

Modelos de Colaboração no Desenvolvimento Distribuído de Software: uma Revisão Sistemática da Literatura

Rodrigo Rocha¹, Catarina Costa¹, Rafael Prikladnicki², Ryan Ribeiro de Azevedo^{1,3}, Ivaldir H. F. Junior, Silvio Meira¹

¹Centro de Informática – Universidade Federal de Pernambuco (UFPE)
50.732-970 – Recife – PE – Brasil

²Faculdade de Informática – Pontifícia Universidade Católica do Rio Grande do Sul
90.619-900 – Porto Alegre – RS – Brasil

³Universidade Federal do Piauí (UFPI)
64049-550 – Picos – PI – Brasil

{rger, csc, rra2, ihfj, srlm}@cin.ufpe.br, rafaelp@puccrs.br

Abstract. *With the rise of Distributed Software Development, software houses attempt to find out the best strategy to improve the development of products and services in a distributed way, but without losing quality at a low cost. This work proposes to identify what models of collaboration are used by organizations to develop software within a distributed context, based on the traditional development life cycle, where the phases thereof are performed (onsite, distributed / offshore and multi-site). The research method used to investigate the forms of cooperation was a systematic literature review that have examined 841 papers published from 2000 to 2009.*

Resumo. *Com a ascensão do Desenvolvimento Distribuído de Software as organizações tentam distribuir da melhor maneira possível suas atividades de desenvolvimento com interesse de melhorá-las. O objetivo deste trabalho é identificar quais modelos de colaboração são utilizados pelas organizações para desenvolver software no contexto distribuído, tendo como base o ciclo de vida básico do desenvolvimento tradicional, e onde as fases do mesmo são realizadas (onsite, distribuído/offshore e multi-site). O método de pesquisa utilizado para levantar as formas de colaboração foi uma revisão sistemática da literatura que analisou 841 trabalhos publicados desde 2000 até 2009.*

1. Introdução

Diante da demanda por software e pelo crescimento do número de empresas que desenvolvem essas soluções, na década passada, como reflexo da globalização, empresas de software começaram a distribuir seus processos de desenvolvimento em lugares diferentes, criando o Desenvolvimento Distribuído de Software (DDS) (Herbsleb, 2007). Também conhecido como Desenvolvimento Global de Software (para projetos que envolvem outros países).

Diversos autores afirmam que o desenvolvimento de sistemas é uma atividade complexa (The Standish Group, 2001), já que desenvolver software envolve incertezas e diversos riscos. E, acrescentando-se a isso distância física e temporal entre os

participantes do processo, os desafios inerentes ao desenvolvimento tendem a ser acentuados e outras dificuldades podem surgir.

Dessa maneira, inúmeras empresas tentam adotar técnicas, metodologias e ferramentas para que possam lidar da melhor forma com as variáveis do contexto distribuído. Em diversos momentos, as empresas que utilizam esse modelo de desenvolvimento necessitam definir quais metodologias utilizarão em um determinado projeto e suas respectivas práticas, pois, estas não provêm de informações que sejam capazes de determinar qual metodologia é a mais adequada para um determinado tipo de projeto ou quais as práticas mais indicadas para o mesmo.

Com a descentralização das empresas e a produção de software acontecendo de forma distribuída, o desenvolvimento de software torna-se mais complexo, exigindo que as organizações necessitem buscar modelos de colaboração (ou formas de trabalho) que consigam satisfazer suas características e necessidades. Smitte e Borzovs (2009) afirmam que os responsáveis pelo projeto devem estar cientes da maneira que a organização das equipes e do projeto podem ser feitas, no sentido de que as formas de trabalho das equipes possam levar o projeto a obter um melhor desempenho. Então, é necessário que haja mais conhecimento dos responsáveis pelo projeto sobre os modelos de colaboração, ou seja, como em seu projeto eles podem dividir as atividades em cada fase do desenvolvimento e onde cada atividade pode ser realizada.

