

Como Evoluem as Organizações de DDS em Ambientes de *Internal Offshoring*? Um Estudo Qualitativo Preliminarⁱ

Rafael Prikladnicki, Jorge Luis Nicolas Audy

Faculdade de Informática (FACIN)
Pontifícia Universidade Católica do Rio Grande do Sul (PUCRS)
90.619-900 – Porto Alegre – RS – Brasil

{rafaelp, audy}@pucrs.br

***Abstract.** Distributed software development has been evolved based on the selection of certain business models. Among the existing models, the offshore outsourcing is one of the best known. But in the last years, it has been observed an increasing on the creation of subsidiaries of companies located in a different country, to provide software development services, in a business model known as internal offshoring. In this context, the purpose of this paper is to present a set of attributes and their evolution in the internal offshoring environment, from the perspective of the subsidiaries. Preliminary results indicate an interesting evolution in the practices of these companies, which sometimes is not ideal, and can compromise the projects performance.*

***Resumo.** O desenvolvimento distribuído de software, na prática, tem sido operacionalizado através de modelos de negócios. Entre eles, o offshore outsourcing é um dos mais conhecidos e explorados em pesquisas. Nos últimos anos, tem sido comum a criação de subsidiárias de empresas, geralmente localizadas em outro país, para prover serviços de desenvolvimento de software, em um modelo de negócio chamado de internal offshoring. Desta forma, o objetivo deste artigo é apresentar, através um estudo de caso de base qualitativa, um conjunto de atributos e sua evolução em um contexto de internal offshoring, sob o ponto de vista das subsidiárias. Resultados preliminares indicam uma evolução muitas vezes de uma forma considerada não ideal, podendo comprometer o desempenho dos projetos.*

1. Introdução

Ao longo dos últimos dez anos, a Engenharia de Software (ES) tem presenciado uma evolução bastante significativa de uma área ainda recente: o Desenvolvimento Distribuído de Software (DDS). Diversas empresas têm distribuído seus processos de desenvolvimento de software, aproveitando-se do avanço da tecnologia de comunicação, da globalização, e da necessidade de se manter competitivo em um mercado cada vez mais globalizado. Com a idéia inicial de diminuição de custos, as operações de DDS acabam criando novos desafios para as equipes envolvidas em projetos desta natureza.

As organizações visam obter vantagens competitivas associadas a custo, qualidade e flexibilidade no desenvolvimento de software, buscando um aumento de produtividade, assim como diminuição de riscos (Sengupta et. al., 2006). Os desafios existentes neste novo contexto estão relacionados não apenas com questões técnicas, mas também questões estratégicas, culturais, e de gestão de conhecimento.

WDDS 2007

I Workshop de Desenvolvimento Distribuído de Software

Muitas vezes, a busca por estas vantagens faz com que as organizações encontrem soluções em outros países (*offshoring*). Os dois principais modelos de negócio em um contexto de *offshoring* são o *offshore outsourcing* (contatação de serviços de uma organização externa localizada em outro país) e o *internal offshoring* (criação de uma subsidiária da própria empresa para prover serviços de desenvolvimento de software). O *offshore outsourcing* é hoje um dos modelos mais conhecidos. Mas com o passar do tempo, e a evolução do DDS, as organizações começaram a perceber dificuldades em estabelecer uma relação com um parceiro estrangeiro, externo, e muitas vezes desconhecido em um primeiro momento. Assim, quando a estratégia de *offshoring* tinha uma visão mais de longo prazo, com a necessidade de um profundo conhecimento do negócio, as empresas passaram a investir na criação dos seus próprios centros de desenvolvimento em países tais como Brasil, China, Índia, Irlanda, Rússia, por exemplo.

