

# **Sistemas sócio-técnicos cooperativos e adequação espacial: uma aplicação da teoria da complexidade ao estudo da atividade de concepção de softwares**

**Iara Sousa Castro**

Arquiteta e Designer, Mestre Engenharia de Produção  
Departamento de Engenharia de Produção / EEUFMG  
e-mail: iarascastro@yahoo.com.br  
fone/fax:(0XX31)-3221-7694

**Eliza Helena de Oliveira Echternacht**

Professora adjunta  
Departamento de Engenharia de Produção / EEUFMG  
e-mail: eliza@dep.ufmg.br

## **Resumo**

Este artigo apresenta resultados de pesquisa vinculada ao campo de interface entre a Ergonomia e a Arquitetura, na qual procurou-se analisar o caráter coletivo da atividade de trabalho aplicada às demandas de adequação espacial das situações de trabalho. Objetivou-se aplicar o modelo da complexidade (Pavard,2000) na análise do processo de reestruturação organizacional ocorrido em uma empresa privada de serviços de Tecnologia da Informação e enfatizar os seguintes aspectos: (1) as relações entre as mudanças das prescrições no sentido da especialização do trabalho e suas influências sobre a dinâmica coletiva da atividade de trabalho de criação e desenvolvimento de *softwares*; (2) as novas condicionantes da atividade geradas e suas repercussões sobre as demandas de adequação espacial e (3) os critérios de adequação espacial propostos no conceito de Espaços de Cooperação Presencial (Benchekroun, 2000).

## **Palavras – Chave:**

Complexidade, ergonomia de concepção, desenvolvimento de *softwares*

## **Abstract**

*Some practical results of the interface between Ergonomics and Architecture are presented. Analyzes of the collective character of work activity where demands for space adequacy exists have been done at a private information technology enterprise. The complexity model of socio-technical systems (Pavard, 2000) have been applied to examine the organizational restructuring process. Three aspects are emphasized: (1) the relationship between prescription changes of work specialization and its influences on software creation and work activity; (2) the new activity conditionings and its repercussions on the demands for space adequacy and (3) the new space adequacy criteria, based on the concept of Presential Cooperation Space (Benchekroun,2000) proposed.*

## **Keywords:**

*Complexity, ergonomics design, software development*

## ***1 - Objetivos***

Este artigo apresenta resultados de pesquisa vinculada ao campo de interface entre a Ergonomia e a Arquitetura, na qual procurou-se analisar o caráter coletivo da atividade de trabalho aplicada às demandas de adequação espacial das situações de trabalho. Objetivou-se aplicar o modelo da complexidade (Pavard,2000) na análise do processo de reestruturação organizacional ocorrido em uma empresa privada de serviços de Tecnologia da Informação e enfatizar os seguintes aspectos: (1) as relações entre as mudanças das prescrições no sentido da especialização do trabalho e suas influências sobre a dinâmica coletiva da atividade de trabalho de criação e desenvolvimento de softwares; (2) as novas condicionantes da atividade geradas e suas repercussões sobre as demandas de adequação espacial e (3) os critérios de adequação espacial propostos no conceito de Espaços de Cooperação Presencial (Benchekroun, 2000).

## ***2 - Contexto***

O estudo concentra-se em uma empresa privada de serviços de Tecnologia da Informação no setor responsável pela criação, produção e implantação de softwares e treinamentos aos usuários . A equipe de trabalho é composta por cem analistas de sistemas, em sua maioria jovens recém-formados, que são mantidos em um esquema de alta rotatividade.

No momento da pesquisa esse setor passa por um processo de reestruturação organizacional com significativas mudanças nos conteúdos das prescrições, especialmente relacionadas à redivisão formal do trabalho que visam, através da especialização das competências, superar problemas relativos ao controle da produção por parte da gerência. O novo contexto organizacional impõe demandas de adequação espacial que se tornaram o ponto de partida desse estudo.

## ***3 - Metodologia e Quadro Teórico***

A metodologia empregada baseia-se na análise da atividade de trabalho em situações reais a partir de observações sistematizadas e das verbalizações dos operadores (Obremdamme e Faverge,1955; Wisner,1967; Theureau,1990; Daniellou,1992). Observou-se os analistas de sistemas em situação de trabalho durante o período de implantação das mudanças, em torno de dois anos (1999-2000) e fez-se entrevistas aos próprios analistas, aos gerentes e diretores da empresa. Durante a fase de sistematização das observações, acompanhou-se integralmente o processo de criação e desenvolvimento de um site para a internet de uma rede de cinemas.

A análise visa comparar os fluxos de comunicação prescritos, conforme as representações do trabalho oriundas da gestão vertical<sup>1</sup> e as demandas reais de comunicação e interação, tais como desenvolvidas pela gestão horizontal<sup>2</sup>. Para a análise dos dados aplicou-se o modelo analítico baseado na aplicação ***da teoria da complexidade ao estudo dos sistemas***

---

<sup>1</sup>A gestão vertical fixa as regras prévias, que são estáveis mas que podem variar a longo prazo, baseadas em três aspectos funcionais: os objetivos da empresa, as regras que definem as tarefas prescritas e as ferramentas disponíveis e suas regras de utilização.( Garza e Weill-fassina, 2000: 220)

<sup>2</sup>A gestão horizontal é caracterizada pelas adaptações e reelaborações das regras e das utilizações informais das ferramentas, pelas transgressões e violações das regras e pelas compensações, a fim de assegurar um equilíbrio do processo. (Garza e Weill-fassina, 2000: 221)

*sócio-técnicos cooperativos* (Pavard,2000). Para integrar a análise da atividade à análise espacial fez-se a leitura das *espacializações* e apoiou-se no conceito de *Espaços de Cooperação Presencial* (Benchekroun,2000). Esses conceitos serão aprofundados nos itens 5 e 6 .

#### **4 – O processo de reestruturação organizacional e as novas prescrições**

As prescrições anteriores fundamentavam-se na designação de cada projeto a um analista de sistemas. Este era responsável por todo o processo de concepção e execução de um software. Para realizar o seu trabalho, o campo de interações necessário à execução de cada projeto se formalizava no núcleo analista - cliente - gerente. Informalmente as interações entre os analistas também se processavam em ajuda ou cooperação.

