

## **Uma Análise Comparativa de práticas de Desenvolvimento Distribuído de Software no Brasil e no exterior<sup>1</sup>**

**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

rafael@inf.pucrs.br, audy@pucrs.br

***Abstract.** More than a decade ago, seeking lower costs and access to skilled resources, many organizations began to experiment with remotely located software development facilities. The purpose of this paper is to present the results of case studies conducted in the last four years with four companies located in Brazil and other countries, where those companies have some distributed development strategies. The analysis is presented according to the strategies, practices and challenges identified. The results are analyzed under the Software Engineering, Software Quality, and Software Project Management point of view.*

***Resumo.** Nas últimas décadas, grandes investimentos têm permitido um movimento de transformação de um mercado local para mercados globais, em um processo que têm criado novas formas de colaboração e competição na área de Engenharia de Software, entre elas o Desenvolvimento Distribuído de Software. O objetivo deste artigo é apresentar os resultados de estudos de caso desenvolvidos nos últimos quatro anos com quatro empresas presentes neste cenário, localizadas no Brasil e no exterior. Apresenta-se a análise, levando-se em consideração as estratégias e práticas adotadas, e os desafios identificados. Os resultados são analisados do ponto de vista da Engenharia de Software, da Qualidade de Software e da Gerência de Projetos de Software.*

### **1. Introdução**

Nas últimas décadas, grandes investimentos têm permitido um movimento de transformação de um mercado local para mercados globais, em um processo que têm criado novas formas de colaboração e competição (Herbsleb et. al., 2001) na área de Engenharia de Software (ES). Neste período, o mercado de desenvolvimento global de software vivenciou grandes crises: muitas falhas em projetos, e o aumento considerável da demanda, em um período onde não havia recursos com as habilidades necessárias para um bom desenvolvimento dos projetos. Assim, o Desenvolvimento Distribuído de Software (DDS) surgiu como uma oportunidade para minimizar as dificuldades encontradas com o crescimento do mercado global. Alguns fatores contribuíram para acelerar este cenário. Entre eles, podemos citar (Carmel, 1999):

- A necessidade de recursos globais para serem utilizados a qualquer hora;

- As vantagens de estar perto do mercado local, incluindo o conhecimento dos clientes e as condições locais para explorar as oportunidades de mercado;
- A grande pressão para o desenvolvimento *time-to-market*, utilizando as vantagens proporcionadas pelo fuso horário diferente, no desenvolvimento conhecido como *follow-the-sun* (24 horas contínuas, contando com as equipes fisicamente distantes).

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). Muitas vezes, a busca por estas vantagens faz com que as organizações encontrem soluções em outros países (*offshore sourcing* ou *offshoring*). E os desafios existentes neste novo contexto estão geralmente relacionados não apenas com questões técnicas, mas também à questões estratégicas, culturais, e de gestão de conhecimento (Herbsleb et. al., 2001).

Segundo a IDC - *International Data Group* (2006), em média, pode se ter uma economia entre 25% a 50% em termos de custo, quando grandes projetos são transferidos para operações *offshore*. Mas custo não é o único benefício que torna o desenvolvimento distribuído atrativo. Especificamente em desenvolvimento *offshore*, outros benefícios tais como profissionais habilitados para trabalhar em um idioma diferente (geralmente a língua inglesa), baixas taxas de *turnover*, e o incentivo de governos locais contribuem para o crescimento desta nova forma de desenvolver software. Além disso, a maioria das organizações que tiveram algum tipo de experiência em *offshoring* reportou também benefícios em melhoria de processo. Desta forma, se for bem planejado, o desenvolvimento distribuído, seja inter ou intra-organizacional, local ou global, pode trazer benefícios consideráveis para as atividades envolvidas no desenvolvimento de software como um todo.

Por este motivo, em termos de estratégia, o *offshore outsourcing* é hoje uma das mais conhecidas para operacionalizar o DDS. Mas ao longo dos últimos anos, novas estratégias surgiram, devido a variedade de possibilidades existentes. Desta forma, o objetivo deste artigo é apresentar os resultados de estudos de caso desenvolvidos nos últimos quatro anos com quatro empresas que possuem alguma estratégia de DDS, analisando as práticas adotadas e as dificuldades encontradas. O principal método de pesquisa utilizado é o estudo de caso e a base empírica da pesquisa envolveu empresas de grande porte e suas unidades de desenvolvimento de software localizadas no Brasil e no exterior. Os resultados são apresentados e analisados do ponto de vista de qualidade de software (com foco no processo de desenvolvimento), engenharia de software (técnicas e ferramentas para o desenvolvimento), e gerência de projetos de software (estratégias de gestão dos projetos distribuídos).

Este artigo está estruturado da seguinte forma: a seção 2 apresenta o referencial teórico de desenvolvimento distribuído de software e os conceitos envolvidos em DDS. A seção 3 apresenta a descrição do estudo de caso realizado, a caracterização das empresas e os resultados encontrados e analisados. A seção 4 apresenta as conclusões, limitações do estudo e pesquisas futuras.

## **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, onde se observou um grande investimento na conversão de mercados nacionais em mercados globais, criando novas formas de competição e colaboração (Herbsleb et. al., 2001). Quando a distância física entre os atores em um ambiente de DDS envolve mais de um país, Carmel (1999) define uma instância do DDS chamada de desenvolvimento global de software (*Global Software Development*<sup>1</sup> – GSD).

Neste sentido, 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 ou países diferentes. 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 problemas e os desafios existentes. Segundo Carmel (1999), não existem dúvidas para qualquer profissional que trabalha na área de ES que tanto o desenvolvimento de software tradicional quanto o distribuído possuem diversas dificuldades. As principais características que os diferenciam são: **dispersão geográfica** (distância física); **dispersão temporal** (diferenças de fuso-horário); e **diferenças culturais** (idioma, tradições, costumes, normas e comportamento).

