NOCR 91] encontramos uma lista mais detalhada de aspectos cooperativos que foram discutidos no workshop sobre CASE em 1990 na Califórnia. Esse artigo expõe o atual estágio do desenvolvimento de ferramentas CASE para apoiar desenvolvimento cooperativo de software, e levanta características de ferramentas que já estão funcionando, o que ainda não funciona, o que deve ser incorporado e o que necessita ser melhor investigado no processo de trabalho em grupo, dentre os quais ressaltamos os seguintes:



O que funciona:

- Redes de comunicação e Bulletin Board Systems (BBS) com baixa conectividade, baixa interação;
- Suporte eletrônico para discussão do grupo sobre descrição e requerimentos de sistemas;
- Tela remota em ambientes limitados;
- Anotação de documentos (com limites).



O que ainda não funciona:

- Coordenação dinâmica do trabalho;
- Desenvolvimento de software sem o uso de papel;
- Redes de comunicação seguras;
- Substituição de tarefas através do uso de ferramentas CASE.

 **O que é necessário:**

- Comunicação rápida através de multimídia;
- Coordenação do projeto atuando em todas as fases do desenvolvimento;
- Rápido acesso às informações do projeto ou produto;
- Ferramentas que apoiem o treinamento de novos componentes do grupo;
- Ferramentas que possam ser modificadas, integradas, separadas de acordo com as necessidades do grupo;
- Sistemas gerenciais fáceis de usar e aprender.

 **O que deve ser mais pesquisado:**

- Identificação de problemas e de como a organização, o projeto, os grupos e os indivíduos interagem;
- Aplicação de ferramentas de apoio a grupos em geral e em grupos de desenvolvimento de software;
- Descrição, formalização e análise de processos que apoiem o trabalho de grupo;
- Que informações as pessoas necessitam para o desenvolvimento dos seus trabalhos;
- Como as equipes de desenvolvimento de software trabalham em grupo;
- Classificação, organização, acesso e apresentação das informações relativas ao desenvolvimento de software;
- Processos de maturação das organizações para implantação de novos procedimentos de trabalho.

Os princípios que regem o desenvolvimento de produtos para uso individual não são os mesmos que guiarão o desenvolvimento de produtos para grupos. Essa idéia é reforçada por King [KING 91] que coloca que a introdução de ferramentas computadorizadas no ambiente de trabalho transforma a natureza do trabalho que deve ser realizado. Isso deve ser bem compreendido, ou o sistema falhará. É importante reafirmar a necessidade de equipes compostas de cientistas da computação, cientistas sociais e cognitivos e usuários para o desenvolvimento de sistemas de apoio ao trabalho cooperativo.

4.4 Conclusões

Neste capítulo buscou-se expor a evolução de metodologias e técnicas de desenvolvimento de software. Partiu-se do desenvolvimento em pequena escala, a organização das equipes de desenvolvimento, os modelos de ciclo-de-vida e as metodologias tradicionais.

Em seguida, detalhou-se mais minuciosamente alguns métodos de desenvolvimento de software que possuem um enfoque cooperativo, comparando-os para que se evidenciasse seus pontos comuns e diferenças. Por fim, apresentou-se uma lista de aspectos que permeiam a construção de apoio automatizado para métodos cooperativos.

No próximo capítulo apresentaremos um estudo de casos que ajudará a elucidar pontos importantes dos métodos acima expostos.

CAPÍTULO 5

Estudos de Casos

5.1 Objetivos

Neste capítulo ilustraremos as teorias apresentadas anteriormente através do estudo de quatro casos. Nos capítulos 2 e 3 levantamos algumas das questões sociais envolvidas e de como a tecnologia apoiaria o Trabalho Cooperativo. Através do estudo de casos buscamos verificar alguns aspectos abordados nesses capítulos.

O estudo é direcionado principalmente no sentido de identificar a utilização de metodologias de desenvolvimento de software tal como apresentado no capítulo 4, visando analisar seus impactos. As principais questões a serem analisadas a partir das experiências relatadas centram-se sobre a possibilidade de perceber se:

- foram usados alguns dos métodos descritos no Capítulo 4;
- a atividade de coordenação é importante;
- trabalho cooperativo gera conflitos;
- foi usado algum método de resolução de conflitos;
- a introdução de novas formas de trabalho ou da ferramenta, gerou resistências vinculadas à medo da perda de poder;
- houve reagrupamento de ocupações fragmentadas;
- o usuário teve realmente participação ativa em todas as etapas do trabalho;
- houve uso de alguma forma de comunicação eletrônica;
- o desenvolvimento de software apoiado em métodos cooperativos pode ser eficaz e atingir seus objetivos sem o apoio de ferramentas automatizadas, contrariando

as constatações de Carneiro [CARN 93] e Pinto [PINT 93] no sentido de que metodologias sem ferramentas de apoio tornam sua implementação bastante complexa;

- a introdução de métodos cooperativos no desenvolvimento de sistemas gera benefícios e vantagens.

As empresas escolhidas foram selecionadas por estarem utilizando métodos cooperativos no desenvolvimento de produtos de software e atuam, respectivamente, nas áreas: petroquímica, financeira, tratamento de informações estatísticas e consultoria. A tabela abaixo descrita, mostra de forma simplificada as atividades principais de cada uma dessas empresas.

Tabela 5.1: Empresas selecionadas e suas principais atividades

CASOS	EMPRESAS	ATIVIDADES
1	Petroquímica	Exploração, perfuração produção, refino, transporte, comercialização de petróleo e derivados
2	Atuante em Estatística	Produção, análise, pesquisa e disseminação de informações de natureza estatística, econômica, geográfica e ambiental
3	Banco Financeiro de Desenvolvimento	Financia o desenvolvimento de empresas nacionais
4	Serviços de Consultoria	Presta consultoria em desenvolvimento de sistemas de informação

5.2 Metodologia Adotada

Na medida em que, no Brasil, só recentemente as metodologias cooperativas de desenvolvimento de produtos de software começaram a ser divulgadas, poucas organizações as utilizam para apoiar o desenvolvimento de seus sistemas. Na escolha das

empresas a serem pesquisadas, não nos pautamos por critérios rígidos (tais como tamanho da empresa, quantidade de funcionários ou origem de capital), e sim, pela existência de uma experiência e possibilidade de obtenção de informações para fins de pesquisa. Sob essa ótica, selecionamos empresas que passaram pela experiência de desenvolver produtos de software baseando-se em métodos com forte conotação cooperativa.

É importante observar que os resultados aqui apresentados são, de natureza qualitativa, fruto da percepção de cada um dos entrevistados, os quais estiveram no centro do desenvolvimento dos trabalhos.

Nos três primeiros casos relatados ressaltamos a dificuldade de entrevistar um maior número de usuários finais. Essas limitações foram colocadas pelas pessoas contactadas inicialmente, as quais, em geral, ocupavam altos cargos de chefia.

A coleta de dados teve por base entrevistas abertas. Tais dados foram sistematizados, facilitando sua posterior análise, permitindo elucidar e detalhar aspectos importantes sobre como as técnicas cooperativas podem auxiliar o desenvolvimento de produtos de software, através do envolvimento dos usuários.

Em suma, os resultados aqui apresentados se apoiam no estudo dos quatro casos citados, os quais, no seu universo de acertos e dificuldades, podem servir de base a futuras experiências.

5.3 Apresentação dos Casos

5.3.1 Empresa Petroquímica

Este caso enfoca uma experiência, desenvolvida numa empresa do setor petroquímico que é formada por um conglomerado de cerca de uma centena de empresas entre subsidiárias, controladas e coligadas, possuindo o monopólio da exploração, perfuração, produção, refino, transporte e comercialização do petróleo e derivados.

A gestão da empresa é feita de forma centralizada com sede no Rio de Janeiro. Sua estrutura é formada por um Conselho Administrativo, uma Diretoria Executiva, cinco diretorias principais: Diretoria de Exploração, Diretoria de Perfuração, Diretoria de Produção, Diretoria de Industrialização e Diretoria de Transportes, e ainda, Superintendências Gerais. Cada órgão possui uma secretaria própria. Além disso, às

diretorias estão vinculadas áreas-fim, tais como departamentos, subsidiárias e áreas de serviço, onde se situa a área de informática.

De forma simplificada, apresentamos a seguir um organograma da empresa:

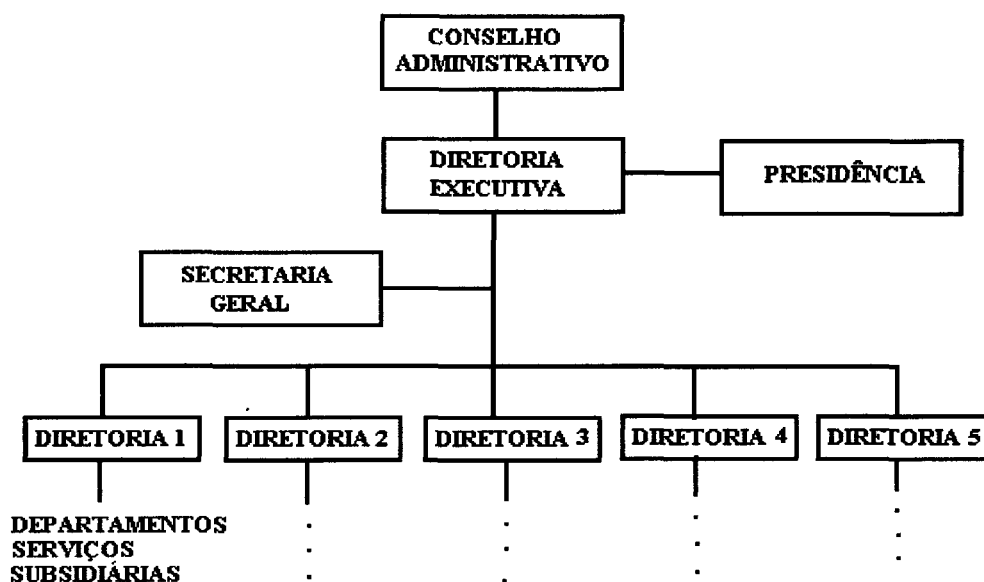


Figura 5.1 : Organograma simplificado da Empresa Petrolífera

A organização das atividades de informática na empresa estão, em grande parte, descentralizadas. Em 1990 foi instituído um órgão que é responsável, entre outras, pelas redes de processamento de dados e pelo desenvolvimento de aplicações corporativas de interesse geral. No entanto, seus diversos setores possuem autonomia para desenvolver aplicações específicas a nível departamental ou local.