Tais interações podem ser ilustradas pelo seguinte diagrama :



**FIGURA 1 – Diagrama de interação do processo de criação e desenvolvimento de softwares.**

O analista interagia com o cliente através da entrevista; da apresentação, aprovação e implantação do projeto; do esclarecimento de dúvidas a respeito do uso do software ou quando resolvia problemas relativos a um mal funcionamento. As interações com o gerente visavam informar sobre o curso do projeto, negociar prazos de entrega e solicitar ajuda frente a dificuldades de solução de problemas. Por outro lado, o gerente acompanhava o trabalho do analista para controlar o processo de trabalho e negociar prazos e custos com o cliente. Embora cada analista estivesse envolvido com o seu próprio projeto, eles também interagiam entre si, quando tinham dificuldades relacionadas a vários fatores, tais como as ferramentas e a linguagem de programação, ou quando tinham dúvidas quanto ao setor da empresa que lhes poderia lhe suporte, etc.

Em conseqüência dessa organização, a gerência convivia com algumas dificuldades. A primeira delas era a concentração das informações relativas a um software em um único funcionário, o que o tornava imprescindível na solução de qualquer problema relacionado ao software produzido. Este aspecto é incompatível com as determinações gerenciais de manter uma equipe jovem, recém-graduada e atualizada quanto às evoluções do ferramental de trabalho e em esquema de alta rotatividade e salários médios.

Outras dificuldades se referiam ao retrabalho relacionado à informalidade do histórico dos projetos, dependente da memória individual. Como esse analista era o responsável por todas as etapas do projeto, ou seja, pela formulação, execução e implantação, freqüentemente

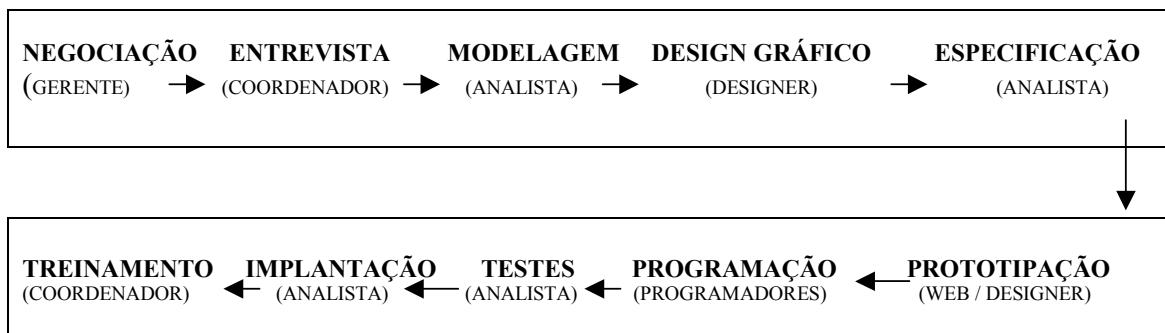
não fazia registros do histórico deste. Assim, os acordos entre o analista e o cliente eram feitos verbalmente e sem documentos para comprovação das decisões tomadas, deixando a gerência vulnerável a pedidos de alterações e modificações. Os analistas se encontravam num contexto que os obrigava a fazer e refazer o mesmo projeto inúmeras vezes, colocando em risco o prazo de entrega e ampliando sua carga de trabalho. Por outro lado, as tarefas de manutenção preventiva ou corretiva dos sistemas demandavam informações somente contidas nestas memórias individuais, o que causava problemas, especialmente quando o analista havia sido transferido ou demitido.

Diante desse contexto a empresa optou por modificar as prescrições .

#### 4.1 - As novas prescrições

As mudanças processadas consistiram no parcelamento do processo de criação de um *software*. O processo foi dividido nas seguintes etapas: entrevista, modelagem, designer, especificação, prototipação, programação, testes, implantação e treinamento. Cada analista é responsável por uma etapa do processo e a realização das tarefas de cada etapa depende da etapa anterior. De acordo com a tarefa que os analistas desenvolvem, eles recebem outras denominações (designer, programador, coordenador, especificador...).

### TAREFAS RELACIONADAS À CONCEPÇÃO DO SOFTWARE



### TAREFAS RELACIONADAS À EXECUÇÃO DO SOFTWARE

FIGURA 2 - Diagrama de interação do processo de concepção e execução de *softwares*

O processo inicia-se na Gerência de aplicação de negócios, com a entrevista ao cliente. A gerência escolhe um analista para ser responsável pelo projeto, ou seja, esse analista vai coordenar a articulação de todas as etapas do projeto. Ele deve estar a par de qualquer dificuldade encontrada em cada uma das etapas, e estar preparado para contornar e propor soluções nessas situações. O analista coordenador vai negociar com o cliente o serviço, o cronograma, os recursos (pessoas), as ferramentas (linguagens e programas), prazos, treinamento, etc. Para isso, o analista vai ter que compreender as necessidades e a atividade contida no trabalho do cliente. Esse conhecimento é adquirido através de várias conversas entre as duas partes. Após essas etapas faz-se uma proposta ao cliente, discriminando todos os serviços que serão prestados.

Aprovada a proposta, o coordenador prepara um esboço da modelagem para que um outro analista possa desenvolver a modelagem propriamente dita. A etapa da modelagem consiste em construir a estrutura do banco de dados do sistema a ser utilizado no restante do processo e, futuramente, na utilização do sistema. O analista responsável por essa tarefa se orienta na proposta e no esboço da modelagem feitos pelo coordenador. Caso ele precise de alguma informação extra, deve perguntar ao coordenador, para este perguntar ao cliente e assim retornar a informação para o analista em questão.

Paralelamente à etapa da modulação, o coordenador passa para o designer as informações da proposta para ele ir elaborando a imagem das páginas do software. Na proposta já é discriminada a funcionalidade de cada página, por isso é possível ele pensar num leiaute para elas. O leiaute passa primeiro pela aprovação do coordenador. Em seguida faz-se uma segunda apresentação para o cliente aprovar ou não a imagem do software. Através das imagens, o cliente tem uma idéia concreta de como as suas necessidades serão atendidas. Se ele não aprova, fazem-se modificações até que se atenda as suas expectativas. Uma vez aprovado, documenta-se o fim dessa etapa e o cliente não pode pedir mais alterações, a menos que ele pague por elas.

Terminada a modulação e aprovado o design gráfico de cada página do software, um terceiro analista ou estagiário começa a fazer a especificação do projeto. O analista vai detalhar a proposta descrevendo o funcionamento da página, os seus objetos, suas propriedades e o que vai acontecer quando o usuário acessá-la. Ele também vai sugerir as fontes de consultas que serão utilizadas para construção de tabelas da codificação e o tipo de ferramenta que deverá ser usada na construção do sistema. É um documento escrito numa linguagem técnica, destinado à leitura de pessoas que entendam de análise e programação de sistemas. Finalizada a especificação, o coordenador do projeto a confere e libera para a Fábrica de Softwares dar continuidade ao projeto.