Neste contexto, nos últimos três anos tem-se procurado observar a evolução da ES nas organizações de DDS, em um contexto de *internal offshoring*, e sob a ótica das subsidiárias provedoras de serviço (Evaristo et al, 2005). Dada a pouca literatura existente, inicialmente foi conduzido um estudo de caso em cinco unidades provedoras de serviços (internas e externas) de empresas multinacionais no Brasil e no Canadá para entender as dificuldades em quatro dos principais modelos de negócio de DDS (Prikladnicki & Audy, 2006; Prikladnicki et al, 2007). Foram identificadas dificuldades relacionadas com a Engenharia de Software, Qualidade de Software e Gerência de Projetos de Software. A partir dos resultados, foi possível comparar os dois modelos de *offshoring*, e identificar uma evolução inicial das provedoras de serviço em ambientes de *internal offshoring*. A identificação da evolução foi realizada analisando-se as duas empresas caracterizadas no modelo de *internal offshoring*. E é exatamente este o objetivo e a contribuição deste artigo, que está organizado em 5 seções. A seção 2 apresenta brevemente os conceitos de DDS. Na seção 3 descreve-se a metodologia de pesquisa. A evolução de *internal offshoring* é apresentada na seção 4. Na seção 5 destacam-se as considerações finais, e por fim as referências bibliográficas.

2. Desenvolvimento Distribuído de Software

O DDS tem se apresentado nos últimos anos como uma alternativa para o desenvolvimento de software. É um fenômeno que vem crescendo desde a última década (Carmel, 1999), onde se observou um grande investimento na conversão de mercados nacionais em mercados globais, criando novas formas de competição e colaboração (Carmel & Tija, 2005).

O DDS tem sido caracterizado pela colaboração e cooperação entre departamentos de organizações e pela criação de grupos de desenvolvedores que trabalham em conjunto, localizados em cidades, países e continentes diferentes (Carmel, 1999). Apesar de muitas vezes este processo ocorrer em um mesmo país, em regiões com incentivos fiscais ou de concentração de massa crítica em determinadas áreas, algumas empresas, visando maiores vantagens competitivas, buscam soluções globais, em outros países, o que potencializa os desafios existentes (Prikladnicki & Audy, 2006).

2.1. A Operacionalização do DDS nas Empresas

A operacionalização do DDS pode ocorrer através de diversos modelos de negócio (Prikladnicki et al, 2007). A decisão de estabelecer uma operação de DDS local (*onshore*) ou global (*offshore*), ou ainda de manter uma relação com uma empresa

WDDS 2007

I Workshop de Desenvolvimento Distribuído de Software

externa (*outsourcing*) ou interna (*insourcing*) depende dos objetivos de cada empresa com o DDS. Isto pode envolver redução de custos, duração das operações (curto ou longo prazo), o fato de estar perto do mercado local, ou ainda acelerar o tempo para um produto entrar no mercado, entre outros. Independente do modelo escolhido, todos possuem algum tipo de impacto nas atividades de Engenharia de Software. Segundo Herbsleb (2007), uma questão aberta neste sentido é a identificação de quais práticas são apropriadas para cada modelo de negócio escolhido.

Em Prikladnicki et al (2007), foi realizado um estudo onde foram comparados os principais desafios das empresas de DDS em quatro modelos de negócio principais, sobre a ótica da ES, Qualidade de Software e Gerência de Projetos. No modelo do tipo *internal offshoring*, uma subsidiária da própria empresa é criada para prover serviços de desenvolvimento (*insourcing*). Esta subsidiária está necessariamente localizada em um país diferente da matriz da empresa, ou empresa contratante (*offshore*). Para um melhor entendimento e comparação com outros modelos possíveis, a figura 1 apresenta de uma forma gráfica como a relação de DDS pode ser estabelecida entre o provedor de serviço e o cliente (independente desta relação ser local ou global).