Estas diferenças refletem-se em diversos fatores. De acordo com Herbsleb et. al. (2001), destacam-se **questões estratégicas** (decisão de desenvolver ou não um projeto de forma distribuída, tendo por base análises de risco e custo-benefício); **questões culturais** (valores, princípios, etc., entre as equipes distribuídas); **questões técnicas** (fatores relativos à infra-estrutura tecnológica e ao conhecimento técnico necessário para o desenvolvimento dos projetos distribuídos, tais como redes de comunicação de dados, plataformas de hardware, ambiente de software, processo de desenvolvimento, etc.); e **questões de gestão do conhecimento** (fatores relativos à criação, armazenamento, processamento e compartilhamento de informações nos projetos distribuídos).

### **2.1. A operacionalização do DDS nas empresas**

Nos últimos anos, a globalização tem proporcionado uma oportunidade para que empresas executem projetos de desenvolvimento de software utilizando equipes geograficamente distribuídas. À medida que as equipes passaram a se organizar desta forma, as empresas começaram a estudar meios para obter um melhor desempenho neste novo cenário. Desta forma, trabalhar em um projeto de DDS pode variar desde a simples distribuição em um mesmo país, ou então em uma distribuição em países ou continentes diferentes, criando assim o desenvolvimento global de software (Carmel, 1999).

Com isso, a pesquisa em DDS tem se tornado bastante comum nos últimos anos. Por este motivo, e devido à diversidade de possibilidades surgidas, existem muitos conceitos envolvidos. Para atuar em um ambiente de DDS, existem algumas características que precisam ser definidas. Entre elas, está o relacionamento das

---

<sup>1</sup> A maioria dos termos utilizados em DDS foi mantida em inglês, pois a literatura existente utiliza os termos tal qual está sendo utilizado neste artigo, e a tradução em alguns casos compromete o uso do termo.

empresas que participam de uma operação como esta, e a caracterização da equipe, em termos de localização geográfica. A união destas duas dimensões cria possibilidades para configurar um cenário de DDS, através de modelos de negócio específicos. Estes modelos variam de acordo com alguns critérios, e apresentam características diferentes em relação às atividades ligadas ao desenvolvimento de software. Desta forma, para melhor entender os desafios, vantagens e limitações dos tipos de DDS, apresenta-se a seguir uma padronização dos conceitos existentes, explorados através do relacionamento entre as dimensões citadas anteriormente. Considerando a relação entre as empresas, esta tem ocorrido através de três formas principais, segundo Robinson et. al. (2004):

- *terceirização (outsourcing)* ou comprar: neste cenário, uma empresa delega o controle sobre uma ou mais atividades para uma empresa externa à quem contratou o serviço. Geralmente é uma análise que envolve a decisão de *make-or-buy*, ou seja, desenvolver na própria empresa ou então delegar para um provedor externo de serviços. Tipicamente, segundo o autor, é uma das formas mais simples de ser implementada e a mais rápida de ser operacionalizada;
- *join-venture* ou colaboração: neste cenário, existe um acordo entre duas ou mais empresas onde, através da união de seus recursos, é criada uma nova entidade para executar um ou mais projetos por um certo período de tempo. Existe um controle nivelado sobre o projeto e sobre os recursos, impactando na redução de custos de todas as empresas envolvidas na *joint venture*;
- departamentos/subsidiárias da própria empresa (*wholly-owned subsidiaries* ou *insourcing*) ou desenvolver: neste último cenário, as empresas criam os seus próprios centros de desenvolvimento de software. Entre os motivos para que isto ocorra está um maior controle, maior flexibilidade, menores preços a longo prazo, além de manter a cultura interna da empresa. Segundo o autor, é o modelo que demanda uma maior complexidade de operacionalização e mais tempo.

Do ponto de vista de distância geográfica, a distribuição ocorre de duas formas:

- *offshore*, ou seja, em um país diferente da matriz da empresa contratante e do cliente. O desenvolvimento *offshore* pode ser realizado em algum centro de desenvolvimento da empresa contratada, ou através da contratação de serviços de uma empresa terceira, localizada em algum outro país. Esta forma de distribuição geográfica é recomendada para projetos que possuem um plano de projeto bem definido e um bom entendimento dos requisitos dos clientes;
- *onshore*, ou seja, no mesmo país onde estão localizados o cliente e a matriz da empresa. Neste caso, podem ocorrer duas situações: a primeira, onde o projeto é desenvolvido em um centro de desenvolvimento da empresa contratada, no mesmo país onde está o cliente, mas distante fisicamente do cliente (*offsite*). E a segunda, onde o desenvolvimento é realizado fisicamente no cliente (*onsite*). Neste caso, a empresa contratada utiliza seus recursos para suportar as atividades fisicamente no cliente (desde especificação de requisitos até implementação e testes). Enquanto que na primeira a vantagem é manter o desenvolvimento perto do cliente, a segunda garante ao cliente um maior controle, e é recomendada para projetos de missão crítica, que são sensíveis ao local onde são desenvolvidos, ou que necessitam uma constante atenção do cliente.

Buscando uma visão unificada das formas de relacionamento entre as empresas e a distribuição geográfica, a figura 1 traz um modelo adaptado de Robinson et. al. (2004).

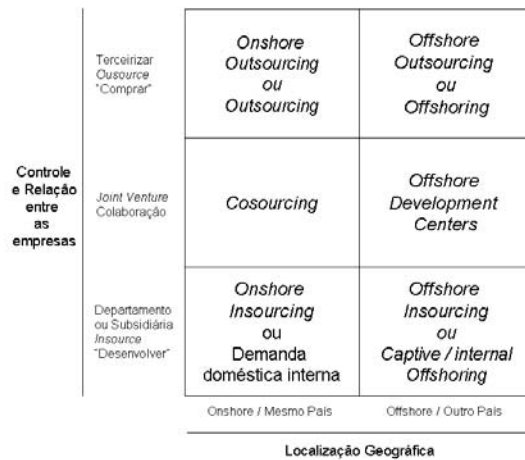


Figura 1. Modelos de Negócio para DDS, adaptado de Robinson et. al. (2004)

Como esta pesquisa teve seu foco direcionado para as relações de terceirização (*outsourcing* ou comprar) e de *insourcing* (desenvolver), as combinações relativas à *joint ventures* foram desconsideradas neste estudo, gerando assim um modelo simplificado, adaptado de Robinson et. al. (2004), representado na figura 2.

