

Aplicando uma Metodologia Baseada em Evidência na Definição de Novas Tecnologias de Software

Sômulo Nogueira Mafrá¹, Rafael Ferreira Barcelos^{1,2}, Guilherme Horta Travassos¹

{somulo, barcelos, ght}@cos.ufrj.br

¹COPPE/UFRJ - Programa de Engenharia de Sistemas e Computação, Cx. Postal 68.511
CEP 21945-970, Rio de Janeiro - RJ – Brasil

²BENQ-SIEMENS – Research and Development Department, Manaus – AM – Brasil

***Abstract.** The transfer of Software technologies to the industrial context could bring undesirable consequences if such technologies are not under an adequate maturity level. This paper aims at to describe the use of an evidence-based methodology that can support the reduction of this risk. The application of such methodology is illustrated through two concrete cases regarding the definition of a software requirements reading technique and a checklist based approach for inspecting software architectural models. Besides, some lessons learned by using the methodology are also described.*

***Resumo.** A introdução de tecnologias de software recém-definidas no contexto industrial pode trazer conseqüências indesejáveis caso a tecnologia não possua um grau adequado de maturidade. Nesse sentido, o presente artigo visa a ilustrar como a utilização de uma metodologia baseada em evidência pode auxiliar a minimizar essa situação. A utilização de tal metodologia é ilustrada através de dois casos concretos relacionados à definição e à avaliação experimental de uma técnica de leitura de requisitos de software e de uma abordagem para inspeção de documentos arquiteturais. Além disso, são discutidas as principais lições aprendidas do uso de tal metodologia.*

1. Introdução

A busca pela melhoria da qualidade no desenvolvimento de software tem sido constante nos últimos anos. Para atingir esse fim, um programa de melhoria em uma organização geralmente caracteriza-se por ser realizado em duas frentes [Kitchenham *et al.* 2004]: por um lado, organizações visam à implantação de processos aderentes a modelos de maturidade, como o CMMI [Chrissis *et al.* 2003] e o MR mps [Weber *et al.* 2004]; por outro, engenheiros de software buscam aprimorar suas qualificações através da obtenção de certificados profissionais, como o PMP [PMP 2000], por exemplo.

Entretanto, ainda de acordo com Kitchenham *et al.* (2004), a utilização de processos de qualidade por engenheiros de software qualificados não é condição suficiente para a melhoria da qualidade no desenvolvimento. O desenvolvimento é dependente de diversas tecnologias muitas das quais não se possuem evidências suficientes sobre potenciais benefícios, limitações, custo de implantação e riscos associados. Em vista desse quadro, gerentes freqüentemente são confrontados com as

seguintes questões quando da adoção de novas tecnologias de software:

- Em qual tecnologia investir, quando todas elas prometem aprimorar a produtividade e a qualidade no desenvolvimento [Shull 1998]?
- Como saber o custo de implantação de uma determinada tecnologia?
- Como determinar o retorno de investimento da implantação de tal tecnologia?
- Sob quais circunstâncias a adoção de tal tecnologia pode ser recomendada?

Essa situação pode ser agravada pela baixa interação muitas vezes observada entre academia e indústria no que diz respeito à definição de novas tecnologias [Kaindl *et al.* 2002]. A alta demanda na indústria por soluções de curto prazo aliada à necessidade de justificar a continuidade de investimento em pesquisa têm pressionado cada vez mais a transferência para a indústria de tecnologias recém-definidas na academia. Como consequência, muitas tecnologias são transferidas para a indústria sem terem passado, durante seu processo de definição, por um processo adequado de avaliação que permitisse a caracterização do seu grau de maturidade.

Visando a minimizar a ocorrência desses problemas, Shull *et al.* (2001) propuseram uma metodologia para a introdução de tecnologias de software na indústria. Tal metodologia apresenta uma série de questões que devem ser tratadas durante a avaliação de uma tecnologia, assim como os tipos de estudos experimentais (estudos primários) que contemplam essas questões. A aplicação de tal metodologia possibilita determinar com níveis razoáveis de segurança as limitações e os pontos fortes da aplicação da tecnologia avaliada. Como resultado, eventuais dificuldades na aplicação da tecnologia poderiam ser exploradas, contribuindo para refinar a técnica antes de disponibilizá-la para a indústria.

