

# Awareness em Sistemas de Groupware

Manuele Kirsch Pinheiro  
(*manuele@inf.ufrgs.br*)

José Valdeni de Lima  
(*valdeni@inf.ufrgs.br*)

Marcos R. S. Borges  
(*mborges@nce.ufrj.br*)

*Instituto de Informática*  
*Universidade Federal do Rio Grande do Sul*  
*Porto Alegre - RS - Brasil*  
*Caixa Postal: 15064 - CEP 91501-970*

*Instituto de Informática*  
*Universidade Federal do Rio Grande do Sul*  
*Porto Alegre - RS - Brasil*  
*Caixa Postal: 15064 - CEP 91501-970*

*Núcleo de Computação Eletrônica*  
*Universidade Federal do Rio de Janeiro*  
*Rio de Janeiro - RJ - Brasil*  
*Caixa Postal 2324 - CEP 20001-970*

## Resumo

*Este artigo apresenta um framework que resume aspectos relevantes no suporte a awareness em sistemas de groupware. Awareness (ou percepção) é o conhecimento sobre o grupo e suas atividades passadas e presentes, e constitui ponto vital para o trabalho cooperativo, sem o qual este trabalho fica descoordenado e perde em qualidade e eficiência. O framework aqui apresentado comporta-se, na verdade, como uma classificação ampla e única para os requisitos necessários a um suporte a awareness adequado. Sobre este framework foram classificadas diversas ferramentas de groupware disponíveis, formando assim uma visão geral das realizações neste domínio.*

## 1. Introdução

A busca por maior eficiência na solução de problemas complexos fez com que atividades passassem a ser resolvidas em grupos, com o objetivo de fazer com que a produção do grupo todo supere a de cada membro isolado. Entretanto, cooperar não é algo fácil. Sua definição de "trabalhar ou agir em conjunto para um objetivo comum" não captura vários aspectos desse processo [2]. Um destes aspectos é a falta de contexto sobre as atividades dos colegas e do grupo, a qual pode gerar redundâncias, inconsistências e contradições dentro do trabalho do grupo. Isso pode gerar um trabalho truncado, sem coesão de idéias, ou incompleto, prejudicando o seu resultado em qualidade e eficiência, podendo até não atingir seus objetivos. Para evitar este tipo de situação, é necessário dar aos membros do grupo acesso a este contexto, ao que se dá o nome de *awareness* ou percepção. *Awareness* é o conhecimento geral sobre as atividades e sobre o grupo. É o conhecimento sobre o que aconteceu, o que vem acontecendo, o que está se passando agora e o que poderá acontecer dentro das atividades do grupo, e sobre o próprio grupo, seus objetivos e sua estrutura.

Dada sua importância, o estudo da percepção e seus mecanismos de suporte têm sido alvo de diversas pesquisas na área de Trabalho Cooperativo Suportado por Computador (CSCW), produzindo uma série de resultados publicados ao longo dos anos. Este artigo objetiva organizar a essência destes resultados através de um *framework*, contendo as principais características para o suporte a *awareness*, apresentando-se como uma proposta de classificação ampla para os mecanismos de suporte a percepção disponíveis. Para tanto, este artigo inicia com uma rápida revisão sobre o que é *awareness* e segue apresentando a classificação, feita através de seis questões básicas que analisam as necessidades de sistemas de *groupware* síncronos e assíncronos, e termina com alguns exemplos de ferramentas de *groupware* classificados e as conclusões deste artigo.

## 2. CSCW e Awareness

A área de CSCW vem buscando, desde o seu princípio, meios de suportar adequadamente o trabalho em equipe. Contudo, este suporte apresenta vários problemas. Um destes é a falta de contexto entre os participantes, que ocorre quando os membros de um grupo de trabalho desconhecem o que seus colegas estão fazendo, ou não sabem onde suas atividades se encaixam no trabalho como um todo, nem qual é a situação desse trabalho. Segundo Borges et al. [3], esse contexto é um ponto importante em sistemas cooperativos,

estendendo-se para não somente o conteúdo das contribuições individuais, mas também ao seu significado para o grupo como um todo e seu objetivo.

O fornecimento deste contexto aos membros de um grupo é chamado *awareness* ou percepção. *Awareness* pode ser conceituada como a contextualização das atividades individuais através da compreensão das atividades realizadas por outras pessoas [1]. Em outras palavras, *awareness* refere-se a ter conhecimento das atividades do grupo, saber o que aconteceu, o que está acontecendo e/ou o que poderá vir a acontecer, além do próprio conhecimento do que é este trabalho e o grupo. Generalizando, "*awareness* significa uma compreensão do estado total do sistema, incluindo atividades passadas, status atual e opções futuras" [22].

Estar atento aos colegas e às atividades por eles desempenhadas representa papel importante na fluidez e na naturalidade do trabalho [14], o que faz da *awareness* peça chave para qualquer forma de cooperação, uma vez que perceber, reconhecer e compreender as atividades dos outros é um requisito básico para a interação humana e a comunicação em geral [22]. Por exemplo, a interação em um mesmo local tem alto grau de *awareness*. Pode-se facilmente saber se um colega está no escritório, se está trabalhando em um projeto ou se está de mau humor e, com o tempo, aprende-se também sobre sua agenda, hábitos e interesses. Tudo isso contribui para conhecer melhor os colegas, criando uma base para as interações, sem a qual estas ficam mais formais e menos fluídas [15]. *Awareness* permite a cada usuário coordenar e estruturar seu trabalho, pois possibilita ao mesmo perceber e compreender no que os demais estão trabalhando. Ela também mostra oportunidades para comunicação informal e espontânea, suportando o estabelecimento e a manutenção de convenções no grupo. Sem percepção, o trabalho cooperativo coordenado é quase impossível [22]. Considerando, por exemplo, o caso do desenvolvimento cooperativo de software, é necessário que os membros tenham noção do contexto de suas atividades no contexto geral do processo para que possam perceber o andamento das atividades realizadas pelos demais e compreender como os resultados gerados por estas atividades podem ser conjugados com os seus próprios, para mais rapidamente chegarem ao resultado final [1]. Neste caso, a ausência de suporte à percepção torna quase impossível a produção de um software consistente e de qualidade, de forma eficiente.

Estas informações de percepção são, segundo Mariani [16], fornecidas por um conjunto de mecanismos, que dão ao usuário indicação das atividades dos demais. Estes mecanismos fornecem oportunidade aos indivíduos de ganharem uma compreensão do trabalho de seus colegas e usar esta informação para coordenar suas atividades para o grupo [10]. Estudando-se vários mecanismos de suporte apresentados na literatura pôde-se identificar um conjunto limitado de características importantes, que vêm sendo observadas nas propostas destes mecanismos. A identificação deste conjunto permitiu a delimitação de um *framework* com tais características importantes, que idealmente deveriam ser observadas pelas ferramentas de *groupware*. Sobre este *framework* pode-se distribuir os mecanismos de suporte à percepção hoje mais difundidos, criando assim uma classificação geral para o assunto, a qual poderá também ajudar na condução de novas contribuições. O *framework* aqui proposto age também como uma reunião das principais contribuições encontradas na bibliografia, fornecendo uma visão geral do que vem sendo desenvolvido nos últimos anos quanto a *awareness*.

