

**Transcrição das fitas de áudio gravadas durante o Workshop:**

**WOSES - 2005**

**“UM OLHAR SOCIOTECNICO SOBRE A ENGENHARIA DO  
SOFTWARE”**

Realizado em 21 de Novembro de 2005, na sede do  
Banco Nacional de Desenvolvimento Economico Social - BNDES  
Rio de Janeiro, Brasil

<http://www.cos.ufrj.br/woses/>

---

**Promoção**

BNDES – Banco Nacional de Desenvolvimento Econômico e Social  
COPPE/UFRJ

**Edição**

Cássio Adriano Nunes Teixeira (BNDES; PESC/COPPE/UFRJ)  
Nicolaas Ruberg (BNDES; PESC/COPPE/UFRJ)

**Organização**

Cássio Adriano Nunes Teixeira  
Henrique Luiz Cukierman  
PESC/COPPE/UFRJ – Linha de Pesquisa: Informática e Sociedade  
Nicolaas Ruberg  
PESC/COPPE/UFRJ – Linha de Pesquisa: Banco de Dados

**Realização**

PESC/COPPE/UFRJ – Linha de Pesquisa: Informática e Sociedade

---

## **Palestra 6:** Proposições para um olhar sociotécnico sobre a Engenharia de Software – UNIRIO

### **[Renata, Unirio:]**

Como nós somos os últimos, a gente não tem previsão para terminar. Mas de qualquer forma a gente trouxe, eu e o Márcio, estamos aqui representando a UNIRIO, e o grupo que trabalha em Engenharia de Software na UNIRIO, a gente trouxe um conjunto grande de slides, eu particularmente porque eu demorei um pouco, puxa qual vai ser a mensagem que eu vou dar para o grupo, haja vista que a gente está justamente tentando aqui montar a nossa mensagem. Então, a gente na verdade reuniu um conjunto de informações sobre os principais projetos e temas de interesse aos quais a gente tem trabalhado nos nossos históricos de pesquisa, aonde a gente, ..., o que é que aconteceu? A gente estudando esses aspectos e a gente esbarrava com a questão social, e em geral a gente puxa e agora como é que eu resolvo?

E a gente meio que dava uma fugidinha assim de banda e o fato é que uma série de projeto que a gente tem trabalhado aqui tem trazido essa conotação social e a gente queria meio que apresentar a vocês e a gente discutir possibilidades e equalizar os nossos vocabulários. Vocês vão me permitir também falar um pouco da UNIRIO, afinal ela é uma Universidade ainda não conhecida, no cenário, pelo menos, de computação, ela é conhecida como uma Universidade em outras áreas, mas o nosso departamento de computação é um departamento ainda recente, a gente completou 5 anos, formamos a primeira turma no ano passado, é uma Universidade federal aonde tem um centro que é o Centro de Ciências Exatas e Tecnologia cujo contexto abriga dois departamentos, departamento de Matemática e departamento de Informática, aonde nós estamos atuando.

Se vocês puderem ver, é o Márcio dando uma aula, a Universidade está situada num lugar super privilegiado, que é a Urca, quando a gente sai ali daquele prédio onde a gente trabalha de um lado a gente vê o Pão de Açúcar e se a gente olha para o outro a gente vê o Corcovado, então, é um lugar bastante agradável para a gente trabalhar. Dentro do departamento hoje a

gente tem o curso de Bacharelado em Sistemas de Informação; a gente tem desde o ano passado, a gente está dialogando com a CAPES no sentido de montar o nosso programa de mestrado, Sistemas de Informação, e diversos projetos de pesquisa tem sido desenvolvidos lá em Engenharia de Software que é um pouco do que a gente vai mostrar aqui, sistemas distribuídos e também redes, e no caso da UNIRIO justamente por ela já ter um histórico com outras áreas de maior conhecimento, inclusive da sociedade em geral, a gente tem uma filosofia bastante,..., uma tentativa de trabalho bastante multidisciplinar com projeto em outras áreas, as áreas de Teatro, Música, Educação, existem projetos também sendo tocados lá, não necessariamente por nós, mas por outros professores, e a gente tentou aqui organizar, dentro dos projetos de Engenharia de Software aqueles que a gente achou que tem uma relação maior com a discussão que a gente podia fazer aqui hoje com vocês. Seriam 5 esses projetos, alguns deles eu estou envolvida mais diretamente, outros o Márcio está envolvido mais diretamente, ele vai falar mais especificamente sobre eles, e começamos com o primeiro.

O primeiro, na verdade, ele retrata um pouco do meu histórico de pesquisas, tanto de mestrado quanto Doutorado, onde a gente buscou discutir as questões que dizem respeito ao apoio, a colaboração em processos de software. E aí a gente começa com uma apresentação que foi feita pelo Humphrey, que foi um dos principais pesquisadores aí na área de Engenharia de Software, um dos criadores do modelo CMM, numa das conferências dos grupos de Engenharia de Software, mais notadamente americanos, Puxa, os Engenheiros de software estão entre as pessoas que eu conheço que são mais brilhantes mas que em geral tem uma reputação muito ruim em relação a cumprir prazos, negociação, estão sempre excedendo custos, nos trabalhos que fazem, e ele conclui que o grande problema é a cultura que existe ou que existiu, eu acho que ainda existe um pouquinho, de desenvolvimento de software.

E concluía também dizendo, enquanto a gente não mudar essa cultura, dificilmente a gente vai conseguir mudar a Engenharia de Software e as suas práticas. E é isso que eu brinco com os meus alunos, acho que todo mundo aqui viu o filme, eu li o livro também, e na verdade eu trago essa idéia, não

querendo generalizar, mas se você que viram o filme já se percebe, mas no livro fica mais claro ainda, que o Michael Keaton ele, de alguma maneira, tenta caracterizar o profissional de software. E pode não ser talvez uma caracterização perfeita, mas talvez seja uma possível caracterização que a sociedade tem do desenvolvedor de software. Quem lembra, ele era gordo, porco, desorganizado, antisocial, sonegador de informações e o pior de tudo, foi ele que também roubou os segredos da ilha a custo de dinheiro.

Então, eu acho que hoje como profissional de software não se veja exatamente como aquele personagem, mas existia, digamos assim, a idéia de pouca confiança e confiabilidade que se podia ter em relação ao profissional de software. E aí o Humphrey começa a enumerar um pouco os problemas,...., ele bota um pouco da culpa na capacitação, no sistema educacional, que é a nossa discussão aqui, privilegia a técnica e desconsidera as questões talvez mais sociais, as questões ligadas a métodos, processos, como é que você controla o seu trabalho, como é que você mede qualidade, e principalmente, que é o ponto que a gente quer discutir mais que é o trabalho em equipe.