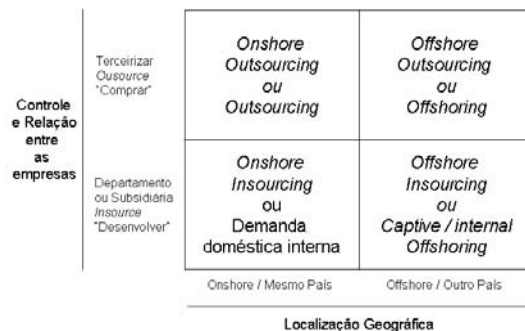


Figura 2. Modelos de Negócio estudados, adaptado de Robinson et. al. (2004)

Em relação aos modelos apresentados na figura 2, foco deste estudo, eles são assim definidos (Robinson et. al., 2004; Carmel et. al., 2005):

- *onshore insourcing* ou demanda doméstica interna: este é um dos mais simples e conhecidos pelos gerentes de projetos. Existe um departamento na própria empresa, ou uma subsidiária no mesmo país (*onshore*), que provê serviço de desenvolvimento de software, através de projetos internos (*insourcing*);
- *onshore outsourcing* ou *outsourcing*: este modelo de negócio indica a contratação de uma empresa terceira (*outsourcing*) para o desenvolvimento de determinados serviços ou produtos de software para uma empresa. A empresa terceira está localizada no mesmo país da empresa contratante (*onshore*);
- *offshore outsourcing* ou *offshoring*: este modelo de negócio indica contratação de uma empresa terceira (*outsourcing*) para o desenvolvimento de determinados serviços ou produtos de software, sendo que a empresa terceirizada está necessariamente localizada em um país diferente da contratante (*offshore*);

- *offshore insourcing* ou *captive / internal offshoring*: este último modelo de negócio indica criação de uma subsidiária da própria empresa para prover serviços de desenvolvimento de software (*insourcing*). Esta subsidiária está necessariamente localizada em um país diferente da matriz da empresa, ou empresa contratante (*offshore*).

## **2.2. Unificação dos conceitos envolvidos na operacionalização do DDS**

O desenvolvimento *offshore* tem sido considerado uma das formas mais comuns para caracterizar o DDS. Segundo Schniederjans et. al. (2005), é também conhecido como *international sourcing* e não é algo novo, pois existe em áreas tais como manufatura há muito tempo. Mas ao longo dos últimos anos, percebeu-se um crescimento bastante significativo e acelerado em um âmbito global, do setor de serviços, tais como serviços de TI, e conseqüentemente o desenvolvimento de software (Bohem, 2006).

Isto faz com que as organizações se interessem por buscar soluções externas em outros países diferentes de onde está a matriz da empresa (*offshore sourcing* ou simplesmente *offshoring*), ou então buscar por soluções no mesmo país (terceirização ou *outsourcing*). Assim, o *international sourcing* também é conhecido e referenciado como *offshoring*. De acordo com Carmel et. al. (2005), *offshoring* é definido como a mudança de execução de um processo de negócio de uma empresa, que anteriormente era realizado no país onde está localizada a sua matriz (*onshore*), para um país estrangeiro, visando vantagens principalmente em relação ao baixo custo em um outro país. O *offshoring* de um processo de negócio independe de o mesmo continuar sendo executado pela mesma empresa (*insourcing*) ou por uma empresa terceira (*outsourcing*).

Freqüentemente, *offshoring* é confundido com *outsourcing*, terceirização, e *offshore outsourcing*. Além disso, estes quatro conceitos têm sido utilizados quase que com o mesmo significado, sem levar em consideração algumas importantes diferenças técnicas. Segundo Carmel et. al. (2005), *outsourcing*, em um contexto corporativo, representa a prática organizacional que envolve transferir uma função organizacional para um terceiro. Segundo Schniederjans et. al. (2005) e Morstead et. al. (2003), *outsourcing* não é um conceito novo, mas uma nova forma de se referenciar à terceirização de atividades e serviços para uma entidade externa especializada. E quando o terceiro está localizado em um outro país, configura o *offshore outsourcing*. Já o *offshoring* representa a transferência de uma função organizacional para um outro país, independente de ser executada dentro ou fora da organização.

Devido à quantidade de termos envolvidos neste cenário de DDS, muitas vezes um conceito é utilizado de forma equivocada. Um dos principais equívocos é pensar que todo o *offshoring* envolve *outsourcing*, ou que o *offshoring* é o *outsourcing* em um contexto global. Mas isto não é verdade. Enquanto *outsourcing* envolve terceirizar processos para uma entidade externa, *offshore* pode envolver tanto a terceirização quanto o desenvolvimento *in-house*, mas em outro país. Por isso, a definição de *offshoring* também inclui empresas que montam seus próprios centros de desenvolvimento, conhecidos como *dedicated captive centers*, em locais de baixo custo.

Em suma, *offshoring* ou *offshore sourcing* é similar ao *offshore outsourcing* quando as empresas contratam empresas terceiras em outros países, mas é diferente quando as empresas simplesmente transferem atividades para a mesma empresa em um

outro país, criando assim os centros de desenvolvimento dedicados (*dedicated captive centers*), conhecidos também como *wholly-owned subsidiary*. Desta forma, o *offshore insourcing* surge como uma opção onde existe a transferência de atividades ou operações para uma entidade interna. Geralmente é uma decisão realizada para manter o controle de certas competências consideradas críticas na empresa. Então, as empresas podem ter a terceirização ou o *outsourcing* sem que para isto elas precisem de uma estratégia *offshore*. Da mesma forma, é possível ter a estratégia *offshore* sem caracterizar a terceirização ou o *outsourcing*. Todas estas possibilidades têm seu reflexo nas atividades que compõem o desenvolvimento de software. Para cada cenário, novos desafios são identificados e novas estratégias para obter um melhor aproveitamento das técnicas de desenvolvimento de software são colocadas em prática.