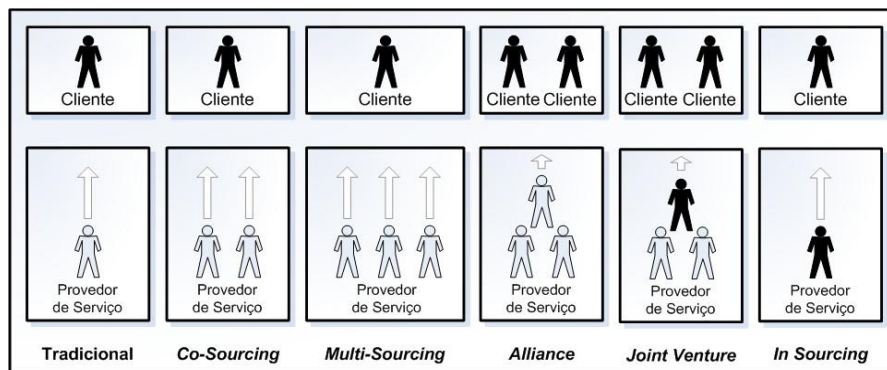


Figura 1. Operacionalização do DDS

É possível observar que a maioria das diferenças está no lado do provedor de serviço, foco do nosso estudo. Na primeira relação, um provedor de serviço entrega um projeto para um cliente. Quando dois provedores trabalham em conjunto, tem-se o *co-sourcing*. O *multi-sourcing* indica a existência de múltiplos provedores de serviço trabalhando para um cliente (o cliente gerencia e integra os serviços). A aliança envolve a colaboração de múltiplos provedores (um deles como responsável) trabalhando para um ou mais clientes. Na *joint venture*, provedores de serviço estabelecem um acordo, através da união de seus recursos, para executar projetos para clientes. Uma entidade independente é estabelecida, e é comum uma empresa *offshore* estabelecer parceria com uma empresa local, bem como o primeiro cliente ser parceiro na *joint venture*.

Finalmente, tem-se a relação de *insourcing*, objeto de estudo neste artigo e conceituada anteriormente. É importante diferenciar dois tipos de *insourcing* existentes na literatura e na prática. De acordo com Hyder et al (2006), esta relação é estabelecida quando um grupo dentro da própria organização-cliente é selecionado como provedor de serviço, mas é largamente gerenciado como uma entidade externa, competindo com outros provedores. Já Evaristo et al (2005) define que nesta relação o provedor de serviço é gerenciado como uma entidade exclusivamente interna, não competindo com provedores externos de serviço, visto que isto não faz parte da estratégia do cliente (que também controla o provedor interno). Neste artigo estudou-se o segundo tipo.

WDDS 2007

I Workshop de Desenvolvimento Distribuído de Software

2.2. Offshore Outsourcing x Internal Offshoring

Tanto *offshore outsourcing* quanto *internal offshoring* envolvem um grande esforço de colaboração entre os atores envolvidos (equipes de projeto, gerentes de projetos, clientes e gerência sênior). Mas existem algumas diferenças significativas entre eles. Do ponto de vista estratégico, o primeiro, por muitas vezes envolver a relação entre diversas empresas, com políticas diferentes, acaba gerando dificuldades tais como segurança de dados e perda de conhecimento organizacional que o segundo muitas vezes não tem. Além disso, no *internal offshoring*, como os colaboradores pertencem à mesma organização, possuem naturalmente uma maior facilidade de comunicação, além de uma maior flexibilidade para trabalhar com requisitos, por exemplo.

Esta maior facilidade do *internal offshoring* em alguns aspectos é compensada pelo alto investimento necessário para estabelecer um centro de desenvolvimento próprio. Além disso, ao mesmo tempo em que é possível encontrar na literatura de DDS diversas experiências relacionadas ao *offshore outsourcing*, ainda é pouca a quantidade de estudos que abordam o *internal offshoring*. Quando o fazem, as discussões geralmente ficam em um nível estratégico. Sendo assim, do ponto de vista de Engenharia de Software, a falta de estudos obriga as empresas a aprenderem com os seus próprios erros, evoluindo de uma forma que muitas vezes não é a ideal, mas que é traçada a partir das experiências mal sucedidas, e na base da “tentativa e erro”.