Muito embora a gente seja obrigada a fazer trabalhos em grupo, dificilmente a gente é formado para esse tipo de trabalho. Enquanto a gente está lá na Universidade. E por outro lado, o projeto de software, acho que ninguém aqui pode negar, são projetos eminentemente colaborativos. Eles envolvem equipes muitas vezes grandes, multidisciplinares, em alguns momentos você está trabalhando sozinho, em outros, você está em grupo, existe uma troca desses momentos bastante constante ao longo de um projeto, muitas vezes não perceptível, o projeto pode envolver diferentes setores de uma mesma empresa, ou diferentes empresas, como foi o caso que o Rafael comentou aqui no início. E ao longo desse processo eu vou estar envolvido com diferentes nuances do que a gente pode chamar de colaboração.

Eu posso estar efetivamente trabalhando em conjunto para atingir um objetivo, e em outros momentos o resultado do meu trabalho é um efeito colateral que ajuda ao trabalho de alguém, e em outros momentos eu estou mesmo num cabo de guerra. Estou negociando alguma coisa com alguém. Isso tudo faz com que esse processo, que é eminentemente colaborativo e que muitas vezes a gente não sabe talvez lidar ou apoiar com essa colaboração

eficientemente. A nossa hipótese, como o professor Ivan estava colocando, a nossa hipótese é que se eu trabalho essa questão da colaboração, se eu trabalho o fato de que as pessoas precisam ter um ponto de vista compartilhado, elas precisam convergir nesses pontos de vista, elas precisam entender o que é que está se passando ao longo daquele projeto; quais são os seus objetivos em comum, tudo diz respeito a fazer as pessoas terem o mesmo entendimento do que está acontecendo ao longo do projeto.

Se eu consigo isso, as chances de que as pessoas se tornem mais produtivas, ou o projeto se torne mais produtivo, e de que eu consiga construir produtos que realmente eu planejei ou que eu pensei inicialmente, podem ser maiores. Então, em cima disso, o que é que a gente pensou? Bom, vamos procurar maneiras de apoiar a colaboração. E a gente tem trabalhado bastante, não só na UNIRIO, mas em uma parceria, porque eu fui ex- aluna COPPE, aluna da UFRJ, então, grande parte do histórico dessas pesquisas foi desenvolvido junto com grupo das UFRJ, com professores, um grupo de pesquisa específico, a gente começou a estudar então, dentro do nosso paradigma, a gente começou a querer resolver esse problema de como é que eu apoio a colaboração, como é que eu estímulo a colaboração, através de que? Ferramentas. Então, vamos construir ferramentas, vamos construir ferramental que ajude as pessoas a colaborarem..

O ferramental, que a gente chama de “groupware” , ferramental de apoio ‘a colaboração ele seria um catalisador dessa compreensão. Então, quando eu desenvolvo uma ferramenta colaborativa, ou um “groupware”, na verdade, quando eu aplico isso num processo de trabalho eu posso estar ajudando essas equipes a atingir a tão sonhada maturidade. Quando a gente vê lá, um modelo CMM, é de maturidade, eu quero que as empresas tornem-se maduras no sentido de que elas conseguem aprender sobre os seus processos, elas conseguem reconhecer esse processo e melhorar continuamente, e elas tem uma certa responsabilidade sobre esse processo. Então a gente faz um paralelo aí; quanto mais imatura a organização, mais a necessidade de uma gerência. Quanto mais madura ele é em termos dos seus processos, essas equipes acabam sendo auto direcionadas, como a gente já discutiu aqui em outros trabalhos.

A própria Cobra, tinha uma estrutura de auto direcionamento, porque, porque tinha um aspecto interessante de responsabilidade da própria equipe com o que estava fazendo. E no caso de “groupware” o que vai ajudar nesse sentido. A gente acha que este tipo de ferramenta vai ajudar a tornar o processo explícito e também a colaboração explícita. E com isso as pessoas vão ter uma maior visibilidade do seu processo. Aqui seria um aspecto mais técnico de “groupware”, entre os diversos aspectos que você pode trabalhar numa ferramenta computacional de apoio ‘a colaboração a gente destacou um conceito que é o conceito de percepção, como é que eu posso fazer as pessoas perceberem o contexto de trabalho, a colaboração que existe através daquela ferramenta que elas estão usando para interagir.

E o que é que a gente começou a trabalhar então, puxa se eu conseguisse criar ferramentas que tornassem o processo mais perceptível, mais visível, a colaboração também mais visível e perceptível ‘as equipes, a tendência é que eu ia tornar as pessoas mais capazes de compreender o processo, identificar oportunidade de melhoria, é aquela história de manter o processo como uma referência e se eu abordo essa estratégia de maneira participativa, onde as pessoas podem efetivamente discutir sobre esse processo, as chances de que eu consiga aceitar melhor a formalização de processos podem ser maiores. Aí a gente começou a explorar isso em ferramentas, uma ferramenta que a gente trabalhou foram as ferramentas de “workflow”, a idéia de que o ferramental de “workflow” pode te ajudar com uma série de recursos que ele tem de visibilidade do processo, de que as pessoa possam ter uma compreensão melhor do processo. Então, eu posso ver a estrutura de atividades, como é que está o andamento, qual é todo o conjunto de fluxo de informações de um processo.

Então, sistemas de “workflow”, por exemplo, podiam ser ferramentas catalisadoras nesse sentido. E a gente começou o trabalho em cima de ferramentas de “workflow” por exemplo, tentando criar outros mecanismos que ajudassem a ter uma percepção ainda maior da colaboração, do tipo, por exemplo, eu estou num processo, eu consigo ver o conjunto de papéis e pessoas que participam daquele processo e consigo ver mais ainda, qual a responsabilidade delas naquele processo?

Qual é a minha responsabilidade e a responsabilidade daquele papel específico, daquela pessoa específica. E mais, como é que os papéis, os grupo se relacionam entre si naquele processo em específico? A gente começou a pensar em mecanismos que ajudassem a trazer essa visibilidade, a gente chegou a fazer alguns estudos de caso, nesse sentido e nos estudos de caso preliminares que a gente fez, a idéia é que a essa visibilidade ajudou a algumas pessoas que não tinham talvez utilizado ainda processos como forma de pautar o seu trabalho, elas conseguiam ter uma clareza de entendimento de qual era o processo, com quem elas colaboravam, quais eram as responsabilidades delas específicas no processo. Então isso foi um conjunto de atividades que a gente desenvolveu nesse sentido de buscar maior visibilidade em relação a colaboração.