A Fábrica de Softwares, ao receber a especificação, encaminha para um coordenador que será responsável pelo sistema. O coordenador passa a especificação para um funcionário que entenda de design e de programação simultaneamente. Esse designer-programador vai preparar as imagens para serem codificadas pelos programadores, ou seja, ele vai fazer cortes nas imagens para que cada programador possa trabalhá-las como tabelas, separadamente, na etapa seguinte do processo.

A programação consiste em codificar as tabelas seguindo a especificação técnica feita pelo analista. Como as informações técnicas recebidas permitem que as páginas sejam tratadas isoladamente durante a sua própria construção, o coordenador pode designar vários analistas para fazer as páginas do software, ou seja, um analista para cada página. Quando as páginas ficam prontas, o projeto é devolvido para a Gerência de Formulação.

O analista responsável pelo projeto começa a fazer os testes do funcionamento do programa. Ele se coloca no lugar do usuário para verificar a ocorrência de erros e falhas durante a utilização do programa. Quando é constatado algum problema de construção, devolve-se o sistema aos programadores para que façam as correções ou modificações. Terminada a fase de “vai e volta”, faz-se a implantação.

A implantação é relativamente rápida, pois já não ocorrem mais mudanças, aprovações e acordos. O analista responsável vai à empresa cliente e faz a instalação do software. Em muitos projetos, essa etapa é o fim do processo, mas existem clientes que negociam uma outra

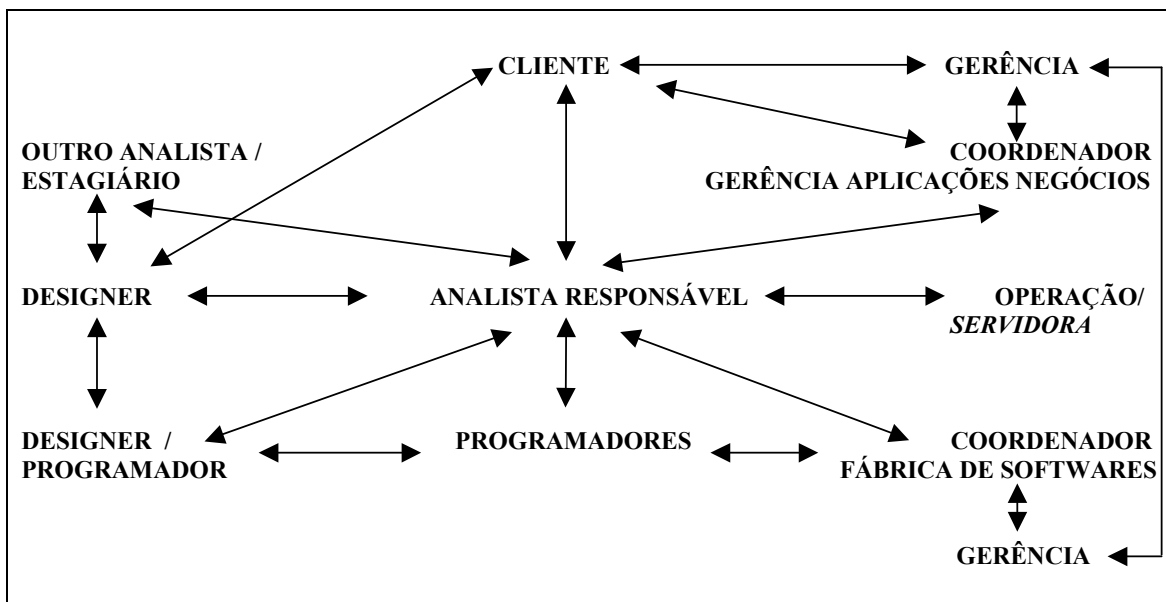
etapa chamada treinamento. Normalmente, o analista responsável formula um pequeno curso sobre o funcionamento do sistema, para ser dado, na própria empresa cliente, aos funcionários ou aos gerentes dos funcionários que irão utilizá-lo.

Para controlar o processo fragmentado, a gerência subdividiu-se em duas células: a Célula de Formulação, que estaria responsável pelas tarefas relacionadas à concepção dos softwares, e a Célula de Construção, que controlaria as tarefas relacionadas à execução dos softwares e que, posteriormente, passou a ser chamada de Fábrica de Softwares.

Após aproximadamente quatro meses, essa nova organização começou a funcionar realmente. Cada célula transformou-se numa gerência, possuindo autonomia para agir com estratégias próprias e para produzir rendimentos independentes para a empresa. A Célula de Formulação, que se transformou em Gerência de Aplicações de Negócios, continuou ocupando o mesmo espaço físico que ocupava anteriormente, mas a Célula de Construção, transformada em Fábrica de Softwares, foi transferida para um outro edifício, próximo à sede da empresa. Aproximadamente dez meses depois, a Gerência de Aplicação de Negócios também foi transferida para o outro prédio, para dar lugar a outro setor da empresa que se expandia, instalando-se no mesmo pavimento onde já funcionava a Fábrica de Softwares.

#### 4.2 – As repercussões sobre a dinâmica coletiva do trabalho

A nova organização do trabalho aumentou o número de pessoas envolvidas no processo e intensificou a interação entre elas. O objetivo prescrito implícito na realização de cada tarefa é cada funcionário executá-la e documentá-la para que o funcionário da etapa seguinte seja capaz de partir desse ponto e dar prosseguimento ao processo, desconsiderando as condições de execução dos fluxos entre as partes e as demandas de comunicação presencial e virtual que aí serão geradas. Isso significa que um analista ao iniciar sua tarefa, baseando-se nos registros das tarefas anteriores, pode constatar ausência de alguma informação necessária ou mesmo perceber erros no projeto, que o impossibilita de executar sua tarefa. Considerando essas possibilidades, o campo de interações necessário à execução de cada projeto pode ser ilustrado pelo seguinte diagrama:



**FIGURA 3 – Diagrama de interação entre as pessoas envolvidas no processo de criação e desenvolvimento de softwares – Gestão Horizontal.**

O gerente interage com o cliente para negociação dos serviços, dos custos e dos prazos. Além do gerente, o cliente se relaciona com o coordenador, com o analista responsável pelo projeto e com o designer para definir suas necessidades e aprovar a execução do site. O analista responsável também ensina o cliente a usar o programa. Este mesmo analista mantém o coordenador informado dos acontecimentos do projeto, e este último mantém o gerente informado. Assim o gerente pode administrar a contratação de novos projetos e a distribuição deles entre os analistas da sua equipe.

O analista responsável também interage com o analista ou estagiário, que faz a especificação, e com o designer do software, que trabalha a imagem das páginas dos sistemas, pois ambos iniciam suas tarefas a partir de informações fornecidas pelo cliente e pelo próprio analista responsável. Caso não haja clareza dos dados, é necessário que esse analista responsável os esclareça ou procure essas informações. O analista da especificação precisa estar muito bem informado sobre os conceitos utilizados pelo designer para não alterar a criação da imagem. Ambos precisam trabalhar informados das suas tarefas, mas também precisam manter o analista responsável a par do que estão fazendo, para que ele possa articular os outros profissionais envolvidos nas tarefas seguintes.