Assim, o objetivo deste trabalho é identificar através de uma revisão sistemática da literatura quais modelos de colaboração são utilizados pela indústria e/ou academia para desenvolver software no contexto distribuído, tendo como base o ciclo de vida básico do desenvolvimento tradicional de software (requisitos, análise, implementação e testes), como também suas variações e se as fases do mesmo são realizadas *onsite* (no cliente), *distribuído/offshore* e *multi-site* (em ambos). Além de apresentar os passos realizados na revisão sistemática da literatura objetivando identificar tais modelos.

Este trabalho está organizado da seguinte maneira: na Seção 2 são apresentados os conceitos de Revisão Sistemática da Literatura; a Seção 3 descreve os passos adotados durante a revisão sistemática, como a questão de pesquisa, a estratégia de busca (chaves da pesquisa, string de busca, fontes de busca, entre outros), assim como seus resultados, logo, os modelos de desenvolvimento no DDS são também apresentados; e por fim, a Seção 4 aborda as considerações finais.

2. Revisão Sistemática da Literatura

As revisões sistemáticas da literatura (*Systematic Literature Review* – SLR em inglês) são parte do paradigma de práticas baseadas em evidências. Muito utilizadas na medicina e ciências da saúde, as revisões sistemáticas estão se popularizando em outras áreas, mas ainda não estão bem estabelecidas na engenharia de software (OATES E CAPPER, 2009).

A essência do paradigma baseado em evidência é coletar e analisar sistematicamente todos os dados disponíveis sobre determinado fenômeno para obter uma perspectiva mais completa e mais ampla do que se pode captar através de um estudo individual. Um dos principais métodos da engenharia de software baseada em evidências são as revisões sistemáticas, classificadas como estudos secundários, já que,

dependem dos estudos primários utilizados para revelar evidências e construir conhecimento (OATES e CAPPER, 2009; TRAVASSOS e BIOLCHINI, 2007).

As Revisões Sistemáticas da Literatura (SLR) são estudos que atuam como norteador para o desenvolvimento de projetos, de forma a direcionar determinada pesquisa, sintetizando todos os estudos relevantes que respondam a determinada questão.

Segundo Travassos e Biolchini (2007) uma SLR é uma forma de realizar uma revisão da literatura de forma não tendenciosa e abrangente, fazendo com que seus resultados tenham cunho científico. Existem alguns aspectos que precedem o início de uma revisão sistemática, os quais devem ser levados em conta, como delimitar o objetivo da mesma, reconhecer a literatura e avaliar os estudos possíveis de serem incluídos. Estes três itens são essenciais ao pesquisador, pois os auxiliam a aperfeiçoar a questão norteadora da revisão, tornando-a bem formulada e clara. (DOMHOLDT, 2005)

O início da SLR se dá, primeiramente, pela determinação do protocolo que enumera as questões a serem pesquisadas e os métodos utilizados para guiar a revisão. Sendo assim, Travassos e Biolchini (2007) assinalam três fases essenciais para a construção da SLR, são elas: Planejamento, Execução e Análise dos Resultados, os quais são detalhados na Figura 1.

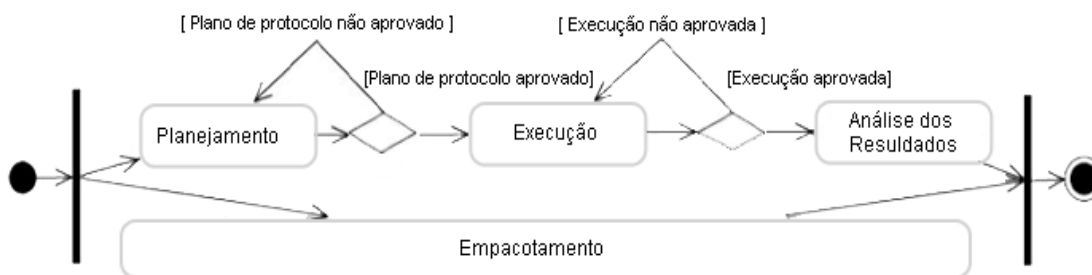


Figura 1 - Processo de Construção da Revisão Sistemática.