Hoje em dia a gente está trabalhando junto com o pessoal da França, no sentido de pensar em, ..., a gente leu bastante o autor Peter que você comentou no início da abertura, e que ele dizia que quando eu quero fazer uma organização aprender na verdade isso se traduz muito, resumindo muito a idéia dele de aprendizagem organizacional, o que eu preciso fazer é fazer com que as pessoas entendam os contextos alheios. Então, eu preciso entender o que se passa com aquela pessoa que trabalha comigo, ou com aquele departamento ou aquele determinado papel, dentro de um contexto de trabalho para que eu possa efetivamente entender o que é que está se passando e efetivamente aprender com as situações que acontecem.

Então a gente começou a trabalhar no sentido seguinte. Puxa como é que eu consigo compartilhar contextos? Então, eu estou desempenhando uma determinada atividade que envolve diferentes pessoas, cada um com seu contexto específico e eu preciso fazer elas convergirem para um contexto único, retirando de cada uma delas informações que sejam pertinentes para o entendimento completo daquela atividade. Não vai dar para eu falar muitos detalhes, mas a idéia é de como é que eu posso gerenciar um contexto dentro de processos.

Estou falando muito rápido? Uma outra iniciativa nossa, que a gente chamou de Redes Sociais para Melhoria de Processo de Software e aqui tentando equalizar o nosso vocabulário, entra aí o rizoma, eu acho que o

rizoma vai aparecer aqui bastante fortemente, a idéia desse projeto é um projeto que está tendo um financiamento do CNPQ e ele vem também de encontro a essa questão de melhoria de processo e de como é que eu altero a cultura atual e a resistência 'as práticas de Engenharia de Software e a implantação de processo em geral e a gente viu o seguinte.

Para melhorar processo de software eu preciso aplicar essas práticas, eu preciso fazer uma mudança cultural; vão acontecer alterações explícitas nas relações sociais de trabalho, eu preciso, exige, bastante treinamento nesse sentido e principalmente, a gente já discutiu aqui, comprometimento e participação. E a gente entende, nesse contexto todo, que tudo o que eu puder fazer em relação a compartilhar conhecimentos sobre essas práticas, compartilhar conhecimentos sobre experiências em relação ao uso dessas práticas que estão sendo colocadas através dos processos eu vou esta ajudando as organizações, as equipes, a refletirem sobre as suas práticas de trabalho usuais. Uma maneira de você facilitar a implantação de processo é fazer as pessoas compartilharem o conhecimento sobre o uso dele.

Por outro lado a gente vê, por parte das empresas, hoje até a Ana Regina discutiu um pouco isso, eu concordo com ela que isso está mudando aos poucos, que as pessoa estão conseguindo mostrar que é possível introduzir processos nas organizações, mas por um grande tempo as pessoa tinham muita insegurança quanto a isso. Puxa, vale a pena adotar um modelo? Será que vai funcionar? E se investia muito dinheiro, altos investimentos para se conseguir introduzir processo nas organizações, tinha uma série de desafios, e grande parte dos espaços para discutir esse desenvolvimento e a aplicação dessas práticas, são em geral, na nossa visão espaços ainda mais acadêmicos e mais informativos. As pessoas vão 'as conferências, vão aos fóruns, para ouvir os cases de outras empresas, mas não há efetivamente uma articulação no sentido de desenvolver coisas em conjunto.

Em cima disso, em paralelo a isso, a gente organiza também desde 1999 um grupo, que é o SpinRio que alguns de vocês aqui até já participaram de algumas reuniões, os grupos Spin são grupos que são incentivados pelo próprio SEI, a idéia dos Spin é Software "Process Improvement Network". É que se possa estabelecer redes de discussão em relação as praticas de



qualidade de software. E a gente estabeleceu um no Rio desde 99, já realizamos várias reuniões, encontros com diversas empresas e diversos temas, e a grande idéia,..., não está dando para ver muito bem, foi a nossa última reunião que aconteceu no Banco Central com a apresentação da EDS, o alcance da EDS no CMMI nível 5. e a grande idéia dessas reuniões é o que? É trazer pessoas que já tiveram experiência de implantação de modelos para uma audiência que está ali no cantinho, não dá para ver muito bem, e essas pessoas poderem se conhecer e discutir essas práticas e um pouco desmistificar um pouco dos problemas que podem existir nessa questão.

Aqui é só para dar água na boca de vocês, a gente completou 6 anos, agora em setembro, esse seria basicamente o grupo que organiza, que fundou o spin, e a gente tomou um champanhe, comeu um bolinho, ano que vem se você quiserem também tomar um champanhe... a idéia desse grupo é o que? Estabelecer uma rede de contatos e de discussão sobre melhoria da qualidade. Com base nesse trabalho que a gente tem feito com spin, a gente tem observado o seguinte, legal, a gente faz as reuniões as pessoas vão, discutem e se conhecem, a gente consegue fazer o “networking”, mas está faltando alguma coisa, está faltando ações. Está faltando a gente se articular no sentido de desenvolver ações voltadas ‘a qualidade de software. Então, a gente pensou em desenvolver esse projeto que a gente chamou de RCC Software, rede de colaboração e conhecimento para melhoria de processo de software, e o objetivo desse projeto, seria estabelecer uma rede de conhecimento e colaboração de profissionais de software para trocar experiências, estabelecer uma massa crítica para avaliação colaboração e conclusão sobre as práticas, ampliar a conectividade entre os profissionais, ter uma infra estrutura que fosse coletiva para a troca de conhecimento, divulgação, discussão, reflexão e principalmente a possibilidade da gente conseguir articular ações.

Que tipo de ações? Organizar treinamentos, desenvolver propostas de alterações, por exemplo, de leis..... a gente fez uma sondagem recentemente e uma das pessoas que respondeu ‘a nossa sondagem disse assim, puxa, uma ação que a gente tinha que começar era fazer uma proposta para que as empresas de software pudessem aderir ao SIMPLES. Que é um problema que

empresas muito pequenas hoje em dia não podem aderir a esse modelo. Enfim, daí surgiu a idéia do projeto e o que é que essa rede talvez teria de diferente, o nome que a gente dá, a outras redes, outros grupos de discussão, e de interesse numa determinada prática. Na verdade aí a gente foi estudar um pouquinho, o conceito de dinâmica de redes sociais que é um conceito que existe aí muito ligado a questão, grupos voltados para ações ambientais, por exemplo, então existe toda uma conceituação do que é que é uma rede social.

Essa conceituação diz o seguinte. Bom, quando você estabelece uma rede não há uma hierarquia prévia, essa rede as pessoas que participam elas tem o objetivo comum, voltado para formar ações em direção a esse objetivo; as pessoas entram e saem dessa rede voluntariamente, a questão de que não há hierarquia, há uma isonomia de papéis, uma multi liderança, é claro que vão existir líderes mas eles não são formalmente indicados dentro dessa rede. Ali deve haver uma certa liberdade de expressão, essa rede se auto-organiza, para articular ações e ela vai tendo a capacidade de atrair novos membros para a sua composição.