Entretanto, a metodologia proposta em Shull *et al.* (2001) não contempla a etapa da definição inicial de uma tecnologia de software. A metodologia parte do princípio de que já existe uma versão inicial da tecnologia a ser avaliada. Nesse sentido, a identificação do conhecimento sobre a área disponível na literatura, que possui fundamental importância para a etapa de definição inicial de uma tecnologia, não é contemplada. Como consequência, esse processo de identificação de conhecimento geralmente é conduzido informalmente, aumentando a possibilidade de viés na pesquisa.

Portanto, o objetivo deste trabalho é apresentar uma extensão à metodologia proposta por Shull *et al.* (2001). Tal extensão é representada pela aplicação de uma abordagem formal para a identificação de evidência (estudos secundários) existente na literatura. A utilização da metodologia estendida é ilustrada através de dois casos concretos, que levam em consideração a definição e a avaliação experimental de uma abordagem para inspeção de documentos arquiteturais baseada em *checklist*, denominada ArqCheck [Barcelos e Travassos 2006a], e de uma técnica de leitura denominada OO-PBR [Mafra e Travassos 2006a].

O impacto da condução de estudos secundários na definição de novas tecnologias é discutido através de lições aprendidas oriundas tanto de projetos de pesquisas que utilizaram tal extensão quanto de projetos que não a utilizaram. A condução de estudos experimentais para caracterizar quantitativamente a contribuição da extensão não é tratada neste artigo, ficando como perspectiva futura de trabalho.

O texto encontra-se dividido em sete seções, incluindo esta que introduz o

artigo. Na seção 2, a metodologia para introdução de tecnologias de software na indústria é descrita. A seção 3 discute as lições aprendidas do uso de tal metodologia durante a definição de duas tecnologias. Essas lições motivaram a extensão proposta neste trabalho, que é descrita na seção 4. Na seção 5, a aplicação da metodologia estendida é ilustrada através da avaliação de ArqCheck e OO-PBR. As lições aprendidas da metodologia estendida são descritas na seção 6. Finalmente, a seção 7 conclui o artigo.

2. Metodologia Original: Estudos Primários

A metodologia experimental para introdução de tecnologias de software na indústria definida em Shull *et al.* (2001) é composta por quatro etapas, que contemplam a avaliação da tecnologia desde sua definição até sua transferência para a indústria. As quatro etapas são representadas por: (a) estudos de viabilidade, (b) estudos de observação, (c) estudo de caso (ciclo de vida), e (c) estudo de caso na indústria.

Os estudos sugeridos para cada uma das etapas estão diretamente relacionados às variáveis que se deseja controlar e os riscos possíveis de serem assumidos na fase em questão, conforme apontado por Silva (2004). Os quatro tipos de estudo são sucintamente descritos nas próximas subseções.

2.1. Estudo de Viabilidade

O objetivo principal de um estudo de viabilidade não é encontrar uma resposta definitiva, mas sim criar um corpo de conhecimento sobre a aplicação da tecnologia. Nesse sentido, seria possível ao pesquisador avaliar se a aplicação da tecnologia é viável, ou seja, se atende de forma razoável aos objetivos inicialmente definidos, de forma a justificar (ou não) a continuação da pesquisa. Além disso, espera-se que o corpo de conhecimento construído forneça subsídios que permitam: (a) o refinamento da tecnologia, e (b) a geração de novas hipóteses sobre sua aplicação a serem investigadas em estudos posteriores [Shull *et al.* 2004].

2.2. Estudo de Observação

Um estudo de observação geralmente é realizado em ambiente no qual a aplicação prática da tecnologia é observada por pesquisadores. Através dessa observação, é possível coletar dados sobre como a tecnologia é aplicada. Dessa forma, os pesquisadores podem adquirir uma compreensão refinada sobre a tecnologia, ao presenciarem eventuais dificuldades que os participantes possam apresentar.