O designer-programador e o(s) programador(es) podem achar falhas na especificação ou não entenderem algum dado, por isso estabelecem, também, interação com o analista responsável. Este último pode também interagir com o(s) programador(es) para que eles façam correções ou modificações nas suas tarefas. O analista pode ou não, intermediar a interação entre o designer-programador e o designer para discutir informações sobre as imagens. O designer-programador interage com o(s) programador(es) para informá-lo(s) e esclarecer dúvidas sobre os cortes das imagens necessárias. Enfim todos eles interagem com o coordenador da Fábrica de Software para informá-lo do andamento do serviço.

O coordenador precisa informar o gerente sobre o andamento do projeto, pedir a compra de algum programa que auxilie a prototipação ou a programação do sistema, solicitar a contratação de mais funcionários, enfim, o coordenador negocia os cronogramas e as metas com a Gerência de Aplicação de Negócios. Essa Gerência e a Fábrica de Softwares precisam interagir, pois, apesar da autonomia de cada uma, o crescimento de uma depende do suporte que a outra pode dar.

Por último, quando um sistema é instalado na servidora da Empresa @, o analista responsável precisa interagir com um analista de outro setor da empresa, chamado de Operação. Nesse caso, o analista responsável depende da disponibilidade e dos serviços dos funcionários desse setor.

Essas interações ilustram como a gestão horizontal se configura para equilibrar o processo linear elaborado pela gestão vertical. O parcelamento do processo implica a interdependência entre os analistas para a realização de suas tarefas, pois o final de uma etapa é o início da outra. Para assegurar a eficiência do processo e a qualidade da produção, os analistas intensificam a comunicação e a cooperação entre eles, configurando um sistema de trabalho onde as propriedades estruturais do sistema de comunicação condicionam o desempenho do coletivo.

Neste contexto, pretendemos verificar como a noção de complexidade, tal como aplicada por Pavard (2000), auxilia a modelização da atividade em questão em seus aspectos coletivos.

## ***5 – A complexidade e suas propriedades***

“Um sistema complexo é um sistema pelo qual é difícil, ou talvez impossível, estudar as propriedades globais a partir de uma análise de um número restrito de seus componentes. Um tal sistema não pode ser estudado a partir do paradigma reducionista clássico que o representaria na forma de um número limitado de variáveis independentes.” (Pavard, 2000: 20)

Pavard utiliza quatro propriedades, características de todo sistema complexo, que são fundamentais para compreendê-los: não-determinismo, decomposição funcional limitada, caráter distribuído da informação e das representações e, por último, suas propriedades de emergência e auto-organização

A seguir, descreveremos cada propriedade e tentaremos ilustrá-las com exemplos práticos, observados em campo, para que se possa verificar a aplicação deste modelo na atividade de criação e desenvolvimento de softwares.

### ***5.1 - Não-determinismo***

Propriedade que diz respeito à impossibilidade de se precisar o comportamento dos sistemas complexos, mesmo quando se conhece perfeitamente todas as funções dos elementos que constituem esses sistemas.

Segundo Pavard (2000: 22), o “pluri-endereçamento”, enquanto “um dos mecanismos mais eficazes para a cooperação” é um importante exemplo de como o não-determinismo pode ter um importante papel de funcionalidade dos sistemas socio-técnicos cooperativos. Este é considerado como um mecanismo não-traçável, ou seja, que não consegue descrever, de maneira explícita, os fluxos informacionais pertinentes para compreender o funcionamento do coletivo, mas contudo, estrutura a memória do coletivo. Esse mecanismo é, provavelmente, o mais importante para compreender a eficácia desse coletivo em situação de co-presença (real ou virtual), pois ele é um dos únicos que permite a partilha de informação. Os fatores que podem influenciar o pluri-endereçamento e contribuir para produzir esse processo não-determinista são vários: o número de pessoas presentes no instante do ato da comunicação, o estado dessas pessoas (interessadas, intrometidas, etc.), sua disponibilidade, o contexto, etc.

Para ilustrar essa propriedade serão descritas duas situações semelhantes, nas quais a analista, responsável por um projeto de um site de uma rede de cinemas para a internet, comunicava-se com a equipe, que estava desenvolvendo o projeto, para que fossem feitas algumas correções. Embora fossem situações semelhantes e que envolviam os mesmos integrantes da equipe, o comportamento deles foram diferentes.

Na primeira situação, a analista responsável pelo projeto ligou para o WEB-designer. Ela queria chamar a atenção de que o texto **REVISTA DE CINEMA** estava escrito de outra maneira: **REVISTADECINEMA**.

O telefone tocou e o coordenador da equipe de WEB atendeu e imediatamente aumentou o tom de voz para mobilizar a atenção dos funcionários que estavam trabalhando nesse projeto:

- “Oi Fulana! Tudo bem?” (referindo-se à analista responsável)

Nesse instante, os dois web-designers e os dois programadores começaram a prestar atenção na conversa para saber para quem era essa chamada.

- “Ah! Você percebeu um erro no texto revista de cinema... e você acha que o problema está no recorte!”

A repetição dos trechos da fala da analista feita pelo coordenador, foi suficiente para que imediatamente o WEB-designer que havia feito o recorte, durante a etapa de prototipação, fosse atender ao telefone. Ele compreendeu a reclamação, desligou o telefone e foi verificá-la. Logo percebeu que o erro não estava no recorte, mas estava no tipo de fonte utilizada no texto. Em seguida ele alterou as fontes e mostrou para os programadores. Um deles achou que as fontes estavam muito grandes e que ele deveria diminuí-las. O WEB-designer executou a alteração. Mais tarde, a analista responsável ligou novamente para o WEB-designer para dizer que a falha estava no tipo de fonte utilizada no texto. Ele comentou que já havia percebido isso e que, também, já havia corrigido.

Na segunda situação, o WEB-designer percebe que o texto “conheça os cinemas” possuía o layer on correto e o layer off sem o “ç”. O texto deve piscar na tela e para que isto seja possível, é necessário construir dois textos com layers diferentes (on e off), que são responsáveis, respectivamente, por dar brilho às letras e por deixá-las foscas. A ausência do “ç” no layer off fazia com que o texto se alternasse na tela entre a grafia correta e a incorreta.