Então, seria uma maneira, um rizoma, Henrique, de um grupo de pessoas se auto organizarem no sentido de articular ações. Isso tem bastantes exemplos, aí eu botei ali o pandinha, a wwf, por exemplo, ela é considerada uma espécie de rede social nesse sentido. a nossa rede, a gente precisava achar um objetivo comum, que unisse as pessoas; foi esse aqui. Eu quero mobilizar as empresas de software, os profissionais de software do Rio de Janeiro, a desenvolverem ações no sentido de consolidar, fortalecer e evoluir a nossa indústria que eu acho que todo mundo que está aqui pelo Rio percebe que é uma indústria que está tendo dificuldades, digamos assim.

Principalmente as pequenas empresas. A gente brinca dentro do processo que a gente quer salvar os pandas do Rio de Janeiro, enfim essa é a idéia. Bom a gente como, novamente o nosso paradigma leva a gente a pensar em ferramentas, a gente está se baseando também numa tese que foi desenvolvida lá na URFJ, no NCE, orientada por um dos membros do projeto, que ele desenvolveu um ambiente voltado ao suporte a esse tipo de estrutura de articulação de redes sociais, e a idéia é que a gente possa montar também

um ambiente par que essa rede possa trocar conhecimentos, articular ações, compartilhar conhecimentos.

O objetivo do projeto é ter esse ambiente em desenvolvimento. A gente está desenvolvendo esse ambiente, já fizemos uma primeira sondagem aonde a gente divulgou os objetivos da rede, a gente já teve um feedback positivo, no sentido de que, puxa, a gente precisa realmente se articular principalmente no sentido de capacitação, legislações, e a gente espera também que esse trabalho traga resultados para pesquisas em outras áreas que a gente também tem interesse. Bom, essa parte aqui talvez seja o que a gente tem, eu nem diria resultados, porque as outras a gente também não tem resultados, são questões que a gente está trabalhando, mas essa tem nos atraído muito recentemente, que é a questão do Software Livre. A UNIRIO, o nosso departamento ele está envolvido bastante fortemente com o movimento de Software Livre, alguns professores e bastantes alunos do departamento tem se envolvido com essa questão e eu recentemente fui um pouco contaminada com essa área. E fui começar a estudar porque a gente pode observar que apesar da idéia de Software Livre já não ser recente, somente recentemente a nossa área de Engenharia de Software tem dado atenção ao processo de desenvolvimento que se estabelece ali, nas comunidades de desenvolvimento de Software Livre.

Então, na verdade se a gente for olhar os produtos do Software Livre que estão sendo colocados aí, à disposição, eles não estão correspondendo a um novo produto, a uma ruptura tecnológica, são produtos que a gente conhece, sistemas operacionais, por exemplo, mas eles trazem uma forma de desenvolver, é software que quebra, tem quebrado bastante os modelos tradicionais de apropriação de conhecimento, e de desenvolvimento tecnológico, baseado eminentemente num processo novamente colaborativo e comunitário de desenvolvimento. E esse tipo de desenvolvimento está se profissionalizando bastante no Brasil, fala-se já em modelos de negócio em Software Livre e a gente precisa dar uma olhadinha em como é que isso está acontecendo.

Existe uma tese bastante interessante do pessoal de São Carlos, do Christian Reis, que é um desenvolvedor de Software Livre participante de

comunidades de desenvolvimento de Software Livre e ele coloca lá, puxa, é curioso perceber que um modelo de desenvolvimento que aparentemente fundamentado no trabalho de amadores e de voluntários, coordenados de maneira pouco formal, usando ferramentas extremamente simples, e mais surpreendentemente ainda é a percepção que existe entre a comunidade de que o software que está sendo produzido tem bastante qualidade. puxa e a gente aqui querendo porque querendo colocar processos, formalizar tudo, esse pessoal, que as vezes não é nem formado, que trabalha na base do voluntariado, consegue gerar produtos que estão sendo usados por outros. E aí a boa e velha questão do comprometimento e da motivação aparecem. E aí eu fui estudar, bom como é que essa comunidade produz?

E existe aí uma literatura clássica, um ponto de quebra que é o trabalho do Raymond, que ele tenta fazer uma comparação entre o desenvolvimento tradicional e o desenvolvimento de Software Livre. e a comparação que ele faz é que se pode trabalhar de duas maneiras. Uma que ele chama de desenvolvimento em Catedral, para quem não conhece essa é a Catedral de Hans na França, onde se coroavam os reis franceses. E a idéia do desenvolvimento em catedral, que seria o desenvolvimento tradicional, que seria o quê? Puxa você tem todo um procedimento para construir um produto que é especial..., em geral essas pessoas tem que ser extremamente especialistas no que estão fazendo. Você tem prazos, você tem custos, enfim a maneira tradicional como a gente costuma a ver o processo de desenvolvimento de software.

E ele diz o seguinte, olha, aqui em Software Livre o desenvolvimento é assim, o desenvolvimento em bazar. As pessoas não necessariamente se conhecem, elas trabalham de maneira comunitária cada uma contribuindo com uma parte do desenvolvimento. A princípio sem nenhuma formalização prévia e sem se conhecerem de antemão. E a gente olhando com mais detalhe, como é que se passa essa produção..., novamente ela está bastante baseada na questão da colaboração em diversos níveis, tanto em programação quanto em teste, como em documentação, traduções, as pessoas contribuem de diversas maneiras... Como é que e a recompensa aqui? A princípio não é recompensa financeira, é recompensa de reconhecimento, de reputação, de prestígio, e isso

estabelece o que eles chamam de uma meritocracia, os líderes nas comunidades são aqueles que demonstram maior mérito, maior reputação, maior reconhecimento, competência no desenvolvimento...

Há todo um processo para tomada de decisões, claro; podem existir comitês, e existe todo um procedimento para tomar decisões, mas o trabalho é bastante voluntário. Bom, é claro que toda essa idéia de Software Livre ela tem uma conotação ainda bastante radical, com toda a nova tecnologia e toda a nova maneira de trabalhar inicialmente surge, mas aos pouquinhos as pessoas estão começando a compreender mais o processo, está se profissionalizando e as pessoas já estão deixando um pouco o radicalismo de lado e estão começando a pensar mais profissionalmente.

Algumas pesquisas, por exemplo, foram feitas tentando entender um pouco mais essa questão da colaboração em desenvolvimento de Software Livre; duas aqui que eu trouxe, uma do Kishnamurti e a outra do próprio Christian Reis, é que eles dizem o seguinte. Puxa o pessoal de Software Livre diz que trabalha em modelos colaborativos, comunitários, só que a gente foi fazer um levantamento e muitos poucos projetos que você encontra lá no "source forge" ou nas grandes comunidades de desenvolvimento reúnem um número grande de pessoas. A grande maioria de projetos é de projetos com poucas pessoas. A grande maioria de projetos, deixa ver se eu tenho um slide sobre isso,..., não; são projetos em que o próprio proponente do desenvolvimento vai ser o futuro usuário daquele produto. Então, aquele nosso grande e enorme problema, que é entender o cliente, perde bastante a sua dimensão.