2.3. Estudo de Caso (Ciclo de Vida)

Atingir esta fase pressupõe que haja indícios de que a tecnologia sendo avaliada é efetiva. Porém, a tecnologia foi avaliada de forma isolada, não sendo possível determinar qual o efeito de sua aplicação em conjunto com outras tecnologias. Para ter acesso a essa informação, um estudo de caso deve ser conduzido com o propósito de caracterizar a aplicação da tecnologia no contexto de um ciclo de vida de desenvolvimento.

2.4. Estudo de Caso (Indústria)

A condução das etapas anteriores permitiria: (a) determinar a viabilidade prática de aplicação da tecnologia, (b) aprimorar o entendimento dos pesquisadores resultando no

refinamento da tecnologia e (c) caracterizar a aplicação da tecnologia no contexto de um ciclo de vida. Como consequência, supõe-se que a tecnologia tenha atingido um grau de maturidade razoável que possibilite a sua avaliação no contexto industrial.

A condução de um estudo de caso na indústria tem como principal objetivo caracterizar a aplicação da tecnologia no ambiente industrial. Na próxima seção, são listadas algumas lições aprendidas sobre a aplicação da metodologia de introdução de tecnologias de software na indústria que motivaram a extensão proposta neste trabalho.

3. Utilizando a Metodologia Original

A metodologia de introdução de tecnologias de software na indústria tem sido tradicionalmente utilizada na pesquisa realizada pelo Laboratório de Engenharia de Software da COPPE/UFRJ [Silva 2004; Kalinowski e Travassos 2004; Lima 2005; Nunez 2005; Mafra 2006; Barcelos 2006]. Através da experiência adquirida com a aplicação dessa metodologia foi possível identificar oportunidades de melhoria que resultaram na proposição de sua extensão, discutida na seção 4.

As lições aprendidas decorrentes da aplicação da metodologia, e a necessidade de se estender tal metodologia, são discutidas através da experiência vivenciada durante a definição das OORT's [Travassos *et al.* 1999], uma família de técnicas de leitura para modelos de projeto OO, e de ISPIS [Kalinowski e Travassos 2004], uma infra-estrutura computacional de apoio a inspeções de software. Ambas as tecnologias foram avaliadas através da condução das quatro etapas contempladas pela metodologia.

3.1. OORT's: Técnicas de Leitura para Modelos de Projeto OO

Durante sua definição, os autores das OORT's não tinham identificado na literatura abordagens de leitura para a inspeção de modelos de projeto OO. Essa carência de iniciativas representou um desafio considerável de pesquisa. Os autores tiveram de definir uma nova tecnologia contando apenas com suas próprias experiências no desenvolvimento de aplicações OO.

Porém, os autores das OORT's possuíam experiência na definição e avaliação experimental de outras técnicas de leitura, entre as quais PBR [Shull 1998], referente a inspeções de requisitos, por exemplo. A viabilidade de aplicação de técnicas de leitura em diferentes tipos de artefatos, aliada à demanda industrial, foi a principal motivação para a proposição de uma família de técnicas de leitura para inspeção de modelos de projeto OO. Contudo, diversas dúvidas permearam a definição das OORT's, entre elas:

- A falta de apoio à leitura de modelos de projeto OO representava de fato um problema para a indústria. Entretanto, como resolvê-lo?
- A abordagem de inspeção representada pela aplicação de técnicas de leitura (que tinha se mostrado adequada para a inspeção de requisitos, por exemplo) seria viável também para a inspeção de modelos de projeto OO?

O desafio representado por essas questões motivou a definição e a utilização de uma metodologia para a avaliação de tecnologias imaturas visando à sua transferência para a indústria. A utilização dessa metodologia no contexto das OORT's contribuiu para a caracterização do grau de maturidade da tecnologia. Resultados experimentais incluindo todos os estudos previstos na metodologia podem ser encontrados em Travassos *et al.* (1999), Shull *et al.* (2001), Melo *et al.* (2001) e Conradi *et al.* (2003).

Naquele momento, as dúvidas e incertezas presenciadas foram consideradas

naturais devido à carência de relatos na literatura sobre experiências anteriores que pudessem auxiliar a tomada de decisão durante o processo de definição das OORT's. Como consequência, a necessidade de se conduzir uma revisão sistemática ainda não tinha sido considerada.