2.2. Base Teórica de Referência do Estudo

O estudo de caso desenvolvido envolveu a coleta de dados a partir de um guia para uma entrevista semi-estruturada. Inicialmente foi realizado um estudo teórico das principais variáveis e dimensões de DDS. Desta forma, foram identificadas variáveis organizadas em doze dimensões, a partir de um conjunto de autores (Tabela 1).

Tabela 1. Dimensões utilizadas na coleta de dados

#	Dimensões	Variáveis	Autores
1	Confiança	Espírito de equipe	Hsieh, 2006, Carmel, 1999
2	Níveis de dispersão	Localização das subsidiárias, localização dos <i>stakeholders</i>	Evaristo et al, 2004 Paasivara & Lassenius, 2003
3	Complexidade	Alocação de tarefas, gerência de projeto	Cusick & Prasad, 2006
4	Colaboração	Sincronização, ferramentas	Herbsleb, 2007
5	Alocação de projetos	Tomada de decisão, negociação, alocação de recursos, modelo de distribuição	Carmel & Tija, 2005, Evaristo et al, 2004
6	Metodologia de desenvolvimento	Processo de desenvolvimento, ciclo de vida do projeto	Sengupta et al, 2006 Komi-Sirviö, 2005
7	Medições	Produtividade, custom	Sangwan et al, 2007
8	Políticas e padrões	Qualidade de software, certificações, avaliações	Meyer, 2006
9	Tipos de Projetos	Desenvolvimento, suporte, melhoria, teste, manutenção preventiva, manutenção corretiva	Carmel & Tija, 2005 Evaristo et al, 2004
10	Distância percebida	Viagens, comunicação	Evaristo et al, 2004 Paasivara & Lassenius, 2003
11	Cultura	Cultura nacional, organizacional e individual	Carmel, 1999
12	Estrutura organizacional	Tipo e número de <i>stakeholders</i>	Damian, 2007, Berenbach, 2006

As entrevistas foram planejadas para explorar questões em cada dimensão, de forma a identificar possíveis atributos e sua evolução em *internal offshoring*.

3. Metodologia de Pesquisa

Um estudo de caso (Oates, 2006) de base qualitativa foi conduzido em duas subsidiárias de empresas multinacionais. A coleta de dados envolveu entrevistas e inspeção de documentação. Foram conduzidas quatro entrevistas em cada subsidiária, todas em

português, em um esforço de 8,5 horas de entrevistas. Também foram revisados planos de projetos, documentos de lições aprendidas e o processo de desenvolvimento de software de cada uma. As entrevistas foram conduzidas *in loco*, com profissionais de três níveis gerenciais. No nível estratégico, foram entrevistados diretores, com conhecimento para responder sobre a estratégia da subsidiária em DDS e a integração com outras subsidiárias da empresa. No nível tático foram entrevistados gerentes de desenvolvimento e gerentes de portfólio de projetos. No nível operacional foram entrevistados gerentes de projeto e líderes técnicos.

Para a análise de dados foram aplicados métodos de análise de conteúdo (Oates, 2006). Antes da análise propriamente dita, os dados foram preparados (transcrição das entrevistas e leitura inicial do texto – em torno de 120 páginas no total). Logo após, iniciou-se a codificação do texto gerado (identificação de categorias), em duas etapas: primeiro, foram identificados trechos das entrevistas que exploravam a evolução de determinados atributos. A segunda etapa envolveu uma leitura cuidadosa destes trechos, buscando mapear em detalhes cada atributo, sua evolução e o quanto a evolução daquela forma era considerada positiva pelas empresas estudadas.