### **3. Estudos de Caso: Análise de Práticas de DDS no Brasil e no Exterior**

Nos últimos anos, diversas empresas estabeleceram estratégias de desenvolvimento distribuído de software no Brasil e no exterior. Alguns estudos realizados com estas empresas em separado (Sengupta et. al., 2006, Pilatti et. al., 2006; Lopes et. al., 2004; Prikladnicki et. al., 2006; Prikladnicki et. al., 2004; Prikladnicki et. al., 2003) constataram a existência de importantes desafios no processo de desenvolvimento. Mas nenhum destes estudos analisou em detalhes os desafios comparando-se os tipos de DDS implantados pelas empresas. Desta forma, este artigo procurou reunir dados de quatro empresas que possuem estratégias de DDS, procurando identificar desafios e estratégias para obter um melhor desempenho em projetos distribuídos. A análise foi feita sob a ótica da Engenharia de Software, da Qualidade de Software e da Gerência de Projetos de Software. O estudo e o método de pesquisa são detalhados a seguir.

#### **3.1. Método de Pesquisa**

Muito embora a ampla revisão teórica desenvolvida, não se tem conhecimento de que o problema apresentado tenha sido abordado sob a mesma perspectiva. Assim, esta pesquisa se caracteriza como um estudo predominantemente exploratório, sendo que o método de pesquisa principal foi o estudo de caso. Neste estudo, pode-se justificar o uso de métodos qualitativos, pois envolve o estudo de DDS no seu contexto real, com a compreensão do estado da arte onde a prática se antecipa à teoria (Creswell, 2003). O método de estudo de caso foi adotado conforme proposto por Yin (2001).

Foram identificados como elementos de análise desafios do DDS sob a ótica da Engenharia de Software, Qualidade de Software e Gerência de Projetos de Software. A coleta de dados foi realizada através de entrevistas *in loco* nas empresas com integrantes de equipes de projetos. O critério inicial para a definição dos entrevistados centrou-se na unidade de análise e nos objetivos do estudo. Neste sentido, foram entrevistados gerentes de projetos, gerentes de unidade e líderes técnicos, devido ao nível de interação destes papéis com colaboradores distribuídos geograficamente. A amostra foi não probabilística, por conveniência, e procurou-se, em conjunto com a organização, uma representatividade dos diversos grupos envolvidos.

Utilizou-se como instrumento de pesquisa um roteiro para uma entrevista semi-estruturada, com questões abertas. Ele foi desenvolvido a partir da teoria estudada e representada pelo protocolo de pesquisa gerado. As questões foram divididas em

tópicos, buscando uma abrangência de todo o processo de desenvolvimento. Além disso, foram desenvolvidos dois questionários, cada um explorando uma dimensão de informações para serem capturadas: dimensão organizacional, com informações da organização como um todo e dimensão de projetos, com informações relacionadas aos projetos vivenciados pelos entrevistados. Devido à limitação de páginas, os roteiros não foram anexos neste artigo, mas estão disponíveis mediante consulta aos autores.

As entrevistas foram realizadas em dois idiomas: inglês e português. Todas foram gravadas, transcritas e analisadas posteriormente, através de análise de conteúdo segundo Krippendorff (2004), buscando-se a replicabilidade e estabilidade dos dados. Para a análise, utilizaram-se as transcrições no idioma original. Buscou-se associar uma pergunta para cada trecho de resposta dada nas entrevistas. Por ser um roteiro semi-estruturado, em alguns casos houve perguntas adicionais, que foram incluídas nas análises. Para cada trecho de resposta, categorias foram identificadas, resultando em um conjunto de categorias, que por sua vez permitiram uma análise profunda do fenômeno.

As questões de qualidade de software basearam-se principalmente na existência e utilização de processos de desenvolvimento nas unidades distribuídas e na adoção de modelos de qualidade. Do ponto de vista da engenharia de software, as questões basearam-se nas técnicas e ferramentas utilizadas para melhorar o desempenho nos projetos distribuídos. Em relação à gerência de projetos, buscou-se avaliar as estratégias utilizadas para obter sucesso na gestão dos projetos distribuídos. Por tratar-se de uma pesquisa qualitativa, devem-se ter claras as limitações existentes, principalmente quanto ao número de empresas estudadas, restringindo a generalização dos resultados obtidos.

### 3.2. Caracterização das Empresas

O estudo foi realizado em quatro empresas ou respectivas unidades, sendo uma delas com matriz no Brasil, duas com matriz no Canadá e uma com matriz nos Estados Unidos. A tabela 1 caracteriza o estudo realizado e os tipos de relações de DDS que foram avaliadas em cada empresa, de acordo com a taxonomia proposta na figura 2.

**Tabela 1. Caracterização das empresas**

Empresa	País de origem	Subsidiária ou parceira	Local da coleta de dados	Modelos de Negócio analisados	Entrevistas
E1	Estados Unidos	Brasil Índia	Estados Unidos Brasil	<i>Offshore Insourcing</i> <i>Onshore Insourcing</i>	5
E2	Brasil	Brasil Estados Unidos	Brasil	<i>Onshore Outsourcing</i> <i>Offshore Outsourcing</i>	5
E3	Canadá	Índia	Canadá	<i>Offshore Outsourcing</i>	4
E4	Canadá	Índia França Inglaterra	Canadá	<i>Offshore Insourcing</i>	3

A empresa 1 (E1) é uma multinacional norte-americana de grande porte. O estudo foi aplicado tanto na matriz da empresa, nos Estados Unidos, quanto na unidade brasileira, localizada no Estado do Rio Grande do Sul. Ela possui mais de 400 colaboradores trabalhando em projetos que atendem as necessidades da área de TI da empresa (demanda interna). A empresa possui dois modelos de negócio: o *offshore insourcing* e o *onshore insourcing*, pois a unidade no Brasil desenvolve projetos internos tanto para a matriz nos Estados Unidos quanto para uma filial no Brasil.



A empresa 2 (E2) é uma multinacional brasileira de desenvolvimento de software, com matriz em São Paulo. O estudo de caso foi desenvolvido na principal unidade de desenvolvimento de software, que possui em torno de 200 colaboradores. Com relação aos clientes, a unidade trabalha apenas com clientes externos à organização. Foram identificados dois modelos de negócio: o *onshore outsourcing*, pois a unidade estudada desenvolve projetos para clientes localizados no Brasil, e também o *offshore outsourcing*, pois a unidade desenvolve projetos para clientes no exterior.