Esse erro foi cometido pelo designer, então o WEB-designer passou um e-mail para o mesmo e outro para a analista responsável pelo sistema comunicando a correção que o designer deveria fazer. Após dez minutos a analista responsável ligou para o Web-designer. O telefone tocou e o coordenador atendeu:

- “Oi Fulana! Tudo bem?” (referindo-se à analista responsável)

O coordenador havia falado o nome da analista num tom de voz diferente e, mais uma vez, todos começaram a prestar atenção na conversa para saber o que ou com quem ela queria falar.

“Ah, sei! Quer dizer que o designer viajou e você quer falar com o Ciclano!” (referindo-se ao WEB-designer)

Nesse instante, o WEB-designer atendeu o telefone, sem que fosse preciso que o coordenador falasse com ele diretamente para atender. A analista havia verificado que o erro realmente existia, mas como o designer não estava na empresa para concertá-lo ela sugeriu:

- “Quem sabe você mesmo não concerta esse erro?” (Analista)

Observa-se que foi necessário sugerir ao WEB-designer que ele tomasse a iniciativa de concertar o erro cometido por outro colega. Logo que ele iniciou a correção, percebeu que a

fonte utilizada no texto não oferecia “ç” e, portanto, o cedilha deveria ser desenhado. Assim, o WEB-designer resolveu não fazer a correção e justificou-se para o coordenador:

- “Como o nosso próprio gerente já nos alertou, não devo fazer esta correção porque depois dou uma solução que fica um pouco diferente do padrão e seremos criticados pelo pessoal da Formulação.” (WEB-designer)

O WEB-designer passou um e-mail para a analista responsável comunicando-lhe que não poderia solucionar o problema e que ficaria aguardando o designer voltar de viagem para dar continuidade a essa página do site.

Na primeira situação não aconteceu uma interrupção do desenvolvimento do processo como na segunda situação. Percebe-se que o mecanismo de “pluriendereçoamento” de situações semelhantes finalizaram-se em diferentes formas, devido à complexidade do erro, do contexto e da mudança de comportamento do WEB-designer para seguir a prescrição imposta pela gerência. Estes exemplos nos revelam, de um lado, a impossibilidade de precisar o comportamento do sistema e, de outro, a funcionalidade deste mecanismo na estruturação dos fluxos de comunicação em contextos cooperativos.

## **5.2 - Decomposição funcional limitada**

Pavard (2000: 24) afirma que um sistema complexo deve possuir uma estrutura dinâmica. Isso quer dizer que é muito difícil estudar as propriedades de um sistema complexo a partir de sua decomposição em subpartes funcionalmente estáticas. A reestruturação funcional de um sistema complexo é permitida pela permanente interação entre o ambiente e suas propriedades de auto-organização.

“...Porém, um verdadeiro sistema complexo não pode ser representado por um conjunto de blocos funcionais isolados, que precisam ser combinados para se obter uma representação do todo. Um dos principais obstáculos à decomposição funcional nos sistemas complexos é o caráter dinâmico e flutuante das funções que o constituem. A interação com o ambiente, os mecanismos de aprendizagem e de auto-organização torna irreal considerá-lo estruturalmente estável.

Por exemplo, uma das propriedades mais interessantes dos sistemas sócio-técnicos é a capacidade de se reorganizar muito rapidamente a sua estrutura funcional. Os agentes podem, em função do contexto, modificar significativamente as regras do jogo e por consequência mudar seus mecanismos de cooperação sem que essa mudança seja programada a um nível central.” (Pavard, 2000: 24)

Tal propriedade pode ser identificada com frequência na atividade de desenvolvimento de softwares em situações semelhantes a esta:

O WEB-designer já havia concluído sua participação num determinado projeto, que chamaremos de projeto P1, e estava fazendo a prototipação de um outro, que chamaremos de projeto P2, seguindo um cronograma estabelecido pela gerência. Dois programadores aguardavam o término desta prototipação, que deveria ser concluída no máximo em sete dias,

para iniciarem a programação do P2. Como a analista responsável pelo projeto P1 comunicou ao WEB-designer que seria necessário, imediatamente, fazer algumas alterações solicitadas pelo cliente desse projeto, a equipe reorganizou as tarefas entre seus componentes por um curto período. Assim, o WEB-designer passou a receber ajuda de um outro programador para terminar o projeto P2 e de dois estagiários, durante dois dias, para realizar as modificações do projeto P1. Com a reorganização da equipe, o prazo de entrega dos dois projetos foi respeitado.

Em situação semelhante, a analista responsável por um projeto P3 recebeu um e-mail do cliente pedindo para ela marcar a apresentação do projeto. Ela conferiu o cronograma e constatou que o cliente estava certo em querer marcar esse encontro. Em seguida, ligou para os programadores que estavam desenvolvendo o projeto para saber quando iriam entregar o software para ela testá-lo. Eles avisaram que estavam atrasados e que não era possível conseguir ajuda de outros colegas porque estavam ocupados. A equipe, em comum acordo, decidiu fingir que a analista não recebeu o e-mail e que ela não ligaria para o cliente. Assim os programadores ganhariam tempo até que algum outro funcionário pudesse ajudá-los.

Nos dois casos citados, as equipes promoveram rearranjos em seus mecanismos de cooperação a fim de solucionar o mesmo tipo de problema (cumprir o cronograma). Essas situações foram imprevisíveis e a auto-organização da equipe para resolvê-las mostrou o caráter dinâmico e flutuante presente nessa atividade.

Os exemplos dados ocorreram em circunstâncias de emergência, mas a reestruturação funcional desse sistema complexo também ocorre de forma inesperada em situações nas quais não existe a emergência. Por exemplo, dois programadores estavam codificando um software seguindo uma especificação que não lhes foi entregue por inteiro. Codificavam, portanto, o sistema na medida em que iam recebendo partes da especificação. Durante esse processo, houve um momento em que os programadores terminaram suas tarefas, antes da transmissão do restante da especificação feita pela analista. Enquanto aguardavam receber o restante da especificação que estavam codificando, eles ofereceram colaboração aos outros programadores da equipe, que trabalhavam em outros projetos. Portanto, a reação dos agentes individuais também não são estáveis, ou seja, esses funcionários poderiam ter tido outro tipo de reação, como por exemplo aproveitar a pausa para descansar, fazer um lanche, ir ao banheiro, verificar os e-mails que chegaram na caixa postal, pesquisar na internet algum assunto sobre o seu próprio trabalho, ou simplesmente conversar com outro colega.

### ***5.3 - Caráter distribuído da informação e das representações***

“A noção de informação distribuída é amplamente polissêmica e veículo de conceitos muito diferentes. Dentro de sua acepção, um sistema é distribuído quando seus recursos são fisicamente ou virtualmente distribuídos em situações e lugares diferentes.” (Pavard, 2000: 26)

No estudo de caso em questão, um analista de sistemas pode distribuir informações virtualmente sobre um determinado software para execução de tarefas diferentes e que podem ser feitas em locais diferentes.