4. A Evolução do DDS em *Internal Offshoring*

É importante destacar que os passos na evolução apresentada em cada atributo não são exclusivos deste modelo de negócio. Apesar de existirem características únicas que o diferencia (tais como a gerência interna do projeto na mesma empresa), existem intersecções com os outros modelos, visto que algumas atividades da ES acabam não tendo diferença prática independente do relacionamento entre as empresas. Desta forma, na maioria dos atributos a evolução acaba iniciando de forma semelhante, com uma diferença significativa a partir dos níveis mais altos. Isto se deve ao fato de que no *internal offshoring* existe uma maior facilidade de padronização em escala global (entre as unidades distribuídas) com o passar do tempo. Assim, foram identificados vinte e dois atributos em quatro categorias, bem como a respectiva evolução. A partir da tabela 2, apresentada na próxima página, alguns resultados interessantes podem ser observados. Inicialmente, a evolução identificada foi baseada na análise tanto de projetos maduros quanto imaturos. Além disso, por terem sido entrevistados profissionais de diferentes níveis de decisão, foi possível traçar a evolução de atributos que afetam diretamente os indivíduos (pessoas), os projetos, o conjunto de projetos (portfólio), e a subsidiária como um todo.

Para cada atributo, ainda identificou-se de uma forma preliminar se aquela era a evolução considerada ideal pela subsidiária, ou seja, se houvesse a oportunidade de iniciar novamente a operação de DDS, os passos na evolução seriam repetidos. Os dados encontrados mostram que alguns atributos (cor cinza) poderiam evoluir de forma diferente em relação a que foi identificada, e isto será explorado em estudos futuros.

Também é possível relacionar algumas evoluções com práticas de algumas áreas de processo de modelos de qualidade de software tais como o CMMI e o MR MPS. Neste caso, percebeu-se que nos níveis iniciais a evolução em alguns atributos acaba sendo orientada por estes modelos, e na medida em que a evolução se torna global, é preciso o suporte de modelos que consideram a relação entre subsidiárias e com a matriz, tais como o eSCM-SP (Hyder et al, 2006). A diferença é que este atua mais no relacionamento entre cliente e provedor de serviço (além de ser recomendado quando existe uma grande independência os dois), e não tanto no contexto da ES em si.