### **3. Framework para Awareness**

A análise de vários mecanismos de suporte à *awareness* propostos na bibliografia identificou um conjunto de características importantes. Estas características giram em torno de 6 questões (o que, quando, onde, como, quem e quanto), cada uma identificando aspectos vitais para o fornecimento de *awareness* dentro de ferramentas de *groupware* síncronas e assíncronas, e que juntas definem o *framework* apresentado neste artigo. Em ambientes síncronos, os membros do grupo precisam estar trabalhando simultaneamente (a interação síncrona descreve a situação onde mais de um usuário está acessando concorrentemente a dados compartilhados [16]), como em videoconferências. Já em um ambiente assíncrono, pode haver um intervalo de tempo entre a atuação de um usuário e sua percepção por seus colegas (a interação assíncrona ocorre quando os usuários estão compartilhando um objeto sobre um período estendido [16]), ou seja, os usuários não precisam estar trabalhando simultaneamente para que o objetivo seja atingido.

Com isso, sistemas de *groupware* síncronos e assíncronos diferem quanto as suas necessidades por *awareness*, uma vez que usuários obrigatoriamente trabalhando ao mesmo tempo terão necessidades de

percepção diferentes daqueles que não precisam trabalhar simultaneamente. Dessa forma, torna-se importante, ao analisar cada uma das seis questões mencionadas acima, observar de qual ambiente se trata.

### 3.1. O Que

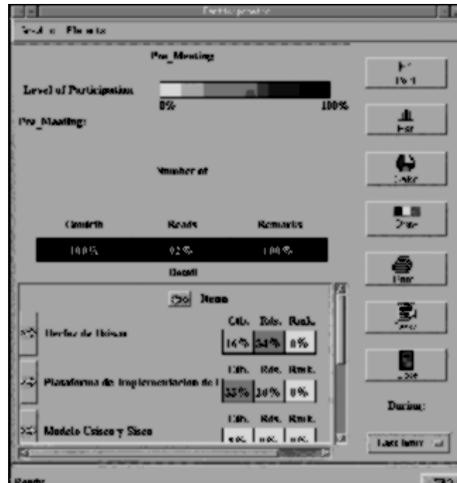
A primeira questão a ser analisada é "o que" e se refere a quais as informações devem ser fornecidas aos usuários. Estas informações estão intimamente ligadas a dois aspectos principais: as atividades e os papéis. As atividades são a base do trabalho cooperativo, pois o objetivo a ser alcançado geralmente é dividido em atividades menores, distribuídas entre os membros do grupo. Conhecer esta distribuição é conhecer as atividades dos colegas, e isto é muito importante para o andamento do trabalho, tanto síncrono quanto assíncrono. Em ambiente síncrono, é mais importante saber detalhes sobre as atividades sendo realizadas no momento, enquanto que em um sistema assíncrono, como não há garantia sobre o momento em que uma tarefa será realizada por um colega, é mais interessante ter uma visão ampla das atividades.

Assim, pode-se dizer que, em ambientes síncronos, o maior interesse está em informações detalhadas, nas atividades vistas em um "micro-nível", pois os membros estão trabalhando ao mesmo tempo, tornando importante a percepção das atividades de seus colegas nos mínimos detalhes. Um exemplo é o Groupkit, um *toolkit* para a construção de *groupware* síncronos e distribuídos para a realização de conferências. O Groupkit fornece uma coleção de *widgets*, como o Telepointer, utilizado principalmente para a comunicação de gestos. Segundo Roseman e Greenberg [18], os gestos são um mecanismo de comunicação rico, que permite aos participantes indicar relações entre artefatos, chamar a atenção para um artefato em especial, etc. Através dos Telepointers é possível visualizar a movimentação do mouse feita pelos colegas, o que fornece uma boa idéia do que cada um está fazendo, fornecendo uma noção em micro-nível das atividades realizadas.

O mesmo não ocorre em ambientes assíncronos, onde um nível de detalhamento tão grande não é tão adequado, podendo atrapalhar o trabalho dos membros. Nestes ambientes, devido à existência de um possível espaço de tempo entre as atuações dos membros, é mais importante ter uma idéia global das atividades realizadas pelos colegas, ter uma visão em "macro-nível" destas. Um exemplo é encontrado dentro do Alliance, um editor cooperativo assíncrono que permite a um grupo geograficamente distribuídos pela Internet editar conjuntamente documentos estruturados. Seu funcionamento se baseia na fragmentação do documento em partes, de acordo com a atribuição de papéis para os autores, definindo seus direitos sobre cada um dos fragmentos [7], [20], [8]. Dentro do Alliance, o suporte a *awareness* é feito por um sistema de notificação, que permite que co-autores recebam os fragmentos inteiros atualizados, de modo que estes não percebam as atividades de seus colegas em detalhes, mas somente em um macro-nível, recebendo apenas o resultado já publicados destas atividades.

Os papéis são elementos de destaque em um ambiente cooperativo, pois representam a noção de hierarquia dentro do grupo. Eles indicam as responsabilidades e possibilidades dos membros sobre o trabalho. Assim, os papéis desempenhados fornecem uma forte indicação sobre o tipo de informação de *awareness* necessária ao seu desempenho. Por exemplo, as informações necessárias para um participante diferem das necessárias a um coordenador. A coordenação é fundamental em atividades cooperativas e o fornecimento de *awareness* específico para a coordenação é importante, tanto em sistemas síncronos quanto assíncronos [10]. Se o coordenador do grupo tem disponíveis mecanismos de percepção próprios, com os eventos relevantes ao seu papel, achará mais fácil a tarefa de conduzir o grupo a um resultado bem sucedido [5],[4].

Para executar suas funções, um coordenador precisa de informações de percepção mais completas que as de um participante. Ele precisa estar atento ao andamento do trabalho como um todo, desde as atividades de cada membro, até o somatório de todas elas. Precisa de informações como prazos e *status* para tomar decisões e dirigir os esforços da equipe para os pontos mais necessários. Um exemplo deste tipo de percepção diferenciada é apresentado na Figura 1, onde o coordenador tem uma visão geral da participação dos membros numa discussão, através desta implementação conhecida como Participameter. O Participameter mostra o nível de participação dos membros como uma percentagem da participação total [5]. Este pode ser observado na Figura 1, incluindo detalhes na parte inferior da figura sobre as percentagens de cada elemento ("contrib", "reads" e "remarks") para cada contribuição. Observa-se também a associação entre valores percentuais e um padrão de cores, permitindo uma rápida assimilação da informação.