Pavard afirma, também, que a noção de representação distribuída, no campo da psicologia cognitiva, recupera o fato de que durante a interação entre o trabalhador e o seu

ambiente, os artefatos (ferramentas) podem ter um papel funcional na organização do pensamento ou da transmissão de conhecimentos. Mas um sistema para ser considerado complexo possui as propriedades comparáveis àquelas dos sistemas distribuídos, no sentido conexionalista.

“Um sistema distribuído, no sentido conexionalista, é um sistema no qual não é possível localizar fisicamente a informação porque ele é mais ou menos uniformemente repartido sobre o conjunto dos objetos ou dos atores do sistema.” (Pavard, 2000: 26)

A distribuição da informação é fundamental para a realização das atividades dos analistas (criação e desenvolvimento de softwares) da empresa estudada, pois o processo é parcelado em etapas, que são interdependentes entre si. O sentido conexionalista dessa propriedade pode ser ilustrado por alguns exemplos que acontecem no cotidiano desses profissionais:

Num primeiro exemplo, o gerente passou determinada tarefa para uma analista. Ela deveria fazer um relatório que, segundo ela, somente outras duas pessoas da equipe de analistas da empresa já haviam feito. Essas pessoas estavam viajando a serviço da empresa. Durante a execução dessa tarefa, a analista encontrou algumas dificuldades que não conseguiu superar. Como a tarefa era nova para os seus colegas de serviços, eles também não conseguiram ajudá-la.

- “Calculei que gastaria no máximo um dia para fazer este serviço, mas estou trabalhando neste relatório há três dias e ainda não consegui acabar porque tenho dúvidas e não acho alguém para me ajudar.” (Analista)

Outro exemplo foi observado durante o período de transição da organização do trabalho anterior para a atual, quando o processo de desenvolvimento de softwares foi segmentado. Segundo um analista, não estava muito claro para ele e para os outros analistas a tarefa pela qual cada um era responsável. Assim, não sabiam com certeza onde terminava e onde começava a tarefa de cada um e não sabiam a quem recorrer no caso de estarem diante de alguma dificuldade.

- “Quando precisamos de ajuda, ficamos de galho em galho, até acharmos a pessoa certa. Enquanto procuramos essas pessoas, o trabalho fica parado. Por exemplo, existe uma empresa em São Paulo que dá suporte à Empresa @, mas existem pessoas dentro da própria Empresa @ que são capazes de resolver a maioria das dificuldades encontradas pelos analistas, sem que seja necessário pedir suporte à empresa paulista. Mas quem são essas pessoas? Onde e quando podemos encontrá-las?” (Analista)

Nas duas situações citadas, os analistas desconheciam as fontes de informação que os ajudariam a resolver a tarefa, ou parte dela, que estavam executando. No contexto de trabalho desses profissionais é muito comum este tipo de dificuldade, pois trabalham com várias ferramentas (linguagens, componentes e programas) que evoluem rapidamente. Além disso, cada software é um novo e diferente desafio, o que coloca o analista vulnerável a novas e diferentes dificuldades. Estas podem ser desconhecidas, também, para os colegas que convivem com esse analista. Nesta situação quem ajudará esse analista?

#### 5.4 - Emergência e auto-organização

“Um sistema complexo comporta a propriedade de emergência que não são diretamente acessíveis (identificáveis, antecipáveis) a partir da compreensão dos seus componentes.” (Pavard, 2000: 21)

A emergência é uma propriedade que ocorre nas situações imprevisíveis, ou seja, nas situações em que nunca se sabe quando e como vão acontecer, embora se tenha conhecimento do funcionamento do sistema. Diante de uma situação de emergência, os trabalhadores envolvidos nela deverão tentar resolvê-la, através de uma auto-organização entre eles.

Por exemplo, é comum aos usuários de microcomputadores que estes parem de funcionar normalmente e os impeçam de realizar suas tarefas. Uma situação semelhante a essa afirmação foi observada na sala de WEB da Fábrica de softwares. Havia um programador que estava trabalhando com problemas na sua máquina:

- “Esta máquina está cheia de problemas e quando ela começa a dar pau eu tenho que desligá-la. Mas não adianta eu dar um boot (desligar o micro) com os aplicativos abertos, pois quando eu tornar a ligar o micro, ele continuará com os mesmos erros. Fechar os aplicativos demora muito, então eu passo o maior tempo fazendo isso. Os técnicos levaram minha máquina para consertar, ficaram com ela durante três semanas e me devolveram do mesmo jeito. Durante esse período, fizemos um remanejamento e todos mudaram de postos: O Fulano acabou ficando sem posto e trabalhando com um notebook. Agora eu só deixarei que levem minha máquina, na condição de trazerem outra para mim.” (Programador)

Esse programador gastava aproximadamente quarenta minutos para fechar os aplicativos, desligar e ligar novamente a sua máquina para reiniciar o trabalho. Isso estava ocorrendo com uma frequência de cinco a oito vezes ao dia. Para que ele não ficasse atrasado com suas tarefas, pois o conserto do equipamento não era imediato, a equipe de WEB se auto-organizou. Ele e os colegas trocaram, temporariamente, de postos seguindo uma certa lógica: cada um passou a utilizar uma máquina que tivesse instalados os mesmos programas que estavam usando nos seus próprios micros.

Após três semanas, cada um voltou para seu posto e, graças a essa auto-organização da equipe, o cronograma não foi abalado. Mas a situação que parecia controlada mostrou-se, novamente, em estado de emergência porque o problema da máquina não havia sido solucionado. Dessa vez, os próprios técnicos preferiram levar outra máquina, com os mesmos softwares instalados, para esse programador trabalhar normalmente.

O exemplo, citado acima, pode ser ilustrado pela figura 4, na página seguinte. A figura 4 mostra, à esquerda, a ocupação normal de cada funcionário em seu próprio posto na sala. Do outro lado, à direita, a figura mostra o deslocamento do funcionário X, que estava com problemas com no seu equipamento, e dos funcionários 1, 4 e 6. O funcionário 4 passou a trabalhar com o lep top no posto do funcionário X.

