

**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 4:** Estudos sociotécnicos de caso: o desenvolvimento de software na COBRA nos anos da reserva – PESC/COPPE/UFRJ - Informática e Sociedade

**[Cássio, BNDES:]**

Agradeço aqui à Margarida, ela é uma das pessoas que ajudou a viabilizar a gente estar usando este espaço aqui e todo o material que a gente está utilizando. Além de ter viabilizado e me ajudado a fazer o mestrado. Eu agradeço mais uma vez. Bom, eu vou apresentar a partir de um caso que a gente deu uma observada, o caso da COBRA Computadores; vou tentar discutir algumas coisas aqui, que no fundo tem um cunho mais filosófico, talvez, da gente tentar nos apropriar de algumas coisas para o nosso campo da Engenharia de Software. O artigo que está ai nos anais, é um artigo que ainda está em elaboração, ele não está totalmente fechado, e ele tem um viés etnográfico – a gente chegar lá e observar o que é que as pessoas estão fazendo.

Na verdade, é só um viés, porque essa experiência já aconteceu, é uma experiência que está lá dos anos 70. Mas a idéia é a gente tentar seguir a Engenharia de Software em ação, não o que consta no manual, mas o que é que o cara esta fazendo na realidade. Tem uma aproximação muito grande, uma proximidade muito grande com os artigos da UFMG também. E o título do artigo é *A COBRA Teve Uma Partitura*. Partitura é o nome de um instrumento que aqueles desenvolvedores criaram lá na época deles. À medida que eu fui entrevistando, o pessoal me disse que tinha uma partitura,..., mas o que é que seria essa partitura?

Partitura é o seguinte, para a gente desenvolver um compilador COBOL, por exemplo, a gente colocava nas linhas as várias fases do compilador e nas colunas a gente colocava o que é que ele gerava, por exemplo, e aí setinhas para cima e para baixo indicavam se aquilo era uma entrada, se era uma saída e quando você olhava de longe parecia uma partitura e a gente começou a chamar aquilo de partitura. Além disso, a gente se vale da partitura como uma metáfora também, e uma das bases para esse artigo é um texto do Deleuze & Guatuari, que eles trazem também a idéia da partitura. Essa aqui [mostrando no PowerPoint], está lá no livro deles, é uma partitura do tal Silvano Bussoti.

Eu parei de me questionar se algum músico consegue tocar a partir disso aqui, eu já abstraí dessa questão.

Mas é muito importante para nós então a partitura como metáfora, para a gente tentar entender a questão do padrão, da flexibilidade, que, por um lado, o cara tem que se ater à partitura, que é uma coisa padronizada, por outro ele tem flexibilidade para criar também. Essa tensão é muito comum, a gente vive como desenvolvedor de software e especificamente aqui, no nosso projeto no BNDES, um projeto de implantação de processo de software, a gente vai estar o tempo inteiro lidando com essa tensão aqui.

Então o cunho desse artigo não é tanto trazer a idéia que a gente está tendo aqui no BNDES, mas é mais trazer um pouquinho de um conhecimento que a gente está tentando buscar de cunho um pouco mais filosófico, para tentar entender melhor a nossa prática de Engenharia de Software. Então, essa foi a introdução zero, a meta introdução, a introdução de como que veio a existir o artigo. E aí a introdução 1 é a do artigo propriamente dito: é que há um ano e meio mais ou menos atrás, a gente estava terminando uma disciplina lá na COPPE, uma disciplina que chama “Computadores como Construções Sociotécnicas” a gente estudava um pouco a história dos computadores e no final de uma aula lá a gente começou a quebrar um pau, quem me conhece aqui, sabe que eu não gosto, aí a gente começou a discutir e alguém vira e fala assim: não, já na minha época, hoje em dia para essas empresas estarem conseguindo fazer essas coisas complicadas e complexas, que elas fazem, é fundamental que elas estejam radicalmente ancoradas nos princípios da Engenharia de Software, nos princípios dos padrões de software e a gente começou a discutir. Uns falavam assim: embora muita gente fala que está seguindo CMMI, tem implantado, tem processo, na prática mesmo muita coisa não é seguida. Mas aí, como esse cara era um cara da antiga, talvez contemporâneo aqui do Arísio, a gente resolveu dar uma olhada, por sugestão do Henrique também. Até acho que ele resolveu me dar uma rasteira, meio assim: vá lá na COBRA, vá tentar descobrir o método que eles tinham lá, o processo, que eles tinham lá.

E eu fui e comecei a entrevistar várias pessoas e comecei a estudar um pouco da história da COBRA, buscando um manual. Eu queria um manual da

metodologia de desenvolvimento deles. Eu jurava que alguém em algum momento ia me mostrar um manual com as folhas amareladas pelo tempo e tal, falar está aqui a nossa metodologia. E não tinha, não tinha. Isso aí foi o Henrique que, olha vai lá, vai buscar, não tinha. Eu jurava que eu ia achar. Eu acreditava que eu ia achar, por conta da formação e tudo.

Então, parti dessa pergunta. Qual o processo que os desenvolvedores da COBRA seguiam? Por que? Porque tem uma série de minicomputadores da COBRA que o software é reconhecido pela qualidade que ele tinha. É isso. Concorda Arísio? Basicamente vou tentar falar muito rapidamente sobre tudo. A busca pelo processo de software da COBRA, o discurso, que aí já é um primeiro conceito que a gente vai tentar ver muito rapidamente, depois tem a bibliografia, para quem se interessar... Então, houve um discurso que foi muito importante para aquele software ter aquela qualidade. Aí a gente começa a se perguntar, puxa vida, existia ou não existia o processo? Aí como a outra metáfora, temos a metáfora do rizoma, que pode ser uma metáfora para rede sociotécnica também, aí a gente vai dizer que sim, existiu [o processo de software na COBRA]. Se olhar num CMMI, vai dizer: não tinha processo algum. Mas se olhar com um enquadramento diferente vai dizer, bom teve processo sim.

E vamos tentar tirar algumas lições que a gente podia aprender como Engenheiros de software; aprender a partir de dois princípios que o Deleuze e Guattari trazem nessa metáfora que eles constroem. O princípio da Cartografia e da Decalcomania. Um pouquinho de reflexão sobre a nossa formação, o vínculo forte que a gente tem com o pensamento moderno e com as implicações que isso tem e os rumos da pesquisa que a gente vai tentar executar, que está em andamento.

Todo mundo que eu entrevistei falou assim, graças a deus não tinha metodologia nenhuma. A gente tinha flexibilidade para criar, era um espetáculo. Mas a medida em que eu fui entrevistando as pessoas uma coisa era comum a todos. Não tinha nenhum dos caras que ao falar da COBRA, ao falar daquela experiência, seus olhos não brilhavam. Eu entendi que foi assim a experiência da vida daquelas pessoas. Elas falavam com orgulho, com gosto, com a bandeira empunhada de defesa, mesmo porque a COBRA ficou

atrelada à idéia do fracasso, porque a reserva de mercado, hoje em dia a gente vê de uma forma pejorativa, a história oficial conta a reserva de mercado como uma coisa pejorativa, então, esses caras tinham uma motivação muito grande.