**Figura 1 Implementação do Participameter**

Logo, ao fazer a pergunta "o que", identificam-se duas características importantes no suporte à percepção: as atividades e os papéis, os quais constituem dois tipos de informações que devem ser característicos para o suporte à percepção. A Tabela 1 apresenta o resumo da questão "o que".

**Tabela 1 Características para a pergunta "o que"**

		<i>Ambientes Síncronos</i>	<i>Ambientes Assíncronos</i>
O que	<i>Atividades</i>	Micro-nível	Macro-nível
	<i>Papéis</i>	Diferenciação das informações de acordo com o papel desempenhado	

### 3.2. Quando

A segunda questão que deve ser feita ao se pensar em *awareness* é "quando": quando ocorrem os eventos geradores das informações de *awareness* e quando se dá a apresentação destas informações. As informações de *awareness* são geradas por eventos que ocorrem durante o trabalho do grupo, e de acordo com seu momento de ocorrência, estes eventos vão ser mais ou menos úteis à percepção. Pode-se dividir a ocorrência dos eventos em quatro momentos: o "passado", para eventos que ocorreram em um intervalo de tempo no passado; o "passado contínuo", para eventos que começaram no passado, mas que continuam válidos até agora; o "presente", para eventos que estão ocorrendo neste momento e o "futuro", representando as opções futuras para o grupo.

O passado inclui eventos que já ocorreram há algum tempo e cujos resultados podem não ser mais válidos. Um exemplo de seu suporte é o SISCO, um sistema distribuído e assíncrono para o apoio a preparação de reuniões, através de um ambiente que permite a discussão dos temas a serem tratados na reunião, sem a obrigatoriedade da tomada de decisão [2]. O SISCO inclui em sua arquitetura uma memória de grupo que armazena as contribuições feitas durante a pré-reunião, facilitando o fornecimento de percepção dos eventos ocorridos no passado. O passado contínuo representa aqueles eventos que, apesar de terem iniciado no passado, continuam valendo até o presente, como são, por exemplo, os fragmentos dentro do Alliance, onde apenas aqueles que ainda não foram substituídas por novos são visíveis aos autores.

O presente compreende os eventos que estão acontecendo agora, como o movimento do mouse ou a alteração da posição de uma barra de rolagem. Um exemplo de seu suporte está no COPSE/CUTE. O COPSE (*Collaborative Project Support Environment*) é um ambiente para o desenvolvimento cooperativo de software, enquanto o CUTE (*Collaborative UML Technique Editor*) é um editor, construído sobre o COPSE, com suporte a aspectos de cooperação na modelagem de softwares orientados a objetos, focando as atividades de diagramação destes [9]. Dentro do CUTE, a inclusão de novos elementos (classes ou relacionamentos), ou mesmo a mais simples alteração em um destes elementos, é imediatamente replicada nos *workspaces* dos demais.

Por fim, o futuro representa as opções do grupo, os eventos que ainda poderão ocorrer, mas que precisam fazer parte da percepção do usuário. É, por exemplo, o caso de um alarme que avise quanto à aproximação dos prazos para as atividades. A chegada do prazo limite é o evento gerador e só ocorrerá no futuro, mas o alarme, que fornece a percepção deste evento, se dá antes.

Alguns destes momentos são mais ou menos importantes de acordo com o ambiente utilizado. Em ambientes síncronos, por estarem os membros trabalhando ao mesmo tempo, é vital fornecer a percepção dos eventos que estão ocorrendo no momento presente, e em ambientes assíncronos, como há um possível espaço de tempo entre a atuação dos colegas, é vital manter o contexto sobre o que foi feito (eventos no passado) e do que ainda está sendo feito (passado contínuo), para que os participantes possam encaixar suas próprias atividades nas do grupo. Em ambos os casos, a percepção de eventos futuros representa uma opção interessante para manter os membros atentos aos possíveis rumos do trabalho.

O interesse pelos eventos ocorridos em um ou outro momento vai determinar o tempo de utilidade da informação e, por consequência, a sua persistência. Até quando as informações de percepção são úteis para o trabalho do grupo? Serão estas informações persistentes? Em sistemas síncronos, o interesse maior se concentra no momento atual, então não há necessidade de manter as informações de percepção por mais tempo do que aquele momento, pois passado este, elas não serão mais úteis. Logo, o tempo de utilidade destas informações é baixo, e por consequência, há a necessidade de uma baixa persistência. Já em sistemas assíncronos, como o interesse maior reside na percepção de eventos no passado ou no passado contínuo, as informações seguem sendo úteis ao grupo, mesmo passado o momento de sua ocorrência, havendo a necessidade por uma alta persistência.

Todavia, deve-se determinar um limite para este armazenamento, um limite onde estas informações perdem sua utilidade. Eventos em um passado contínuo só são úteis ao grupo enquanto forem verdadeiros, podendo ser descartados assim que deixarem de ser. Já eventos no passado são úteis mesmo quando não forem mais verdadeiros, e devem ser mantidos até certo ponto, exigindo a definição de um critério de caducidade, que inutiliza estas informações. Este critério pode ser um tempo de vida fixo, o fornecimento das informações para um ou mais membros, ou mesmo combinações de eventos.

Outra característica importante com relação à questão “quando” refere-se ao momento da apresentação das informações de percepção. Em sistemas síncronos, como o interesse está nos eventos presentes, quanto mais cedo for dada a percepção destes eventos, melhor (imediate). Já em sistemas assíncronos, há um intervalo de tempo entre os eventos geradores e os demais membros os perceberem, logo, a informação é apresentada possivelmente em um momento posterior à ocorrência dos eventos, sendo adequado permitir o próprio usuário decidir o momento de receber estas informações. Exemplos são o CUTE, que por trabalhar com eventos presentes, possui uma baixa persistência e apresenta as informações dos eventos imediatamente a sua ocorrência, e o Alliance, que apresenta eventos no passado contínuo, mantendo uma alta persistência nestas informações através de seus fragmentos.

A Tabela 2 resumiu o que foi discutido sobre a questão “quando”: quando ocorrem os eventos, até que ponto as informações geradas são úteis e quando apresentar estas informações.

**Tabela 2 A questão "quando" para percepção**

		<i>Ambientes Síncronos</i>	<i>Ambientes Assíncronos</i>
Quando	<i>Eventos</i>	Presente	Passado
		Futuro	Passado Contínuo
			Futuro
	<i>Apresentação</i>	Imediata	Posterior (a critério do usuário)
	<i>Persistência/Tempo de Utilidade</i>	Baixa	Alta (critério de caducidade)

### 3.3. Onde

A terceira questão que deve ser feita quando o assunto é percepção é onde as informações são geradas e apresentadas. Ao se trabalhar em grupo, trabalha-se sobre um conjunto de objetos dispostos dentro de um

espaço de trabalho (*workspace*), compartilhados entre os membros. A percepção do que está ocorrendo neste espaço compartilhado é chamada de *workspace awareness*. Trata-se do conhecimento do que está acontecendo no *workspace* compartilhado no momento atual, e é um elemento natural dentro de ambientes físicos de trabalho. Para Gutwin e Greenberg [13],[14], um participante, em uma reunião face a face, pode ver todo o *workspace* físico e quem estiver nele. Pode interagir com artefatos físicos, fornecendo aos demais informações sobre a natureza e o progresso das atividades. Estes são conhecimentos naturais e inerentes em ambientes físicos, mas não em um *workspace* virtual, onde os recursos de dispositivos de entrada e saída são menores, se comparados aos ambientes físicos, mas impõem menos restrições sobre as interações [14], [13].