3.2. ISPIS: Infra-estrutura Computacional para Inspeções de Software

Diferentemente do cenário presenciado durante a definição das OORT's, a pesquisa sobre inspeções de software possuía considerável pesquisa disponível na literatura. Como consequência, os autores de ISPIS puderam realizar uma revisão criteriosa da literatura levando em consideração resultados experimentais relacionados a processos de inspeção de software.

Através dos resultados obtidos, ISPIS foi definida e avaliada experimentalmente. A descrição e os resultados dos quatro tipos de estudo a que ISPIS foi submetida podem ser encontrados em Kalinowski e Travassos (2004) e Kalinowski e Travassos (2005).

Entretanto, mesmo a presença relativamente extensa na literatura de estudos experimentais sobre inspeções de software não contribuiu para a redução de incertezas durante a definição de ISPIS. Como exemplo das dificuldades encontradas pode ser citada a falta de consenso sobre a condução de inspeções síncronas ou assíncronas.

3.3. Lições Aprendidas

Como principal lição aprendida dessas pesquisas, verificou-se que a simples presença de estudos experimentais na literatura sobre uma determinada área de pesquisa não seria suficiente para minimizar a dificuldade na definição de uma nova tecnologia; resultados experimentais devem ser caracterizados: (a) para que sejam melhor entendidos, e (b) para que seja possível identificar tendências e oportunidades de pesquisa.

Para isso, um processo de caracterização de evidências deveria ser formalizado com o objetivo de se identificar e caracterizar eventuais resultados experimentais existentes relacionados à tecnologia a ser desenvolvida. A formalização desse processo permitiria que essa etapa fosse repetível e passível de ser melhorada.

Essa necessidade motivou a extensão da metodologia proposta em Shull *et al.* (2001) de forma a contemplar a adoção de uma abordagem baseada em evidência para a caracterização de resultados experimentais. Tal abordagem é representada pela condução de estudos secundários (revisões sistemáticas). A extensão é descrita na próxima seção.

4. Estendendo a Metodologia Original: Estudos Secundários

Esta seção descreve a importância da condução de estudos secundários no processo de geração de evidências na pesquisa científica. Além disso, é descrito como estudos secundários são complementares a estudos experimentais (estudos primários) nesse processo de geração de evidências. Na sequência, é descrito como a condução de estudos secundários pode contribuir para a metodologia proposta em Shull *et al.* (2001).

4.1. Revisões Sistemáticas em Engenharia de Software

A importância da aplicação de uma metodologia baseada em evidência na pesquisa científica pode ser ilustrada pela experiência vivenciada na Medicina. Durante muito tempo, a área médica esteve repleta de revisões que não utilizavam métodos para

identificar, avaliar e sintetizar a informação existente na literatura [Cochrane 2003]. No final da década de 80, estudos conduzidos para avaliar a qualidade das publicações médicas chamaram a atenção para a baixa qualidade científica [Cochrane 2003].

Uma provável consequência dessa falta de rigor pôde ser observada no final dos anos 80s. Naquela ocasião, estudos apontaram que, de um lado, a falha em organizar a pesquisa médica em revisões sistemáticas podia custar vidas [Cochrane 1989]. Por outro lado, julgamentos clínicos de alguns especialistas foram considerados inadequados quando comparados aos resultados de revisões sistemáticas [Antman *et. al* 1992].

Desde então, o reconhecimento da necessidade da condução de revisões de forma sistemática e formal em Medicina tem crescido rapidamente. Este fato pode ser comprovado pela grande quantidade de revisões formais publicadas a cada ano na área médica [NHSCRD 2003]. Tais revisões, denominadas de revisões sistemáticas, são revisões rigorosas da literatura à procura de indícios que possam levar à identificação de evidências sobre um tema de pesquisa ou tópico na área em questão.

Foi o trabalho de Kitchenham *et al.* (2004) o primeiro a estabelecer um paralelo entre Medicina e Engenharia de Software, no que diz respeito à abordagem baseada em evidência. Segundo os autores, a Engenharia de Software Baseada em Evidência deve prover meios pelos quais melhores evidências provenientes da pesquisa possam ser integradas com experiência prática e valores humanos no processo de tomada de decisão considerando o desenvolvimento e a manutenção do software.