WDDS 2007
I Workshop de Desenvolvimento Distribuído de Software

Tabela 2. Os atributos identificados e a sua evolução

Atributo	Evolução identificada	Categoria	Avaliação
Diferenças culturais	<ol style="list-style-type: none"> 1. Não existe diferença cultural entre as equipes de DDS 2. As pessoas entendem os impactos das diferenças culturais e os compartilham 3. As pessoas são treinadas em como lidar diferenças culturais de forma independente em cada subsidiária 4. Iniciativas globais são desenvolvidas para lidar com as diferenças culturais 	Pessoas	Repetiria a evolução
Aquisição de confiança	<ol style="list-style-type: none"> 1. Não existe confiança 2. As pessoas entendem a necessidade da confiança nas equipes distribuídas 3. As pessoas são treinadas em aquisição de confiança de forma independente em cada subsidiária 4. Iniciativas globais são desenvolvidas para lidar com as diferenças culturais 	Pessoas	Repetiria a evolução
<i>Awareness</i> de atividades (quem está fazendo o que), processo (quem faz quando), e disponibilidade (que está disponível quando)	<ol style="list-style-type: none"> 1. Existe mecanismo de <i>awareness</i> independente em cada subsidiária, e é reativo (estimulado por falta de informação, padrões, e conflitos) 2. Existe mecanismo de <i>awareness</i> independente em cada subsidiária, e é pró-ativo (estimulado por planejamento prévio) 3. Existe infra-estrutura tecnológica de suporte à <i>awareness</i> de forma independente em cada subsidiária 4. Existe uma infra-estrutura tecnológica padrão global de suporte à <i>awareness</i> 	Pessoas	Evolução deveria ser diferente
Gestão de conhecimento	<ol style="list-style-type: none"> 1. A gestão de conhecimento depende de iniciativas individuais 2. A gestão de conhecimento depende de iniciativas das equipes 3. Existem iniciativas independente em cada subsidiária, com sistemas locais para gestão de conhecimento 4. Existe uma infra-estrutura tecnológica global para a gestão de conhecimento 	Pessoas	Evolução deveria ser diferente
Treinamento	<ol style="list-style-type: none"> 1. Os treinamentos são focados em conhecimento técnico, sob demanda 2. Os treinamentos são balanceados entre técnicos e não-técnicos, sob demanda 3. As subsidiárias criam programas de treinamentos, técnicos e não técnicos 4. Existem iniciativas globais para treinamento técnico e não-técnico 	Pessoas	Evolução deveria ser diferente
Habilidades globais e de DDS	<ol style="list-style-type: none"> 1. As pessoas não estão preparadas para trabalhar em ambientes globais 2. As pessoas são treinadas sob demanda 3. As subsidiárias possuem programas de treinamentos obrigatórios em habilidades de DDS 4. Os programas de treinamentos obrigatórios evoluem em escala global 	Pessoas	Evolução deveria ser diferente
Gerência de projeto	<ol style="list-style-type: none"> 1. Existe um gerente de projeto com autoridade local, na subsidiária responsável, para gerenciar o projeto 2. Gerentes de projetos são replicados em todos os locais distribuídos onde existem integrantes da equipe do projeto 3. Gerentes de projetos são replicados em locais distribuídos considerados críticos para a gestão do projeto 4. Existe um único gerente global para o projeto, com autoridade sobre toda a equipe 	Projeto	Repetiria a evolução
Fases do ciclo de vida de desenvolvimento de software	<ol style="list-style-type: none"> 1. A subsidiária executa atividades de codificação 2. A subsidiária executa atividades de codificação e teste 3. A subsidiária executa atividades de especificação de requisitos, codificação e teste 4. A subsidiária executa todas as atividades do ciclo de desenvolvimento de software 	Projeto	Repetiria a evolução
Especificação de requisitos	<ol style="list-style-type: none"> 1. A especificação de requisitos é executada sempre de forma distribuída 2. A especificação de requisitos é executada de forma distribuída ou presencial, dependendo do projeto 3. As subsidiárias criam técnicas para facilitar a especificação de requisitos 4. Existem iniciativas globais para a difusão de técnicas padrão para especificação de requisitos 	Projeto	Repetiria a evolução
Gerência de configuração	<ol style="list-style-type: none"> 1. Não existe infra-estrutura de gerência de configuração 2. Existe uma infra-estrutura local de gerência de configuração em cada subsidiária 3. Existe uma infra-estrutura global de gerência de configuração, que não está integrada entre todas as subsidiárias 4. Existe uma infra-estrutura global e integrada de gerência de configuração 	Projeto	Evolução deveria ser diferente