O fornecimento de *workspace awareness* em sistemas síncronos é vital para que os membros do grupo trabalhando simultaneamente tenham noção de todas as atividades realizadas dentro do *workspace* compartilhado. Neste contexto, o Groupkit oferece vários widgets para o fornecimento de *workspace awareness*, como telepointers, multi-user scrollbars e o Gestalt Viewer, para a localização dos colegas no *workspace*, apresentado na Figura 2.

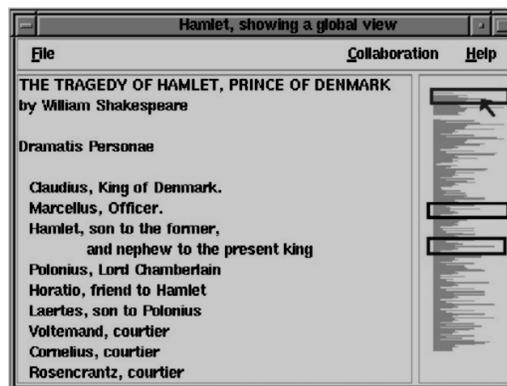


Figura 2 Exemplo de Widget para *workspace awareness* [18]

Em ambientes assíncronos, não há como garantir a presença dos colegas no *workspace* em um intervalo de tempo, assim, o foco de *awareness* localiza-se nos objetos compartilhados por estes colegas. É através destes objetos que a comunicação entre os membros se dá, através de sua manipulação e de seu histórico, que mostram o que houve e está acontecendo dentro do trabalho em grupo, criando assim o contexto para as atividades de cada membro. Segundo Dourish [10], o artefato compartilhado é, em ambientes assíncronos, essencialmente o único espaço compartilhado disponível aos participantes, sendo peça chave na colaboração assíncrona. Um exemplo disto é o projeto POLITeam, que desenvolveu um *groupware* integrado, com suporte para o trabalho cooperativo assíncrono e distribuído, com componentes de workflow, e processamento coordenado de documentos e tarefas [17],[22]. O POLITeam mantém um histórico de eventos gerados dentro do ambiente compartilhado, atrelando cada evento ao objeto sobre o qual o evento foi realizado.

Tanto em sistemas síncronos quanto assíncronos é de inteira responsabilidade do projetista desenvolver um ambiente capaz de dar suporte adequado, seja a *workspace awareness*, seja a objetos compartilhados. Para tanto, é necessário apresentar uma metáfora adequada ao tipo de informação desejada. Podem ser usadas tanto metáforas de escritório quanto de sistema. As metáforas de escritório, ideais para sistemas síncronos, representam elementos do ambiente de trabalho real, presentes no dia-a-dia dos participantes, como salas e mesas, minimizando a possibilidade de má interpretação. Um exemplo é o *groupware* DIVA, um protótipo de ambiente de escritório virtual.

As metáforas de sistema relacionam o *groupware* com versões monousuárias do sistema, como o Alliance, cuja interface se assemelha a de um editor HTML padrão. Esta metáfora utilizada afeta o modo como as informações são capturadas pelos participantes, havendo a necessidade de enriquecê-la adequadamente com as informações de *awareness*, a fim de permitir aos participantes uma percepção mais clara das atividades do grupo, enfatizando os aspectos de cooperação e fornecendo aos usuários a sensação de estarem realmente trabalhando em grupo.

A Tabela 3 resume as considerações colocadas sobre a questão “onde”, onde as informações são geradas e apresentadas, relacionando-se ao espaço de trabalho e à metáfora utilizada neste espaço.

**Tabela 3 A questão "onde" para a percepção**

		<i>Ambientes Síncronos</i>	<i>Ambientes Assíncronos</i>
Onde	<i>Espaço</i>	<i>Workspace awareness</i>	Objetos compartilhados
	<i>Metáfora</i>	Escritório	Sistema

### 3.4. Como

A próxima questão sobre percepção é "como" e indica como as informações são apresentadas aos usuários, como é sua interface. A interface com o usuário é a responsável pelo fornecimento das informações de *awareness*, devendo apresentá-las de forma resumida, a fim de evitar a sobrecarga dos membros e permitir uma rápida assimilação. Os usuários não podem se sentir soterrados pelas informações de percepção, nem tê-las omitidas. Estas informações também devem ter uma natureza passiva, surgindo direto das atividades de cada pessoa, ao invés de ser gerenciada ou procurada explicitamente [10]. Uma interface mal projetada pode causar tanto a sobrecarga, quanto a omissão, por ser inadequada ao tipo de informação apresentada.

Portanto, para fazer o projeto de uma interface balanceada, deve-se procurar por elementos de interface adequados. Estes elementos podem ser ícones ou cores associados a informações específicas, como papéis e participantes, gráficos representativos do progresso do trabalho, ou ainda uma combinação de elementos como estes. O importante é que estes elementos apresentem as informações de percepção de forma resumida, sem sobrecarga, nem perda de conteúdo significativo. Para tanto, estes elementos deverão fazer uma filtragem ou um agrupamento das informações, mostrando apenas aquilo que for mais útil e interessante a cada participante. Estes processos de filtragem e agrupamento podem utilizar vários critérios desde o papel até preferências pessoais de cada membro, como são, por exemplo, os *awareness* profiles utilizados pelo POLITeam. Estes profiles fazem uma seleção sobre os eventos ocorridos, fazendo com que somente aqueles que estiverem no conjunto de profiles assinados pelo usuário sejam apresentados. O POLITeam também faz uso de cores e ícones sobrepostos para agrupar as informações e apresentar perifericamente ao usuário as atividades realizadas sobre um objeto e o tempo transcorrido.

Além disto, há a questão da apresentação dos objetos e do espaço de trabalho compartilhado. Se todos os membros vêem os objetos e o *workspace* da mesma forma ou se há diferenças nas visões de cada um, isto afeta a percepção do grupo, pois manter uma mesma visão para todos significa que os membros vão ter disponíveis as mesmas informações (quando a representação diverge, o contexto pode se perder [13]). Forma-se, então, uma disputa entre as necessidades do grupo e dos indivíduos. Segundo Gutwin e Greenberg [13], diferentes representações do espaço de trabalho e seus objetos podem tornar as tarefas individuais mais fáceis, mas também podem restringir a comunicação sobre os objetos, sendo mais fácil manter a percepção quando uma mesma representação é mantida. Assim, o projetista de um *groupware* deve decidir entre beneficiar atividades individuais ou priorizar a percepção das atividades coletivas. Em sistemas síncronos, o trabalho é feito pelo grupo simultaneamente, fazendo com que uma percepção uniforme do *workspace* e dos objetos seja mais importante do que a flexibilidade para o usuário isolado, sendo importante ter a oportunidade de ver onde seu colega está trabalhando agora e o que está fazendo em detalhes. Assim, costuma-se usar, nestes casos, interfaces com maior acoplamento, como interfaces WYSIWIS (*What You See Is What I See*), e WYSIWIS relaxadas, ou ainda recursos como múltiplas visões.