Enfim, tem uma série de coisas interessantes para se discutir, o fato é que essa comunidade está querendo se colocar como um modelo de negócio, uma maneira de desenvolver produtos de software, haja vista, por exemplo que grandes empresas, IBM, SUN e outras, elas estão digamos tentando pelo menos compreender ou aderindo a essa movimentação e a gente se questiona da seguinte maneira. Se a gente for olhar também essas pesquisas que estão sendo feitas, tentando caracterizar o desenvolvimento de software livre, as próprias pessoas das comunidades dizem: puxa, eu tenho problemas aqui, como é que eu documento os meus produtos. Como é que eu faço

avaliação de usabilidade? Como é que eu gerencio todo esse trabalho aqui em colaboração? E aí a gente se pergunta, puxa, tudo o que a gente já aprendeu em Engenharia de Software, o que é que a gente pode adaptar e aplicar para esse tipo de desenvolvimento. Aí eu coloquei lá os pinguinzinhos, o pessoal do Software Livre, indo lá pedir aulas, para quem de competência sabe fazê-lo.

E por outro lado, como é que esse modelo de desenvolvimento pelo Bazar,...., caramba, só tem 5 minutos, pode ajudar e influenciar aí a nossa maneira de desenvolvimento, haja vista que é um modelo que produz resultados de uma maneira bastante diferente do que a gente consegue hoje, com a Engenharia de Software tradicional. E isso aí é uma afirmação de uma pessoa da IBM, César Taureon que é responsável por novas tecnologias na IBM, inclusive Software Livre e que ele diz, puxa a conciliação desses dois modelos o modelo mais anárquico proposto pelo método bazar e os conceitos de Engenharia de Software talvez gerem um dos mais intensos esforços de compreensão e desenvolvimento nos próximos anos. Eu acho que aí está um pouco refletida a idéia de como é que você mistura o social com o técnico. Márcio, ainda tem algumas palavrinhas para colocar para vocês, eu peço perdão por ter me estendido.

**[Márcio, Unirio:]**

-Bom, eu vou usar as transparências aqui só como referência, mas eu vou passar mais rápido por elas. Uma outra linha que nós temos trabalhado na UNIRIO, também advinda já em algumas variantes de coisas que nós tínhamos feito antes, por exemplo, a parte de modelagem de projetos de software é alguma coisa que venho desenvolvendo desde 99-2000, que também foi motivo da minha tese de Doutorado, mas que depois se abriu em outras áreas, abriu em outras direções que estamos continuando a desenvolver na UNIRIO. A idéia no geral é que nós temos duas, temos muitos problemas com desenvolvimento de software, em geral, você vê muitos casos de insucesso, você vê muitos problemas sendo apresentados, relatórios, tipo “calls repor”t, retratando uma indústria ainda nos primórdios que apresenta muitas dificuldades em conseguir atingir resultados desejados e você tem duas linhas básicas tentando explicar as razões desse insucesso.

Uma linha é mais tecnológica e diz que os problemas estão no fato de nós ainda não conhecermos muito bem isso que nós estamos criando, nós não temos tecnologia suficiente para isso. O outro lado preza mais o social, que seria problema de gerência. Não sabemos administrar um número tão grande de pessoas com criatividade para desenvolver um produto. Dentro dessa direção você vê vários problemas apresentados pelos projetos de software, em particular a necessidade de você considerar variáveis que você não pode medir diretamente. Considerações subjetivas, em particular os fatores humanos que estão envolvidos em todo o processo de desenvolvimento.

A questão é, nós temos que tomar decisão. Gerenciar um projeto de desenvolvimento de software consiste em você tomar certas decisões no momento em que você precisa, e são vários esses momentos ao longo de um ciclo de desenvolvimento, em particular, em projetos grandes. O que acontece aqui é que as decisões levam muito em conta esses fatores subjetivos, essa capacidade humana de considerar vários fatores ao mesmo tempo, entre eles aqueles mais observáveis e outros mais subjetivos de forma geral, e você tem que juntar isso tudo, você tem que trazer isso tudo para algum modelo, para alguma forma para você poder decidir em cima disso. O que acontece é que hoje nós não temos muito sucesso nessa decisão, visto que os projetos em si não têm atingido o nível de sucesso que se espera.

E parte disso acredita-se que é por conta de você não ter uma forma clara de representar esse conhecimento, trazer esse conhecimento para alguma coisa que seja mais passível de ser trocada, de ser compartilhada com outras pessoas e ajudar a treinar essas pessoas para que esse conhecimento seja mais facilmente distribuído e usado na prática. Então, a idéia desse projeto era representar esse conhecimento gerencial de uma forma que pudesse ser mais compartilhado, de forma que você pudesse construir artefatos, novamente aí ferramentas, que ajudassem a transferir esse conhecimento, que ajudassem a usar esse conhecimento já representado de uma forma explícita, de forma a que nós possamos atingir um maior sucesso no gerenciamento dos projetos.

Com isso foi definido o que nós chamamos de gerenciamento baseado em cenários, onde os gerentes experientes desenvolverem modelos de seus

sistemas, descrições formais dos seus sistemas e deixam essas decisões guardadas para que outras pessoas, no caso gerentes, não tão experientes, têm mais benefícios em usar essas ferramentas venham a utilizar esses modelos e fazer a análise dos seus projetos com base nesses modelos. Eu vou pular aqui os detalhes de como são falados esses modelos, e vou entrar direto nas aplicações. Existem duas linhas principais de aplicação nesses modelos que nós estamos explorando no momento. Uma delas trata de experimentação em Engenharia de Software que é uma coisa que está muito em voga hoje, quer dizer, dentro da Engenharia de Software nós tentamos trazer evidências de que as teorias que nós propomos, as ferramentas que nós criamos, os métodos e processos que são desenvolvidos e são apresentados na literatura de Engenharia de Software, realmente funcionam.

Então, esses experimentos têm uma ligação muito forte com os experimentos que são feitos na medicina, por exemplo, onde você busca evidências claras de que um remédio funciona, de que um tratamento funciona, enfim, e em Engenharia de Software nós temos um problema com esses experimentos que é o envolvimento de muitas pessoas. Você normalmente para testar determinadas técnicas, você precisa de grupos grandes, você precisa de características específicas nesses grupos, que muitas vezes são difíceis de se reunir numa visão mais laboratorial de uma técnica, de um tratamento.