Em vista disso, para atingir um nível adequado de evidência a respeito da caracterização de uma determinada tecnologia em uso, a Engenharia de Software Baseada em Evidência deve fazer uso basicamente de dois tipos de estudos: estudos primários e estudos secundários [Mafra e Travassos 2006b].

Por estudos primários, entende-se a condução de estudos que visem a caracterizar uma tecnologia em uso dentro de um contexto específico. Nessa categoria encontram-se os estudos experimentais, entre os quais experimentos, estudos de caso e *surveys* [Wöhlin *et al.* 2000]. A metodologia de Shull *et al.* (2001) contempla a condução de estudos primários para a avaliação de uma tecnologia de software.

Por estudos secundários, entende-se a condução de estudos que visem a identificar, avaliar e interpretar todos os resultados relevantes a um determinado tópico de pesquisa, fenômeno de interesse ou questão de pesquisa. Revisão sistemática é um tipo de estudo secundário [Kitchenham *et al.* 2004; Biolchini *et al.* 2005]. Nesse sentido, resultados obtidos por diversos estudos primários correlatos atuam como fonte de informação a ser investigada por estudos secundários (Figura 1).

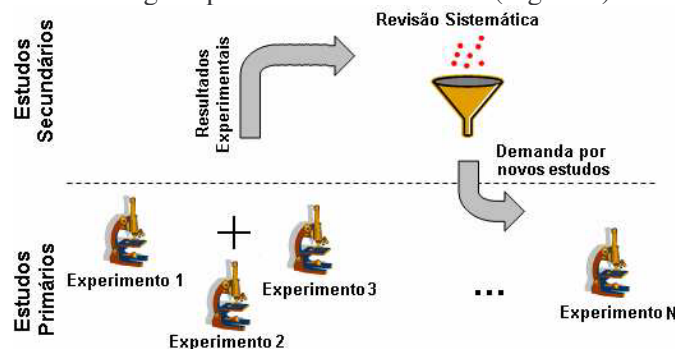


Figura 1 - Interação entre Estudos Primários e Secundários.

Além disso, é válido ressaltar que estudos secundários não podem ser considerados uma abordagem alternativa para a produção primária de evidências, representada pelos estudos primários [Biolchini *et al.* 2005]; estudos secundários são complementares a estudos primários nesse processo. A precisão e a confiabilidade proporcionadas pela condução de estudos secundários contribuem para a melhoria e para o direcionamento de novos tópicos de pesquisa, a serem investigados por estudos primários, num ciclo iterativo.

4.2. Proposta de Extensão à Metodologia Original

A proposta apresentada nesse trabalho consiste em estender a metodologia experimental de introdução de tecnologias de software na indústria, definida em Shull *et al.* (2001), através da condução de revisões sistemáticas (Figura 2). Nesse sentido, a metodologia resultante fica dividida em dois passos: Definição e Refinamento da Tecnologia.