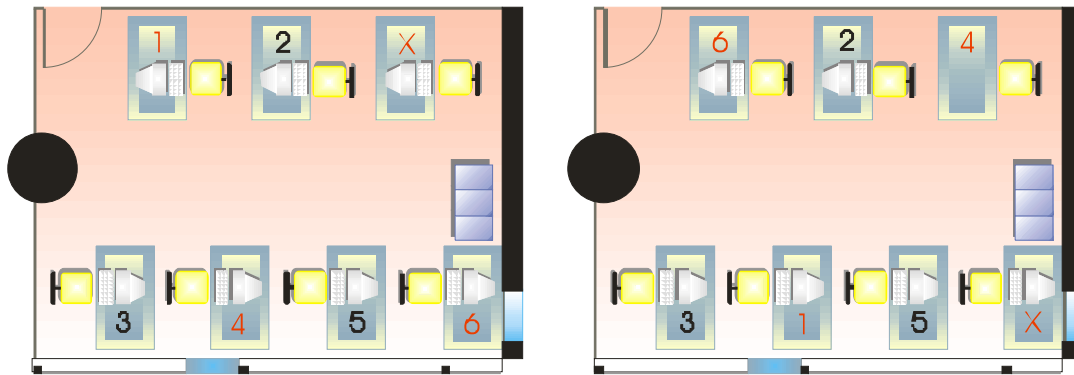


FIGURA 4 – rearranjo de postos da sala de desenvolvimento de *softwares* em *WEB*

Um exemplo que mostra uma outra situação de emergência pôde ser observado quando o gerente da Fábrica de Softwares percorreu todas as salas do seu escritório para perguntar aos seus funcionários se um determinado programa estava instalado nos seus micros. A Empresa @ não tinha licença para utilizar esse programa e em poucas horas chegaria um fiscal para vistoriar as máquinas.

- “Temos que verificar todas as máquinas e deletar esse programa.” (Gerente)

Imediatamente os funcionários interromperam a execução de suas tarefas para verificar a sua própria máquina e, também, aquelas que não estavam sendo utilizadas. Após duas horas, o fiscal fez a vistoria e não constatou uso ilegal do programa sem licença, devido à cooperação e auto-organização dos funcionários da empresa.

Após utilizar a noção de complexidade, tal como aplicada por Pavard (2000) em atividades coletivas, percebeu-se que o fluxo de informação e a cooperação estão diretamente relacionados com a eficiência do trabalho coletivo envolvido nas atividades de criação e desenvolvimento de softwares.

#### **6 - As novas demandas espaciais e as perspectivas de adequação**

O caráter dinâmico deste sistema de trabalho se processa em interação com o espaço, onde as características dos meios de trabalho (espaço, distribuição dos postos de trabalho, disponibilidade de recursos para comunicação presencial e não-presencial) configuram as possibilidades de eficiência do coletivo, ao mesmo tempo em que podem ampliar ou reduzir a carga de trabalho gerada.

Os fluxos de informação da atividade dos analistas de sistemas acontecem presencialmente, através dos deslocamentos dos analistas de um posto de trabalho, virtualmente, através do uso do e-mail, ou pelo uso do telefone. Estas formas de comunicação concretizam-se no “espaço de produção” e este pode influenciar a eficiência dos fluxos de informação e induzir a própria maneira de comunicação, devido ao leiaute, às distâncias, às definições de circulações e ao dimensionamento dos postos.

A separação dos analistas, segundo esta organização, provoca deslocamentos dos mesmos de uma sala para a outra, a fim de se comunicarem. Durante o deslocamento, os

analistas estão sujeitos a se dispersarem com conversas ou com solicitações de ajuda de outros colegas, que estão com alguma dificuldade para executar a sua própria tarefa. A distância física entre os dois escritórios aumenta as chances de contratempos, pois torna-se necessário deslocar do vigésimo andar de um edifício, atravessar uma avenida de tráfego intenso localizada no centro da cidade, percorrer quatro quarteirões, até chegar no oitavo andar do outro edifício. Isto influencia negativamente o processo porque aumenta o tempo de execução de uma tarefa e de interrupções, comprometendo a tranquilidade do ambiente de trabalho de profissionais que precisam se concentrar durante a atividade. A concentração dos analistas é prejudicada quando os postos de trabalho e as circulações entre eles são subdimensionados, não havendo espaço suficiente para aglomerar dois ou mais analistas reunidos em volta de um micro. Nestas situações, os analistas dos postos vizinhos cedem seus postos momentaneamente ou ficam sujeitos a esbarrões e a serem envolvidos pela reunião vizinha.

Para evitar os deslocamentos dos analistas, a empresa determina que eles se comuniquem virtualmente (através de e-mail). Mas eles não conseguem seguir essa prescrição porque existem situações em que é necessário reunir analistas em volta do micro ou é necessário obter uma resposta imediata. Nessas situações, é imprescindível que eles se desloquem ou usem o telefone.

Como a distância física entre os escritórios é significativa, o telefone é utilizado paralelamente à comunicação virtual e à presencial. Quando um analista envia um e-mail para um outro, imediatamente o primeiro telefona para o segundo para certificar se a sua mensagem foi lida ou para apressar o retorno do e-mail enviado. Do mesmo modo, quando um analista precisa se deslocar para outra sala a fim de encontrar outro analista, ele telefona antes de efetivar o deslocamento para se certificar de que seu colega está presente e disponível para recebê-lo, intensificando a comunicação e a interrupção, através do telefone.

Neste contexto o espaço se revela repleto de inadequações, destacam-se o ruído constante oriundo das vozes e o toque incessante do telefone, a alta frequência dos deslocamentos, as interrupções, o desconforto postural das situações de compartilhamento de meios projetados para uso individual.

Sendo a comunicação e a interrupção inerentes à atividade, os analistas se apropriam do “espaço de produção” para adequá-lo a ela. A organização espacial imposta pela gerência condiciona a utilização do espaço, mas os analistas têm liberdade para escolher o posto em que vão trabalhar, para organizar os seus pertences no posto, para trocar de posto quando for necessário, para definir em quais postos de trabalho vão situar os aparelhos de telefone e os trajetos que vão percorrer no deslocamento de um posto para o outro. São pequenas modificações do espaço ou da forma de utilizá-lo, cuja finalidade é facilitar a realização do trabalho. O espaço físico, portanto, sofre influência da “espacialização” dos seus usuários.