A empresa 3 (E3) é uma multinacional canadense de desenvolvimento de software. O estudo de caso foi realizado na matriz da empresa no Canadá, que possui em torno de 100 colaboradores. Foi identificada a estratégia de *offshore outsourcing*, uma vez que a empresa terceiriza parte do desenvolvimento para uma empresa externa localizada na Índia. Esta empresa na Índia, quando os dados foram coletados, possuía em torno de 20 colaboradores trabalhando com parte da equipe no Canadá.

A empresa 4 (E4) é uma empresa de grande porte localizada no Canadá. O estudo foi realizado na unidade de desenvolvimento de software localizada na matriz da empresa, que possui 45 colaboradores. Foi identificada a estratégia de *offshore insourcing*, pois a empresa desenvolve produtos para vender no mercado, mas os clientes destes produtos são unidades de negócio da própria empresa, que identificam oportunidades nos escritórios de pelo menos quatro países diferentes (Tabela 1).

### **3.3. Resultados dos Estudos de Caso**

Os estudos realizados consideraram os desafios de DDS sob o ponto de vista da Engenharia de Software, Qualidade de Software e Gerência de Projetos de Software. Além disso, considerou-se para análise o tempo de atuação das empresas em projetos de DDS. Desta forma, a empresa E1 atua há quatro anos, a empresa E2 há sete anos, a empresa E3 há três anos e a empresa E4 há um ano. A seguir são apresentados os dados coletados em cada uma das empresas, de acordo com o modelo de negócio analisado.

**Engenharia de Software:** durante as entrevistas, foram feitas perguntas específicas sobre técnicas e ferramentas utilizadas nas atividades de engenharia de software. Os principais resultados encontrados em cada empresa foram:

Empresa 1 (E1): do ponto de vista de *onshore insourcing*, foram comentadas dificuldades em relação ao entendimento dos requisitos. Neste modelo de negócio, a equipe de projeto não tinha muitos problemas de comunicação com o cliente devido ao idioma ser o mesmo e a distância geográfica não ser significativa. Por outro lado, existia uma tendência à falta de formalização de requisições de mudança, antes de a unidade investir na definição de processos de desenvolvimento e implantação de modelos de qualidade. Em relação ao modelo de *offshore insourcing*, foram apontados desafios bem diferentes. Como a equipe no Brasil interagiu com um cliente que possuía conhecimento técnico nos Estados Unidos, houve conflitos em relação a padrões de código a serem utilizados, e diversos desentendimentos em relação aos requisitos a serem identificados, analisados, desenvolvidos e gerenciados. A arquitetura do software também foi uma dificuldade enfrentada pela equipe no Brasil, visto que exista a necessidade de se adequar a arquitetura já existente nos Estados Unidos. Por fim, identificou-se que a manutenção dos sistemas como um importante desafio, visto que a empresa estava recém iniciando um esforço de implantação de processos e gestão de conhecimento.

Empresa 2 (E2): do ponto de vista de *onshore outsourcing*, a unidade estudada terceirizou o desenvolvimento de certos projetos internos para empresas externas. Entre os principais desafios encontrados, está a necessidade de utilizar padrões de codificação definidos pela empresa. Como futuramente a unidade poderia decidir continuar o projeto internamente, ou ainda contratar uma outra empresa, foi necessário um treinamento inicial da empresa contratada nas técnicas e ferramentas de engenharia de software utilizadas. Em relação a outra estratégia existente, a de *offshore outsourcing*, a empresa tinha um cliente nos Estados Unidos, e decidiu desenvolver o projeto na unidade do Brasil. O principal desafio do projeto em relação à Engenharia de Software foi a engenharia de requisitos, principalmente na gestão dos requisitos. Houve diversas requisições de mudanças, motivadas principalmente por dificuldades de comunicação e de falta da correta documentação dos requisitos.

Empresa 3 (E3): nesta empresa foi analisado apenas o modelo de negócio *offshore outsourcing*, onde parte do trabalho foi terceirizado para uma empresa externa, na Índia. Foram citados como desafios a dificuldade em se unificar técnicas de desenvolvimento de software quando parte da equipe de projeto está distribuída. Especificamente, a empresa teve dificuldade com integração de diferentes módulos de um sistema, e com a padronização do desenvolvimento.

Empresa 4 (E4): nesta empresa foi analisado apenas o modelo de negócio *offshore insourcing*. E aqui o principal desafio em termos de Engenharia de Software foi o uso de ferramentas padrão para análise e projeto do sistema dentro da equipe distribuída. Além disso, como a empresa estava recém no início de sua experiência com projetos distribuídos, foi bastante comentada a falta de padronização entre as equipes.

**Qualidade de Software:** durante as entrevistas, foram feitas perguntas específicas sobre a existência de processos bem definidos, sobre como eram realizadas melhorias nos processos, caso existissem, sobre gerência de configuração, políticas da empresa para desenvolvimento de software, além de informações sobre a existência de práticas de gestão de conhecimento. Os resultados encontrados em cada empresa foram:

Empresa 1 (E1): para a estratégia de *onshore insourcing*, a empresa citou como desafio principal a dificuldade em definir um processo de desenvolvimento de software. Já para *offshore insourcing*, além da definição de um processo, surgiu com bastante importância a dificuldade de convergir dois ou mais processos diferentes em um conjunto de práticas para serem utilizadas em um projeto.

Empresa 2 (E2): tanto para a estratégia de *onshore outsourcing*, como para a estratégia de *offshore outsourcing*, os principais desafios encontrados em relação a qualidade de software foram idênticos. Eles se concentraram principalmente em questões de gerência de configuração, visto que as empresas contratada e contratante trabalhavam com ferramentas diferentes para tal finalidade. Além disso, a gestão do conhecimento também foi apontada como um desafio, visto que grande parte do conhecimento estava ficando na empresa contratada, e a empresa estudada recém estava criando estratégias para armazenar e disseminar as informações internamente.

Empresa 3 (E3): para a estratégia de *offshore outsourcing*, a empresa apontou como desafios principais a necessidade de convergir para um processo de desenvolvimento padrão entre as equipes distribuídas. Além disso, apesar de a

contratada na Índia possuir processos com base em modelos de qualidade de software tipo CMMI, os processos da empresa estudada no Canadá não eram baseados em modelos de qualidade, o que, segundo os entrevistados, contribuíram para as dificuldades vivenciadas. Além disso, comentou-se que a gerência de configuração foi um dos maiores desafios dos projetos desenvolvidos nesta estratégia, pois as empresas possuíam estruturas diferentes, e conseqüentemente ferramentas diferentes para desempenhar esta atividade. Além disso, as políticas das empresas para desenvolvimento de software eram sensivelmente diferentes.