Nas interfaces WYSIWIS, todos os membros ativos têm a mesma imagem do *workspace* e seus objetos, o que garante o contexto das atividades. Entretanto, estas são interfaces muito restritivas e podem prejudicar o andamento do trabalho por tolher demais os indivíduos. Para estes casos, uma alternativa são as interfaces WYSIWIS relaxadas, que garantem maior liberdade aos indivíduos, como, por exemplo, a livre navegação no *workspace*, ao custo de uma possível perda de informações, especialmente em grandes *workspaces*. O GroupKit é um exemplo de toolkit que permite a construção de ferramentas de *groupware* com interface WYSIWIS, e o CUTE/COPSE, um exemplo de *groupware* com interface WYSIWIS relaxada.

Além das interfaces WYSIWIS, os sistemas síncronos podem ainda contar com o recurso de múltiplas visões, onde os participantes podem optar em visualizar o *workspace* e seus objetos de várias maneiras, permitindo, por exemplo, que um participante mantenha uma visão completa do *workspace* e outra visão adequada às suas atividades individuais.

Em ambientes assíncronos, não há sentido em se utilizar interfaces WYSIWIS puras, que enfocam as ações dos colegas presentes, pois não há garantias de que eles estarão trabalhando ao mesmo tempo. Todavia, a presença de mais de um usuário no sistema é uma boa oportunidade para cooperação e deve ser explorada. Para estes casos, o mais adequado são interfaces WYSIWIS relaxadas, que mantém a liberdade de ação de cada participante, sem deixar de mostrar o que os colegas ativos estão fazendo no momento. Outra possibilidade é unir interfaces WYSIWIS relaxada com a idéia de múltiplas visões, fornecendo aos participantes a oportunidade de escolher entre uma visão mais adequada a sua atividade individual e outra que favoreça sua colaboração. O Alliance é um exemplo de *groupware* que oferece o recurso de múltiplas visões a seus usuários.

Uma última possibilidade é a utilização de interfaces desacopladas, que não são acopladas às interfaces dos colegas. Estas permitem a cada membro adaptar a interface às suas necessidades individuais, sem que os demais sejam notificados disto, mas não impedindo a interface de incluir elementos para percepção. Um exemplo disso é o SISCO, onde a interface de um usuário não mantém ligação com a dos demais, mais ainda assim mantém indicadores para *awareness*.

A Tabela 4 resume a questão "como".

**Tabela 4 Percepção de "Como"**

		<i>Ambientes Síncronos</i>	<i>Ambientes Assíncronos</i>
Como	<i>Interface</i>	WYSIWIS	WYSIWIS relaxado
		WYSIWIS relaxado	Múltiplas Visões
		Múltiplas Visões	Desacopladas
	<i>Balanceamento</i>	Filtragem	
		Agrupamento	

### 3.5. Quem

A penúltima questão a ser feita é "quem": quem está trabalhando e atento no momento. Este é um conhecimento importante para o bom andamento das atividades no grupo, pois age como facilitador da cooperação, à medida que estimula a interação e a comunicação informal entre os membros. Segundo Smith [21], boa parte do trabalho em um ambiente de escritório é baseada na comunicação informal, a qual, em muitos casos, permite às pessoas ficarem conscientes das atividades dos colegas e fornece também um contexto para as atividades relacionadas ao trabalho.

A noção de presença dos outros participantes é vital em ambientes síncronos, pois é inviável realizar uma tarefa simultaneamente com um grupo de pessoas sem saber quem são estas pessoas. Nestes ambientes, os participantes precisam (obrigatoriedade) estar conscientes da presença dos demais para que o trabalho prossiga e obtenha resultados satisfatórios. Já em ambientes assíncronos, a noção de presença comporta-se como uma oportunidade de cooperação a ser explorada, pois a presença de todos não é obrigatória para que o trabalho prossiga. Explorando estas oportunidades, os membros podem trocar idéias, experiências e dúvidas, enriquecendo o conhecimento do grupo, a própria cooperação e o trabalho como um todo.



Figura 3 Informações detalhadas sobre um participante [11]

A Figura 3, mostra um exemplo widget do GroupKit [11], [18], [19], que apresenta uma lista com os participantes presentes e permite a consulta de maiores informações sobre os colegas, como foto, telefone, email. Fornecer informações sobre os colegas é importante para o fortalecimento da noção de grupo, tanto em ambientes síncronos quanto assíncronos. Esse conhecimento cria um senso maior de comunidade, à medida que os membros vão conhecendo melhor seus colegas, e o uso de mecanismos de comunicação reforçam a coesão entre estes membros.

A noção da presença envolve também saber se alguém está ou não atento ao sistema, pois estando os membros geograficamente distantes, a mera presença não garante que o colega esteja realmente atento. De posse deste conhecimento, é possível a um membro conversar, trocar idéias, pedir auxílio ou mesmo resolver possíveis conflitos através de ferramentas de comunicação, permitindo tornar as relações entre os participantes mais pessoais e interativas, menos formais. Estas ferramentas de comunicação devem ser adaptadas ao tipo de sistema onde estão inseridas. Em sistemas síncronos, elas devem priorizar a comunicação síncrona, incluindo ferramentas no estilo conferência, como o CUTE que inclui uma ferramenta de chat. Já em ambientes assíncronos, destaca-se o uso de ferramentas assíncronas, como email, quadro de avisos e notas, que garantem aos membros a oportunidade de manter uma comunicação informal com seus colegas. O uso de ferramentas síncronas, como *chats*, fica sujeito à eventual presença de mais de um membro, agindo somente como um recurso extra para a comunicação.

A Tabela 5 resume a questão “quem”, com as características da percepção expostas acima.

Tabela 5 A questão "quem" para percepção

		<i>Ambientes Síncronos</i>	<i>Ambientes Assíncronos</i>
Quem	<i>Presença</i>	obrigatoriedade	oportunidade
	<i>Ferramentas de Comunicação</i>	Essencialmente ferramentas síncronas	Essencialmente ferramentas assíncronas

### 3.6. Quanto

A última das questões a se fazer sobre percepção é “quanto” e destaca uma característica essencial que deve ser observada no suporte à percepção: qual a quantidade ideal de informações que deve ser apresentada ao usuário, a fim de lhe prover percepção sobre o grupo e suas atividades.