Dentro da empresa, um dos aspectos administrativos considerado crítico era o processo de tomada de decisão. Com a gerência centralizada, quando havia necessidade de deliberação sobre alguma requisição vinda de uma área-fim ou diretoria, a proposta percorria os vários níveis até chegar ao órgão que possuía poder de decisão. Então, a resposta percorria todos os níveis, em sentido contrário, até chegar ao requisitante. Esse processo era longo e complexo devido as dimensões da empresa. Como a empresa atravessa uma fase de modernização e reorganização de sua estrutura, foi criada uma Assessoria de Modernização responsável por um projeto de reorganização e um programa de modernização empresarial composto por várias etapas. O primeiro projeto posto em prática visou otimizar o apoio à alta administração da companhia, na tomada de decisões gerais. Um grupo de trabalho foi criado para desenvolver esse primeiro projeto. Participaram do grupo representantes do Gabinete da Presidência, do Conselho

Administrativo, da Diretoria Executiva e demais Diretorias. Esse grupo foi coordenado por um representante da Assessoria de Modernização. Os participantes e o coordenador foram escolhidos pela alta administração. Os critérios de escolha não ficaram transparentes para o grupo que era composto por sete pessoas e dois assistentes. Dentre as sete pessoas, uma era ligada ao órgão central de informática. As pessoas que fizeram parte dessa equipe inicial também eram usuários finais, só que ocupavam cargos de chefia. Durante as reuniões não havia rigidez protocolar, a participação era livre e bastante estimulada. O projeto começou a ser desenvolvido em julho de 1992.

Esse grupo inicial fez um levantamento das relações e aspectos críticos do processo de tomada de decisões, racionalizando atividades e dispensando aquelas que não agregavam valor (análise de negócio). Depois disso, foi desenvolvido um protótipo na linguagem CLIPPER, que foi apresentado aos usuários finais de todos os conselhos, secretarias, três departamentos e cinco áreas de serviço. Os usuários puderam opinar fornecendo novas idéias e mudanças. O protótipo voltou, então, para a alta administração que novamente pode alterar os aspectos que julgavam necessários, integrando as expectativas dos usuários com o interesse estratégico da empresa. O protótipo mostrava todo o modelo conceitual até chegar ao sistema de informação. Depois disso, o órgão de informática ocupou-se da fase de desenvolvimento do sistema, havendo um afastamento dos usuários, sendo o coordenador o elo que unia usuários e desenvolvedores. O sistema foi desenvolvido num VAX, utilizando a linguagem C e Banco de Dados RDB, sendo que o protótipo inicial foi completamente descartado, pois foi desenvolvido em computadores de pequeno porte e o sistema definitivo teria que ser implementado no VAX.

Neste caso foram entrevistados o coordenador do projeto, o chefe da Assessoria de Modernização e o chefe da secretaria de uma área de serviço, onde o sistema está sendo primeiramente implantado. Essas pessoas são, também, usuárias do sistema.

O coordenador do projeto teve uma atuação, que foi considerada eficaz e dinâmica, tornando-se pessoa-chave, coordenando as reuniões iniciais, atuando como intermediário entre a equipe desenvolvedora e os usuários e entre os superiores e a equipe desenvolvedora. Ele possuía autonomia para desenvolver o projeto segundo os critérios que desejasse, não sofrendo interferências de superiores. A opção pelo tipo de metodologia adotada partiu dele que, em cada etapa do ciclo-de-vida, usou um método de trabalho, buscando em cada uma deles o que tinha de positivo e de adequado às tecnologias disponíveis. Foram usadas a Análise Estruturada, Projeto Estruturado e Orientação à Objetos (teorias de encapsulamento). Durante a fase de análise, na etapa de levantamento de requisitos, foi usado o PC-CASE, mas foi logo abandonado, pois somente o coordenador do projeto conhecia bem a ferramenta.

A equipe desenvolvedora era formada por um analista e um programador, tendo o apoio de um segundo analista, mais ligado a aspectos administrativos e gerais, ou seja, atuando como analista de negócios. Neste caso, as funções de análise e programação estavam bem separadas, não havendo reagrupamento de funções.

Os usuários não tiveram contato com a equipe desenvolvedora e não participaram em nenhum momento da codificação nem dos testes de sistema, pois o coordenador intermediava as relações entre os dois grupos. Apesar disso, os usuários mostraram-se satisfeitos com o coordenador interligando os dois grupos. Segundo os entrevistados, os usuários apontaram que houve aprendizado mútuo através do relato de experiências, ou seja, sem vivências práticas. Como os componentes da equipe foram indicados pela empresa, foi abordada a falta de um trabalho inicial de esclarecimento e motivação dessa equipe, para que a participação fosse mais intensa, e com isso as pessoas envolvidas pudessem perceber a importância da troca de informações como forma de enriquecimento pessoal e profissional.

A equipe inicial manifestou a existência de resistências tanto com relação à forma de trabalho que teria que ser adotada quando o sistema fosse implementado, quanto em relação à introdução da informática. Não houve resistências contra a nova metodologia de trabalho para o desenvolvimento do produto. Segundo o coordenador, "ninguém, aparentemente, percebeu a nova metodologia". Mas ele considerou que, "desenvolver um trabalho como esse é muito difícil" pois, "os usuários tinham poder de derrubar o projeto". Devido a não ocorrência de maiores divergências, não houve necessidade de aplicação de nenhum método de resolução de conflitos, tudo era resolvido através de discussões e acordos. Como as pessoas envolvidas ocupavam altos cargos administrativos, não houve problemas relativos à liberação para a participação dos encontros, mas foi ressaltado pelo assessor de modernização que houve dificuldades na conciliação de horários para as reuniões.

Já no grupo de usuários finais, as resistências foram muitas, principalmente, geradas por receios na mudança da forma de trabalho e mudanças na estrutura de poder que o novo sistema aportaria. As pessoas temiam uma transferência de função, ocasionando perda do poder. Os conflitos foram resolvidos através de discussões onde buscava-se a coesão de idéias. Mas, segundo o coordenador: "quando há conflitos é que começam a surgir idéias". Como a equipe desenvolvedora teve pouco contato com o grupo de usuários, não surgiram conflitos entre eles, assim como não foi assinalada troca de experiências entre essas duas equipes.

No apoio a comunicação entre os participantes não foi usado nenhum meio eletrônico, assim como também, não houve suporte de nenhuma ferramenta que automatizasse procedimentos.

Não houve treinamento sobre o método cooperativo para as pessoas envolvidas no projeto. O treinamento limitou-se aos usuários finais, no sentido de aprender a utilizar o sistema desenvolvido.

Com relação a vantagens, os entrevistados enfatizaram a percepção da melhoria do nível de qualidade do sistema em relação aos métodos utilizados anteriormente.

Os fatores citados como os que mais influenciaram no sucesso do projeto foram: a determinação de desenvolvê-lo e a forma como foi tratado; o comprometimento dos usuários do sistema, pois o processo cooperativo tornou todos um pouco responsáveis pelo sistema; a perspectiva de gerar um produto com alta qualidade, facilitando sua implantação na área administrativa da empresa.

A maior dificuldade considerada pelo coordenador do projeto, foi gerenciar o medo das pessoas que participaram do grupo inicial, pois em princípio, haveria mudanças na estrutura organizacional de alguns setores. Isso gerava receios e inseguranças, pois eram os próprios participantes que encabeçavam tal estrutura organizacional. Por fim percebeu-se que os temores eram infundados, pois o novo sistema, mais rico e ágil, pode ser absorvido dentro da estrutura atual. Além disso, outro problema referia-se a interação com novas tecnologias, no caso os computadores, sendo que, esse aspecto foi abordado por todos os entrevistados. Foi colocado, também, que atualmente as pessoas estão bem mais habituadas com as novas tecnologias e estão interessadas em que o sistema seja logo implantado em toda a organização, e mesmo, exercendo pressões para que isso se dê o mais rápido possível.

Análise dos Resultados

Este caso foi considerado por todos os entrevistados com um caso bem sucedido, que vem alcançando o objetivo definido a priori, que foi: otimizar o apoio à alta administração da empresa, na tomada de decisões gerais. Percebemos que o objetivo desse projeto de modernização centra-se na possibilidade de melhorar processos administrativos estratégicos, buscando diminuir o tempo entre as decisões e a burocracia associada.

O coordenador do projeto, considerado pessoa-chave, buscou associar aspectos cooperativos no desenvolvimento do ciclo-de-vida do produto, levantando relações de trabalho e apoiando-se num protótipo, ou seja, atuando como propõem

Andersen et al. [ANHO 91], estendendo a participação aos usuários dos diferentes setores envolvidos no projeto. Apesar da experiência ter sido gerada a partir de decisões provenientes dos altos níveis hierárquicos, observou-se que as pessoas envolvidas no projeto mostraram-se bastante motivadas, facilitando o desenvolvimento dos trabalhos. Durante as reuniões não havia rigidez de papéis, todos podiam participar.

Confirmando o que é ressaltado por Greenbaum et al. [GRKY 91] e Tou et al. [TOBE 94], entre outros, o uso de prototipagem foi uma estratégia para integrar todos os usuários, que não participaram da primeira fase do projeto, tornando-os um pouco responsáveis pelo sistema, reforçando a coesão do grupo. Foram consultados usuários de vários níveis hierárquicos, de diferentes diretorias, áreas de serviço, departamentos, etc. Mas essa integração de usuários não se deu em todas as fases do ciclo-de-vida de desenvolvimento, constatando-se que a cooperação não atingiu os níveis propostos por um grande número de autores, entre eles Kyng [KYNG 91], Bannon [BANN 91], Greenbaum et al. [GRKY 91] que colocam que os usuários devem participar de todas as etapas do desenvolvimento dos projetos, como meio de aumentar a qualidade dos resultados.

Kyng [KYNG 91], Carmel et al. [CAWG 93], entre outros, assinalam a importância das trocas de vivências de trabalho entre a equipe desenvolvedora e o grupo de usuários gerando trocas de experiências em seus campos de atuação profissional, o chamado aprendizado mútuo. Neste caso não houve tal vivência, mas as pessoas mostraram-se satisfeitas com o nível de interação alcançado.