Empresa 4 (E4): para a estratégia de *offshore insourcing*, a empresa identificou desafios relacionados principalmente a unificação dos processos nas unidades distribuídas. Além disso, citou que, para cada tipo e tamanho de projeto, um conjunto de processos era necessário, e isto necessariamente deveria ser feito no início de cada projeto. Na sua pouca experiência com projetos distribuídos, a empresa não possuía uma estratégia padrão de gerência de configuração, o que gerava problemas no desenvolvimento dos projetos. Um outro fator importante mencionado foi em relação ao ciclo de vida de desenvolvimento utilizado. A empresa acreditava que a estratégia de *offshore insourcing* era a única que permitia o uso de um ciclo iterativo e incremental, pois os colaboradores tinham um nível de interação maior do que uma estratégia de *offshore outsourcing*, por exemplo, e nesta última eles só enxergavam a possibilidade de desenvolver projetos seguindo um ciclo de vida do tipo cascata.

**Gerência de Projetos de Software:** durante as entrevistas, foram feitas perguntas sobre as atividades relacionadas com a gerência de projetos. Foram incluídas todas as dificuldades relacionadas com a integração da equipe de projeto, e fatores decorrentes da distribuição, entre eles os chamados aspectos não-técnicos (diferenças culturais, confiança, idioma, entre outros). Os principais resultados encontrados foram:

Empresa 1 (E1): para a estratégia de *onshore insourcing*, a empresa identificou desafios relacionados ao planejamento do projeto. Já para *offshore insourcing* os desafios se concentraram em questões tais como idioma diferente, diferenças culturais e falta de compartilhamento de contexto, além da necessidade de haver um processo de aquisição de confiança e gerenciamento de expectativas entre as equipes distribuídas.

Empresa 2 (E2): para a estratégia de *onshore outsourcing*, foram comentadas dificuldades e desafios relacionados diretamente às atividades da gerência do projeto, tais como planejamento, padronização de relatórios de acompanhamento entre outros. Já na estratégia de *offshore outsourcing*, surgiram alguns desafios adicionais, decorrentes da distribuição geográfica ser em outro país. Surgiram de forma bastante importante questões relacionadas à comunicação, e idioma diferente entre as equipes distribuídas, *awareness* (percepção do que cada colaborador está executando), além dos desafios já existentes relacionados às atividades da gerência de projetos. Adicionalmente, o desafio de adquirir confiança apareceu de forma idêntica ao encontrado na empresa 1. Um ponto interessante neste aspecto é que, mesmo estando a mais tempo trabalhando com DDS, o fato de trabalhar com clientes externos à empresa fez com que o processo de aquisição de confiança entre as equipes demorasse mais tempo em relação ao que ocorreu na empresa 1, cujos clientes eram internos.

Empresa 3 (E3): para a estratégia de *offshore outsourcing*, a empresa identificou alguns desafios importantes, tais como a necessidade de padronização de atividades

gerenciais, interfaces entre a empresa contratante e a empresa contratada e um maior controle das tarefas sendo realizadas. Além disso, comentou-se sobre as dificuldades com o idioma e a confiança entre as equipes distribuídas, e a necessidade de se realizar viagens no início dos projetos para aumentar a confiança das equipes, melhorando os processos de engenharia de requisitos e de planejamento do projeto.

Empresa 4 (E4): para a estratégia de *offshore insourcing*, a empresa citou como desafios principais a necessidade de ter ferramentas integradas de gestão de projeto, além da correta distribuição das atividades para os colaboradores e a percepção (*awareness*) sobre o que cada um está executando. Outro comentário foi a necessidade de caracterizar as equipes distribuídas como uma única equipe dentro de um projeto, o que não vinha ocorrendo devido o início recente da empresa neste cenário.

### 3.4. Análise dos Resultados Encontrados

Diversos desafios relacionados ao ambiente de DDS foram identificados nas empresas estudadas. A tabela 2 apresenta um resumo das principais características encontradas, de acordo com a estratégia avaliada, sob a ótica da Engenharia de Software.

**Tabela 2. Resumo dos desafios de Engenharia de Software encontrados**

	Empresa 1 (E1)	Empresa 2 (E2)	Empresa 3 (E3)	Empresa 4 (E4)
<b>Onshore Outsourcing</b>		- Padrões de código - Definição de ferramentas		
<b>Onshore Insourcing</b>	- Gerência de Mudanças			
<b>Offshore Outsourcing</b>		- Engenharia de requisitos - Gerência de Mudanças	- Unificação de técnicas de desenvolvimento - Integração de sistema	
<b>Offshore Insourcing</b>	- Padrões de código - Engenharia de requisitos - Arquitetura do software - Manutenção de software			- Definição de ferramentas - Padrões de código e de desenvolvimento de software

A tabela 3 apresenta um resumo das principais características encontradas nas empresas, de acordo com a estratégia avaliada, sob a ótica da Qualidade de Software.

**Tabela 3. Resumo dos desafios de Qualidade de Software encontrados**

	Empresa 1 (E1)	Empresa 2 (E2)	Empresa 3 (E3)	Empresa 4 (E4)
<b>Onshore Outsourcing</b>		- Gerência de configur. - Gestão do conhecimento		
<b>Onshore Insourcing</b>	- Definição de um processo desenvolvimento de software			
<b>Offshore Outsourcing</b>		- Gerência de configur. - Gestão do conhecimento	- Integração de processos - Gerência de configuração - Unificação de políticas de desenvolvimento - Ciclo de vida em cascata	
<b>Offshore Insourcing</b>	- Definição de um processo desenvolvimento de software - Integração de processos			- Integração de processos diferentes - Gerência de configur. - Ciclo de vida iterativo e incremental

A tabela 4 apresenta as principais características encontradas nas empresas, de acordo com a estratégia avaliada, sob a ótica da Gerência de Projetos de Software.