E aí a gente está cansado de ler nos nossos manuais clássicos de Engenharia de Software o seguinte: a motivação é fundamental, o envolvimento das pessoas é fundamental. O comprometimento das pessoas é fundamental. Esses caras tinham isso, e muito. Agora, de onde é que vinha isso? O que é que gerava esse orgulho esse gosto pela história da COBRA. Aí começamos a investigar e vimos que isso pode ser descrito muito facilmente através da concepção de discurso desse camarada aí, desse Paul Edwards. E o que é que vem a ser discurso? Discurso é também o que a gente está acostumado a entender como discurso, como a oratória, a retórica, mas não é só isso.

O discurso para esse camarada aí repousa numa materialidade muito forte. Então o discurso para ele tem tecnologia, tem empresa, tem técnica, tem linguagens, metáforas, tem a prática do dia a dia, e tem o discurso como a gente entende também. E esse discurso para ele, à medida que ele vai crescendo, ele também vai gerando poder, vai gerando comportamento e vai gerando outros fatos e outros artefatos, e outras lógicas, outras tecnologias que reforçam mais ainda esse discurso. Então, a gente podia pensar numa espiral crescente que o discurso vai num crescendo vai se tornando mais robusto, quando ele se torna mais robusto, ele gera mais tecnologia, gera mais comportamento, e por aí vai. Bom, mas que discurso foi esse. Naquela época existia discurso da autonomia tecnológica nacional, de uma forma geral, mas existia o discurso da autonomia tecnológica em informática. Então, aquele pessoal,..., nós estamos fazendo a indústria nacional de informática.

Vamos dar uma olhada muito rápida em alguns elementos desse discurso de autonomia tecnológica nacional em informática, sendo que aquele conceito que o Henrique abordou hoje de manhã, de traduções ou translações é o que viabiliza esse discurso. É o que faz esse discurso acontecer. Então esse discurso, como não é só peça de retórica, tinha vários grupos. Por exemplo, tinha a Marinha, tinha o BNDES, tinha o Serpro, o governo, e tem

mais. Por exemplo, o Henrique falou hoje de manhã, que tradução é uma coisa que envolve um certo ajustamento de objetivos e interesses. Eu tenho um objetivo, o outro tem um objetivo. A gente para conseguir uma determinada coisa, eu me afasto um pouquinho da minha posição, e o outro também se afasta da posição dele, tentando uma síntese, tentando uma tradução para se materializar, ou um fato ou um artefato, no caso aqui a marinha estava preocupada porque ela tinha comprado umas fragatas e não dominava a tecnologia do computador embarcado nelas.

E eles estavam preocupados que se não dominassem aquela tecnologia, a gente estava expondo a nossa própria soberania, se a nossa fragata de guerra, se ela não dominar isso, se a gente entrar em guerra com a Inglaterra, por exemplo, nós estamos perdidos. Aí o BNDES falou assim, legal, esse negócio de construir computador eu acho importante, esse aí é o meu papel, mas olha só, Marinha, não é do jeito que você quer. Não é exatamente esse computador que você quer não. Eu vou entrar no negócio com você, mas é um computador próximo. Não é esse que você quer. Isso só para demonstrar o seguinte, que os interesses deles [BNDES e Marinha, no caso] não são exatamente coincidentes mas vai se compondo, cada um tem um interesse no início e ao final, se compõe uma tradução para materializar no caso o suporte do discurso: o próprio computador. Então você tinha vários atores envolvidos, os oficiais, os engenheiros, a elite de engenharia que existia que não tinha muito campo de trabalho aqui no Brasil, e isso começa a se materializar em várias coisas também, a ter um suporte material.

Então, começa-se a produzir equipamentos mesmo. Terminal inteligente, um negócio que veio lá do Serpro, processador com ponto flutuante, foi desenvolvido no NCE, que é um processador de ponto flutuante para máquina IBM, que evitava a troca rápida daquela máquina, e várias coisas... Seminários, publicações, sociedades... então, esse discurso ia desenvolvendo essas coisas todas e era desenvolvido por isso tudo. Havia interesses altamente antagônicos: tinha o pessoal que estava lutando contra a ditadura, mas que queria um computador, e a ditadura também queria o computador, porque achava importante aquilo.

Aí os caras lá da Universidade, por exemplo, começaram a ver que aquela luta pra se desenvolver o mini computador brasileiro era muito próximo da luta, misturava-se com a luta, de acabar com a ditadura. E por aí vai. Aí a gente tem metáforas, tipo ameaça a nossa soberania, e uma série de coisas. Então, tudo isso aí são elementos que compõem o discurso.

E, no final surge a COBRA. Que ela, além de ter sido uma empresa, ainda existe até hoje, mas não aquela COBRA, existe a COBRA até hoje, mas uma outra COBRA. Ela existe como uma metáfora também. É uma metáfora muito importante no discurso, porque a metáfora ordena o nosso modo de pensamento. Então, surge a COBRA, como a metáfora por excelência desse discurso, uma metáfora materializada numa planta industrial e que pelo próprio nome é uma metáfora.

Por exemplo, COBRA, aqui no BNDES sobre Java, chama o Gladstone lá. COBRA em saber tudo sobre o financeiro. Chama o Zé Marcos. O cara é COBRA no negócio. Então, não é a toa que essa empresa de computadores brasileiros serve como metáfora, mas ela foi uma metáfora materializada numa planta industrial. E desse discurso todo aí, tínhamos o suporte maior do próprio discurso, que é a série de mini computadores COBRA 500. O discurso é isso tudo e ele surge disso tudo também. Não dá para tentar olhar o software do COBRA 500, tentar saber porque é que ele tinha qualidade que atingiu, sem ter a sensibilidade para com esse discurso que envolveu todos os atores e que alinhou todos os esforços de todos os atores como aconteceu.

Então está aí o discurso de autonomia tecnológica em informática que fez surgir uma efetiva indústria nacional, desde a concepção, desde o projeto até a construção, a fabricação, a distribuição de hardware e de software. E aqui vocês me permitam ler um pouco, só para a gente consolidar essa idéia do discurso lá do Paul Edwards. Que pelo discurso ele aponta uma dimensão sóciopolítica da tecnologia. Ele atenta para os elementos materiais que moldam o universo político e social e o discurso é o modo de conhecimento, o discurso é um arcabouço de suposições e acordos sobre como é que a gente deve interpretar a realidade. E além de ser suportado por metáforas paradigmáticas, o discurso é suportado por técnicas, tecnologias que são incorporadas às instituições sociais. Então, o discurso te ajuda a ver o mundo

de uma forma diferente, e ele não é simplesmente retórica mas tem todo esse suporte material também.

Isso aí, nas palavras de um dos ex-funcionários da COBRA, que usou o termo discurso também, mas no modo que a gente entende mais. Falou: *sem xenofobia nenhuma, nosso discurso era afirmativo. Ele é um cara empolgado. Era um discurso afirmativo, de avanço, de conquistas e realizações. Esse negócio de complexo de inferioridade nem passava perto da COBRA. Nem passava perto. Porque a gente estava ali pau a pau com os americanos, o nosso negócio era desenvolver também, o que a gente conseguiu, e não sei o que lá...*

**[Henrique Cukierman, COPPE:]**

*Só observar rapidamente porque quando a Ana Regina falou na parte da manhã, ela falou do alto escalão de uma empresa. Nós não estamos falando do alto escalão não. Isso aqui é o seu corpo de engenheiros que estava fazendo independente, muitas vezes, do alto escalão, que em alguns momentos tinha uma direção contrária ao seu chamado médio escalão, também era o operariado.*

**[Cássio, BNDES:]**

Exatamente, era o pessoal que concebia, tanto o hardware quanto o software. Que concebia, que programava, e tudo mais. Então, gente, é no embalo desse discurso que se dá a produção de software do COBRA 500. então, era uma turma extremamente motivada, envolvida, comprometida que vai desenvolver o software. Diferente de você tentar implantar um processo de software dentro de uma empresa onde ninguém está afim e vai ser mais uma aporrinhção para o cara. Isso faz parte do problema.