Pinto [PINT 92] constatou que a possibilidade do esforço de aprendizado de uma nova metodologia provoca rejeição, mas neste caso, como colocou o coordenador, as pessoas não perceberam a inovação do método de desenvolvimento de software aplicado, não havendo percepção de rejeição. Já a introdução dos computadores na rotina diária de trabalho, provocou temores que foram sendo amenizados na medida em que, as pessoas conheciam melhor o novo sistema. Neste caso não houve apoio de nenhum tipo de sistema de comunicação eletrônico, evidenciando as colocações de Wilson [WILS 91], que o sucesso de um projeto que envolve cooperação não ocorre somente devido ao apoio tecnológico, e sim, por vários fatores que englobam, principalmente, as necessidades humanas.

Neste caso os conflitos foram considerados positivos, geradores de novas propostas e corroborando Mathias et al. [MAFU 92], que colocam que os conflitos podem fomentar a discussão de idéias não amadurecidas, abrindo novos caminhos.

Outro ponto fortemente ressaltado por Grudin [GRUD 91], [GRUD 94] e Greenberg [GREN 91], é a importância das ferramentas para o sucesso do método, esse

aspecto não pode ser avaliado, pois neste caso, não houve utilização de nenhum suporte eletrônico ao desenvolvimento do ciclo-de-vida do produto. Vale assinalar que houve uma tentativa inicial de utilização do PC-CASE, logo descartada por falta de pessoas que conhecessem melhor seu funcionamento. Ou seja, o desenvolvimento dos trabalhos foi apoiado em materiais simples, constatando-se a falta de ferramentas de apoio ao desenvolvimento cooperativo de software, tal como foi discutido por Norman et al. [NOCR 91] e apresentado no capítulo 4.

Do ponto de vista geral, nos parece que o gerenciamento do medo das pessoas representa a maior resistência à realização de projetos. No caso que temos em apreço, percebe-se claramente tal aspecto, reforçando o que foi colocado no capítulo 3 (pp.22), onde é abordada a necessidade de "buscar-se sistemas que expandam as opções dos trabalhadores, de tal forma, que eles possam desenvolver estratégias de trabalho que permitam uma participação consciente e democrática", amenizando o medo das mudanças estruturais e conseqüente perda do poder.

Não constatou-se a tendência verificada por Friedman [FRID 93], da convergência entre as atividades de análise e programação. Entretanto, percebe-se que a função de analista de sistemas se expandiu criando um profissional mais ligado à aspectos administrativos e gerais, intrínscos ao projeto, corroborando as colocações de Tierney [TIER 91] sobre a emergência do analista de negócios.

5.3.2 Empresa atuante em Estatística

O segundo caso refere-se a uma experiência realizada numa empresa que tem por finalidade básica a produção, análise, pesquisa e disseminação de informações de natureza estatística (demográfica e social), econômica, geográfica, geodésica e ambiental. Sua sede situa-se na cidade do Rio de Janeiro e suas Unidades Regionais estão distribuídas nos diferentes estados e em Brasília.

A implantação da informática na empresa iniciou-se nos anos 70 de forma centralizada, com um computador de grande porte IBM. Nos anos 80 iniciou-se um processo de descentralização, que foi consolidado nos anos 90, e que demandou recursos de informática que possibilitassem a migração das aplicações existentes. São duas as diretorias ligadas ao setor de informática, as quais, são divididas em departamentos: uma diretoria de informática, que é responsável pelo desenvolvimento de aplicativos, fornecendo

ferramentas e suporte técnico é dividida em dois departamentos: um departamento técnico e outro de atendimento (responsável pelo atendimento da demanda de software dos especialistas temáticos). Na outra diretoria, ligada às pesquisas estatísticas, estão os especialistas temáticos que são responsáveis pelos resultados das várias pesquisas empreendidas pela empresa.

Esta experiência divide-se em duas fases:

- ♦ a primeira refere-se ao projeto e desenvolvimento de uma ferramenta cooperativa, através da prototipagem cooperativa onde os usuários(especialistas temáticos e desenvolvedores) participaram analisando e validando o protótipo;
- ♦ a segunda, o desenvolvimento de sistemas específicos de análise e processamento de dados de pesquisas usando a ferramenta cooperativa desenvolvida no primeiro momento, envolvendo especialistas temáticos e desenvolvedores de sistemas.

Foram entrevistadas sete pessoas: um chefe do departamento de informática responsável pelo desenvolvimento da ferramenta, um gerente desse departamento, um chefe do departamento de especialistas temáticos, um supervisor da equipe de especialistas temáticos, um gerente de projeto e dois chefes do departamento de desenvolvimento. Sendo duas pessoas do departamento de especialistas temáticos, duas do departamento que desenvolveu a ferramenta e três do departamento de desenvolvimento.

A experiência envolveu as duas diretorias e refere-se ao desenvolvimento e implementação de uma ferramenta cooperativa que permite aos especialistas temáticos gerarem o plano de críticas dos dados recolhidos nas pesquisas. Antes desta ferramenta ser desenvolvida, eram os funcionários responsáveis pelo desenvolvimento dos sistemas (analistas) que especificavam as críticas. Através da ferramenta os especialistas temáticos que são responsáveis pelo processamento dos questionários das pesquisas, geram o plano de críticas como parte dos programas, que por sua vez, são desenvolvidos por uma equipe técnica de informática. As duas diretorias possuem um nível de interação bastante alto devido a dependência de serviços que uma possui em relação à outra. Os especialistas temáticos precisam dos sistemas desenvolvidos pelos departamentos técnicos e de atendimento e estes, precisam dos especialistas temáticos para obter informações específicas. O protótipo foi desenvolvido numa linguagem que também foi desenvolvida "in house".

De acordo com um dos entrevistados, os recursos financeiros são insuficientes para a compra de ferramentas e as disponíveis no mercado não se adequam às necessidades específicas da apuração descentralizada realizada pela empresa. Por isso, na empresa, existe uma grande variedade de ferramentas desenvolvidas "in house".

No caso em estudo, a equipe responsável pela ferramenta cooperativa interagiu sistematicamente com os especialistas temáticos e equipe de desenvolvedores, para que a ferramenta preenchesse as expectativas de todos. Utilizou-se a técnica de prototipagem evolutiva, com a participação ativa dos usuários. Na figura abaixo descrita, pode-se visualizar de maneira mais clara, as relações e tarefas empreendidas pelas diferentes equipes:

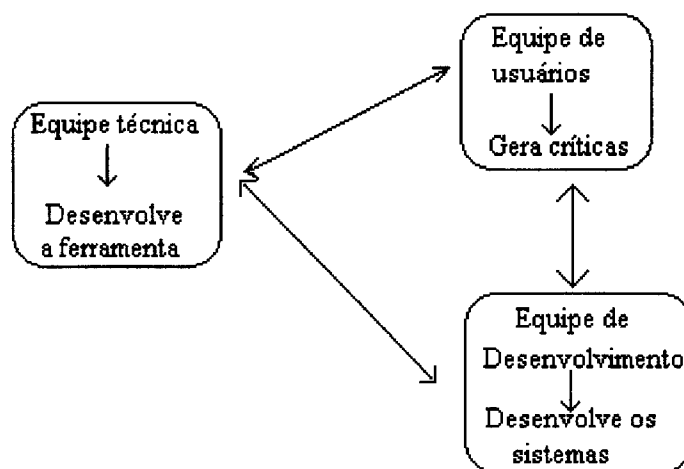


Figura 5.2: Relação entre as equipes envolvidas no desenvolvimento e uso da ferramenta

Normalmente, na empresa, as equipes técnicas que são responsáveis pelo desenvolvimento de sistemas para cada pesquisa estatística específica, são escolhidas de acordo com o perfil e as necessidades de cada projeto. Geralmente são compostas por um gerente, analistas, programadores e documentador. Na fase de concepção do projeto os programadores não são envolvidos, mas, na fase de codificação os analistas também programam. Em geral, o desenvolvimento se dá utilizando o modelo cascata com as seguintes etapas: análise de viabilidades, projeto lógico, implementação e manutenção, não adotando-se ferramentas CASE.

Na primeira fase (desenvolvimento da ferramenta), não foram assinalados conflitos e as pessoas envolvidas ressaltaram a forte troca de informações, unificando vocabulário e proporcionando uma maior auto-confiança nos especialistas temáticos para discutirem aspectos técnicos com os membros da equipe desenvolvedora. Ou seja, houve percepção de aprendizado mútuo. Mas não houve integração dos especialistas temáticos em todas as etapas do desenvolvimento da ferramenta. Tampouco houve treinamento, os usuários eram apresentados ao protótipo, forneciam suas opiniões e a equipe desenvolvedora realizava as modificações. O fator considerado como incentivador para o desenvolvimento e o posterior uso da ferramenta foi a expectativa de usar e gerar produtos com maior qualidade.

Na segunda fase, a equipe técnica responsável pelo desenvolvimento da ferramenta, ofereceu treinamento para especialistas temáticos e desenvolvedores de como usar a ferramenta cooperativa para desenvolver os sistemas específicos para as pesquisas. Tal treinamento foi considerado suficiente, não havendo relato de nenhum tipo de problema referente a liberação de pessoal para participar do treinamento.

Não houve qualquer tipo de conflito assinalado e sim um bom entendimento entre todas as pessoas envolvidas. Todos os entrevistados reforçaram o fato da existência de um ambiente amigável entre os funcionários da empresa. Durante as entrevistas pudemos constatar esse clima positivo com relação aos trabalhos desenvolvidos de forma cooperativa. Com isso, foi assinalado que não houve nenhum tipo de resistência nem por parte da equipe desenvolvedora, nem por parte dos especialistas temáticos, sendo que o desempenho da equipe não foi medido.

Como os especialistas temáticos passaram a descrever as críticas dos dados, eles consideraram que essa atividade favoreceu um aumento de conhecimento em informática. Não foi assinalado qualquer tipo de resistência à introdução da nova ferramenta. Num primeiro momento houve uma dificuldade de comunicação entre eles e a equipe desenvolvedora, mas com o tempo o vocabulário técnico unificou-se, facilitando a comunicação. A equipe desenvolvedora considera que passou a conhecer mais sobre estatística, assim como, os especialistas temáticos passaram a conhecer mais de informática, havendo dessa forma, trocas de conhecimento, sendo que os membros da equipe técnica tiveram alguma vivência do trabalho dos especialistas temáticos de forma a integrar as críticas aos programas que desenvolvem. Não foi assinalado nenhum problema relativo a medo da perda do poder, talvez amenizado pelo clima de camaradagem constatado durante as entrevistas. Um aspecto levantado pelos especialistas temáticos foi a insegurança que havia, antes da implantação da ferramenta, na descrição das críticas dos questionários, pois em caso de erro, a responsabilidade recaía sempre sobre eles. Hoje esse quadro foi

amenizado com o uso da ferramenta, que facilita as alterações, sem que a equipe desenvolvedora precise ser contactada.