**Tabela 4. Resumo dos desafios de Gerência de Projetos encontrados**

	Empresa 1 (E1)	Empresa 2 (E2)	Empresa 3 (E3)	Empresa 4 (E4)
<b>Onshore Outsourcing</b>		- Planejamento do projeto - Padronização de artefatos de gestão		
<b>Onshore Insourcing</b>	- Planejamento do projeto			
<b>Offshore Outsourcing</b>		- Comunicação - Idioma - Confiança - <i>Awareness</i>	- Padronização de artefatos de gestão - Idioma - Confiança - Controle do projeto	
<b>Offshore Insourcing</b>	- Idioma - Diferenças culturais - Contexto - Confiança			- Ferramentas integradas de gestão - <i>Awareness</i> - Alocação de atividades - Percepção de equipe

A partir da análise das tabelas apresentadas, algumas conclusões puderam ser identificadas. Do ponto de vista da Engenharia de Software, pode-se destacar que:

- A gerência de mudanças, a necessidade de padronização da atividade de codificação e a utilização de ferramentas para apoiar as atividades de ES distribuídas são desafios que não podem ser menosprezados, independente da estratégia de DDS;
- A engenharia de requisitos tem seus desafios mais acentuados quando a estratégia de DDS possui uma característica *offshore*;

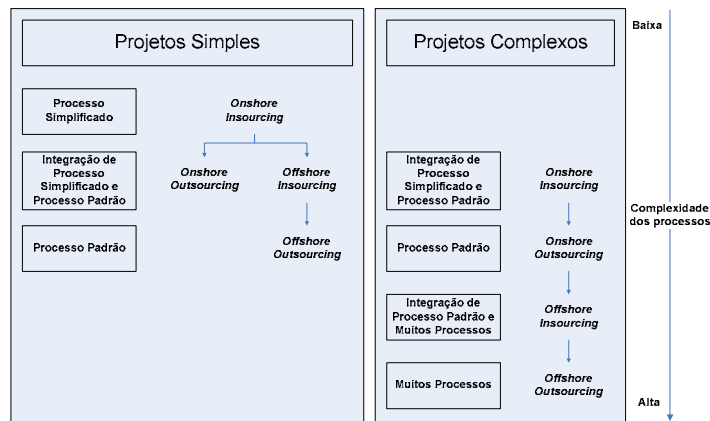
Algumas conclusões também foram identificadas do ponto de vista da Qualidade de Software. Desta forma, pode-se destacar que:

- A gerência de configuração é um dos principais desafios, sendo que as dificuldades são acentuadas quando a estratégia envolve o *outsourcing*;
- De acordo com as empresas estudadas, a estratégia de DDS do tipo *outsourcing*, independente da localização geográfica, geralmente facilita o uso de um ciclo de vida em cascata. Por outro lado, a estratégia do tipo *insourcing* torna mais fácil a utilização de um ciclo de vida iterativo e incremental;
- Apesar de ser importante em todas as estratégias, a gestão de conhecimento do processo de desenvolvimento é mais crítica em estratégias do tipo *outsourcing*;
- A adoção de um processo único, ou a adaptação de processos dos diversos locais distribuídos é de extrema importância e faz a diferença em um projeto de DDS, independente da estratégia adotada pela empresa;

Por fim, também é possível destacar algumas conclusões do ponto de vista da Gerência de Projetos de Software, quais sejam:

- A necessidade de percepção (*awareness*) em relação ao que as equipes estão executando aumenta quando a estratégia tem a característica *offshore*. Nesta estratégia, questões relacionadas ao idioma, à cultura, à comunicação e à confiança são acentuadas;
- O planejamento do projeto, a padronização de artefatos, o processo de gestão, e a existência ferramentas são importantes, independente da estratégia de DDS adotada;

Além disso, um ponto bastante interessante que chamou a atenção durante a análise dos dados foi a relação entre o tamanho e a complexidade dos projetos com a necessidade de processos para obter melhores resultados na sua execução, no contexto de cada modelo de negócio ou estratégia de DDS. Em todas as empresas, questionou-se sobre a existência de processos durante todo o ciclo de desenvolvimento de software. Buscou-se então associar os processos com os tipos de projetos e respectivos modelos de negócios utilizados. Desta forma, como resultado, identificou-se que as empresas trabalhavam com dois tipos de projetos principais: projetos simples e projetos complexos. A classificação de cada tipo variava um pouco em cada empresa, mas era baseada em características idênticas. A complexidade dos projetos era avaliada em relação ao tamanho da equipe e ao número de horas contratadas. Com isso, foi possível desenvolver uma representação da necessidade dos processos de acordo com a complexidade dos projetos e com o modelo de negócio utilizado (Figura 3).



**Figura 3. Complexidade dos Projetos X Modelo de Negócio X Processos**

Identificaram-se três variações do conjunto de processos que eram utilizados nos projetos: processo simplificado (subconjunto de práticas do processo padrão), processo padrão (processo definido pela empresa), e muitos processos (criação de práticas adicionais além das existentes no processo padrão da empresa). Analisando a figura 3, é possível perceber que para projetos simples, as empresas utilizaram processos que variam de um processo simplificado para um modelo de *onshore insourcing*, até o uso de um processo padrão, no modelo de *offshore outsourcing*. Já em relação aos projetos complexos, identificou-se uma clara diferença no uso dos processos em cada modelo de negócio estudado, e uma necessidade de processos mais formalizados.

É interessante observar que para projetos simples não foi necessário o uso do que as empresas chamaram de muitos processos. Por outro lado, para projetos complexos, não se identificou o uso de um processo simplificado nos projetos estudados. Verificou-se ainda a necessidade do uso de uma grande carga de processos quando o modelo de negócio de desenvolvimento de software era caracterizado como *offshore outsourcing*. Então, do ponto de vista de necessidade de uso e complexidade do processo, tanto para projetos que as empresas consideraram simples, como para os projetos considerados complexos, o modelo de *onshore insourcing* representava o nível simples no uso dos processos, enquanto que o modelo de *offshore outsourcing* representou o nível mais crítico e com uma maior carga e necessidade de formalização de processos necessários para o bom andamento e conseqüente sucesso dos projetos.