WDDS 2007

I Workshop de Desenvolvimento Distribuído de Software

			Projeto	Evolução deveria ser diferente
Ferramentas para comunicação	<ol style="list-style-type: none"> São utilizadas ferramentas assíncronas (correio eletrônico), e pobres em contexto (correio eletrônico) São utilizadas ferramentas síncronas (mensagens instantâneas, vídeo conferência), e pobres em contexto (MSN, ICQ) São utilizadas ferramentas síncronas e ricas em contexto (vídeo conferência) 		Projeto	Evolução deveria ser diferente
Ferramentas para colaboração	<ol style="list-style-type: none"> Não existem ferramentas de suporte à ES colaborativa Existem algumas ferramentas de suporte, mas são iniciativas individuais e sob demanda em alguns projetos As equipes utilizam ferramentas de mensagens instantâneas como foram de colaborar As subsidiárias implantam ferramentas colaborativas para o suporte às atividades de ES 		Projeto	Repetiria a evolução
Infra-estrutura de suporte ao DDS	<ol style="list-style-type: none"> A infra-estrutura para DDS é pobre e planejada sob demanda ou a partir de problemas enfrentados A infra-estrutura para DDS é particular de cada subsidiária A infra-estrutura para DDS é padrão e global 		Projeto	Evolução deveria ser diferente
Gerência de risco	<ol style="list-style-type: none"> Não existe gerência de risco nos projetos A gerência de risco existe, mas é particular de cada subsidiária Existe um processo padrão para gerência de risco nos projetos, compartilhado pelas subsidiárias A gerência de risco ocorre tanto em nível de projeto quanto em níveis estratégicos (decisões de desenvolver um projeto de DDS e locais) 		Projeto	Evolução deveria ser diferente
Atividades do projeto	<ol style="list-style-type: none"> As atividades são alocadas para as subsidiárias, sem dependência e interação entre as equipes distribuídas As atividades são alocadas para as subsidiárias, com dependência e alguma interação As subsidiárias trabalham de forma paralela, com bastante interação A empresa implementa o desenvolvimento de software 24 horas no formato <i>follow the sun</i> 		Projeto	Repetiria a evolução
Tipos de projetos	<ol style="list-style-type: none"> Os projetos são apenas do tipo "correção de erros", sem necessidade de conhecimento do negócio Os projetos são também do tipo "novos projetos", com a necessidade de um conhecimento mínimo do negócio Os projetos são também do tipo "manutenção" e "melhorias", com grande necessidade de conhecimento do negócio Os projetos são também do tipo "teste de aplicações desenvolvidas por outras subsidiárias" 		Portfólio	Repetiria a evolução
Alocação de projetos	<ol style="list-style-type: none"> Não existe um processo formal de alocação de projetos para as subsidiárias Existe um processo informal de alocação de projetos, baseado em opinião pessoal Existe um processo formal para alocação de projetos em cada subsidiária Existe um processo global e padrão para alocação de projetos baseado na situação e histórico de cada subsidiária 		Portfólio	Evolução deveria ser diferente
Escritório de gerenciamento de projetos (PMO)	<ol style="list-style-type: none"> Não existe PMO Cada subsidiária cria o seu próprio PMO Existe um PMO global, não integrado entre todas as subsidiárias e a matriz Existe um PMO global e integrado 		Portfólio	Repetiria a evolução
Relação com a matriz	<ol style="list-style-type: none"> A subsidiária precisa mostrar que é capaz de satisfazer as expectativas da matriz A subsidiária precisa mostrar que é capaz de trabalhar com diferentes tecnologias A subsidiária precisa mostrar que é capaz de trabalhar para diferentes áreas de negócio e portfólios A subsidiária precisa mostrar que é capaz de trabalhar em parceria com outras subsidiárias 		Organização (subsidiária)	Repetiria a evolução
Relação com os clientes	<ol style="list-style-type: none"> A equipe técnica é replicada na matriz para fazer a interface com o cliente A gerência do projeto é replicada na matriz para fazer a interface com o cliente Os analistas de negócio são replicados na matriz para fazer a interface com o cliente A subsidiária faz o contato direto com o cliente 		Organização (subsidiária)	Repetiria a evolução
Melhoria de processo	<ol style="list-style-type: none"> São criados processos diferentes para cada equipe ou portfólio de projetos São criados processos para cada subsidiária, com a criação de um grupo responsável em cada local (SEPG local) São criados processos padrão e globais, com a criação de um grupo global responsável pelos processos (SEPG global) 		Organização (subsidiária)	Evolução deveria ser diferente
Estrutura organizacional	<ol style="list-style-type: none"> As equipes trabalham de forma integrada entre as subsidiárias, mas papéis e responsabilidades não são padrão As equipes trabalham de forma integrada entre as subsidiárias, e existe um padrão na nomenclatura de papéis e responsabilidades Existe uma única equipe em cada projeto, composta por profissionais de diversas subsidiárias 		Organização (subsidiária)	Evolução deveria ser diferente