Fonte: Travassos e Biolchini (2007)

No Planejamento estão incluídos os objetivos principais da pesquisa, as questões e os critérios que influenciaram na inclusão ou exclusão dos achados, as estratégias de investigação, a escolha prévia dos artigos obtidos, a seleção final dos mesmos e o direcionamento da revisão. É nele que se define o protocolo de revisão, onde é sugerida a revisão do mesmo por especialistas ou até mesmo através de um teste de execução.

É na fase da Execução que se realiza a investigação nas fontes pré-estabelecidas prosseguindo, então, para o estudo dos trabalhos selecionados e classificação dos mesmos segundo os critérios de inclusão e exclusão determinados no protocolo. Caso haja restrições no decorrer das buscas, podem ser feitas adaptações para satisfazer quaisquer limitações.

Na última fase, a Análise de Resultados, é realizada a coleta dos dados extraídos dos artigos encontrados e selecionados segundo os critérios estabelecidos no protocolo. A partir disto, estes dados são analisados e sintetizados de acordo com o método

escolhido. Ao final de uma SLR obteremos um mapeamento da conjuntura real do conhecimento, facultando um melhor planejamento.

3. Processo de Identificação dos Modelos

Para dar início à descrição do processo da Revisão Sistemática que foi utilizado nesse trabalho, foi essencial a construção do protocolo, como forma de direcionamento da mesma buscando avaliar os escritos que respondessem à questão da pesquisa. Para tanto, seguem abaixo os principais tópicos do protocolo da revisão sistemática que foram utilizados nessa pesquisa.

Uma Revisão Sistemática tem o intuito de responder a uma questão de pesquisa, e neste caso a questão investigada foi: “Quais são os modelos de desenvolvimento para o DDS?”. Esta questão tem o objetivo de assimilar quais são os modelos utilizados pela indústria e/ou academia para desenvolver software no contexto distribuído. Tendo como base o ciclo de vida básico do desenvolvimento tradicional de software (requisitos, análise, implementação e testes) e suas variações, e também, onde cada fase do ciclo é realizada.

3.1. Estratégia de Busca

Ao se iniciar a pesquisa dos estudos primários, Kitchenham (2007) sugere que uma estratégia deve ser usada, através da definição das palavras chaves, bibliotecas digitais, jornais e conferências.

3.1.1. String de Busca

A partir de combinações feitas com palavras-chaves relevantes, foi construída a *string* de busca padrão direcionada. Os termos e sinônimos identificados são apresentados abaixo:

- **Desenvolvimento Distribuído de Software:** Distributed Software Development, Distributed Software Engineering, Distributed Software Teams, Global Software Development, Global Software Engineering, Global Software Teams, Collaborative Software Development, Collaborative Software Engineering, Collaborative software teams, Globally Distributed Work, Distributed Development, Distributed Teams, Global Software Teams, Globally Distributed Development, Geographically Distributed Software Development, Offshore Software Development, Offshore, Offshore Outsourcing, Dispersed Teams;

- **Modelos de desenvolvimento:** Collaboration models, Collaboration Model, Models of Collaboration, Model of Collaboration, Development Model*, Models of Development*, Collaboration form*, Form* of Collaboration, Development proces*, Process* of Development, Work form*, Form* of Work, Life Cycle Models*.

3.1.2. Definição das Fontes de Busca

Para fazer a seleção das fontes, foi definido alguns critérios: os artigos devem estar disponíveis na web, mecanismos de busca através de palavras-chave e com garantia de resultados únicos ao mesmo conjunto de palavras-chave. Os artigos também podem ser obtidos por pessoas com experiência no assunto. O idioma das fontes deve ser em inglês assim como o idioma dos artigos, por ser o idioma mais comum entre as principais conferências do tema investigado.