Na empresa cada projeto possui um gerente responsável e a ele foi atribuída a responsabilidade pelos bons resultados obtidos. O chefe do departamento técnico da diretoria de informática foi considerado como a pessoa-chave no desenvolvimento dos produtos através das propostas cooperativas.

Finalizando, ressaltamos que os principais problemas citados foram: dificuldade de fornecer suporte técnico; dificuldade de cumprir prazos e alcançar as metas; e apesar de ter havido referência ao treinamento ter sido considerado suficiente, especialistas temáticos abordaram a dificuldade encontrada no aprendizado da ferramenta.

Enquanto que as vantagens foram: rapidez no desenvolvimento; facilidade de manutenção; facilidade para realocar membros da equipe desenvolvedora. Foi ressaltado que houve percepção de aumento de produtividade, de qualidade dos produtos e do próprio trabalho. Mas não conseguiu-se que os prazos fossem cumpridos e não houve nenhuma avaliação relativa à custos.

Análise dos Resultados

Este caso foi considerado um total sucesso por todas as pessoas contactadas, atingindo os objetivos pretendidos. Ele é dividido em duas etapas: desenvolvimento de uma ferramenta cooperativa usando metodologias cooperativas e posterior implantação nos setores de especialistas temáticos e de desenvolvimento de sistemas.

Na primeira fase foi utilizada a prototipagem cooperativa, sendo que neste caso, o protótipo foi sendo refinado e tornou-se o produto final, corroborando Gronbaek [GRON 89], que coloca que para que a prototipagem cooperativa seja eficaz é necessário realizá-la com ativa participação de desenvolvedores e usuários, permitindo que o usuário avalie a funcionalidade do sistema durante o seu desenvolvimento. Para que o produto preenchesse as expectativas de todos os envolvidos (desenvolvedores da ferramenta, especialistas temáticos e desenvolvedores de sistemas), os três grupos interagiram coordenados por um analista de sistemas, chefe do Departamento Técnico de Informática, que participou de todas as etapas de desenvolvimento da ferramenta até sua implantação, e ainda, forneceu treinamento. As reuniões para avaliação do protótipo se davam sob sua orientação, não havendo nenhum tipo de rigidez protocolar.

Na segunda fase, essa mesma pessoa fornecia o treinamento para os usuários usarem a ferramenta. Essa ferramenta serviu para que os dois grupos envolvidos nesta fase

conhecessem mais profundamente seus trabalhos, de modo a não somente agir mecanicamente, mas seguindo as propostas de Bannon [BANN 91], conhecendo realmente as etapas de seus trabalhos.

Em ambas as fases não foram assinalados casos de conflitos, muito pelo contrário, o que constatou-se foi a manutenção de um ambiente de alta camaradagem. Contudo, entre as equipes, havia uma diferenciação de termos técnicos devido a distinção de áreas de atuação: especialistas temáticos trabalham com estatística e a equipe de desenvolvimento com informática. As reuniões para trabalhar o protótipo juntamente com o posterior treinamento para utilizar a ferramenta, colaboraram para unificar vocabulário, pois segundo Chen [CHEN 94], a diferença de vocabulário cria muitas dificuldades no desenvolvimento de sistemas. De acordo com Bodker et al [BOGR 91], a prototipagem cooperativa colabora para o desenvolvimento de uma linguagem comum e uma compreensão partilhada.

A equipe técnica, responsável pelo desenvolvimento da ferramenta, vivenciou o trabalho dos especialistas temáticos e da equipe de desenvolvimento para que a ferramenta pudesse preencher os requisitos de ambas as partes, com percepção de aprendizado mútuo, seguindo o que é colocado por grande número de autores, entre eles, Kyng [KYNG 91] e Bodker et al.[BOGR 91], que colocam que o aprendizado mútuo é estabelecido a partir de vivências práticas.

Não houve referências a problemas de disputa de poder, sendo que, a boa relação entre os diversos grupos envolvidos nesta experiência pode ter sido incrementada pela prototipagem cooperativa, pois segundo Tou et al. [TOBE 94], as equipes que desenvolvem suas atividades baseadas na cooperação transformam a natureza do trabalho, mudando o foco da competição para a ajuda mútua.

Neste caso, também não houve apoio de ferramentas automatizadas nem de comunicação eletrônica.

A empresa desenvolve seus produtos "in house" forçada pela falta de verbas que custeiem a compra de produtos do mercado. Um aspecto que nos chamou a atenção, são as reclamações sobre os baixos salários pagos pela empresa em contraste com a alta motivação dos entrevistados para desenvolver seus trabalhos. Talvez esse quadro possa ser explicado pela constante possibilidade de desenvolvimento de produtos que estimulam a criatividade e que aumentam o conhecimento das pessoas envolvidas.

Foi constatado que, em geral, os analistas de sistema, e mesmo os gerentes de projeto, se ocupam das tarefas de programação, corroborando os resultados encontrados

por Friedman [FRID 93] sobre a crescente convergência entre as atividades de análise e programação.

5.3.3 Casos do Banco de Desenvolvimento

Neste caso enfocaremos **duas experiências** realizadas num Banco de Desenvolvimento que financia empresas nacionais, sendo que nenhuma delas foi totalmente finalizada.

O banco é um empresa pública federal com capital 100% nacional, sendo que, sua organização possui uma estrutura mista entre centralizada e distribuida. Sua estrutura hierárquica é formada por um Conselho de Administração, a Diretoria, secundados pela Presidência. No quarto nível encontram-se doze áreas, dentre as quais, a área de Administração, que é dividida em Departamentos e Gerências. É neste nível que encontra-se o Departamento de Sistemas, que é responsável pelo desenvolvimento de sistemas corporativos voltados principalmente para aplicações comerciais e financeiras. Cada área possui uma Gerência de Sistema, que desenvolve alguns sistemas locais de forma descentralizada, dando suporte aos usuários das diversas atividades de negócios, sendo que, a empresa investe continuamente em novas tecnologias. Apresentamos, a seguir, um organograma simplificado da empresa.

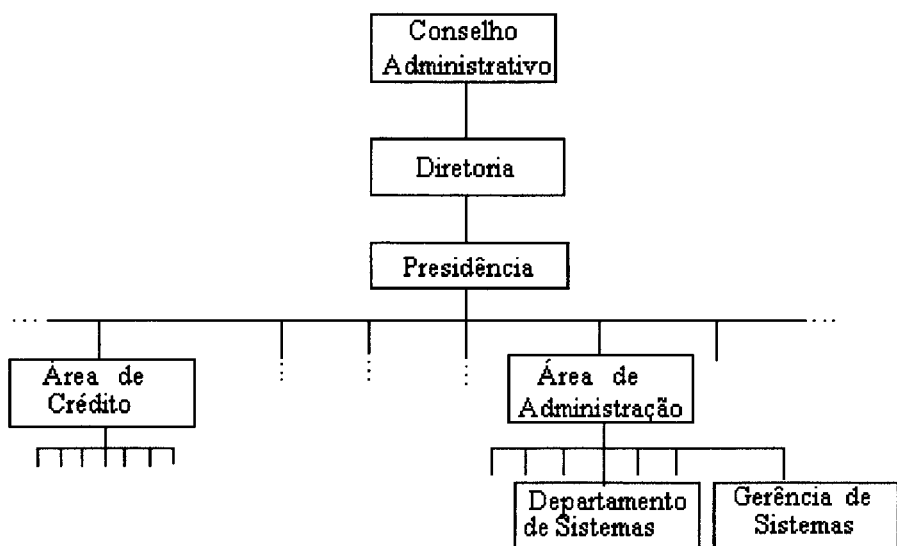


Figura 5.3: Organograma simplificado do Banco de Desenvolvimento

Nesta pesquisa foi entrevistada uma pessoa que na época do desenvolvimento das experiências, ocupava o cargo de gerente do departamento de sistemas. Foi utilizado, também, material obtido por ocasião do estudo realizado na mesma área por Carneiro [CARN 93], e que se adequa perfeitamente aos propósitos desta pesquisa.

Primeira Experiência

A empresa tinha como objetivo desenvolver "in-house" uma metodologia customizada de desenvolvimento de produtos de software que viabilizasse a padronização dos sistemas, gerando produtos de maior qualidade, que satisfizessem plenamente os seus usuários. Com isso, os chamados "donos" de sistema, teriam seu controle limitado, possibilitando melhorias nos processos de manutenção e documentação. A documentação era ineficiente devido a grande rotatividade de pessoal das empresas de terceirização, que eram responsáveis pelo desenvolvimento de software da empresa. O projeto-piloto iniciou-se por volta de 88/89.

A chefia do Departamento de Sistemas foi quem primeiro apoiou a iniciativa. Esse departamento, que era dividido em três gerencias, ficou responsável pelo projeto, entretanto apenas uma das gerencias se mostrou interessada, tornando-se órgão chave de todo o processo. Tal gerencia disponibilizou recursos de forma a garantir a ajuda de uma consultoria externa. O projeto foi dividido em duas etapas:

- na primeira, buscava-se levantar o diagrama organizacional da empresa com a relação entre órgãos;
- na segunda, a meta era o desenvolvimento da metodologia.

Como na primeira etapa estariam envolvidas diversas pessoas de vários níveis hierárquicos, buscou-se o apoio de uma consultoria externa para coordenar as reuniões. Foi utilizado, então, o método JAD para o levantamento de informações. O grupo participante da primeira etapa era basicamente formado por técnicos, analistas e membros da Administração Geral. A equipe se manteve bastante motivada durante o desenvolvimento dos trabalhos. Ao final de cada encontro as informações geradas eram transferidas para o PC-CASE, de forma que na reunião seguinte o grupo tinha acesso a informações bem organizadas, geradas na reunião anterior. As reuniões transcorreram sem grandes conflitos e as divergências de opiniões acabavam gerando novas propostas, não havendo treinamento da equipe para o uso do método JAD, as pessoas aprendiam a técnica através de sua utilização e com o apoio do coordenador. Durante as reuniões não era

utilizado o apoio de nenhum equipamento eletrônico, somente o quadro branco. A contratação do consultor externo gerou alguma insatisfação, logo dissipada pela sua ativa e proveitosa participação.

Os membros da equipe não foram conhecer o trabalho dos outros participantes, mas através das interações para o desenvolvimento do projeto, todos aprenderam muito sobre áreas que não estavam diretamente ligadas com as suas, ou seja, consideraram que houve algum "aprendizado mútuo".