Vamos começar aqui a ver algumas coisas assim desse, bem modestamente, deste cunho filosófico que a gente foi tentar enxergar esse desenvolvimento lá. Se a gente chegar lá e falar assim, o processo de desenvolvimento da COBRA era “ad hoc”, cada um fazia do seu jeito. Era CMMI nível 1, não tem processo. Mas por outro lado, se a gente tentar um enquadramento diferente, tentar entender, mas puxa vida, falar que não tem



processo, a coisa era meio complicada demais para ter resultado prático e de qualidade como teve.

Aí se a gente falar a gente pode dizer que não existiu porque não tinha não estava materializado nenhum manual. Como eu esperava encontrar. Não tinha aquele manual baseado numa, hoje seria um manual baseado na UML, e nos autores que por aí estão. Não teria isso naqueles autores daquela época. Mas esse processo, ele existia nas relações dos desenvolvedores, nas práticas e nos artefatos que esses desenvolvedores, eles próprios, desenvolveram. Por exemplo, a tal da partitura. Então, existia materializada em algumas outras coisas que eles fizeram. Mas, no entanto, esse processo, a gente não consegue atingir a essência dele. Ah não, processo baseado na Engenharia, na Análise Essencial. Ou o processo baseado na orientação por objeto. Ele não tinha. Ele existia numa multiplicidade que é um termo que esses dois camaradas aqui usam para tentar descrever a metáfora que eles trazem, que é a metáfora do rizoma.

Esses caras trazem a metáfora do rizoma e vão fazer uma série de constructos para tentar explicar o que é que eles estão querendo dizer com essa metáfora do rizoma. Então a gente vai usar alguns desses constructos.

Rapidinho, para tentar entender de uma forma muito breve, muito rápida. A metáfora do rizoma se contrapõe à metáfora da árvore. Que é a metáfora por excelência que a gente está acostumado e que a gente é iniciado, principalmente nos cursos de estrutura de dados. A árvore tem uma raiz, e os seus descendentes. Então, você sempre pode remontar uma raiz. Essa raiz é uma chave, uma explicação, ou é uma essência como está chamado ali. A questão é a que metáfora do rizoma diz que qualquer ponto pode se ligar a qualquer ponto, é um outro tipo de ordenação, é uma estrutura de dados, a gente não conhece, na qual nós não estamos iniciados. E cuja representação bem desordenada apareceu logo de cara naquela partitura.

Aquela partitura dá a idéia desse emaranhado que é impossível você dizer onde começa onde termina. Para você fazer uma trajetória do ponto original até o filho de enésimo nível.

Então, a multiplicidade é basicamente uma coisa que não tem uma essência. Ah a qualidade do software existia porque tinha o processo tal que

era baseado na metodologia tal. Não tinha isso lá. E nas palavras dos entrevistados, um lá me falou, olha só, nós éramos micro empresas verticalizadas de desenvolvimento. Cada equipe era uma microempresa. Com liberdade, com autonomia, e ligado a todas as outras. A gente por exemplo conseguia que os colegas do Serpro testassem as coisas para a gente. Tudo pelo canal informal. Porque se fosse depender de eu falar para a minha Diretoria, gerar um memorando, mandar para a diretoria do Serpro, descer o memorando, para alguém testar o negócio lá para a gente, não ia acontecer. Eu pegava o telefone, conhecia a turma lá e eles testavam para mim.

Então, o Deleuze e o Guattari falam da conexão e eles falam do sistema a-centrado, ou seja, não tinha um centro. Não tinha essa árvore, se a gente pensar num organograma, uma árvore de comandos centralizados que você vai subir lá, opa, quem manda está aqui, está bem definido aqui. Não tinha isso. E isso ficou muito claro lá, também de acordo com o que eles narraram: olha, a gente estava ligado com todo mundo; a gente resolvia tudo e era um sistema a-centrado. No entanto, mesmo sem ter esse comando centralizado, mesmo sem ter essa idéia da árvore, onde você tem um ponto a partir do qual a ordenação se explica, tudo se auto-coordenava e a resultante final era sincronizada. Os computadores saíam na época que tinham que sair, o software rodava na época que tinha que rodar, a coisa no final dava certo.

Quem olhasse lá ia falar, puxa vida, qual é a ordem aqui? Não sei, é caos, mas do caos tem uma ordem também e esses conceitos também do Deleuze e do Guattari eles nos ajudam a lidar com essa coisa, porque a gente é muito confortável em lidar só com a coisa extremamente ordenada. Desse ponto em diante, abriram-se as possibilidades de entendimento do processo que existiu. Tem um exemplo que eu achei extremamente interessante, tem vários exemplos lá que os caras narraram, mas esse..., sabe como foi feito o compilador do COBRA 500 que a gente estava falando?

O Compilador era o seguinte: a gente estava fazendo em linguagem C, só que a linguagem C não estava acabada não. E era outra equipe que fazia linguagem C. Mas não acaba por aí não. O pessoal que fazia a linguagem C, eles usavam linkeditor, que também não estava pronto não. Era outra equipe que fazia. Você está brincando? Não, é sim... pode perguntar, pode conversar.

Não bastasse, tudo rodava num sistema operacional que não estava pronto ainda... Não, você está brincando, não é possível... Não, ainda tem mais. O hardware também não estava pronto, a gente estava fazendo tudo ao mesmo tempo.

Então, era isso, essa foi a experiência do cara, isso foi o que eles nos contaram. E isso aí, no Deleuze e Guattari, é o rizoma, é o real, é o múltiplo, são as várias conexões. Deus é brasileiro, no final tudo dá certo. (...) o mundo real é assim. Aí, puxa vida, eu ainda relutei, não é possível, cadê o processo? Tem que ter o processo, cadê o processo... até que eu me conformei e aí eu posso dizer que eu tive uma luz, a partir desse ponto eu comecei a entender bem melhor essas questões do sociotécnico, quando realmente resolvi abandonar a idéia do processo, materializado num manual.

E é impossível alcançar a genealogia dessa qualidade da COBRA. Porque se você for procurar uma chave explicativa, isso aqui explica. A hora que você vai ver aquele ponto de explicação ele vai se abrir em muitos outros. E isso vem da experiência que cada uma das equipes tinha já anteriormente e a gente não consegue delimitar de onde é que veio aquela qualidade, porque ela repousa nas diversas inter-relações não lineares, nas palavras de Deleuze e Guattari, no rizoma, no real. Então, descrever o processo de software da COBRA passa muito mais por descrever cada experiência, o caso a caso dessas experiências, dessas microempresas como eles se auto intitulavam. E aí não tem uma coisa definida a priori, antes do cara desenvolver já existe aquele processo, e já vai saber o que é que ele tem que fazer na frente. Não. Então, para descrever o processo a gente tinha que passar por isso e é mais fácil do que tentar partir de uma definição essencialista baseada num modelo padrão, a UML hoje em dia, e que em linha sucessiva de derivação ela ia determinar qual era o processo daquela empresa. Hoje em dia, nós temos um processo, não era nada disso. Para os informatas é o Unified Process que é um processo muito comum hoje em dia que se estabelece quase como padrão hoje em dia.