5. Considerações finais

Neste artigo apresentou-se uma análise preliminar da evolução de organizações de DDS em *internal offshoring*, em um contexto de ES, e sob a ótica das subsidiárias. Além das limitações típicas de estudos qualitativos, os dados apresentados representam uma fotografia das empresas analisadas, independente de ser a melhor evolução para cada atributo. Como continuidade do estudo, está prevista a aplicação de uma *survey* em três empresas com dois objetivos: confirmar os dados encontrados (tanto os atributos quanto a evolução), identificando também atributos adicionais; e identificar se a evolução é considerada adequada pelas empresas, ou se os motivos pelos quais as empresas evoluem em certos atributos acabam sendo determinados pela pressão e complexidade do ambiente de DDS. Desta forma, será possível mapear de uma forma consistente quais atributos evoluem, como e por que evoluem, e se a evolução deveria ser diferente.

Referências Bibliográficas

- Berenbach, B. (2006). "Impact of Organizational Structure on Distributed Requirements Engineering Processes: Lessons Learned", In: Workshop on GSD at ICSE, Xangai, China.
- Carmel, E., Tija, P. (2005). "Offshoring Information Technology: Sourcing and Outsourcing to a Global Workforce", UK: Cambridge.
- Carmel, E. (1999). "Global Software Teams – Collaborating Across Borders and Time-Zones",.
- Cusick, J., Prasad, A. (2006). "A Practical Management and Engineering Approach to Offshore Collaboration", IEEE Software, 23(5), pp. 20-29.
- Damian, D. (2007). Stakeholders in Global Requirements Engineering: Lessons Learned from Practice", IEEE Software, 24(2), pp. 21-27.
- Evaristo, R., Audy, J. L. N., Prikladnicki, R., Avritchir, J. (2005). "Wholly Owned Offshor Subsidiaries for IT Development: A Program of Research". In: HICSS, pp. 258b, Havaí.
- Evaristo, R., Scudder, R., Desouza, K., Sato, O. (2004). "A Dimensional Analysis of Geographically Distributed Project Teams: A Case Study", JET-M Journal. 21(3), pp. 75-89.
- Herbsleb, J. D. (2007). "Global Software Engineering: The Future of Socio-technical Coordination", In: 29th ICSE, Minneapolis.
- Hsieh, Y. (2006). "Culture and Shared Understanding in Distributed Requirements Engineering", In: ICGSE, pp. 101-105, Florianópolis, Brasil.
- Hyder, E. B., Heston, K. M., Paulk, M. C. (2006). "The sCM-SP v2.01: The eSourcing Capability Model for Service Providers (eSCM-SP) v2.01 – Model Overview". CMU.
- Komi-Sirviö, S., Tihinen, M. (2005). "Lessons Learned by Participants of Distributed Software Development", Knowledge and Process Management, 12(2), pp. 108-122.
- Meyer, B. (2006) "The Unspoken Revolution in Software Engineering", IEEE Computer, 39(1).
- Oates, B. J. (2006). "Researching Information Systems and Computing", Sage Publications.
- Paasivara, M., Lassenius, C. (2003) "Communication needs, practices and supporting structures in global inter-organizational software development projects", SPIP, 8(4), pp. 183-199.
- Prikladnicki, R., Audy, J. L., Damian, D., Oliveira, T. (2007). "DSD: Practices and challenges in different business strategies of offshoring and onshoring", In: ICGSE, To appear.
- Prikladnicki, R., Audy, J. L. N. (2006). "Uma Análise Comparativa de Práticas de Desenvolvimento Distribuído de Software no Brasil e no exterior", In: XX SBES.
- Sangwan R., Bass, M., Mullick, N., Paulish, D. J., Kazmeier, J. (2007). "Global Software Development Handbook", Auerbach Publications.

ⁱ Estudo realizado pelo grupo de pesquisa em Desenvolvimento Distribuído de Software, do PDTI, financiado pela Dell Computadores do Brasil Ltda., com recursos da Lei Federal Brasileira nº 8.248/91.