A dificuldade apontada como principal, foi a falta de iniciativa da maior parte dos participantes para resolver situações não previstas, as quais, necessitaram a intervenção do coordenador.

A introdução do novo método de trabalho estabeleceu aspectos positivos: a alta mobilização e participação dos usuários, mesmo que esse processo tenha se limitado a pessoal que ocupava níveis hierárquicos mais elevados e que as pessoas de níveis mais inferiores não tenham sido envolvidas, não havendo participação de todas as pessoas que seriam atingidas pela nova metodologia.

Os recursos financeiros eram obtidos facilmente, assim como a liberação das pessoas para os encontros se dava sem maiores problemas.

Os fatores que foram considerados como os que mais fortemente contribuíram para o sucesso da implantação da técnica participativa neste caso foram: "a decisão ter partido de cima", ou seja, a alta chefia do departamento estar interessada e patrocinar todos os requisitos necessários para o bom desenvolvimento dos trabalhos.

A segunda etapa não foi iniciada, pois houve alterações estruturais e a gerência que deveria assumir o prosseguimento dos trabalhos não se interessou em fazê-lo. Hoje a idéia da metodologia esta sendo revisada para a elaboração de uma nova proposta.

Segunda Experiência

A segunda experiência, iniciada pouco tempo após a primeira, visava desenvolver um sistema de informações voltado para a área de crédito. A equipe era formada por técnicos de informática (que eram responsáveis pela programação), analistas de sistemas e usuários da área de crédito. Neste caso foi usado JAD, mas sem o apoio de consultoria externa, pois algumas das pessoas envolvidas já tinham adquirido experiência participando no primeiro caso. Todos estavam motivados e participaram ativamente, mas pouco tempo após os trabalhos terem começado, houve um processo de descentralização organizacional da empresa e foram criadas gerencias de sistemas que passariam a

desenvolver sistemas locais. O gerente que ficou então responsável pela gerência relativa a área de crédito, não se motivou em continuar o desenvolvimento do sistema, pois considerava que o sistema era mais amplo do que sua gerência poderia abranger e tudo foi interrompido. Apesar da experiência ter sido interrompida, as reuniões realizadas geraram a definição dos processos, as entidades relacionais e os processos de entrada/saída.

Hoje o JAD não é mais usado na empresa pois a atividade de desenvolvimento está diminuindo.

Apesar da segunda experiência não ter sido finalizada, o uso de técnicas participativas (no caso JAD), nos dois casos, geraram percepção de vantagens que listaremos a seguir: viabilizar discussões que antes do uso da técnica seriam difíceis de serem realizadas; aumentar a produtividade; aumentar a qualidade dos produtos, pois eles eram decisão de todos e com isso não haviam reclamações posteriores.

Análise dos Resultados

Esta experiência é dividida em dois casos, sendo que os analisaremos em conjunto. O primeiro caso foi dividido em duas etapas: uma que buscava levantar o diagrama organizacional da empresa e outra que visava o desenvolvimento de uma metodologia de desenvolvimento de produtos de software. A primeira etapa foi terminada com sucesso. A contratação de uma consultoria externa, inicialmente, gerou resistências, logo dissipadas pelo desempenho satisfatório do consultor corroborando as afirmações de Nunamaker et al. [NUDV 93], que colocam que um líder/coordenador desempenha um papel muito importante, melhorando o processo de reunião e colaborando para o êxito. O consultor atuou como coordenador/líder de reuniões bem estruturadas, baseadas em perguntas e em atitudes que foram expostas por Minucucci [MINI 87] e detalhadas no capítulo 2.

Neste caso foi usado o PC-CASE, para a organização dos dados gerados durante as reuniões, constatando-se que sua utilização encontra-se de acordo com o que é descrito no Capítulo 3: "o uso de ferramentas e metodologias formais para apoiar reuniões e o desenvolvimento dos trabalhos, facilita a manipulação de informações, gera uma linguagem comum, colaborando para a melhoria da comunicação e conseqüente integração entre os membros do grupo".

As pessoas aprendiam as técnicas do método JAD durante a realização dos trabalhos, ou seja, não houve nenhum tipo de treinamento anterior. Durante as reuniões as

peças envolvidas, ou sejam: técnicos, analistas e membros da administração geral, trocaram informações e experiências, se enquadrando na proposta do método que não pressupõe vivências das práticas de trabalho. Não foi assinalada a existência de conflitos, entretanto, foi apontado um medo inicial de participar das discussões, exigindo que o consultor estimulasse uma participação mais ativa.

Como neste caso não houve desenvolvimento do sistema, não foi constatado reagrupamento de funções de análise/programação. Contudo, em geral, na empresa, essas funções são bem distintas.

A primeira etapa da primeira experiência pode ser finalizada por influência da gerência que a patrocinou, evidenciando o que coloca Soares Neto [SOAR 93] que um fator vital para o sucesso do método JAD é o comprometimento dos níveis superiores da empresa com o processo.

Não houve nenhum tipo de apoio eletrônico de comunicação, pois as reuniões se davam numa sala e segundo Soares Neto [SOAR 93], JAD independe de recursos, bastando haver espaço para se desenhar o modelo, já que a comunicação visual é fundamental.

A segunda etapa desta experiência não foi iniciada por causa de mudanças estruturais na empresa. A gerência que ficou responsável pela continuidade do trabalho não se interessou, corroborando as constatações de Carneiro [CARN 93], sobre a importância do patrocínio para a continuidade de desenvolvimento de projetos dentro de empresas.

A segunda experiência foi interrompida, também, por alterações estruturais. A princípio foi formada uma equipe composta de analistas de sistemas e usuários, todos bastante motivados. Mas a descentralização provocada pela alteração estrutural, alterou o poder das gerências, inviabilizando a continuidade do projeto. Apesar disso, um aspecto positivo constatado é a experiência ter deixado como herança pessoas capazes de liderar/coordenar encontros apoiados em JAD.

5.3.4 Empresa que Presta Consultoria

Este relato não pode ser considerado um estudo de caso, pois não refere-se a uma experiência específica, mas sim, a um conjunto de percepções obtidas a partir de várias experiências vivenciadas por um especialista em JAD, e que foi o consultor externo do caso

anterior. Tais informações colaboram para reforçar alguns aspectos expostos no método JAD e no caso acima exposto.

De acordo com o especialista, o grande número de projetos desenvolvidos com JAD, tem mostrado que o material necessário para a utilização do método resume-se a bloco de papel, fita gomada e canetas, não sendo necessário o apoio de nenhum meio eletrônico.

Os grupos devem possuir um coordenador capacitado metodologicamente e que estimule a criatividade dos membros, os quais, devem possuir, principalmente, as seguintes características: disponibilidade, competência, autoridade, ser representativo, estar motivado, persistência, consciência metodológica. Na prática, o coordenador deve definir um conjunto de perguntas e a linguagem de modelagem para representar as respostas, assim como, fazer muitas perguntas sendo disciplinado, para não interferir nas respostas. Não foi considerado necessário fornecer treinamento aos participantes para usarem o método, pois segundo o entrevistado, "o treinamento é a própria experiência de desenvolver os trabalhos". A participação efetiva de todos os membros do grupo, deve ser incentivada pelo coordenador, de modo que, ao final dos trabalhos todos se sintam um pouco "donos" do resultado.

É importante ressaltar a importância do escriba, ou seja, da pessoa responsável pelas anotações de todas as decisões tomadas. Ele precisa ter capacidade de resumir sem alterar as idéias, ser rápido e atento à todas as modificações propostas pelos participantes, reter, claramente, todas as conclusões e precisa ser organizado, para que ao final de cada reunião se tenha todas as decisões claramente anotadas de forma a otimizar a próxima reunião.

O aspecto levantado como o mais importante para o desenvolvimento de projetos usando JAD é o patrocínio da alta administração da empresa, ou seja, deve existir alguém que apoie totalmente os trabalhos e seus resultados. A técnica JAD é considerada ineficaz se não há patrocínio.

Em geral, durante as reuniões, se apresentam poucos conflitos entre os participantes, mas as vezes, se mostra necessário a intervenção de um árbitro técnico ou político, de modo a desfazer dúvidas e reforçar a coesão das idéias geradas. Sendo que, normalmente, esse arbitro é escolhido pelo próprio grupo. A coesão de opiniões apoia a visão coletiva do produto que está sendo gerado, e o especialista assinala que, nesse aspecto, "uma visão coletiva é mais do que a soma das visões individuais".

A troca de informações que acontece durante as reuniões faz com que cada participante tome conhecimento de aspectos que não fazem parte de seu universo de

trabalho. Esta interação resulta num aumento das fronteiras de informações de cada participante, gerando aprendizado mútuo.

Segundo o entrevistado, em relação as experiências já vivenciadas em empresas no desenvolvimento de produtos de software, o uso de JAD tem-se mostrado vantajoso em relação aos procedimentos usados anteriormente. Há aumento da produtividade, pois as decisões são mais definitivas e as discussões mais rápidas do que nos métodos tradicionais; há aumento da qualidade dos produtos gerados, sendo este um dos fatores que favorece fortemente o uso da técnica; há diminuição da manutenção dos produtos pois as informações vão sendo validadas durante sua geração, diminuindo o número de requisitos incompletos ou inválidos. Na maioria dos casos não há medidas para avaliar os custos, mas acredita-se que haja uma queda dos custos de desenvolvimento gerada pela rapidez com que os trabalhos se desenvolvem. Todos esses aspectos colaboram para que os sistemas possuam documentação mais organizada e que não precisa ser alterada constantemente.

O tamanho dos grupos não é considerado como um fator que influencie os resultados, visto que, a empresa já teve experiências com grupos pequenos de 6 e 7 pessoas, grupos médios de 15 e 18 pessoas e grupos grandes com 40 pessoas

5.4 Conclusões

Neste capítulo os estudos e posterior análise de casos apresentados, permitiu-nos submeter as diferentes experiências às colocações de vários autores.

Como forma de melhorar a visualização dos resultados, apresentaremos, de forma reduzida, uma tabela comparativa dos casos estudados, classificando as questões de acordo com a Tabela 4.1 e complementando-a com alguns critérios abordados durante a análise dos casos. Devido ao grande número de informações, a tabela será apresentada em duas páginas.