As fontes utilizadas foram:

- IEEEExplore Digital Library (<http://ieeexplore.ieee.org/>)
- ACM Digital Library (<http://portal.acm.org>)
- Elsevier ScienceDirect (www.sciencedirect.com)
- EI Compendex (www.engineeringvillage2.org)

Além das fontes supracitadas, foi incluído também no estudo o capítulo do livro *Infonomics for Distributed Business and Decision-Making Environments: Creating Information System Ecology* (SMITE e BORZOVS, 2009), já que o mesmo trata diretamente das formas de colaboração para o DDS.

3.2. Definição dos Critérios de Inclusão de Estudos

A análise para inclusão de artigos foi feita por títulos, palavra-chave, resumo e conclusão do trabalho. Os seguintes critérios de inclusão para os artigos foram definidos:

- a) estar disponível na internet;
- b) ser escrito em Inglês;
- c) ter sido publicado entre 2000 e 2009 (quando DDS começou a ser realmente consolidado);
- d) estudos que apresentem, primária ou secundariamente, modelos de desenvolvimento para projetos distribuídos de software relacionados ao ciclo básico de desenvolvimento (requisitos, projeto, codificação, testes);
- e) possuir projetos comerciais/industriais/acadêmicos;

3.3 Tipos de Estudo

Os tipos de estudos são classificados como:

- Experimentais ou *Empirical Studies* (trabalhos onde dados empíricos coletados e analisados);
- Teóricos (estudos conceituais baseados em um entendimento de uma área, referenciando outros trabalhos relacionados);
- Revisões Sistemáticas (trabalhos que usam uma metodologia bem definida para identificar, analisar e interpretar evidências relacionadas a uma questão de pesquisa específica);
- Relato de Experiência Industrial (*Industrial Experience Report*).

3.4 Resultados

A Tabela 1 descreve os resultados obtidos através da Revisão Sistemática, abordando desde a questão da pesquisa até a definição dos estudos primários. Após a busca nas 4 fontes definidas, os trabalhos encontrados foram selecionados, primeiramente, com base no título e palavra-chave. Dos 95 trabalhos encontrados no **IEEEExplore Digital Library**, 58 estavam inseridos nos critérios de busca da pesquisa. No **ACM Digital Library**, dos 443 resultados obtidos apenas 139 satisfizeram a questão da presente pesquisa. Foram encontrados 235 artigos no **Elsevier ScienceDirect**, dos quais somente 48 se encaixavam nas especificações estabelecidas e, dos 67 estudos do **EI Compendex**, 42 foram aceitos para a segunda fase da seleção.

Tabela 1 – Método de Busca e Escolha de Estudos Primários

Fontes	Método de Busca e Escolha de Estudos Primários					Estudos Primários
	Busca	1ª Seleção (Título e Palavra-chave)	2ª Seleção (Resumo e Conclusão)			
			Irrelevante	Repetido/Duplicado	Incompleto	
Questão de Pesquisa						
IEEEXplore Digital Library	95	58	52	0	1	5
ACM Digital Library	443	139	134	0	4	1
Elsevier ScienceDirect	235	48	47	0	0	1
EI Compendex	67	42	31	2	9	0
Infonomics for Distributed Business and Decision-Making Environments: Creating Information System Ecology	1	1	0	0	0	1
TOTAL	841	288	264	2	14	8

Com a primeira seleção, os 841 trabalhos retornados na busca, se resumiram a 288 potenciais estudos que foram analisados novamente, agora através do resumo e da conclusão. A partir disto, os estudos que atendessem ao critérios de inclusão, era incluídos como estudos primários e os demais não aceitos eram subdivididos em Irrelevante, Repetido/Duplicado e Incompleto. Os trabalhos selecionados como Estudos Primários foram avaliados segundo seu ano de publicação. Dos 8 resultados retornados, 1 foi publicado no ano de 2001, nenhum foi datado entre os anos de 2002 e 2005 e em 2008, 2 foram de 2006, 4 de 2007 e 1 foi publicado em 2009.