“A espacialização é uma atividade, em parte intencional, de concepção do espaço pelos usuários em vista de sua utilização e de sua utilidade. Ela se traduz pela exploração, transformação e a criação de possibilidades de interação entre os usuários, e entre os usuários e o mundo psíquico. Assim definida, ela pode ser vista igualmente, como um processo permanente de otimização das formas e das modalidades das atividades individuais e coletivas, principalmente a comunicação, a coordenação e a cooperação.” (Benckroun, 2000: 37)

Através da leitura das “especializações” é possível perceber quais são as demandas da atividade e adequar o espaço a elas. Para adequar o espaço à atividade e à organização do trabalho adotada pela empresa @ é necessário criar um “espaço de produção” mais flexível, onde os analistas, envolvidos com um mesmo projeto, serão reunidos em um mesmo local. Este pode ser caracterizado segundo a definição de “Espaço de Cooperação Presencial” – ECP, empregada por Benckekroun (2000) para caracterizar espaços nos quais acontece o trabalho coletivo:

“Os ECP concentram os pequenos grupos, cujos membros dividem uma missão global, desenvolvem os cursos das ações fortemente interdependentes, constroem os ambientes cognitivos partilhados, arranjam os meios de trabalho e de ferramentas comuns. Essas atividades são certamente finalizadas por seus objetivos de performance e de eficácia, todas as vezes elas representam igualmente o papel do desenvolvimento do coletivo e de uma comunidade de práticas e de ofícios.” (Benckekroun, 2000: 36)

Com o agrupamento de analistas que estão envolvidos em um mesmo projeto num mesmo espaço, o fluxo de informação tornar-se-á mais direto e a cooperação ocorrerá principalmente de maneira presencial, facilitando a comunicação verbal e gestual (Benckekroun, 2000: 37). Estas ocorrerão somente entre as pessoas interessadas no projeto sem envolver outras, que por estarem em um mesmo local, são envolvidas necessariamente no processo de comunicação. Além disto, a reunião de várias competências em um mesmo espaço favorecerá o trabalho coletivo, porque a troca de informações e experiências ajudam aos trabalhadores a tomarem decisões com mais segurança durante a atividade (Barthe, 2000: 251, Weill-fassina e Benckekroun, 2000: 3).

Ainda que os analistas sejam subordinados a duas gerências diferentes, que representam faturamentos independentes, a organização do escritório em vários ECP implicará a fusão espacial delas. Isto quer dizer que mesmo que os analistas pertençam a diferentes gerências, eles estarão compartilhando o mesmo espaço de acordo com o projeto que estejam desenvolvendo. Esta solução é viável porque as gerências estão, atualmente, ocupando um mesmo pavimento de um edifício.

## **7 - Conclusão**

As mudanças da prescrição baseadas na especialização do trabalho ampliaram significativamente as demandas de interação, a cooperação e a comunicação entre os analistas envolvidos no processo, estando os fluxos de informação diretamente relacionados com a eficiência do trabalho coletivo. Neste contexto, o modelo analítico baseado na aplicação da teoria da complexidade aos sistemas sócio-técnicos complexos (Pavard, 2000) revelou-se pertinente e potencializador da análise das situações de trabalho.

O pluriendereço e a auto-organização se revelaram como importantes mecanismos de potencialização da eficiência desse trabalho. A funcionalidade destes mecanismos na estruturação dos fluxos de comunicação no contexto cooperativo deve-se especialmente a:

(1) a difusão das informações por entre as partes envolvidas e o freqüente desconhecimento das fontes de informação que os ajudariam a resolver a tarefa (caráter

distribuído da informação e das representações). No contexto de trabalho desses profissionais é muito comum este tipo de dificuldade pois trabalham com várias ferramentas (linguagens, componentes e programas) que evoluem rapidamente. Além disso, cada software é um novo e diferente desafio, o que coloca o analista vulnerável a novas e diferentes dificuldades;

(2) sua eficiência na redução de um condicionante freqüente neste contexto, as interrupções da atividade que se processam continuamente frente às demandas de comunicação, especialmente em contextos de emergência.

Os contextos de emergência, relacionados à imprevisibilidade do funcionamento do sistema, se originam especialmente (1) nos disfuncionamentos do sistema técnico, quando é comum os microcomputadores alterarem seu funcionamento normal e (2) na inadequação ou na falta da ferramenta adequada a uma determinada tarefa. Tais contextos demandam uma reestruturação funcional desse sistema submetido a mecanismos de ajustamentos em tempo real onde a auto-organização se destaca.

Neste sentido, a espacialização (Benckroun, 2000) processada pelo coletivo foi observada e inserida como determinante da solução espacial necessária ao contexto. Consideramos o pluriendereço e a auto-organização como os principais condicionantes para diretrizes de projeto frente às demandas de adequação espacial. Esses são mecanismos que exigem o compartilhamento presencial de objetivos. A compreensão de gestos ou palavras soltas de um diálogo só fazem sentido para pessoas envolvidas em um mesmo contexto e a capacidade de auto-organizarem diante de uma situação se deve às tomadas de decisões compartilhadas em tempo real.

Concluimos, portanto, serem os Espaços de Cooperação Presencial (Benckroun, 2000) um bom modelo para a redução dos condicionantes neste cenário produtivo, na medida em que permitem concentrar em um mesmo espaço todas as etapas do processo relacionadas a um mesmo projeto, e que, se separadas pelas prescrições, tal como aqui ocorre, reduzem as possibilidades de comunicação, de coordenação e de cooperação.

## **8 - Bibliografia**

- 1 BARTHE, B. Travailler la nuit au sein d'un collectif: quels bénéfices?. In: BENCKROUN, T. e WEILL-FASSINA, A. *Le travail collectif: perspectives actuelles en ergonomie*. 1.ed. Toulouse: Octares editions, 2000. Cap. 11, p. 235-255.
- 2 BENCKROUN, T. Les espaces de coopération proxémique. In: BENCKROUN, T. e WEILL-FASSINA, A.. *Le travail collectif; perspectives actuelles en ergonomie*. 1.ed. Toulouse: Octares editions, 2000. Cap. 2, p.35-53.
- 3 DANIELLOU, F. *Le statut de la pratique et des connaissances dans l' intervention ergonomique de conception*, Tese de Livre-docência, Université de Toulouse- Le Mirail, 1992.
- 4 GARZA, C. e WEILL-FASSINA, A. Régulations horizontales et verticales du risque. In: BENCKROUN, T. e WEILL-FASSINA, A.. *Le travail collectif; perspectives actuelles en ergonomie*. 1.ed. Toulouse: Octares editions, 2000. Cap. 10, p217-234.

- 5 OMBREDANE, A. e FAVERGE, J.M. - L'analyse du travail, facteur d'économie humaine et de productivité, Paris, PUF,1955.
- 6 PAVARD, B. Apport des théories de la complexité à l'étude des systèmes coopératifs. In: BENCHEKROUN, T. e WEILL-FASSINA, A.. *Le travail collectif*; perspectives actuelles en ergonomie. 1.ed. Toulouse: Octares editions, 2000. Cap. 1, p19-34.
- 7 THEUREAU, J. - *Introduction à l'étude du cours d'action*, Paris, 1990, Tese de habilitação, Universidade de Paris XIII.
- 8 WISNER, A. *The operating models and the choise of variables in ergonomics field research*. Rapport n. 28, Laboratoire d'Érgonomie du CNAM, Paris, 1967.