Esta preocupação é uma constante no desenvolvimento de um suporte eficaz à percepção, afetando todas as questões anteriormente discutidas. O ponto chave desta preocupação é fornecer a quantidade certa de informações para a percepção, sem causar nem a insuficiência, que poderia deixar os usuários sem um contexto para as suas atividades, nem a sobrecarga, que pode prejudicar os participantes à medida que interrompe o seu trabalho e exige atenção excessiva para filtrar as informações de seu interesse. É necessário buscar um meio termo entre a insuficiência e a sobrecarga de informações, extrapolando a idéia de filtragem e agrupamento para além da interface (“como”), para dentro da escolha das informações (“o que”), para a escolha do local e do momento da geração e apresentação (“onde” e “quando”), e para e com quem apresentar (“quem”) estas informações.

Dessa forma, esta questão afeta a todas as demais (o que, quando, como, onde e quem), caracterizando-se como uma dimensão paralela, influenciando cada questão de forma isolada, e também o somatório de todas as informações extraídas das outras cinco questões, o qual não pode ficar em nenhuma das extremidades do “quanto”, nem na insuficiência, nem na sobrecarga. Outra forma de visualizar esta questão é como função da

densidade de informação. Imaginando que cada item nas questões discutidas (como atividades em micro-nível, eventos no passado e interface WYSIWIS) possa variar entre não fornecer informação (não ter este suporte), fornecer um pequeno conjunto, e disponibilizar um grande conjunto de informações, a dimensão “quanto” representaria o conjunto de todas estas informações, o qual poderia ser um conjunto mais ou menos denso. A Figura 4 resume esta idéia central da questão “quanto” como uma dimensão paralela às demais, definida por todo o conjunto de informações indicados pelas outras questões e cujo ideal para um sistema de *groupware* seria alcançar um ponto entre um conjunto muito denso e um muito pouco denso de informações. Ou seja, procurar um ponto com informações suficientes para manter o contexto do grupo, sem que o seu trabalho seja prejudicado, seguindo os interesses desse grupo e do próprio *groupware*.



Figura 4 Dimensão "quanto" - variação na densidade da informação

### 3.7. O Framework

Discutidas as seis questões "o que", "quando", "onde", "como", "quem" e "quanto", forma-se, com a união destas questões, o *framework* apresentado na Figura 5. Este *framework* resume as características consideradas mais importantes para percepção, apresentadas durante esta discussão. Com isso, chega-se a uma classificação para os mecanismos de suporte à percepção representada pelo próprio *framework*, o qual também forma uma visão geral do que é necessário observar e optar durante o desenvolvimento deste tipo de suporte. Pode-se observar que a Figura 5 é formada na verdade pela união das Tabelas 1 a 5 com a Figura 4, representando a formação deste *framework* pelo conjunto das 6 questões discutidas.

		Ambientes Síncronos	Ambientes Assíncronos
<b>O que</b>	Atividades	micro-nível	macro-nível
	Papéis	diferenciação das informações de acordo com o papel desempenhado	
<b>Quando</b>	Eventos	Presente Futuro	Passado Passado Contínuo Futuro
	Apresentação	Imediata	Posterior (a critério do usuário)
	Persistência/Tempo de Utilidade	Baixa	Alta (critério de caducidade)
<b>Onde</b>	Espaço	workspace awareness	objetos compartilhados
	Metáfora	escritório	sistema
<b>Como</b>	Interface	WYSIWIS WYSIWIS relaxado Múltiplas Visões	WYSIWIS relaxado Múltiplas Visões Desacopladas
	Balanceamento	Filtragem Agrupamento	
<b>Quem</b>	Presença	obrigatoriedade	oportunidade
	Ferramentas de Comunicação	Essencialmente ferramentas síncronas	Essencialmente ferramentas assíncronas

Figura 5 - Framework para suporte à percepção

## 4. Exemplos Classificados

Para a confecção deste trabalho, vários sistemas de *groupware* foram estudados, alguns dos quais citados no decorrer do artigo. Estes se encontram classificados na Tabela 6, de acordo com o *framework* da Figura 6. Cada um é representado por uma letra grega acompanhada de um sinal de “-”, se o suporte não for adequado,

e suas principais características encontram-se descritas abaixo. Convém apenas lembrar que para esta rápida análise não foi considerada, mostrada na Tabela 6, a questão “quanto”.

**Tabela 6 Classificação de alguns sistemas de *groupware***

		<i>Ambientes Síncronos</i>	<i>Ambientes Assíncronos</i>
O que	<i>Atividades</i>	Micro-nível ( $\alpha$ -) ( $\epsilon$ -) ( $\eta$ )	Macro-nível ( $\delta$ ) ( $\epsilon$ ) ( $\phi$ ) ( $\gamma$ )
	<i>Papéis</i>	diferenciação das informações de acordo com o papel desempenhado ( $\delta$ -) ( $\phi$ ) ( $\gamma$ -)	
Quando	<i>Eventos</i>	Presente ( $\alpha$ ) ( $\epsilon$ ) ( $\eta$ )	Passado ( $\delta$ ) ( $\gamma$ )
			Passado Contínuo ( $\delta$ ) ( $\epsilon$ ) ( $\phi$ ) ( $\gamma$ ) ( $\eta$ -)
	Futuro		
	<i>Apresentação</i>	Imediata ( $\alpha$ ) ( $\eta$ )	Posterior ( $\delta$ ) ( $\phi$ ) ( $\gamma$ )
<i>Persistência/Utilidade</i>	Baixa ( $\alpha$ ) ( $\eta$ )	Alta ( $\delta$ ) ( $\epsilon$ -) ( $\phi$ -) ( $\gamma$ )	
Onde	<i>Espaço</i>	<i>workspace awareness</i> ( $\alpha$ -) ( $\epsilon$ -) ( $\eta$ )	objetos compartilhados ( $\delta$ ) ( $\phi$ )
	<i>Metáfora</i>	Escritório ( $\epsilon$ )	Sistema ( $\alpha$ -) ( $\delta$ -) ( $\phi$ )
Como	<i>Interface</i>	WYSIWIS ( $\eta$ )	Desacopladas ( $\delta$ ) ( $\gamma$ )
		WYSIWIS relaxado ( $\alpha$ ) ( $\eta$ )	
		Múltiplas Visões ( $\delta$ -) ( $\epsilon$ ) ( $\phi$ )	
	<i>Balanceamento</i>	Filtragem ( $\delta$ )	
		Agrupamento ( $\delta$ )	
Quem	<i>Presença</i>	Obrigatoriedade ( $\alpha$ ) ( $\epsilon$ ) ( $\eta$ )	Oportunidade ( $\delta$ -) ( $\gamma$ -)
	<i>Ferramentas de Comunicação</i>	Ferramentas síncronas ( $\alpha$ ) ( $\epsilon$ ) ( $\gamma$ ) ( $\eta$ )	Ferramentas assíncronas ( $\delta$ -) ( $\epsilon$ ) ( $\gamma$ -)

➤COPSE/CUTE ( $\alpha$ ): O COPSE é um ambiente para o desenvolvimento cooperativo de software e também um *framework* para a construção de aplicações *groupware*, enquanto o CUTE é um editor UML com suporte a aspectos de cooperação, construído sobre o COPSE [9]. O CUTE trabalha com eventos presentes, com apresentação imediata, não havendo persistência para as informações de percepção. Ele também mantém a noção de presença no *workspace*, graças ao recurso de “*user list*”, e mantém ferramentas de comunicação síncronas, através de um sistema de mensagens e de um *chat*. Além disso, ele possui uma interface WYSIWIS relaxada, mantendo informações como o posicionamento e detalhamento (visualização de atributos e métodos) das classes e comuns a todos os participantes, mas a navegação pelo espaço de trabalho é livre, e adota uma metáfora de sistema, assemelhando-se a um sistema de diagramação.