Tabela 5.2: Tabela de características dos casos estudados

	Empresa Petrolífera	Empresa atuante em estatística	Banco de Desenvolvimento
Objetivo	Reorganização e modernização empresarial, otimização de processos	Desenvolvimento de ferramenta cooperativa, implementação da ferramenta	Desenvolver metodologia customizada de desenvolvimento de produtos de software
Critérios	Qualitativos: participação de usuários para aumentar qualidade do produto gerado	Qualitativos: melhorar as relações entre usuários e equipe desenvolvedora	Qualitativos: aumentar a qualidade dos produtos
Base	Dinâmica de grupo	Prototipagem e aprendizado mútuo	Dinâmica de grupo e relações de trabalho
Temas	Democracia do ambiente de trabalho, participação, responsabilidades	Negociação, democracia no ambiente de trabalho, aceleração de projeto	Participação
Dinâmica das reuniões	Livre	Livre baseadas no protótipo	Estruturadas

	Empresa Petrolífera	Empresa atuante em estatística	Banco de Desenvolvimento
Responsável pela coordenação	Coordenador do projeto	Analista Chefe	Consultor externo
Estimulou aprendizado mútuo?	Não	Sim	Não
Apoio de protótipos	Sim	Sim	Não
Perspectivas sobre os usuários	Grupo limitado contendo alta chefia	Técnicos em estatística, analistas e técnicos em informática	Analistas, programadores e membros da direção
Apoio eletrônico de comunicação	Não	Não	Não
Utilização de ferramenta automatizada	Não	Não	Sim
Percepção de reagrupamento de funções fragmentadas	Não	Sim	Não

Comparando as tabelas 4.1 e 5.2 constatamos a dificuldade de identificar os casos acima descritos com os métodos cooperativos apresentados no capítulo 4.

No caso da empresa de consultoria, não é apresentado nenhuma experiência específica, e sim, um resumo de percepções de um especialista em JAD, bem sucedido na implantação e disseminação desse método.

O caso do Banco de Desenvolvimento é declarado como experiência em JAD, sendo que podemos observar características do caso que não se enquadram perfeitamente àquelas propostas por tal método, como por exemplo a questão "Critérios".

Já os casos da empresa Petroquímica e Estatística são mais complexos, com suas características bem diversificadas, não sendo possível estabelecer uma classificação formal, e sim, aproximativa. O caso da empresa estatística possui oito questões em comum com o Projeto Participativo, mas os "Objetivos" divergem justamente no aspecto que é considerado base do Projeto Participativo: melhorar a conscientização sobre a natureza e os postos de trabalho. No caso em questão não percebemos tal preocupação.

O caso da empresa Petroquímica, possui três aspectos que se aproximam na proposta de JAD e quatro em Projeto Participativo, tendo dois que não se identificam com nenhuma questão. Talvez em sua essência ele se aproxime mais de Projeto Participativo do que de JAD por proporcionar reuniões mais livres, sem estruturas rígidas.

Alguns métodos cooperativos buscam integrar os usuários finais em todas as etapas do desenvolvimento. Neste estudo pudemos perceber que o usuário participou do desenvolvimento dos projetos, contudo, em nenhum dos casos, foi observado sua participação integral em todas as fases do desenvolvimento dos projetos, enfraquecendo as vantagens democráticas proporcionadas pelo método e dificultando seu total envolvimento.

Em relação ao uso de ferramentas de apoio ao trabalho cooperativo, Bock [BOCK 92] descreve uma pesquisa onde foi identificado que o correio eletrônico é a primeira ferramenta de apoio ao trabalho cooperativo, entretanto em nosso estudo não foi constatado o uso de nenhuma ferramenta de comunicação.

A seguir listaremos os benefícios citados e que foram obtidos a partir da introdução dos métodos cooperativos nos casos estudados:

- o processo cooperativo tornou todos um pouco responsáveis pelo sistema, aumentando a qualidade dos produtos, pois eles eram fruto de todos e com isso não haviam reclamações posteriores;
- a perspectiva de gerar um produto com alta qualidade, facilitando sua implantação na área administrativa da empresa;
- rapidez no desenvolvimento dos produtos, apesar de um dos casos não ter conseguido cumprir os prazos;
- facilidade para realocar pessoas;
- viabilizou discussões que antes do uso do método eram difíceis de serem realizadas;
- aumento da produtividade, pois as decisões são mais definitivas e o desenvolvimento como um todo é mais rápido do que nos métodos tradicionais;

- diminuição no trabalho de manutenção dos produtos pois as informações vão sendo validadas durante sua geração;

- na maioria dos casos não há medidas para avaliar os custos, mas existe percepção de queda dos custos de desenvolvimento gerada pela rapidez com que os trabalhos se desenvolvem.

CAPÍTULO 6

Conclusões

Neste capítulo, pretende-se apresentar os principais resultados encontrados nesta pesquisa empírica e as condições que deles derivam.

A área de trabalho cooperativo vem recebendo muita atenção por parte de pesquisadores, aplicando-se à diferentes domínios do conhecimento. Neste estudo buscamos identificar elementos cooperativos que se aplicassem à Engenharia de Software, contribuindo para aumentar a qualidade dos produtos e a satisfação do usuário, através de seu total envolvimento no processo de desenvolvimento. Nesse sentido, o aspecto considerado mais importante do processo de implantação de métodos cooperativos é a possibilidade de intensificar a participação de todas as pessoas que serão atingidas pelos sistemas.

Verifica-se que, no Brasil, já existem várias iniciativas, principalmente em universidades, mas a difusão do enfoque cooperativo nas empresas é muito pequeno, com experiências isoladas e parca divulgação de resultados. Esse aspecto ficou evidenciado nesta pesquisa.

As experiências, que formam o suporte de nossas conclusões, foram realizadas em empresas estatais, onde existe um contexto favorável, com ambientes mais democráticos e menos competitivos, em contraste com o clima de disputa instalado em grande parte das empresas privadas. Essas empresas desenvolvem sistemas de informação, que geram influências na organização do trabalho, não havendo casos de estudo que tivessem abordado o desenvolvimento de software básico. A estrutura democrática é fundamental para que as pessoas possam cooperar sem se sentirem ameaçadas em seu poder, favorecendo a união de forças de todos os envolvidos, no intuito de alcançar a meta desejada.

Em geral as experiências aconteceram sem nenhum tipo de apoio automatizado, seja através de sistemas eletrônicos de comunicação, decisão, coordenação ou ferramentas CASE. Na verdade, é primordial que a adoção de ferramentas seja

precedida de uma mudança cultural da empresa ou dos setores envolvidos, enfatizando a importância do contexto social antes do eletrônico. Com esta ótica procura-se preparar o ambiente de modo a vencer resistências que a introdução de novas ferramentas pode suscitar. Mostra-se de suma importância a existência de uma etapa, anterior a introdução de novas tecnologias, que deve privilegiar a otimização e a racionalização dos processos de trabalho.

No enfoque da Engenharia de Software, a introdução de métodos cooperativos demanda que haja uma fase anterior à adoção do novo processo de desenvolvimento, em que sejam esclarecidas as vantagens de tais métodos, para que as pessoas envolvidas se sintam motivadas e seguras em relação às mudanças que possam ocorrer na organização de suas tarefas. Elas devem ter consciência de que a participação gera poder em relação a possibilidade de interferir no desenvolvimento dos sistemas que influenciarão suas práticas de trabalho.

Temos que ressaltar que tais metodologias ainda não foram suficientemente testadas e adotadas para que possamos afirmar qual delas é a mais abrangente ou melhor. Cada uma delas se adequa aos interesses específicos segundo as linhas de atuação das organizações no mercado econômico, social e político em que estão imersas. Corroborando as constatações do capítulo 5, segundo as quais, a complexidade de classificação de cada experiência com um método específico é muito grande, sendo que cada método fica sujeito as adaptações das instituições onde são aplicados.

Em relação aos estudos de casos, gostaríamos de assinalar a dificuldade de análise provocada por uma visão muito otimista que os entrevistados tinham de suas experiências, e que colaborou para encobrir possíveis problemas. Outro aspecto que pode ter contribuído para as conclusões positivas que encontramos, foi a impossibilidade de entrevistarmos os usuários finais, que são aqueles que poderiam ter apontado alguma dificuldade que não estivessem compartilhando com seus superiores. Tais restrições nos foram colocadas pelas pessoas contactadas inicialmente, em geral, responsáveis diretos pelo desenvolvimento dos sistemas e que não abriram espaço para que se realizassem entrevistas com usuários que ocupavam níveis hierárquicos mais baixos. Apesar da fraca motivação e esclarecimento em relação aos métodos participativos, as pessoas entrevistadas, em geral, não opuseram resistências à introdução de novos métodos, mas, paradoxalmente, mostraram-se inseguras em relação a introdução dos computadores no ambiente de trabalho.

O estudo de casos se mostrou revelador ao evidenciar como alguns aspectos abordados pelos diferentes pesquisadores, se comportam na sua aplicação prática:

- ♦ na maioria, as reuniões se passavam sem maiores problemas não havendo necessidade de utilização de métodos de resolução de conflitos;
- ♦ o coordenador foi considerado pessoa-chave para que os trabalhos se desenvolvessem harmoniosamente e alcançassem os objetivos esperados;
- os usuários finais ainda têm sua participação limitada, observando-se uma tendência a escolha de pessoas que ocupam cargos de chefia;
- ♦ o uso de ferramentas também é pequeno e não constatamos a utilização de nenhum apoio eletrônico de comunicação, não evidenciando as constatações de Carneiro [CARN 93] e Pinto [PINT 93] sobre a complexidade da implantação de novas metodologias sem o apoio de ferramentas;
- ♦ a introdução das novas formas de trabalho geraram resistências vinculadas ao medo da perda do poder.

Em suma, gostaríamos de apontar alguns critérios que evidenciaríamos a natureza cooperativa do trabalho e não somente, pessoas trabalhando juntas:

- ♦ os trabalhos devem ser integrados e dependentes, não pode ser uma divisão de tarefas de modo que ao final, reunido-se todas as tarefas obtenha-se o trabalho completo, para que todos se sintam um pouco "donos" do resultado;
- ♦ no desenvolvimento de produtos de software os analistas, programadores e usuários trabalhem realmente juntos, em todas as etapas do ciclo-de-vida;
- ♦ importância do trabalho inicial de conscientização, necessidade de mudança cultural e união de forças para se alcançar, mais rapidamente e com maior eficácia, os objetivos;
- ♦ os softwares desenvolvidos têm que ter como objetivo potencializar as capacidades humanas e não substituir o trabalho humano.