Os tipos de estudo são classificados em: experimentais, teóricos, revisões sistemáticas ou relatos de experiências industriais. A maioria dos estudos selecionados foram estudos experimentais, no caso, cinco. O segundo tipo mais encontrado, com dois, foi de estudos teóricos. Relato de experiências industriais apareceu em um dos estudos, e nenhum trabalho utilizou uma revisão sistemática como tipo de estudo.

3.5 Modelos de Colaboração Identificados

Na Tabela 2 são apresentados os modelos de desenvolvimento dos trabalhos encontrados na revisão sistemática. Na coluna do lado esquerdo são expostas todas as fases de desenvolvimento que foram encontradas na SLR. Ao lado, os estudos e as formas de desenvolvimento são apresentados, onde é possível perceber, por exemplo, que o estudo 1 (T1 –significando Trabalho 1) possuiu cinco fases de desenvolvimento, sendo a fase de requisitos realizada *multisite*, a fase de projeto e codificação realizada distribuída ou *offshore*, os testes foram *multisite*, e a implantação *onsite*. Da mesma forma que T1 foi descrito, os outros estudos (T2, T3, T4, T5, T6, T7) podem ser, seguindo o mesmo entendimento, onde cada um pode realizar atividades e onde as mesmas são realizadas, no cliente ou de forma distribuída.

Tabela 2 – Modelos de desenvolvimento distribuídos de software encontrados na SLR

Fases	Estudos	T1		T2		T3		T4		T5		T6		T7	
	Formas	On	Dis	On	Dis	O n	Dis	On	Dis	On	Dis	On	Dis	On	Dis
Requisitos															
Cron. Proj. e Plan. Releases															
Seleção do Time e Contrato															
Análise															
Projeto															
Codificação															
Testes															
Gar. Qualidade															
Trein. Time Offshore															
Avaliação Usabilidade															
Implantação															

Através dos resultados foi possível encontrar sete modelos de desenvolvimento distribuído, onde as fases do desenvolvimento e o local onde essas são realizadas variam de trabalho para trabalho.

O oitavo trabalho (T8) é o de Smite e Borzovs (2009), que pode ser considerado como um trabalho relacionado. Nessa pesquisa, foram identificados dezenove modelos de colaboração através de uma pesquisa de campo com trinta e oito projetos distribuídos, porém, diferentemente desta pesquisa, a pesquisa de Smite e Borzovs foca apenas em quatro fases de desenvolvimento (Análise, Projeto, Codificação e Testes). Isso pode limitar o estudo, uma vez que cada projeto é único, e seu contexto e características variam de projeto para projeto.

A maioria dos trabalhos encontrados na revisão sistemática diz respeito à descrição dos projetos e seus resultados. Ou seja, não são trabalhos que busquem identificar os modelos de desenvolvimento que existem e suas formas de colaboração. Dessa maneira, é possível afirmar que a inexistência de trabalhos nessa área e um incentivo para pesquisadores e organizações investirem nesse segmento, já que é um campo aberto e ainda sem muita exploração.

4. Considerações Finais

Esta pesquisa apresenta os modelos de desenvolvimento para ambientes distribuídos que tanto a Indústria de Software como a Academia utilizam. Esses modelos foram identificados através de uma revisão sistemática da literatura realizada entre setembro de 2009 até janeiro de 2010, envolvendo 841 trabalhos, publicados entre 2000 e 2009.

Embora este estudo tenha seguido uma metodologia rigorosa, o mesmo possui algumas limitações, como a quantidade de estudos da revisão sistemática da literatura, que tomou como base, apenas 8 trabalhos. Esse pequeno número de trabalhos pode se explicar pela imaturidade da área de DDS, tendo em vista que ela ascendeu nas últimas duas décadas porém, pela academia, trabalhos sobre o tema só surgiram nos últimos dez ou doze anos.