➤POLITeam ( $\delta$ ):O projeto POLITeam objetiva desenvolver um *groupware* com que suporte ao trabalho cooperativo assíncrono e distribuído, incluindo componentes de workflow, e processamento coordenado de documentos e tarefas [22],[17]. O POLITeam usa a metáfora de containers de objetos, como documentos, ferramentas e outros containers, assemelhando-se aos desktops padrão, usados em diversos sistemas, podendo ser classificado como metáfora de sistema. Sobre esta metáfora, o *workspace* compartilhado pode ser visualizado através de três visões diferentes, caracterizando múltiplas visões, mas todas estas são desacopladas das visões mantidas pelos colegas. A noção de presença é fornecida no sentido de oferecer uma oportunidade, mas não há um suporte adequado à comunicação. Quanto aos eventos, seus usuários tem acesso a eventos no passado e no passado contínuo, com apresentação posterior e alta persistência, com as informações de *awareness* localizadas junto aos objetos compartilhados. Já as atividades são visíveis em um macro-nível, uma vez que somente é possível saber que foram feitas determinadas operações nos objetos como um todo, e não que parte foram alteradas.

➤DIVA ( $\epsilon$ ): O DIVA é um protótipo de ambiente de escritório virtual, com foco no fornecimento integrado de *awareness*, tanto em modo síncrono quanto assíncrono. Ele usa uma metáfora de escritório, modelando elementos como “pessoas”, “salas”, “mesas” e “documentos”. As “mesas” funcionam como local para o trabalho dentro das salas, podendo ser usadas para preservar o contexto de trabalho [22], permitindo a definição de diferentes modos de acoplamento entre as interfaces dos usuários, caracterizando a presença de múltiplas visões. O DIVA também fornece uma noção sobre a presença e a

disponibilidade dos colegas no escritório, a qual, juntamente com as ferramentas de comunicação síncronas, por canais de áudio e vídeo, e assíncronas, via notas que podem ser deixadas para os colegas, funcionam como uma obrigatoriedade.

- Alliance (  $\phi$  ): O Alliance é um editor cooperativo assíncrono que permite a um grupo de pessoas geograficamente distribuídas e conectadas a Internet editarem conjuntamente documentos estruturados. Seu funcionamento se baseia na fragmentação do documento em partes, de acordo com a atribuição de papéis para os autores, definindo seus direitos sobre cada um dos fragmentos [20]. Dessa forma, o Alliance trabalha com o conceito de papéis e atividades em um macro-nível, operando com a atualização de fragmentos completos, os quais também garantem a apresentação de eventos no passado contínuo, com alta persistência e apresentação posterior. O Alliance também apresenta uma metáfora de sistema, assemelhando-se a um editor HTML tradicional, e o recurso de múltiplas visões, permitindo a visualização e a edição do documento em várias visões.
- SISCO (  $\gamma$  ): O SISCO é um sistema distribuído e assíncrono para o apoio a preparação de reuniões, através de um ambiente que permite a discussão dos temas a serem tratados na reunião, sem a obrigatoriedade da tomada de decisão [2],[6]. Sua arquitetura inclui uma memória de grupo que armazena as contribuições feitas durante a pré-reunião, facilitando o fornecimento de percepção de eventos ocorridos no passado ou no passado contínuo, com apresentação posterior e persistência alta, implementada através do uso do SGBD para a manutenção desta memória. Quanto à comunicação, o SISCO trata a presença dos colegas como uma oportunidade, apresentando uma ferramenta de comunicação síncrona, por troca de mensagens rápidas e não armazenadas, e a própria memória de grupo como uma forma de comunicação assíncrona, além de uma interface totalmente desacoplada.
- GroupKit (  $\eta$  ): O GroupKit é um *toolkit* para a construção de *groupware* síncronos e distribuídos para a realização de conferências, incluindo também aplicações de conferência propriamente ditas e gerentes de sessão, que permitem aos usuários criarem e gerenciarem suas conferências [18],[19]. Através de seus *widgets*, o GroupKit permite às aplicações de conferências construídas sobre suas bases apresentar informações sobre as atividades em um micro-nível, com eventos presentes, de apresentação imediata, mas ainda com um recurso de manter o estado da conferência desde a saída de seu último membro até a entrada do próximo, o que pode ser considerado como um suporte à eventos em um passado contínuo não muito adequado. Já os últimos desenvolvimentos dentro do GroupKit têm vindo na direção de um suporte mais adequado à *wospace awareness*, com o desenvolvimento de novos *widgets* para isto. Atualmente, há um suporte já preparado a interfaces WYSIWIS puras e WYSIWIS relaxadas, embora a escolha por um destes modelos e da própria metáfora a ser seguida depende da aplicação desenvolvida. Quanto a presença, o GroupKit já traz em sua distribuição um *widget* para isso, o “Participantes”, e uma aplicação de conferência, chamada “Post It”, para a troca de mensagens textuais síncrona.

## 5. Conclusões

Este artigo apresentou um *framework*, resumindo as principais características para o suporte à percepção dentro de ferramentas de *groupware*. Estas características podem ser divididas em seis questões (o que, quando, onde, como, quem e quanto), sempre analisadas sob os pontos de vista de sistemas síncronos e assíncronos. A análise de várias ferramentas de *groupware* a luz deste *framework*, mostrou que diversos aspectos do suporte à *awareness* encontram-se descobertos, e muitas pesquisas ainda podem ser desenvolvidas sobre estes aspectos. Dentre estes pontos, pode-se destacar claramente a questão do suporte a eventos no futuro. A Tabela 6 mostrou que nenhuma das ferramentas analisadas apresentou este tipo de suporte, o qual é importante para manter a noção dos rumos e prazos do trabalho. Outro aspecto que também não tem recebido a devida atenção é a questão dos eventos no passado. Estes não foram muito abordados nas ferramentas analisadas, apesar da importância que decisões e atividades passadas podem ter nas atividades presentes e futuras do grupo. Além disso, o *framework* apresentado aqui não chegou a atacar a questão da estruturação do trabalho cooperativo, bastante comum em ferramentas de *workflow*. As necessidades de percepção para este tipo de ambiente devem ainda ser alvo de estudos.