Aí vamos continuar sem aprofundar nas lições usando a metáfora do Deleuze e Guattari e os conceitos que eles usam para descrever aquela metáfora. Então eu já falei. Se a gente fosse para um enquadramento CMMI,

olha só, não tem processo definido na COBRA, nem definido, muito menos gerenciado. Mas o Deleuze vai falar que o rizoma, o real, a multiplicidade e no caso o rizoma que a gente está olhando é o processo de software, ele não pode ser justificado por nenhum modelo estrutural, nenhum modelo gerativo.

O que Deleuze e Guattari vão falar, o rizoma, o processo de software, você não tem nada que consiga explicar porque a origem dele é uma origem múltipla. Você não consegue apreendê-la. O rizoma é mapa. Aí ele começa a trazer uma outra metáfora para explicar a metáfora dele. É o mapa. E o CMMI, ISO, o mpsbr, isso tudo é decalque. Então, o Deleuze vai falar do mapa e vai falar do decalque. Vamos tentar ver o que é que é isso. Ele fala o seguinte. Um mapa,..., ele para tentar explicar o que é que é o rizoma, para tentar falar o que é que é a multiplicidade e isso tudo para tentar descrever como é que é o real, como é que pode ser o real, ele faz o seguinte. O mapa não é uma coisa que representa a realidade, mas o mapa te permite construir a realidade. Construir rotas. À medida que você constrói rotas você constrói a realidade de uma viagem que você queira fazer, por exemplo. Eu posso sair aqui do ponto A, e posso ir diretamente ao ponto B. Mas eu posso ir do ponto A para o D, E, C, eu posso fazer N possibilidades de rota aqui, isso é o que o Deleuze fala. Um mapa te dá N possibilidades. Múltiplas possibilidades.

Mas o decalque, não. o decalque já é um negócio que vem com uma ordem a priori. Então, se isso daqui é um mapa, e isso aqui é um decalque, quando eu pego esse decalque e coloco em cima do mapa, aí eu pus uma ordem já definida a priori. Estou falando, vou de A até B, passando por C e essa ordem está definida a priori aqui no decalque. Então, ele vem dizer o seguinte. O processo é mapa. CMMI, ISO, essas coisas todas, é o que eu estou chamando aqui de decalque. Porque vem uma ordem pré-definida. Uma ordem sobredeterminada.

Aí vocês imaginam que aqui é o decalque do processo de software da COBRA, na própria COBRA. Aí a gente teria a brilhante idéia, ah encontramos o processo, está aqui. O manual da COBRA está aqui. Encontramos o Manual. Vamos levar para o BNDES. Vamos... aí implantamos. No que a gente implanta, o mapa aqui do BNDES é um pouco diferente. E aí a gente tem um caminho não realizado. Aí a gente vai dizer o seguinte, ah, pelo amor de Deus,

esse negócio aqui é tecnicamente perfeito. Se tem alguma coisa que não funcionou aqui no BNDES, o problema é que o BNDES tem algum problema. O problema é um problema social.

Essa percepção dos decalques e do mapa, essa sacada que o Deleuze e Guattari podem nos dar, é muito importante para a gente começar a refletir sobre os nossos modelos, CMMI, e por aí vai, e como lidar com esses modelos. O que é que a gente faz quando a gente implanta um processo de software? Quando a gente implanta um processo de software, nós estamos construindo uma realidade, construindo um mapa, mudando a relação de trabalho das pessoas, com o próprio trabalho delas. Então, uma vez que o processo de software entrou, o trabalho dela é outro. Mudou e o processo de software também mudou. Quando você implanta o CMMI aqui o CMMI é outro, não é aquele CMMI.

Então a gente tem que pensar no decalque e no mapa, fazendo uma outra analogia, para confundir, mais a gente pode pensar num mapa e no decalque como a partitura, aquela que a gente falava lá. Porque a partitura não pode ser tomada como a própria sinfonia. Porque a sinfonia para existir, tem que ter músico, maestro e tal e tem que executá-la para que a sinfonia exista. Então, se a gente for pensar no CMMI, ele não é um processo de software, ele é um modelo, ele é decalque, ele traz o enquadramento, e os decalques não podem ser tomados como mapa. Tipo o seguinte, olha só. Para ter qualidade de software, implanta isso aqui. Isso aqui garante a qualidade. Não é assim. Mesmo porque o decalque não abrange toda a realidade. Aqui a gente tem o mapa, o decalque é uma simplificação, é um modelo da realidade. É complicado, mas se vocês lerem e tiverem paciência de ler o texto do Deleuze e Guattari, vocês vão ver que é muito mais complicado ainda...

Mas a partir daí, a gente então começa a perceber uma nova compreensão, uma proposta para a compreensão dos chamados fatores críticos do sucesso. Hoje a professora Ana nos trouxe aqui, e foi super legal, questões que não têm nada de técnico, não tem nada de técnico... é comprometimento, é não sei o que, e acho que nós estamos aqui porque temos essa sensibilidade, que está havendo um limite para esse ganho técnico que pode se ter para as respostas técnicas. Então lá, os decalques, os CMMI

não cobrem o mapa, a realidade, a multiplicidade. Usualmente, eles focalizam o problema sobre o viés tecnicista, relegando, dentre outros, os chamados fatores culturais, que o Rafael também já trouxe, os fatores sociais, fatores de comunicação, fatores de política, ele relega isso tudo.

Agora se a gente começa a perceber que o CMMI e Cia Ltda, são decalques, isso pode nos ajudar nos nossos trabalhos de implantação desses processos. Isso pode ajudar muito. Porque uma das formas da gente estruturar a realidade, criar o mapa, criar o rizoma, é exatamente através do decalque. Eu posso entrar pelo decalque para tentar estruturar a realidade que é uma coisa que o Serjão está tentando fazer aqui no BNDES, baseado no CMMI e vai tentando dar aquela mexida... às vezes os beques não jogam como a gente pediu para eles jogarem, mas tudo bem. Então, é isso aí.

Então o decalque, essa ordem sobredeterminante é uma das formas para a gente construir a realidade, o mapa, o rizoma. Ou o processo de software. Aí vamos tentar algumas reflexões aqui também, e eu não fico, pela formação que eu tenho, eu não fico muito à vontade para essas reflexões. Esse texto, essa formação moderna, eu nem sabia que eu tinha uma formação moderna. No Mestrado foi espetacular porque eu descobri que eu tenho uma formação moderna. Num primeiro momento eu achei que isso era bacana, hoje eu acho que não é tão bacana mais não, mas vamos lá... mas é interessante.

Esse camarada aqui ele fala o seguinte. Projetista, Engenheiro de software, pesquisador, tem mania de não querer saber de teoria nem de filosofia, mas isso é fundamental se a gente quiser dar um outro passo na Engenharia de Software. É fundamental a gente abordar essas questões aqui. E eu vejo que, se eu não tivesse,..., se eu tivesse ido com o olhar do Engenheiro de software lá na COBRA, eu não teria enxergado nada, eu ia falar que não existiu processo lá. Então, a gente tem que perceber que a Engenharia de Software é uma construção. O Arísio hoje falou, porque não a Arquitetura de Software?