Alguns trabalhos podem ser desenvolvidos futuramente a partir deste estudo, como: pesquisas mais aprofundadas sobre os modelos de desenvolvimento, que possibilitem identificar qual modelo seria mais adequado para determinado contexto;

analisar melhor os resultados da SLR a fim de identificar alguns porquês, como o caso do modelo realizar mais tarefas distribuídas do que no cliente, embora algumas dessas questões tenham sido identificadas; procurar investigar outros modelos, a fim de classificá-los e caracterizando-os de uma melhor forma.

Referências

- Domholdt E. (2005). *Rehabilitation research: principles and applications*. Missouri: Elsevier Saunders.
- Herbsleb, J. D. (2007). *Global Software Engineering: The Future of Socio-technical Coordination*. IEEE Computer Science. p188-198.
- Oates, J. B.; Capper G. (2009). Using systematic reviews and evidence-based software engineering with masters students. *International Conference on Evaluation & Assessment in Software Engineering, EASE*.
- Pfleeger, S. and Kitchenham, B. (2001). Principles of survey research: part 1: turning lemons into lemonade. *ACM SIGSOFT Software Engineering Notes*, 26(6):44–45.
- Prikladnicki, R.; Audy, J. (2004). *Munddos: Um Modelo de Referência para Desenvolvimento Distribuído de Software*. 18º Simpósio Brasileiro de Engenharia de Software.
- Smite, Darja., Borzovs, Juris. (2009). *New Forms of Work in the Light of Globalization in Software Development. Infoeconomic for Distributed Business and Decision-Making Enviroments: Creating Information System Ecology*. Business Science Reference – Blekinge Intitute of Technology, p346.
- The Standish Group. (2001). *CHAOS 2001: A Recipe for Success*.
- Travassos, G., Biolchini J. (2007). *Revisões Sistemáticas Aplicadas a Engenharia de Software*. In: XXI SBES - Brazilian Symposium on Software Engineering, João Pessoa, PB, Brasil.

Apêndice: Estudos Primários

T1	Gaurav Caprihan. (2006) <i>Managing Software Performance in the Globally Distributed Software Development Paradigm</i> , Proceedings of the IEEE international conference on Global Software Engineering, p.83-91, October 16-19.
T2	James Cusick, Alpana Prasad. (2006). <i>A Practical Management and Engineering Approach to Offshore Collaboration</i> , IEEE Software, v.23 n.5, p.20-29.
T3	Alvin W. Yeo. (2001). <i>Global-software development lifecycle: an exploratory study</i> , Proceedings of the SIGCHI conference on Human factors in computing systems, p.104-111.
T4	Rizwanjameelqureshi, M., Hussain, S. (2008). <i>An adaptive software development process model</i> . <i>Advances in Engineering Software</i> . Volume: 39, Issue: 8, Pages: 654-658.
T5	Setamanit, S., Wakeland, W., Raffo, D. (2007). <i>Improving Global Software Development Project Performance Using Simulation</i> . In: <i>Portland International Conference on Management of engineering and Technology</i> Portland, OR, USA.
T6	Faiz, M. F., Qadri, U. Ayyubi, S.R. (2007). <i>Offshore Software Development Models</i> . <i>Information and Emerging Technologies (IET 2007)</i> . pp. 1-6.
T7	Leszak, Marek., Meier, Mandref. (2007). <i>Successful Global Development of a Large-scale Embedded Telecommunications Product</i> . <i>International Conference on Global Software Engineering (ICGSE 2007)</i> . pp. 23-32.
T8	Smite, D., Borzovs, Juris. (2009). <i>New Forms of Work in the Light of Globalization in Software Development. Infoeconomic for Distributed Business and Decision-Making Enviroments: Creating Information System Ecology</i> . Business Science Reference – Blekinge Intitute of Technology, p346.