Em resumo, a *awareness* mostrou-se um campo ainda fértil para as pesquisas na área de CSCW, sempre na busca por um suporte mais adequado ao trabalho em equipe.

## 6. Referências Bibliográficas

- [1] Araújo, R.M.; Dias, M.S.; Borges, M.R.S. Suporte por Computador ao Desenvolvimento Cooperativo de Software: Classificação e Propostas. In: XI Simpósio Brasileiro de Engenharia de Software. *Anais*. Fortaleza, CE, outubro, 1997. P. 299-314
- [2] Bellassai, G.; Borges, M.R.S.; Fuller, D.D.; Pino, J.A.; Salgado, A.C. SISCO: A tool for improve meetings productivity. In: CRIWG'95. *Proceedings*. 1995. P. 149-161.
- [3] Borges, M. R.S.; Cavalcanti, M.C.R.; Campos, M. L. M. Suporte por computador ao trabalho cooperativo. In: XV Congresso da Sociedade Brasileira de Computação. XV Jornada de Atualização em Informática. JAI'95. *Anais*. Canela, RS. Julho, 1995.
- [4] Borges, M.R.S.; Pino, J.A.; Fuller, D.; Salgado, A. Key issues in the design of an asynchronous system to support meeting preparation, In: *Decision Support Systems*.
- [5] Borges, M.R.S.; Pino, J.A. Awareness Mechanisms for Coordination in Asynchronous CSCW. In: 9th Workshop on Information Technologies and Systems. Charlotte, North Carolina, dez., 1999. *Proceedings*. Charlotte, 1999.
- [6] Cavalcanti, M.C.; Borges, M.R.S., Endo, M.Y. SISCO-RIO: Um sistema assíncrono para apoiar a preparação de reuniões. In: XI Simpósio Brasileiro de Engenharia de Software. *Anais*. Fortaleza, CE, outubro, 1997. P. 475-479.
- [7] Decouchant, D.; Salcedo, M. R.; Quint, V. *Structured Cooperative Authoring on the World-Wide Web*. Disponível em: <<http://www.w3.org/pub/Conferences/WWW/Papers/91/>>. Acesso em: set., 1998.
- [8] Decouchant, D.; Salcedo, M. R.; Serrano, M. *Principes de Conception d'une Application d'Édition Coopérative de Documents sur Internet*. Disponível em: <<ftp://ftp.inrialpes.fr/pub/opera/publications/IHM97.ps.Z>>. Acesso em: set., 1998.
- [9] Dias, M.S.; Xexéo, G. Editor Cooperativo para Diagramação de Software OO. In: XI Simpósio Brasileiro de Engenharia de Software. *Anais*. Fortaleza, CE, outubro, 1997. P. 499-502.
- [10] Dourish, P. Extending awareness beyond synchronous collaboration. In: CHI97 Workshop on Awareness in Collaboration Systems. Position Paper. *Proceedings*. Atlanta, USA. Março, 1997. Disponível em: <<http://www.best.com/~jpd/chi97-awareness.html>>. Acesso em: julho, 1999.
- [11] Greenberg, S.; Roseman, M. Groupware toolkits for synchronous work. In: Beaudouin-Lafon, M. (editor). *Computer-Supported Cooperative Work: trends in software series*. cap.6, p. 135-168. John Wiley & Sons, 1998. Disponível em: <<http://www.cpsc.ualgary.ca/grouplab/papers>>. Acesso em: jan., 2000.
- [12] Gutwin, C., Roseman, M. A usability study of workspace awareness. In: CHI96. Short Paper. *Electronic Proceedings*. Disponível em: <<http://www.acm.org/sigchi/chi96/proceedings/shortpap/Gutwin4/cg4txt.html>>. Acesso em: outubro, 1999.
- [13] Gutwin, C.; Greenberg, S. Design for individuals, design for groups: tradeoffs between power and workspace awareness. In: ACM Conference on Computer Supported Cooperative, 1998. *Proceedings*. Disponível em: <<http://www.cpsc.ualgary.ca/grouplab/papers/>>. Acesso em: setembro, 1999.
- [14] Gutwin, C.; Greenberg, S. A framework of awareness for small groups in shared-workspace groupware. *Technical Report 99-1*, Department of Computer Science, University of Saskatchewan, Canadá. Disponível em: <<http://www.cpsc.ualgary.ca/grouplab/papers/1999/99-AwarenessTheory/html/theory-tr99-1.html>>. Acesso em: setembro, 1999.
- [15] Hudson, S.E.; Smith, I. Techniques for fundamental privacy and disruption trade-off in awareness support systems. In: ACM Conference on Computer Supported Cooperative Work - CSCW'96. Boston, Massachusetts, USA, 1996. *Proceedings*. Novembro, 1996. P. 248 - 257.
- [16] Mariani, J.A. SISCO: Providing a cooperation filter for a shared information space. In: ACM SIGGROUP Conference on Supporting Groupwork - GROUP97. *Proceedings*. Phoenix, Arizona, USA. Novembro, 1997.
- [17] Prinz, W. *The POLITeam Awareness Client*. Disponível em: <<http://orgwis.gmd.de/projects/POLITeam/poliawac/>>. Acesso em: outubro, 1999.
- [18] Roseman, M.; Greenberg, S. Building real time groupware with GroupKit, a groupware toolkit. *ACM Transactions on Computer Human Interactions*. n.3, v.1, p. 66-106, mar., 1996. ACM Press. Disponível via WWW em <<http://www.cpsc.ualgary.ca/grouplab/papers>> (jan.,2000)
- [19] Roseman, M.; Greenberg, S. Building groupware with GroupKit. In: Harrison, M. (editor). *Tcl/Tk Tool*. p. 535-564. O'Reilly Press, 1997. Disponível em: <<http://www.cpsc.ualgary.ca/grouplab/papers>>. Acesso em: jan., 2000
- [20] Salcedo, M. R. *Alliance sur l'Internet: support pour l'édition coopérative des documents structurés sur un réseau à grande distance*. Grenoble, France: INPG, 1998.
- [21] Smith, I. Toolkits for multimedia awareness. In: CHI96. *Proceedings*. Disponível em: <[http://www.acm.org/sigchi/chi96/proceedings/doctoral/Smith/ies\\_txt.html](http://www.acm.org/sigchi/chi96/proceedings/doctoral/Smith/ies_txt.html)>. Acesso em: julho, 1999.
- [22] Sohlenkamp, M. *Supporting group awareness in multi-user environment through perceptualization*. Dissertation. Fachbereich Mathematik-Informatik der Universität - Gesamthochschule - Paderborn. Fevereiro, 1998. Disponível em: <<http://orgwis.gmd.edu/projects/POLITeam/poliawac/ms-diss/>>. Acesso em: julho, 1999.