O Henrique também falou, por que não a Arquitetura de Software. No almoço eu estava falando, puxa, Engenharia de software, para ter esse cunho científico, para ter esse valor, porque podia ser Astrologia de Software... quando chega, quanto tempo vai demorar,..., só chamando a mãe Diná para

saber quanto tempo vai demorar, nem sei quanto tempo... se fosse Astrologia de software não ia ter o cunho de ciência, não ia dar certo. Mas a gente tem que entender que nós estamos construindo o que é a Engenharia de Software. E questões importantes na Engenharia de Software estão fora do escopo dela. Eu acho que estão, eu acho que é por isso que a gente está aqui. Porque tem uma série de questões que a gente fala de não técnicas, que estão nos incomodando, e que a gente sabe que a própria Engenharia de Software não está tomando conta dessas questões.

Agora, se não está tomando conta a gente fala isso não é Engenharia de Software, depende da construção de Engenharia de Software que a gente está querendo fazer. Nós temos que colocar isso para dentro da Engenharia de Software.

Então, como o René Descartes lá, o vínculo que a gente tem com esse pensamento moderno. Com René Descartes a gente viu que o método passou a ser a única diretriz para a verdade. No método científico no caso dele. Para ser científico, para ser verdadeiro, tem que seguir um método, outra pessoas tem que conseguir repetir o que você fez, porque aí a coisa vai tomar ares de verdade. Isso se entranhou tanto na Matemática e nas Ciências Naturais que passou a dominar na nossa cultura o que é que conta ou não conta como conhecimento. Daí a Engenharia de Software quis tirar os contaminantes ditos sociais para ela estar na conta de uma disciplina científica também. Então, ela começou a tirar a questão social, política e tal, para se manter como uma disciplina puramente técnica, porque essa coisa aqui vai dar mais certo com as questões ditas mais técnicas. Então, vamos tentar clarear um pouco esse ponto. O pensamento moderno muito baseado no mecanicismo, positivismo; e hoje a gente vive esse tecnicismo, a era da técnica e da tecnologia, a gente acha natural que as nossas idéias de desenvolvimento de sistemas, por exemplo, sistemas de software, elas se baseiam nessas idéias de representação, ordem, formalização. Vamos começar a ver.

Representação... como Engenheiros de software, como Analista de sistemas, a gente acha normalmente, os livros ensinam para a gente, vá lá, elicite os requisitos, faça uma representação, implemente. Ordem... vá lá e elicita o requisito. Uma vez que você fez o seu modelo, a ordem você já pegou,

aquilo ali é imutável, não está mudando. A gente sabe que não é assim. Mas a questão da formalização, tem que ter linguagens para formalizar as coisas. A regularidade dos fenômenos, a possibilidade de controle, a separação entre o sujeito que está fazendo que está praticando a Engenharia de Software e o objeto que basta a ele descobrir. Então, se a gente pensar nas ciências, por exemplo, vamos pensar na Física newtoniana, que aí fica mais fácil, você tem a idéia da ordem.

Você tem as 3 leis da física. As 3 leis da Mecânica. O sistema é ordenado. Aquilo lá é imutável e você pode fazer modelos, você representa esses modelos, você consegue entender o mundo a partir dali e você consegue controlar o mundo a partir dali. Toda a lógica moderna se baseia nessas idéias e a Engenharia de Software também está baseada nisso daí. A gente pratica isso. Hoje, aqui, eu tenho certeza, todo mundo concorda, não tem essa de você pegar requisitos que eles não vão mudar. A gente sabe que nem tem jeito de você pegar os requisitos a priori. Porque quando você desenvolve o sistema e você coloca lá, ele já altera a realidade, com outros requisitos também. Mas essas idéias da modernidade estão muito presentes na Engenharia de Software. Então, com isso a gente gera o enquadramento da Engenharia de Software, um enquadramento ou um decalque.

A gente sabe o seguinte: a Engenharia de Software está aqui, a gente pode,..., o objeto da Engenharia de Software pode ser representado, ele pode ser formalizado, o Gladstone pode chegar lá, elicitando os requisitos e fazer um modelo de requisitos, define tudo e não vai mudar nada, e depois a gente vai conseguir controlar, é só botar o sistema seguindo o método, a gente tem a nossa racionalidade, nós estamos fazendo uma coisa racional, vamos construir um programa, e também está baseado na questão da burocracia, que é você definir perfis, definir papéis a priori, quer dizer a Engenharia de Software que a gente é formado ela está muito enquadrada nesse tipo de coisas. E aí nós estamos aqui por que?

A gente começou a ser sensível pela questão da interpretação. Por exemplo em senso comum, cada um interpreta de um jeito, aí tem também o imprevisto... Ih, isso não estava no modelo. E agora, Gladstone, o que é que a gente faz? E o comprometimento. Nós não falamos que tem que estar



comprometido? Mas o que é que tem comprometimento a ver com a razão, por exemplo. E a questão da cultura. Funciona num lugar, funciona no outro... ih, a prática mudou.

Antes a gente fazia Análise Essencial, agora é orientação por objeto. Ih mas esse problema aqui é o caso a caso. Como o Arísio falou. E o contingente, olha, aconteceu isso aqui. Além disso o patrocínio,... fundamental, se não fosse o patrocínio, do chefe, a gente não estava aqui. E as decisões não racionais. Vai falar que não tem... vai falar que os nossos projeto de desenvolvimento é tudo decisão tecnicamente racional. Claro que não... pinta de vermelho, mas vermelho é o dobro do preço. Ta bom, eu quero que seja vermelho, vermelho eu gosto. Vai e pinta de vermelho. E o caos. Existe o caos.

E as lutas de poder. Opa, não vai implementar esse sistema aqui do DRCA, porque senão eu vou perder o meu serviço aqui. Vou ser mandado embora. E o conflito de interesses. Eu quero que seja assim, o outro quer que seja assado. Então, a gente podia ficar tentado, eu mesmo fiquei tentado, faço mea culpa aqui, a pensar o seguinte: não, então a gente tem que gerar um outro enquadramento mais abrangente. A gente tem que tentar gerar um outro enquadramento que vá pegar mais essas coisas e nós vamos ser felizes para sempre. Mas a idéia, a questão é que é impossível, o que está errado aqui, dentro dessa abordagem sociotécnico, o que está errado, o conflito, o que impede, o que faz a coisa não ir bem, não funcionar bem é essa divisão é impossível de ser feita. É impossível.

Na verdade, a gente lida com isso aí. O técnico está no social, o social está no técnico, o comprometimento, o controle, a razão com a decisão, o racional, o método com a cultura, está tudo misturado. É o que o Henrique falou hoje cedo. Que o camarada olhando a placa de um computador ele viu aquela complexidade, aquela burocracia, caramba, isso daqui é a mesma coisa do organograma da empresa que está construindo. As coisas são misturadas. E essa é a dificuldade que a gente tem por conta da nossa formação. A gente tem a tendência, a gente foi formado para entender o mundo com as grandes divisões. Para nós é difícil, para mim é extremamente complicado, partir do princípio que não tem divisão nenhuma.

E que vamos fazer. Mas isso é o sociotécnico também. Mas qual seriam os desafios aí dos Departamentos de Informática hoje em dia? O ambiente é cada vez mais mutante; o ambiente cada vez mais incerto e a complexidade das tarefas cada vez maior. Só que a gente aplica nossa Engenharia de Software baseado nas premissas do mecanicismo. E as premissas são que o mundo é imutável, ou que pelo menos ele muda, mas a gente sabe a lei com a qual está mudando. Então, eu sei prever o futuro. E o mundo é ordenado. Então, essas são as premissas que tem baseado, que tem, melhor balizado o desenvolvimento da Engenharia de Software e a gente está aqui porque a gente, eu acho que nós concordamos que existem uma série de fatores ditos sociais que esse enquadramento não dá conta.