Então, nós através desses modelos nós buscamos em algum momento lá na frente, quando o nosso conhecimento de Engenharia de Software, ou principalmente das relações humanas que existem dentro de Engenharia de Software for tal, que nós possamos criar modelos computacionais pra representar esses conhecimentos e fazer os experimentos como nós chamamos em sílico&&& que é fazer os experimentos das novas técnicas em cima de modelos computacionais. Essa é uma realidade a se buscar, é meio estranho o termo, mas na verdade, uma meta a ser atingida a médio e longo prazo, mas hoje já se explora bastante a modelagem do comportamento das pessoas dos fatores sociais que existem dentro de um projeto de software, das relações que existem dentro desses projetos, busca-se modelar algumas



dessas relações para que se possa chegar a essa base de experimentação mais adiante.

Outra aplicação muito forte também é o desenvolvimento de jogos de treinamento. Eu vou colocar essa telinha pra dar uma idéia do que é que é isso... Essa é uma experiência bastante interessante que nós já estamos desenvolvendo há uns 3 anos. Nós construímos jogos de gerência. Uma espécie de ambiente em que você possa colocar as pessoas para gerenciar projetos como se elas estivessem na realidade, no mundo real. Melhor, tentando chegar próximo do mundo real.

E aí você vê nitidamente a diferença que é você dar uma aula de gerência usando uma ferramenta desse tipo e não usando uma ferramenta desse tipo. enquanto que você está dando aula de gerência, ensinando certos princípios de gerência de você exaurir muito o desenvolvedor numa fase inicial de um projeto de software, ele vai tender a gerar erros que vão se propagar ao longo de diversos artefatos, ao longo do ciclo de vida, e isso vai te sair caro lá no final, nas fases de teste, de inspeção, etc, e na hora que você tem um modelo descrevendo isso, mesmo que esse modelo não seja preciso, como um modelo da Física, como um modelo da Matemática, mas que ele pelo menos retrate essa constatação, essa observação para o estudante, você percebe as pessoas entrando no clima do que é que é o jogo, no clima do que é que é o projeto.

Começando a se preocupar que o tempo está terminando e ela tem que desenvolver; o dinheiro está acabando, e as pessoas não estão rendendo o que deveriam, e o que é que está acontecendo, porque o cara está cansado, porque não está trabalhando as horas que ele devia, enfim, você levanta essa discussão e você traz, você aflora essas características que você não consegue retratar muito claramente no ambiente de sala. Porque você não tem aquele ambiente em que os eventos vão acontecendo, em que as situações de projeto podem ser diretamente observadas, de modo a você poder explicar, porque uma teoria faz com que o sistema reaja de um jeito, e porque outra faz com que ele reaja de outro.

Então, essa experiência de trazer essa modelagem para características humanas, e aplicar isso em ferramentas de ensino, tem sido bastante

gratificante dentro do nosso contexto. Bom eu não sei se nós temos mais tempo,... eu acho que isso é uma outra dimensão que nós estamos explorando também, alocação de recursos humanos em projetos de software, a idéia é que esse é um aspecto fundamental, por tudo o que nós já vimos aqui hoje, você trazer as pessoas certas para as atividades certas, trazer as pessoas certas para formarem determinados grupos, são atividades fundamentais dentro de um projeto que vão ditar basicamente a sua capacidade de produzir um software com sucesso em termos de tempo, cronograma, qualidade, em termos de você formar uma equipe melhor do que ela era antes do início do projeto.

Então nós tentamos aplicar modelagem também nessa descrição do projeto e trazer uma forma, metodologia, uma técnica de você alocar recursos humanos nessas tarefas. O processo de alocação como um todo ele vai envolver identificação das competências e da capacidade das relações entre as pessoas, a seleção de um conjunto de recursos humanos normalmente a partir de um grupo maior do que você tem disponível para selecionar a equipe de um projeto, a monitoração de como a alocação dos recursos humanos está sucedendo, está funcionando num determinado projeto, de modo que você possa ajustar isso numa próxima alocação, e finalmente a avaliação dos recursos humanos para que você tenha um feedback e melhore a própria base de competências a medida em que as pessoas vão participando em novos projetos, passando por ciclos de treinamento, enfim... nós então, modelamos o problema de alocação de Recursos Humanos como um problema de satisfação de restrições onde você tem um conjunto de atividades que devem ser associadas a pessoas ou que devem ser atribuídas a pessoas, você tem desenvolvedores e você tem um ponto de junção que são as dependências de conhecimento que as atividades precisam e o conhecimento que as pessoas detêm para cumprir determinadas atividades.

E o objetivo da técnica que foi proposta é fazer a junção desse lado das atividades que devem ser cumpridas como os desenvolvedores que você tem disponíveis para cumprir essas atividades. Bom, o que é que são essas competências? O que caracteriza o desenvolvedor e a necessidade das tarefas? Tanto aspectos técnicos, conhecimento em Java, conhecimento em UML, conhecimento em processo tal, tal e tal, quanto aspectos sociais,

aspectos de relacionamento dentro do projeto. qual é a característica de tal pessoa que essa tarefa exige? Que características da outra pessoa exhibe?. Como é que essas pessoas aqui trabalham em conjunto? Se relacionam bem, não se relacionam bem, tem características que geram conflito, tem características que não geram conflito, essas caracterizações não técnicas são também informação relevante no momento em que você faz a alocação das pessoas nos projetos.

Como nós tínhamos um modelo e esse modelo ele leva em conta a proximidade das pessoas, as demandas das atividades, nós implementamos também uma solução por otimização, em que você pode configurar a sua equipe com relação a tentando aprimorar fatores de custo, tempo para conclusão do projeto, qualificação necessária para realização das tarefas, sub alocação, enfim, tentar reduzir sub alocação das pessoas nas tarefas que dependam assim de uma capacitação muito menor do que a pessoa poderia oferecer, e temos aqui um exemplo de como se poderia utilizar a técnica, que é um determinado processo aqui, um conjunto de atividades num projeto com suas demandas do que é que cada uma dessas atividades precisa,... aqui nós temos um conjunto de potenciais desenvolvedores apresentando suas características como se fosse uma espécie de currículo dos desenvolvedores,, a maior parte da caracterização ainda é técnica, mas algumas características aqui de desenvolvedores aparecem questões não necessariamente técnicas, como negociação, relacionamento com público, e como é que você poderia fazer o match.

A dificuldade de fazer esse match entre as atividades e as pessoas é o volume de, o número de potenciais alocações que você poderia ter. aqui, nessa primeira tarefa, dentro daquele subconjunto ainda pequeno de desenvolvedores, só haveria um profissional habilitado. Mas por exemplo nessas 3 tarefas aqui, você poderia ter 7 profissionais daquele conjunto se não me engano de 8 capacitados a realizar essas atividades. Dado esse número muito grande de possibilidades que você tem para unir, para relacionar, essas atividades com esses profissionais, você teria uma variabilidade grande também nos resultados a serem atingidos, de acordo com a seleção que for feita. Por exemplo, em termos de tempo para conclusão do projeto que usa

também um outro modelo por trás dos panos, para a partir da caracterização das pessoas identificar quanto tempo levaria para concluir o projeto, você teria algo variando entre 39 dias e 46 dias.