Finalmente, nossa proposta é recomendar uma importante pesquisa das questões envolvidas e formulação de alternativas da aplicação da cooperação em desenvolvimento de software, assim como, os avanços na área de CSCW poderão fornecer subsídios mais sólidos para apoiar o desenvolvimento de ferramentas de groupware que suportem as etapas do ciclo-de-vida dos produtos. Outra direção possível de pesquisa é a incorporação de sistemas especialistas ou sistemas baseados em conhecimentos nos groupwares.

Bibliografia

- [ALLE 90] ALLEN,C. "Definitions of Groupware", Applied Groupware, Vol 1, pp.1-2, 1990.
- [ANHO 91] ANDERSEN,Peter; HOLMQUIST,Berit. "Language, Perspectives and Design", Design at Work: Cooperative Design of Computer Systems, pp.91-119, 1991.
- [AUST 90] AUSTIN,Laurel; LIKER,Jeffrey; MCLEOD,Poppy. "Determinants and Patterns of Control over Technology in a Computerized Meeting Room", Proceedings of CSCW, pp. 39-51, out 1990.
- [BANN 91] BANNON,Liam. "From human factors to human actors: The role of psychology and human-computer interaction studies in system design", Design at Work: Cooperative Design of Computer Systems, pp.25-44, 1991.
- [BASC 91] BANNON,Liam; SCHIMIDT,Kfeld. "CSCW: Four Characters in Search of a Context", Studies in Computer Supported Cooperative Work, Editado por John Bowers e Steven Benford, Elsevir Science Publiser, pp.3-16, Inglaterra, 1991.
- [BARN 68] BARNARD, C. "The Functions of the Executive", Harvard University - Edição de 30º aniversário, Press, 1968.
- [BASF 92] BATISTA,Thais; SANTOS,Heloísa; FUKS,Hugo. "Cooperatividade na Manipulação de Documentos Multimídia", Monografia em Ciência da Computação, PUC/RJ, nº 3, pp.1-20, 1992.
- [BEDA 91] BERSOFF,Edward; DAVIS,Alan. "Impacts of Life Cycle Models on Software", Communications of the ACM, Vol 34, nº 8, pp. 104-118, ago 1991.
- [BJER 93] BJERKES,Gro. "Some PD Advice", Communications of ACM, Vol 36, nº 4, pp. 39 - junho 1993.

- [BOCK 92] BOCK, Geoffrey. "Groupware: The Next Generation for Information Processing ?", IEEE Computer Society Press, Califórnia, 1992.
- [BOGR 91] BODKER,Susanne; GRONBAEK, Kaj. "Design in Action: From Prototyping by Demonstration to Cooperative Prototyping", Design at Work: Cooperative Design of Computer Systems, Lawrence Erlbaum, NY, pp.197-218, 1991.
- [BOG2 91] BODKER2,Susanne; GRONBAEK, Kaj. "Cooperative Prototyping: Users and Designs in Mutual Activity", International Journal Man-Machine Studies, vol 34, nº 2, pp.453-477, março 1991.
- [BOEH 76] BOEHM,Barry. "Software Engineering", IEEE Trans. Computers, pp.1226-1241, Dez 1976.
- [BOEH 91] BOEHM,Barry. "A Spiral Model of Software Development and Enhancement", Validation and Verifying Knowledge-Based Systems, editada por Uma Gupta, pp.202-213, USA, 1991.
- [BORO 89] BOEHM,Barry; ROSS,Rony. "THEORY-W Software Project Management: Principles and Examples", IEEE Transactions on Software Engineering, - vol 15, nº 7, pp.902-916, julho 1989.
- [CAMA 92] CAMARGO,Cláudia. "FLECHA: Um Editor Gráfico Cooperativo para o Modelo de Objetos do Geotaba", Tese de Msc, COPPE-Programa de Engenharia e Computação/UFRJ, dez 1992.
- [CAME 83] CAMERON,John. "Tutorial-JSP&JSP: The Jackson Approach to Software Development", IEEE Computer Society, New York, The Institute of Electrical and Eletronics Engineering, 1983.
- [CAWG 93] CARMEL,Erran; WHITAKER,Randall; GEORGE,Joey. "PD and Joint Application Design: A Transatlantic Comparison", Communications of ACM, vol 36, nº 4, pp.40-48, junho 1993.
- [CARN 93] CARNEIRO,Denise. "Adoção de Tecnologias de Processo no Desenvolvimento de Software: Estudo de Casos", Tese de Msc., COPPE-Programa de Engenharia de Sistemas e Computação/UFRJ, nov. 1993.
- [CAFU 92] CARVALHO,Sônia; FUKS,Hugo. "Groupware: A Incorporação de Aspectos Sociais ao Software", Monografia em Ciências da Computação PUC/RJ, nº 6, pp 1-23, 1992.
- [CADR 91] CASTRO,Leirton; DRUMMOND, Rogério. "SISTRAC: Sistema de Suporte a Trabalho Cooperativo", Projeto A_HAND, Departamento de Ciência da Computação, UNICAMP, pp. 1-17 , 1991.

- [CHEN 94] CHEN,Hsinchun. "Colaborative Systems: Solving the Vocabulare Problem", COMPUTER, pp.58-66, maio 1994.
- [CHIT 91] CHITMAN, Edith. "Estudo e Classificação de modelos de ciclo de vida", Tese Msc., COPPE, Programa de Engenharia em Sistemas e Computação/ UFRJ, maio 1991.
- [CLBE 93] CLEMENT,Andrew; BESSELLAAR,Peter. "A Retrospective Look at PD Projects", Communications of the ACM, Vol 36, nº 4, pp.29-37, junho 1993.
- [COTH 91] COCKBURN, A.J.G.; THIMBLEBY, H. "Reflexive perspective of CSCW", SIGCHI Bulletin, vol 23 , nº 3, pp.22-24, julho 1991.
- [CSTB 90] COMPUTER SCIENCE AND TECHNOLOGY BOARD. "Scaling-up: A reserch Agenda for Software Engineering", Communications of ACM, Vol 33, nº 3, pp.281-293, março 1990.
- [CRIS 91] CRISPIM,Emilia. "Avaliação de Métodos para o Desenvolvimento de Software", Tese de Msc., COPPE, Programa em Engenharia de Sistemas e Computação/UFRJ, maio 1991.
- [CURT 91] CURTIS,Bill. "Technie as Non-Technological Factors in Software Engineering ?", 13º International Conference Software Engineering, pp. 147-148, 1991.
- [DATA 92] DATANEWS. "Terapia de Grupo na Informática", Datanews, pp.10-12, 06 de outubro 1992 .
- [DEMA 78] DEMARCO, Tom. "Structured Analysis and System Specification", New York, Yordon Press, 1978.
- [DEMA 91] DEMARCO, Tom. "Non-Technological Issues in Software Engineering", 13º International Conference Software Engineering, pp.149-150, 1991.
- [DUFL 92] DUARTE,Renato; FUKS,Hugo; LUCENA,Carlos. "Software Design Cooperativo", Monografia em Ciências da Computação PUC/RJ, pp.1-16, 1992.
- [ELGR 91] ELLIS,C.A; GIBBS,S.J; REIN,G.L. "Groupware: Some Issues and Experiences", Communication of the ACM, Vol 34, nº 1, jan. 1991.
- [FARI 91] FARIA,Eduardo. "O sistema de hipertexto X", Oficinas de Informática na Educação, COPPE, Programa de Engenharia de Sistemas e Computação/UFRJ, nov 1991.

- [FIUR 81] FISHER,R; URY,W. "Getting to Yes", Boston, MA, Houghton-Mifflin, 1981.
- [FRID 93] FRIEDMAN,Andrew. "Computer Systems Development: History, Organization and Implementation", Jonh Wiley & Sons Ltda, Inglaterra, Reprinted fev 1993.
- [FRCO 93] FRIEDMAN,Andrew; CORNFORD,D.S. "Computer Systems Development: History, Organization and Implementation", Jonh Wiley & Sons Ltda, Inglaterra, 1993.
- [GHJM 91] GHEZZI,Carlo; JAZAYERI,Mehdi; MANDRIOLI,Dino. "Fundamentals of Software Engineering", Prentice Hall Inc., 1991.
- [GREE 91] GREENBERG,Saul. "Computer-supported cooperative work and groupware : an introduction to the special issues", Int J. Man-machine studies n°34, pp.133-141, 1991.
- [GRE2 91] GREENBERG,Saul. "An Annotated Bibliography of Computer-Supported Cooperative Work", SIGCHI Bulletin, Vol 23 n°3, pp.29-62, jul 1991.
- [GRKY 91] GREENBAUM,Joan; KYNG,Morten. "Introduction: Situated Design", Design at work: Cooperative design of Computer Systems, Lawrence Erlbaum Associates, pp.1-24, 1991.
- [GRON 89] GRONBAEK,Kaj. "Rapid Prototyping with fourth Generation Systems-An Empirical Study", OFFICE: Technology and People n° 5, pp.105-125, 1989.
- [GRUD 92] GRUDIN,Jonathan. "A Tale of two cities: reflections on CSCW in Europe and the United States", SIGCHI Bulletin, vol 23, n°3, pp.22-24, julho 1992.
- [GRUD 88] GRUDIN,Jonathan. "Why CSCW Applications Fail: Problems in the design and Evaluation of Organizational Interfaces", Proceeding of the second conference on Computer-supported Cooperative Work, ACM Press, pp.85-93, 1988.
- [GRUD 91] GRUDIN,Jonathan. "On Computer-Supported Cooperative Work-CSCW Introduction", Communication of ACM, Vol 34 N° 12, pp.30-34, dez 1991.
- [GRUD 94] GRUDIN,Jonathan. "Groupware and Social Dynamics: Eight Challenges for Developers", Communication of the ACM, Vol 34, n° 1, pp.92-105, jan 1994.