#### **4. Considerações Finais**

O desenvolvimento de software sempre se apresentou de forma complexa. Existem diversos problemas e desafios inerentes ao processo. Em paralelo, a distribuição das equipes no tempo e no espaço tem tornado os projetos distribuídos cada vez mais comuns. O software é cada vez mais indispensável para a sociedade moderna, onde a globalização é uma característica fundamental. Neste contexto, o DDS surge como um grande desafio para a área de ES e tem levado os pesquisadores a defrontar-se com a necessidade de novos conhecimentos e uma abordagem mais multidisciplinar e até interdisciplinar. O estudo realizado buscou contribuir para aumentar a ainda pequena, mas crescente literatura da área, identificando desafios de quatro estratégias de DDS, comparando-se empresas no Brasil e no exterior, com diferentes tempos de experiência.

Os resultados deste estudo, assim como de estudos anteriores, apresentam oportunidades para aprofundar pesquisas voltadas para esta nova classe de problemas que o DDS está trazendo (Sengupta et. al., 2006). Do ponto de vista científico, a legitimação dos resultados é decorrente do processo de pesquisa como um todo. Entre os principais resultados, constatou-se a semelhança de alguns desafios, independente da estratégia. Por outro lado, é clara a necessidade de um maior esforço durante todo o ciclo de vida de um projeto, quando a estratégia possui uma característica *offshore*.

Além disso, percebeu-se uma tendência cada vez maior de investimentos na criação de centros de desenvolvimento próprios, caracterizando assim uma operação do tipo *insourcing*. Isto ficou evidente ao analisar o tempo de experiência das empresas na área. Estudos anteriores (Carmel et. al., 2005) apontam o *offshore insourcing* como o último estágio a ser alcançado em uma representação de maturidade de modelos de negócio para DDS. Mas o estudo realizado mostrou que mesmo empresas atuando há apenas um ano neste contexto avaliam como positivo o investimento em *insourcing*.

Em relação à Engenharia de Software, cabe ressaltar que para cada modelo de negócio adotado, dificuldades diferentes podem surgir. Assim, recomenda-se estabelecer planos de ação no início de cada projeto, evitando surpresas ao longo do ciclo de vida. Especificamente em relação ao DDS no Brasil, Carmel et. al. (2005) identificam o Brasil como uma potência em ascensão. Segundo os autores, o país está no segundo time de possíveis destinos para operações distribuídas, mas em constante crescimento. Perde apenas para países tais como Índia e China. E em relação a práticas de DDS adotadas no Brasil e em outros países, nota-se uma maior experiência nos outros países. Por isto recomenda-se também o investimento na formação de profissionais para atuar globalmente e orientados para o DDS, evitando o foco apenas na formação técnica.

Como estudo futuro, pretende-se expandir a pesquisa para outras empresas e outros modelos de negócio. Pretende-se também identificar características essenciais que as empresas que investem em DDS possuem ao longo do tempo, desde o momento em que iniciam uma operação deste tipo, seja ela qual for, até atingir um estágio onde se pode perceber um amadurecimento significativo nas práticas de desenvolvimento de software neste contexto. Além disso, outras oportunidades envolvem o aprofundamento em temas específicos de DDS, associados a cada modelo de negócio explorado. Entre os temas mais críticos identifica-se engenharia de requisitos, gerência de risco, gerência de configuração, gestão de conhecimento, definição de processos globais, manutenção de software, projetos distribuídos em micro e pequenas empresas e teste de software.

## **Referências Bibliográficas**

- Bohem, B. (2006), “A View of 20th and 21st Century Software Engineering”. In: 28<sup>th</sup> 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”. Prentice Hall, EUA, 269p.
- Creswell, J. W. (2003), “Research Design: Qualitative, Quantitative, and Mixed Methods Approaches”. EUA: SAGE Publications.
- Herbsleb, J. D., Moitra, D. (2001), “Global Software Development”, IEEE Software, March/April, EUA, p. 16-20.
- IDC – International Data Group (2006), Disponível em <http://www.idc.com/>, abril/2006.
- Krippendorff, K. (2004), “Content analysis: an introduction to its methodology”. EUA: Sage Publications.
- Lopes, L., Prikladnicki, R., Audy, J. L. N. (2004), “Distributed Requirements Specification: Minimizing the Effect of Geographic Dispersion”. In: VI ICEIS, Cidade do Porto, Portugal, p. 531-534.
- Morstead, S., Blount, G. (2003), “Offshore Ready: Strategies to Plan & Profit from offshore IT-Enabled Services”. EUA: ISANI Press.
- Pilatti, L., Audy, J. L. N., Prikladnicki, R. (2006), “Software Configuration Management over a Global Software Development Environment: Lessons Learned from a Case Study”. In: Workshop on Global Software Development for the Practitioners at ICSE, Xangai, China.
- Prikladnicki, R., Evaristo, R., Audy, J. L. N., Yamaguti, M. H. (2006), “Risk Management in Distributed IT Projects: Integrating Strategic, Tactical, and Operational Levels”. Aceito para publicação no International Journal of e-Collaboration, special issue on Collaborative Project Management.
- Prikladnicki, R., Audy, Jorge L. N. (2004), “MuNDDoS: Um Modelo de Referência para Desenvolvimento Distribuído de Software”, In: 18<sup>o</sup> SBES, Brasília, Brasil.
- Prikladnicki, R., Audy, J. L. N., Evaristo, R. (2003), “Global Software Development in Practice: Lessons Learned”, Journal of Software Process: Improvement and Practice – Special Issue on Global Software Development, Wiley, 8(4), Oct-Dec, p. 267-281.
- Robinson, M., Kalakota, R. (2004), “Offshore Outsourcing: Business Models, ROI and Best Practices”. EUA: Mivar Press.
- Schniederjans, M. J., Schniederjans, M. A., Schniederjans, D. G. (2003), “Outsourcing and Insourcing in an International Context”. EUA: M. E. Sharpe.
- Sengupta, B., Chandra, S., Sinha, V. (2006) “A Research Agenda for Distributed Software Development”. In: 28<sup>th</sup> ICSE, Xangai, China.
- Yin, Robert. (2001), “Estudo de Caso: planejamento e métodos”. SP: Bookman, 2001.

---

<sup>i</sup> 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.