Só para a gente pensar, por exemplo, o engenheiro de software, a realidade que ele vai ter que fazer, o sistema que ele vai construir já é uma realidade que existe antes de ele o construir? Tipo o cara vai achar um tesouro. O tesouro já está lá afundado. Basta o cara descobrir onde, qual caverna qual navio que está afundado lá, resgatar o tesouro, aquele tesouro já existia... O objeto com que o Engenheiro de software lida já existe antes? É uma coisa notoriamente construída em conjunto com os usuários. Construída em conjunto com os processos de negócios. Construída em conjunto com a própria organização. Construída em conjunto com a própria tecnologia.

Se a gente estivesse desenvolvendo hoje só em Cobol, a gente estaria fazendo de uma forma, mas hoje a gente desenvolve diferente porque a gente tem outras coisas. Então, isso tudo está envolvido. Então, o Engenheiro de software e a realidade independem um do outro? E isso vai muito em choque com as premissas da modernidade que escoram o desenvolvimento da Engenharia de Software como a gente conhece.

Bom, eu já estou terminando, basicamente o que eu queria dizer aqui agora é o seguinte. Como é que a gente está pensando em continuar a pesquisa? A gente fez aquele primeiro estudo de caso lá da COBRA, que também não está completo ainda, mas é um estudo um pouco mais histórico, e a gente pretende agora estudar outros casos. Um caso natural é um que eu estou hoje envolvido num projeto de implantação de um processo de software e eu quero olhar pelo menos mais dois casos. Mas basicamente o nosso

momento é o seguinte. É aceitar que a complexidade do objeto da Engenharia de Software é uma complexidade sociotécnica. Não adianta falar que é só técnica, ou que é social. É uma complexidade sociotécnica e a gente está pretendendo ver o processo de software, mapa ou rizoma, ou rede sociotécnica, tudo isso seria uma metáfora para enxergar essa realidade a gente está tentando ver como uma infra estrutura, que é uma outra metáfora. Essa a gente está puxando dos noruegueses. Mas basicamente por que? Porque eles têm um,..., esse camarada aqui, ele tem essa seguinte,..., isso daqui foi a adaptação de uma frase que ele tem lá mas que é mais ou menos o seguinte: a gente tem visto implantação de processos de software apenas como a gerenciável tarefa de especificar e seguir padrões técnicos, deixando de lado ou tratando muito pouco, ou falando que é problema de outra disciplina, deixando de lado a grande e incontrolável tarefa de alinhar, as inovações, alinhar a tecnologia, alinhar as políticas, as condições de mercado, as condições organizacionais, a cultura, ou seja, o sócio programa todo envolvido com essas discussão. Aqui só para dar alguns tópicos de definição da infraestrutura ele vai dizer que a infra estrutura é uma coisa que tem que dar conta da tensão que existe entre a padronização e a flexibilidade. Isso é muito claro. Como padronizar o processo de trabalho de um desenvolvedor de software, mas manter a flexibilidade de criação dele? Que os caras da COBRA resolveram com a tal da partitura, de uma certa forma eles se entendiam mas cada um compunha dando o seu toque pessoal ali.

Uma infra-estrutura como um processo de software, é um sistema aberto, dinâmico, mutável, eu não sei as delimitações desse sistema a priori. Uma infra-estrutura como um processo de software é uma coisa que habilita e suporta a realização de outras atividades. Um processo de software para as pessoas construírem software aqui no BNDES. Construir sob um determinado entendimento melhor. É para isso que a gente está tentando fazer isso. Obviamente uma infra-estrutura é compartilhada por diversos grupos de usuários. Essa questão de ser um sistema aberto é interessante, até onde que vai o processo de software e a partir dali começa o processo de negócio, a modelagem do negócio, então a coisa é complicada. E a infra-estrutura ela

existe fisicamente, incorporada em arquiteturas, padrões, ferramentas,... Vamos instalar a suíte da Rational, por exemplo, bota lá.

A infra-estrutura para desenvolvimento de software vai estar lá e obviamente ela deve dar um retorno econômico. Ela tem que se pagar. E além dela se pagar... e uma coisa interessante é que a base instalada de uma infra-estrutura e de um processo de software, a base instalada também conforma o processo de software ou a infra-estrutura. Por exemplo, quando você vai mudar o seu processo de software, obviamente você está herdando você está mantendo talvez a compatibilidade ou você está sendo influenciado com tudo aquilo que já existia.

Basicamente eram esses pensamentos, que não estão muito estruturados talvez, mas basicamente era para dar uma idéia do que é que a gente está tentando discutir lá na COPPE. Agradeço a atenção. Se alguém quiser perguntar aproveita enquanto o orientador ainda está perto porque sozinho eu não respondo não.

**[Arísio, A&S Consultoria:]**

*Querida fazer um comentário, oi, Arísio, consultor independente, só lembrar... eu fiz um treinamento de Análise Estruturada... deve ter sido no início dos anos 70. uma transparência me marcou muito. Eu guardei essa transparência até hoje, na minha cabeça, eu só não tenho o desenho aqui. Mas estava escrito. Tudo está fortemente interligado. Se não me engano isso era atribuído ao Tom de Marco. É tudo o que a gente está falando aqui. No fundo é isso que a gente está falando aqui. E isso era dos anos 70 e poucos. Então, só quero ressaltar, só estou fazendo um comentário, para dizer que a gente não tem grandes novidades não. Os nossos problemas são antigos, a gente já os conhece há bastante tempo. A gente está pisando nesse barro já há um bocado de tempo e não está saindo tijolo, por enquanto.*

**[Cássio, BNDES:]**

*Mais ninguém?*

**[Renata, UNIRIO:]**

Na verdade, é também um comentário, não é tanto uma pergunta, é interessante a gente ver que as coisas se interligam aqui nas apresentações e com o que a gente já vem estudando, e é engraçado ver como um discurso completamente distinto, talvez do que eu estou acostumada a estudar e a ver, como que as coisas se encaixam. Então, eu tenho muita, acontece muito comigo em cursos que a gente vai falar de processos, software, vai falar de modelos, que os alunos querem deixar a gente meio que numa saia justa, e falam assim, isso é tudo besteira, a gente não precisa de processos, as pessoas tem que trabalhar mesmo de maneira livre, etc e tal; e a gente tenta convencer as pessoas da seguinte maneira.

Olha, pelo menos é a minha visão, o processo nem tanto é uma camisa de força, uma coisa que você tem que seguir explicitamente, como por muito tempo se tentou vender a idéia de processo, mas mais do que isso ele é uma referência, ele é a tal da infra estrutura que você quer colocar aí, seria ótimo se eu pudesse trabalhar em todas as empresas no ambiente que a Cobra tinha, pessoas altamente motivadas, capacitadas, creio eu, com uma sensação de querer produzir alguma coisa grande e importante para o país... Isso não é verdade sempre, a verdade é que nem sempre você tem as pessoas totalmente capacitadas, você nem sempre tem as pessoas totalmente motivadas, e você precisa do processo para ajudar.