- [GRU2 94] GRUDIN,Jonathan. "Computer-Supported Cooperative Work: History and Focus", COMPUTER, pp. 19-25, maio 1994.
- [HANA 89] HANATA,Shuetsu. "Management of Software Production", Japanese Perspectives in Software Engineering, Addison Wesley, pp.279-301, 1989.
- [HASS 91] HASSARD, Jack. "Cooperative Learning and the teaching of science", Georgia State University, 1991.
- [HIRS 85] HIRSCHHEIM,Rudy. "Office Automation: A Social and Organizational Perspective", John Wiley & Sons Ltda, 1985.
- [ISHI 92] ISHII,Hiroshi. "Computer and Video Fusion Approach of Team Workstation", SIGOIS Bulletin, Vol 13, n° 1, pp.11-12, abril 1992.
- [JEVA 93] JESSUP,Leonard; VALACICH, Joseph. "On the Study of Group Support Systems: An Introduction to Group Support Systems Research and Development", Group Support Systems, Editado por Jessup e Valacich, Macmillan Publishing Company, NY, pp. 3-7, 1993.
- [JOJO 82] JOHNSON-LENZ,Peter; JOHNSON-LENZ,Trudy. "Groupware: The Process and Impacts of Design Choices, Computer-Mediated Communications Systems: Status and Evaluations", E.B. Kerr e S.K. Hiltz, Academic Press, NY, 1982.
- [JOJO 91] JOHNSON-LENZ,Peter; JOHNSON-LENZ,Trudy. "Post-Mechanistic Groupware Primitives: Rhythms, Boundaries and Containers", International Journal Man-Machine Studies, Vol 34 n° 2, pp.395-417, março 1991.
- [KELL 91] KELLNER, Marc I. "Non-Technological Issues in Software Engineering", Panel Session Overview, 13° International Conference Software Engineering, USA, pp. 144-146, 1991.
- [KING 91] KING,Morten. "Design for Cooperation: Cooperation for Design", Communication of ACM, Vol 34, n° 12, pp. 65-73, dez 1991.
- [KLIN 91] KLING,Rob. "Cooperation, Coordination and Control in Computer Supported Work", Communications of ACM, vol 34, n° 12, pp. 83-88, dez 1991.
- [LEWE 92] LEWE, Henrik; BARENT,Volker. SIGOIS BULLETIN, Vol 13, n° 1, pp. 19-20, Abril 1992 .

- [LEOM 90] LEWIS, Ted; OMAN, Paul. "The Challenge of Software Development", IEEE Software, pp. 9-12, Novembro 1990.
- [MCPA 91] MCMENAMIN, Stephen; PALMER, John. "Análise Essencial de Sistemas", Ed. Mc Graw Ltda. e Makron Books Ltda., 1991.
- [MACR 90] MALONE, Thomas; CROWSTON, Kevin. "What is Coordination Theory and how Can it Help Design Cooperative Work Systems ?", Proceedings of CSCW, pp. 357-370, out 1990 .
- [MALU 91] MANNING, M. L. ;LUCKING, R. "The what, why and how of cooperative learning", The Social Studies, Vol 82, nº 3, maio/junho 1991.
- [MARC 91] MARCA, David. "Augmenting SADT to Develop Computer Support for Cooperative Work", IEEE, pp. 94-103, 1991.
- [MABO 92] MARCA, David; BOCK, Geoffrey. "Groupware: Software for Computer-Supported Cooperative Work", IEEE Computer Society Press, Califórnia, 1992.
- [MARI 92] MARIANO, Sandra. "Automação de Escritório e Competitividade Empresarial: Um estudos de Casos", COPPE, Programa em Engenharia de Sistemas e Computação/UFRJ, RJ, março 1992.
- [MAFU 92] MATHIAS, Adolfo; FUKS, Hugo. "Resolução de Conflitos e Trabalho em Grupo", Monografias em Ciência da Computação PUC/RJ, nº 5, pp. 1-29, 1992.
- [MINI 87] MINICUCCI, Agostinho. "Técnicas do Trabalho de Grupo", Editora Atlas, SP- 1987.
- [MOLI 93] MOYSÉS, Elizabeth; LIMA, Luiz. "Design Participativo de Sistemas, Enfoque do Usuário", monografia da área de informática e sociedade, COPPE/UFRJ, 1993.
- [MUNF 88] MUNFORD, Enid. "El Diseño Participativo de Sistemas: Prática e Teoria", La Automatization y el Futuro del Trabajo, Ministério de Trabajo y Seguridad Social, pp. 609-626, Madrid, 1988.
- [MUSS 87] MUSSIO, Piero, "Introdução à Informática", Ed. Vozes, 1987.
- [NAAN 92] NAKAUCHI, Yasuchi; ANZAI, Yuichiro. "Open CSCW Systems for Distributed Environments", Sigois Bulletin, Vol 13, Nº1, pp 23-25, abril 1992.

- [NORM 91] NORMAN,Donald. "Collaborative Computing: Collaboration First, Computing Second", Communications of ACM, Vol 34, n° 12, pp.88-90, Dez 1991.
- [NOCR 91] NORMAN,Ronald; CHIKOFISKY,Elliot; RUBENSTEIN,Burt. "CASE at the Start of the 1990's", International Conference in Software Engineering, pp. 128-139, 1991.
- [NUDV 93] NUNAMAKER,J.F.; DENNIS,Alan; VALLACICH,Joseph; VOGEL,Douglas; GEORGE,Jeoy. "Group Support Systems Research: Experience from the Lab and Field", Group Support Systems-New Perspectives, Editado por Jessup e Valacich, Macmillan Publishing Company, NY, pp. 125-145, 1993.
- [PINT 93] PINTO,Paulo E. "Um Estudo Sobre o Processo de Difusão de Engenharia de Software: O Caso da Petrobrás, Tese de mestrado, COPPE/UFRJ, Programa em Engenharia de Sistemas e Computação, RJ, 1993.
- [PRES 92] PRESSMAN,Roger. "Software Engineering", A Practioner Approach, 3° edição, McGraw-Hill International Editions, 1992.
- [RESC 90] REDER,Stephen; SCHWAB, Robert. "The Temporal Structure of Cooperative Activity", Proceedings of CSCW , pp. 303-316, out 1990.
- [ROAB 87] ROCHA,Ana; AGUIAR,Teresa; BLASCHEK,José. "Ambientes para Desenvolvimento de Software: Definição de Termos", Relatório Técnico ES-137/87, COPPE, Programa de Engenharia de Sistemas e Computação/UFRJ, 1987.
- [ROCH 92] ROCHA,Ana Regina. "Processo de Desenvolvimento de Aplicações", Instituto Latino-americano de Tecnologia, RJ, 1992.
- [RODD 92] RODDEN,Tom. "ECSCW'91 Developers Workshop", Sigois Bulletin, vol 13, n° 1, pp. 3-5, abril 1992.
- [ROSS 85] ROSS,Douglas. "Applications and Extentions of SADT", IEEE Computer, Vol 18, n° 4, pp. 25-34, abril 1985.
- [SCHL 92] SCHLICHTER,Johann. "Collaboration in Distributed Document Processing", SIGOIS Bulletin, Vol 13, n° 1, pp. 32-33, abril 1992.
- [SCHU 91] SCHLUMBERGER,Maurice. "Software Engineering Mangement", 13° International Conference Software Engineering, pp. 152-153, 1991.

- [SHAW 90] SHAW, Mary. "Prospects for an Engineering Discipline of Software", IEEE Software, pp. 15-24, Nov 1990.
- [SHME 88] SHLAER, Sally; MELLOR, Sephen. "Object-Oriented Systems Analysis", Englewood Cliffs, New Jersey, Prentice-Hall, 1988.
- [SILF 92] SILVA, Sidney; FUKS, Hugo. "Sistemas de Informação de Escritório Usando Redes de Petri", Monografia PUC/RJ nº 7/92, 1992.
- [SIVI 90] SIVIERI, Rodolpho. "Engenharia de Software: Gerenciar Mudanças, o grande desafio", Artigo Técnico da Petrobrás, pp. 1-7, 1990.
- [SOAR 93] SOARES NETO, Horácio, "Análise Vital de Sistemas", DATAMEC, 1993.
- [STEV 88] STEVENS, Wayne. "Projeto Estruturado de Sistemas", Editora Campus, RJ, 1988.
- [STHA 91] STHAL, Marimar. "Ambientes de ensino-aprendizagem computadorizados: da sala de aula convencional ao mundo da fantasia". Oficina de trabalho ambientes de ensino/aprendizagem apoiados por computador, COPPE, Programa de Engenharia de Sistemas e Computação/UFRJ, 1991.
- [TIER 91] TIERNEY, Margaret. "The Formation and Fragmentation of Computing as an Occupation: A Review of Shifting 'Expertise'", Working Paper Series of the Programme on Information & Communication Technologies, Edinburg PICT nº 25, pp. 1-31, 1991.
- [THEM 92] THE MOCCA GROUP, An Environment for CSCW Applications, Sigois Bulletin, Vol 13, Nº1, pp 21-23, abril 1992.
- [TOSO 92] TORNAGHI, Alberto; SOUZA, Jano. "MULEC, Multieditor Cooperativo, Workshop em Trabalho Cooperativo", Pensamento Crítico e Software Educacional, COPPE, Programa de Engenharia de Sistemas e Computação/UFRJ, 1992.
- [TOBE 94] TOU, Ivan; BERSON, Steven; ESTRIN, Gerald; ETEROVIC, Yadrán; WU, Elsie. "Prototyping Synchronous Group Applications", COMPUTER, pp.48-56, maio 1994.
- [TRAV 93] TRAVASSOS2, Guilherme. "Trabalho Cooperativo e o Desenvolvimento de Produtos de Software: Estudo de Caso com o Método de Coad/Yordon", artigo em preparação COPPE, Programa de Engenharia de Sistemas e Computação/UFRJ, junho 1993.

- [TULL 91] TULLY, Collin. "A failure of Management Nerve and Vision" -13^o International Conference Software Engineering, pp. 154-155, 1991.
- [WALT 88] WALTON, Richard. "Nuevas Perspectivas en el mundo del trabajo", La Automatization y el Futuro del Trabajo, Ministério de Trabajo y Seguridade Social, pp. 629-642, Madrid, 1988.
- [WEXE 92] WEXELBLAT, Alan. "Groups Without Groupware", SIGOIS Bulletin, vol 13, n^o 1, pp. 33-35, abril 1992.
- [WHBC 91] WHITAKE R, S.; BRENNAN, Susan; CLARK, H.H. "Coordinating Activity: an Analisis of Interaction in Computer-suported Cooperative Work", ACM-CHI, 1991.
- [WILS 91] WILSON, P. "Computer Supported Cooperativa Work (CSCW): Origins, concepts and research initiatives", Computer Networks and ISDN systems, n^o 23, pp. 91-95, 1991.
- [WIFL 86] WINOGRAD, Terry; FLORES, Fernando. "Understanding Computers and Cognition", Ablex Publishing CO; Norwood, NJ, 1986.
- [YAMA91] YAMAMOTO, Lídia; DRUMOND, Rogério. "Sistema de Teleconferência", Projeto A_HAND, Departamento de Ciência da Computação, UNICAMP, pp.18-28, 1991.