Note, esses modelos não são perfeitos, ninguém aqui está retratando uma verdade absoluta, mas é uma aproximação, é alguma coisa com a qual você pode difundir um pouco mais o conhecimento, você pode difundir um pouco mais a maneira de se pensar e como aqueles fatores estão agindo dentro do projeto. Então você tem um conjunto grande de soluções possíveis... a partir do momento em que você aplica um conjunto de restrições basicamente determinadas pela caracterização das pessoas nas demandas das tarefas, você tem um espaço de busca menor, e finalmente dentro daquele espaço de busca você tem um conj ainda menor que representa as soluções ótimas de acordo com uma dessas escolhas em termos de otimização e de acordo com o modelo que está por trás desse processo.

Bom, com base nesse ferramental que foi criado, nos fizemos um estudo experimental em que se notou que as pessoas têm dificuldades em fazer essas tarefas de alocação particularmente quando mais de um fator, dentre aqueles que foram apresentados anteriormente, é utilizado nesse processo de otimização. Daí a ferramenta se mostrou útil para definir essas equipes quando as premissas do modelo são aceitas como base para se tomar a decisão de alocação. Bom isso aqui é uma panorâmica rápida do tipo de coisas que a gente está trabalhando, acho que agora temos um tempinho ainda para as perguntas.

**[Cássio, BNDES:]**

*Na nossa programação a gente reservou a última hora para a gente conversar sobre os próximos passos do evento. Então, eu acho que a gente podia tentar fazer uns 10 minutos de discussão aqui com o pessoa e depois a gente dá esse encaminhamento. Quem quer fazer alguma pergunta?*

**[Margarida Sa Freire, BNDES:]**

É que na verdade aqui a gente tem uma experiência que você nunca tem muita opção de escolha, a equipe já é dada, é aquilo e pronto. Por

exemplo, eu vi lá. Se eu fosse escolher, é claro que a primeira equipe 39 dias, para 46, 4 pessoas para 7, é claro,..., não precisa nem pensar... mas é muito difícil você ter um grupo de opções, você poder escolher todos esses profissionais para você montar uma equipe. Olha, eu quero que você faça isso e você tem duas pessoas para fazer. Essa é que é a realidade que a gente vive, enfim.

**[Márcio, Unirio:]**

É verdade, em várias situações isso acontece, mas mesmo quando você tem, talvez duas talvez seja um extremo em que a ferramenta ou a técnica não vá trazer muito benefício, mas digamos 4 pessoas se você tem um grupo grande de atividades, mesmo como distribuir essas quatro pessoas dentre essas atividades pode trazer um ganho. E aí o conhecimento dessas relações, o conhecimento dessas capacidades e demandas das atividades pode ser benéfico nesse sentido. Nós ainda não exploramos a quanto essa variação no número de pessoas, número de atividades vai inferindo um maior benefício da ferramenta ou se a pessoa, um gerente experiente teria basicamente o mesmo resultado da ferramenta.

O que eu acredito é que a medida em que o número de tarefas vá aumentando e quando se pensa em implantar processos e seguir uma receita mais definida, esse número de tarefas aumenta até pela segregação de grupos de tarefas que eram consideradas subjetivamente antes, os benefícios vão se somando, os benefícios vão aumentando a medida em que você tem uma melhor caracterização tanto das tarefas, quanto das pessoas. Com certeza, as tarefas ainda são, apesar da definição e esse é um caráter complicado nesse tipo de técnica, apesar da definição das tarefas ser difícil, caracterizar o que é que demanda uma tarefa é complicado, caracterizar o conhecimento de uma pessoa é mais complicado ainda e caracterizar as relações entre as pessoas é uma dimensão a mais, uma camada a mais; mais complicado ainda.

**[Francisco Lima, DEP/UFMG:]**

*Eu estava pensando exatamente sobre essa complicação, quando você falou quando as variáveis vão aumentando as pessoas não conseguem*

*otimizar. Isso é verdade. Mas aí está se tratando mais de uma complexidade de natureza matemática e o que pressupõe que as variáveis sejam matematizáveis. O que a gente tem percebido na prática é que as pessoas até tentam formalizar, elas criam instrumentos, de formalização, para montar equipe, mas na hora de tomar decisão a intuição, esse conhecimento, eles conseguem sintetizar e fazer o julgamento de maneira mais global que é mais eficiente que o julgamento analítico. É aquilo que já foi falado aqui, a idéia de referência ela é interessante, as pessoas vão formalizando até um certo ponto, isso ajuda a balizar algumas escolhas, mas na hora de bater o martelo a coisa é tão complicada, tão imponderável que o instrumento deixa de ajudar.*

*Essa fábrica de software que a gente está acompanhando tem desde pessoas que começaram a trabalhar com equipe fala com fulano; eu não consigo trabalhar, eu não consigo ter um bom desempenho com ele, sem que ele consiga objetivar o que é que quer dizer essa relação, até o ar condicionado. Existe uma separação entre pessoa que toleram o ar condicionado, pessoas que toleram mais ou menos, e pessoas que não toleram. Então, toda a racionalidade técnica foi pelos ares, por causa dessa variável,..., é ruído; a maneira como as pessoas conversam ao telefone, uns são mais sensíveis, alguns reagem bem, outros não conseguem lidar com isso, então, eu acho que é importante tentar conhecer porque quando a gente conhece essas intervenções, quando a gente estava projetando um novo espaço, do local, sem esse conhecimento a gente não ia fazer e a idéia inicial deles é:, primeiro, vamos definir a organização, separar analistas de programadores, do pessoal de teste, manutenção etc, que é toda a organização e depois eles foram vendo que as variáveis eram muito mais complexas do que isso, era muito mais importante resgatar a história anterior, de como que essas equipes foram se constituindo com todas essas variáveis do que implantar e que são justamente quais variáveis são relevantes. E aí?*

**[Márcio, Unirio:]**

O fato de existir um modelo que ajuda... ou, ele guia, serve como referência ou serve até como uma fonte de incentivo para que essas variáveis sejam identificadas. O ar condicionado, o ruído do telefone, o fato de estar

perto de uma pessoa mais barulhenta, o fato de estar perto de pessoas que são muito caladas, são informações que, de contexto para contexto, são importantes. E a existência do modelo te leva a discutir essas informações. Porque, realmente, nem todo mundo precisa da lanterna, nem todo mundo precisa do modelo para tomar a decisão, e muitas vezes, se você olhar no projeto como um todo, limitações do modelo, os gerentes, as pessoas com maior experiência, vão tomar decisões melhores do que as que o próprio modelo indica.