Então, eu costumo sempre a dizer assim, o processo é uma referência, ele está lá para todo mundo ver, para todo mundo ver, entender como é que deveria acontecer o trabalho internamente e ele está lá justamente como o tal decalque, para você ir tentando mudar. Se você observa determinadas práticas não funcionam 100%, aquela questão da flexibilidade, ele é a sua referência para mudar também. Então eu acho interessante ver os conceitos os nomes e a maneira como se apresentam essas coisas, que de fato é como a gente tenta praticar.

**[Cássio, BNDES:]**

*Para a gente começar a produzir tijolo, talvez a gente não produza porque no fundo a gente está muito amarrado na questão da modernidade, a*

*gente está muito amarrado, no mecanicismo, no positivismo. Se a gente começa a afrouxar essas coisas e começa a dialogar com essa turma dos pós-modernos esses conceitos, a gente consegue perceber então, aceitar que a gente não precisa ter sempre o modelo sobredeterminante, que o caso a caso é importante de ser tratado, também que isso é ciência, que isso pode ter uma eficiência, depende tudo do enquadramento.*

**[Henrique Cukierman, COPPE:]**

Eu acho que você [Renata] fez uma observação muito importante, que é em relação, acho que tua observação dizia respeito ao vocabulário, tem um vocabulário, e eu acho isso extraordinariamente importante. As palavras,..., a gente costuma comentar isso, a gente tem que ter uma certa artesanaria com as nossas palavras e tentar expandir o nosso vocabulário para descrever realidades complexas para as quais ou diante das quais, os velhos vocabulários não dão mais conta. O que eu digo é assim. Nos tivemos a parte da manhã, fazendo já um balanço, eu sinto assim a vingança do não técnico sobre o técnico. Ah, isso aí não é técnico, quer dizer, poderia dar a impressão de que o não técnico venceu o técnico, se vingou, massacrou, o que também nós estamos limitados por essas palavras, por isso é que quando a gente sugere o sociotécnico, não é só uma medida retórica. As palavras elas são veículos importantes de comunicação, de enquadramento, de compreensão, a gente sabe disso.

Então, fico feliz de você fazer essa observação. A segunda coisa também, que eu queria observar é que não está claro, nem na apresentação nem no artigo, quem já leu ou quem for ler, o que é que,..., veja bem, tem uma insinuação poderosa aí, que é a seguinte. O que teria acontecido se nós tivéssemos tido a oportunidade histórica, e aí eu falo em termos de Brasil, falo com uma certa emoção, eu estou no prédio do BNDES, que tem um papel na história desse país, se nós tivéssemos tido a oportunidade histórica de prosseguir com a experiência de uma juventude mobilizada em torno da construção de um conhecimento local brasileiro?

Essa é a pergunta. Porque muito provavelmente, é o que eu discuto com o Cássio. Talvez se tivesse chegado a um decalque brasileiro. Um

decalque, mas brasileiro, e talvez o que esteja insinuado é que, qual o discurso com o qual contamos atualmente para desenvolver o nosso conhecimento em software e entre parênteses, “snif, snif” , porque software a gente ainda tem a sensação de que estamos produzindo, bem ou mal, e em hardware... outro dia vieram me falar, vamos perder outra chance, que é o software embarcado. Que depende da sinergia entre software e hardware.

Nesse momento você tinha uma comunidade produzindo hardware e software e tinha um discurso. De uma certa maneira lembra um pouco aquela música do Cazuza, ideologia eu quero uma para viver, mas eu acho que isso é o que está insinuado no artigo. Então, você tem o que é capaz uma comunidade mobilizada e uma comunidade de jovens engenheiros brasileiros, da qual eu, com muita honra, fiz parte, eu sei,..., por isso é que eu falei, vai lá ver como é que é, quer dizer,..., eu não gostaria que se concluísse que se está defendendo a prevalência do não técnico sobre o técnico. Eu prefiro, por isso ficamos ontem a noite discutindo o slide, a imagem do slide, aquele da coisa enquadrada ali. O que é que o sociotécnico? Esse é o esforço. Eu acho que a gente tem aí um futuro pela frente, para a gente ir afinando, discutindo, estamos só começando, e eu acho que os vocabulários nos ajudam muito, muitíssimo.

**[Marcus Vinícius, DBA:]**

*-Eu queria só fazer uma referência ao vocabulário, voltando a insistir como professor. O que acontece na verdade, eu concordo com o Henrique na questão do sociotécnico contra o técnico, e discordo que nós fizemos alguns tijolos, sim. Que ainda hoje a realidade da nossa Universidade, o aluno querer sair programando. E eu falo muito, você faz isso na SOS computadores. Eles têm o curso, não precisa vir para a Faculdade para aprender a programar. Basta saber ler, pegar o manual e aprender qualquer linguagem para programar. E o nosso aluno tem esse enfoque, de querer programar. Sem método nenhum, sem decalque nenhum, como a gente queria colocar.*

*Então, eu acho que o técnico nos ajudou a pelo menos dar uma formação melhor ao nosso pessoal. A saber, pelo menos, o que é que eles estão fazendo. Concordo que ainda falta muita coisa não técnica pra esse*

*produto ser de qualidade. E outra coisa que eu queria colocar em relação ao que o Henrique falou, é do nosso contexto Brasil. A gente teve durante muito tempo e ainda temos essa inércia do contágio do nosso pessoal, das nossas mentes no sentido de nós podermos nos libertar, a gente passou muito tempo sendo decalque do mundo aí fora. E acho que hoje a gente tem pessoas, a gente tem cabeças, temos Universidades, cursos, que têm condições de nos dar isso. E é hora da borboleta realmente fazer alguma coisa.*

**[José Marcos, BNDES:]**

Bom, eu sou Zé Marcos; sou aqui do BNDES, já fui da área de sistemas, já me aposentei e eu queria falar muito da questão da placa, do organograma da placa. Porque quando você começou a falar da COBRA, eu trabalhei uns quatro anos com o cobrinha lá e fiz o curso do sistema operacional e quando você estava falando da estrutura de equipe lá eu me lembrei da estrutura dos módulos do sistema operacional. Antes até de você falar do organograma da placa. Então, acho que isso é importante, porque a gente está amarrado a esses modelos, porque a gente traz isso do nosso dia a dia, da nossa formação, da nossa educação, de tudo.

Eu acho que os modelos hoje, a gente se sente meio preso por eles, porque eles prendem mesmo. A nossa criatividade está limitada aos modelos porque se vendeu muito que os modelos são melhores do que a gente e na COBRA, com certeza, naquela época você tinha gente, e gente motivada para fazer trabalho com ideal de trabalho mesmo. Eu acho que aqui no BNDES a gente tenta buscar isso, você é um que, graças a Deus está aqui dentro, a gente tem dificuldade porque a nossa cultura também está moldada em coisas que são antagônicas, mas eu acho que a questão básica é saber, na hora da flexibilidade,..., eu acho que a gente está numa fase em que a gente tem que buscar mais a flexibilidade. Porque os padrões você os tem cada vez mais poderosos. Agora a questão é saber qual o que eu preciso usar.

Será que eu preciso usar um padrão agora? Porque às vezes está na hora de eu fazer uma expansão e despadronizar algumas coisas; porque senão você não anda. E a gente morre de medo, depois que a gente teve um sucesso com um padrão, de sair do negócio. Eu, na época que trabalhava com



o COBRA, eu trabalhava na Marinha, a gente implantou uma metodologia de desenvolvimento pequena, no núcleo, que hoje desse núcleo ele acabou virando o núcleo de desenvolvimento. Mais de apoio, software ligado à finanças, ao contábil, operacional da Marinha. Mas a gente não teve, em nenhum momento, a pretensão de virar isso.

A gente estava focada muito no dia a dia. Acho que a grande armadilha é essa. Quando é que o tecnicismo tira a gente da realidade. Se ele tira a gente da realidade, a gente fica num mundo dos sonhos, do método pelo método, buscando a racionalidade sempre do método e às vezes você perde, assim, a banda passa e a sua oportunidade de trabalho foi embora. Então, acho que a questão básica é essa. Você tem que refletir na hora em que você está fazendo, o uso do método, se você está fazendo aquilo. Porque você precisa dele naquele momento, ou se você está fazendo aquilo porque você está amarrado a ele como uma ferramenta que você acha que vai usar depois. E essa é, tentar ajudar aí nessa reflexão, porque eu depois de trabalhar na Marinha, eu tentei reproduzir o modelo que tinha dado certo lá e nunca deu. Então, tentei reproduzir uma vez, depois eu desisti, eu falei eu tenho que olhar onde eu estou, o que é que tem de bom onde eu estou, e qual é o potencial dele crescer, se é que ele tem que crescer.

**[Henrique Cukierman, COPPE:]**

*-Você quer comentar, Cássio?*

**[Cássio, BNDES:]**

*-Amén*

**[Henrique Cukierman, COPPE:]**

*-Deixa eu fazer só dois comentários breves, porque o nosso tempo também já era... Ele fez um comentário ali, não está ali, eu me esqueci o nome dele, saiu com o celular, esse celular é um problema... Eu acho que a gente não pode esquecer só que a produção de conhecimento ela não é um fenômeno isolado e nem é um fenômeno exclusivamente acadêmico. Ela depende aí novamente do rizoma, do país, você precisa de política,..., a COBRA era a expressão de uma política industrial, por exemplo. A idéia da*

*reserva surge no momento em que se percebe que não bastava saber fazer uma geringonça.*

*As coisas tinham sido feitas assim, estou exagerando, é preciso entrar em escala industrial. Conceber, produzir, vender e dar manutenção. Coisa que, aliás, a COBRA era péssima, para dar manutenção, era um problema grave. Era um calcanhar de Aquiles da COBRA. Você precisa ter esse ciclo inteiro. Esse ciclo inteiro envolve política industrial, política de ciência e tecnologia, política universitária, de formação de recursos. Isso é uma primeira observação que eu acho importante. Ali, respondendo ao,..., não basta a formação, não é só a Universidade que forma, a formação é esse rizoma de política industriais e de ciência e tecnologia, entre outras coisas.*

*Agora eu queria responder o negócio do nosso aluno, da formação do nosso aluno. Eu sou Coordenador do processo de seleção do programa de Engenharia de Sistemas lá da COPPE. Recebi um currículo de um estudante chinês, querendo vir fazer pós-graduação na COPPE. Eu não sei o que é que um estudante chinês quer fazer de pós-graduação no Brasil, ou ele arrumou uma namorada brasileira, ou deve ser um tráfico qualquer de alguma coisa, vir fazer,..., (...) não é o primeiro chinês? Então, tem um trânsito de chineses, tem uma história de chineses, obrigado,...*

*E aí ele mandou o currículo eu distribuí lá o currículo, ninguém quis ficar com o chinês. Mas, enfim, eu me interessei muito pelo chinês, o currículo do chinês é realmente de você cair no chão. O cara é formado em Computer Sciences, a nossa ciência da computação, é um curso de,..., se você disser para fazer isso no Brasil, vão dizer que você está maluco, esse cara pirou, delirou no sociotécnico. Ele estuda do lado de Linguagem C, arquitetura de computadores, estrutura de dados, ele estuda introdução à filosofia zen budista, literatura chinesa, o pensamento econômico de Mão Tse Tung, é tudo junto, o curso é quase meio a meio. Quase meio a meio.*

*Aí você se pergunta, porque é que o chinês tem essa formação? Por que é que uma Universidade chinesa resolve formar o seu aluno; eu sei porque eu dou aula na Graduação, eu dou aula na Engenharia de Computação e Formação, eu dou aula em um curso de Computador e Sociedade. Ele passa quatro anos estudando um monte de coisas e tem um semestre de*

*Computador e Sociedade. Esse aluno é tecnicista por formação. Não tem saída, não tem saída. Você vai falar do rizoma, da rede, ele vai sair, ele não consegue ter uma manipulação mínima de símbolos, de vocabulários, de metáforas, pra entender o que é a modernidade. Agora o que é que acontece?*

*Não é à toa que a China está aí, a China está formando negociadores com a modernidade. E não uma mão de obra para ficar executando coisa. Essa é uma tentativa de responder a sua pergunta, não sei se os outros chineses vieram nessa mesma linha, mas eu acho que se eles vem nessa linha, vem mais para ensinar do que para aprender. Acho que eu fui muito além do tempo.*

**[Cássio, BNDES:]**

Mas até é interessante essa questão do Arquiteto, no artigo do Chicão, o Arquiteto estuda artes.

**[Arísio, A&S Consultoria:]**

*Ele falou, eu quis insinuar isso de manhã, quando eu falei porque é que não é Arquitetura de software e sim Engenharia de Software?*

Houve um movimento há um tempo atrás de criação desse profissional. Discutia que não se deveria fazer Engenharia, se deveria fazer arquitetura de software. De uma certa maneira isso descambou um pouquinho mais internamente, porque hoje se fala do Arquiteto de software, mas o arquiteto que lida com a estrutura do software em si e não é aquele cara que vai lidar talvez com as questões de usabilidade e coisas do gênero.

*Tem sempre também uma constante de comparar a Engenharia de Software de certa maneira com artes, há uma corrente que privilegia, que considera, não só XP, XP tem algumas diretrizes, mas em geral o processo de processos ágeis olha um pouquinho e começa a falar, acho que você devia estudar um pouco de literatura, estudar um pouco mais, porque nós temos muita proximidade com artes, estamos sempre fazendo alguma coisa que é muito nova. O nosso coeficiente de repetição de software, hoje, ainda é muito baixo.*

*Em algumas correntes do tipo Sistemas de Informação, você já tem um coeficiente de repetição muito alto, onde faz sentido você definir padrões, você definir processos, mais rigorosos para isso. Agora, software não é só isso, o escopo em que se desenvolve software, o tipo de software que se desenvolve, são muitos, muito maiores do que o escopo dos sistemas de informação, que por si já é bem grande. E nesses outros escopos você ainda vê muita arte e pouca Engenharia. E a própria Engenharia de Software não tem recursos desenvolvidos hoje para abordar todas essas diferentes dimensões que o software se aplica. Então, até por falta de instrumentos, você vai, lança mão de usar algum artesanato em diversas situações. Esse paralelo de software no horizonte que vai do extremo da Engenharia, para o extremo do artesanato, ainda é bastante discutido.*

Obrigado, eu acho que agora é a hora de passar o microfone adiante para o pessoal,...

***[Henrique Cukierman, COPPE:]***

Posso te interromper? A gente já está longe, posso te interromper, porque a gente atrasou um pouquinho aqui, passamos 5 minutos. Depois você comenta. Convidar o pessoal da Engenharia de Produção da UFMG, o Francisco, agora já me esqueci o sobrenome, e o André. (...) Chicão mesmo, então, dá no mesmo.