Mas aquele conhecimento vai ficar ali, com aquela pessoa. Ele não vai ser facilmente compartilhado. Outras pessoas vão cometer erros que possivelmente aquela pessoa não cometeria. Uma vez que você tenha um modelo desses e consiga colaboração dessa pessoa, e isso é extremamente fundamental, para tentar identificar esses fatores e tentar relacionar esses fatores, isso sim vai te trazer um benefício em termos de você formalizar a informação que vai poder se transformar em conhecimento na cabeça de uma outra pessoa, nos modelos de decisão de outra pessoa, isso aí tem um valor muito grande. Pelo menos, a minha visão hoje é nesse sentido.

**[Cássio, BNDES:]**

*-Temos espaço para mais uma pergunta. Se ninguém se habilita, eu tenho uma pergunta. Ninguém se habilitou?... Tirou a minha pergunta.*

**[Nicolaas, BNDES:]**

Ôh Márcio, a minha pergunta é simples: será que modelo é uma coisa que nós analistas procuramos, mas que o resto do mundo não quer saber?

**[Cássio, BNDES:]**

*-Se a resposta for rápida, ainda tem a minha.*

**[Márcio, Unirio:]**

Não sei se a resposta é rápida não. É difícil dizer Nicolaas. Um modelo é uma expressão simplificada da realidade, simplificada pelo fato de nós não

termos capacidade de ir mais além do modelo. Mas eu vejo o modelo como um degrau de uma escada; nós usamos aquilo para descrever uma visão do mundo que nós queremos compartilhar com outras pessoas. Eu provavelmente tenho algumas idéias de desenvolvimento de software dadas pela experiência que eu tenho, dadas pelos projetos em que eu já participei e pelas situações que eu vivi. Você tem um conjunto de outras e que o ideal seria que eu pudesse usar somente as suas experiências porque você me passaria um retrato muito mais fiel do que se você desenhar um modelo, se nós sentarmos para conversar e você me explicar determinadas situações.

Mas como isso não é possível hoje, o modelo é a via de comunicação. O modelo não é o objetivo, o modelo é o meio através do qual as idéias são trocadas. E isso... modelos em diferentes níveis. Em algumas situações você usa a matemática porque você busca otimizar, você busca um nível de formalidade que talvez seja um exagero, mas que normalmente é um degrau. Quem sabe, lá na frente, nós não seremos capazes de descrever modelos fiéis da realidade usando estes construtores. Outras vezes o diagrama no papel ou as idéias, ou o texto ou o livro, ou... enfim, são, de certa maneira, modelos. Menos formais ou mais formais.

Então, eu não sei realmente responder esta sua pergunta. No meu ponto de vista, eu acho o modelo importante porque ele é a via de comunicação. É a forma através da qual nós trocamos idéias. E as idéias vão ser a base do conhecimento, a idéia transmite a informação, que junto da sua internalização, junto da sua interpretação, vai se transformar no conhecimento. Então, eu não sei se eu tenho a resposta; eu acho importante como essa via de comunicação.

**[Cássio, BNDES:]**

*- Bom, então eu acho que a gente parte para... Eu posso fazer uma pergunta então... Eu acabei de... A pergunta provavelmente vai ser para ela mesma porque é essa questão do software livre, do modo de desenvolvimento do software livre, como que a gente pode... o que é que a gente pode apropriar desse desenvolvimento.*



*Nós temos a questão central que no artigo do pessoal de Belo Horizonte trata que é como elicitar os requisitos, e talvez a gente tenha que passar a falar em como construir os requisitos. Porque no software livre, quem está construindo, ele está definindo os requisitos. Ele já está construindo o software e os requisitos, estou fazendo um sistema que eu quero usar. Então um sistema que a gente já sabe mais ou menos o que é que é, constrói-se aquele núcleo e aquilo vai se expandindo.*

*Agora, será que isso que acontece na comunidade de software livre vai dar conta de contribuir, e em que grau, com esse problema básico da gente construir os requisitos com o usuário?*

**[Renata, Unirio:]**

É. Tem duas coisas aí. Na verdade, hoje, se você for olhar aí as comunidades de software livre, principalmente aquelas maiores, as mais expressivas, tipo LINUX, etc., hoje elas já estão se configurando como em consórcios e instituições, digamos assim, né, desenvolvimento desses produtos. Então, é meio como se você tivesse... essas comunidades estão se organizando como, eu diria, fábricas, mas como instituições capazes de desenvolver software e produtos que vão ser usados por terceiros, e vão obviamente continuar existindo aquelas comunidades menores que vão fazer outro tipo de produto ainda para uso próprio.

Então essas que a gente imagina que são as que estão se organizando no sentido de oferecer efetivamente um modelo de negócio, eu acho que essas..., a tendência é que se amparem num conjunto de práticas, digamos assim, não só para recuperar requisitos, mas também de todo o processo mesmo de desenvolvimento. Por exemplo, ferramentas que são muito usadas em comunidades de software são as ferramentas de gerências de configuração, as mais usadas, que são as ferramentas que a gente usa também em engenharia de software. Então eu acho que essa intersecção aí entre o que a gente aprendeu em engenharia de software e essas comunidades que estão se organizando um pouco mais, eu acho que vai acabar acontecendo.

Mas eu acho que tem um outro lado também; coisas que eu já imaginei, quer dizer, como é que a gente pode se apropriar da maneira que essas comunidades desenvolveram dentro, organizacionalmente. Coisas que eu já imaginei sem pensar muito, por exemplo, Teste. Por que é que eu não posso pensar em comunidades de teste internamente dentro de uma organização no modelo que é feito do bazar? Eu tenho um produto lá disponível para uma comunidade e que as pessoas testem intensivamente aquilo ali, tem uma massa crítica enorme, e confirmam um pouco mais de qualidade àquele produto. Então, tem também possibilidade de se apropriar desse modelo acho que também internamente em organizações.

Esse paralelo que eu acho que tem aí um desafio grande da gente... de usar como um domínio para a gente discutir não só as questões de engenharia de software mas também as SócioTécnicas, digamos assim.

**[Cássio, BNDES:]**

*-Vamos agradecer ao pessoal da UNIRIO. Então agora a gente passa para a última parte do nosso encontro de trabalho, onde a gente tenta pensar nos próximos passos para essa rede que a gente está tentando construir, que é uma rede que tem a possibilidade de discutir a engenharia de softwares sob esse ponto de vista sócio-técnico. Qual contribuição que a gente pode trazer para essa engenharia que tem esse enquadramento tão tecnicista?*