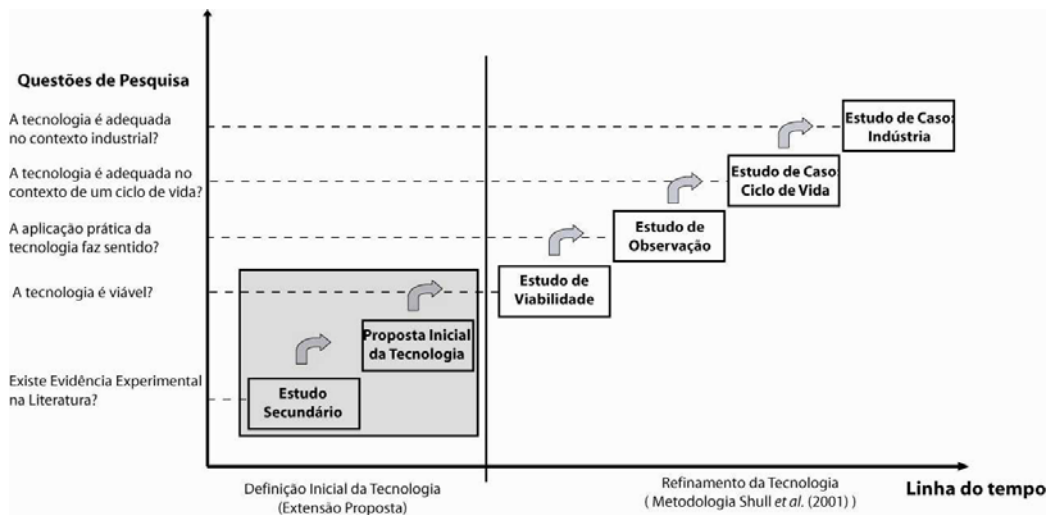


Figura 2 - Extensão da Metodologia Experimental para Introdução de Tecnologias de Software proposta em Shull *et al.* (2001).

O primeiro passo consiste na Definição Inicial da Tecnologia. Durante esse passo, a condução de uma revisão sistemática seria a primeira etapa a ser seguida. O planejamento da revisão sistemática representado pela definição de um protocolo de pesquisa ajudaria a delimitar o escopo da pesquisa. A presença de tal protocolo, definindo de forma explícita as questões de pesquisa, os critérios de seleção de fontes bibliográficas, os critérios de inclusão e exclusão de artigos, e os critérios de qualidade a serem observados nos estudos, ajudaria a amadurecer o problema sendo investigado.

Além disso, os procedimentos a serem seguidos durante a condução da revisão sistemática, no que se refere à identificação e à caracterização da evidência existente na área, poderiam reduzir drasticamente riscos associados à definição de uma tecnologia. Como consequência dessa caracterização, oportunidades de melhoria pertinentes à área de pesquisa poderiam ser identificadas. Além disso, tal caracterização possibilitaria identificar as lições aprendidas dos estudos identificados, minimizando a possibilidade de repetir erros passados.

Com base no conhecimento adquirido e nas evidências identificadas através da condução da revisão sistemática, o pesquisador define uma proposta inicial da

tecnologia. A partir dessa versão inicial da tecnologia, as etapas relacionadas à série de estudos de avaliação propostos na metodologia original são realizadas, visando a um refinamento sucessivo e a um amadurecimento da tecnologia em questão.

Na próxima seção, a aplicação da metodologia estendida é ilustrada através de dois casos concretos.

5. Aplicando a Metodologia Estendida

A metodologia estendida foi utilizada no contexto da definição e avaliação experimental de duas novas tecnologias de software. As subseções a seguir descrevem como a condução de revisões sistemáticas foi fundamental para a definição dessas tecnologias e para a posterior avaliação experimental à qual elas foram submetidas.

5.1. ArqCheck: Abordagem de Inspeção de Documentos Arquiteturais

Uma das motivações para a identificação de abordagens para inspeções arquiteturais na literatura foi decorrente da necessidade de um projeto acadêmico referente ao desenvolvimento de uma infra-estrutura para condução de estudos experimentais – Projeto eSEE [Mian *et al.* 2005]. Naquele momento, havia interesse na aplicação de uma abordagem para a avaliação da arquitetura eSEE em relação aos requisitos.

Além disso, a cada ano vem aumentando o interesse de empresas brasileiras de desenvolvimento de software por abordagens de melhoria de processos e de produtos [MTC/SEPIN 2005]. A necessidade por uma abordagem de inspeção arquitetural que fosse extensível, simples de ser aplicada, genérica, e adaptável [Barcelos e Travassos 2006a] visando à utilização no contexto dessas empresas foi outra motivação para a realização do trabalho. Portanto, uma revisão sistemática foi conduzida, e com base nos resultados obtidos, cada abordagem foi caracterizada de acordo com tais critérios. A revisão sistemática é sucintamente apresentada na próxima subseção.

5.1.1. Estudo Secundário: Revisão Sistemática sobre Abordagens de Inspeção Arquitetural

A revisão sistemática conduzida no contexto da definição de ArqCheck teve como principal propósito a obtenção de informações sobre o estado da arte em relação às abordagens de inspeções de documentos arquiteturais existentes na literatura.

O objetivo da revisão pode ser resumido como: *analisar* abordagens que avaliam documentos arquiteturais *com o propósito de* caracterizá-las *com respeito às* características presentes em técnicas de inspeção de software *do ponto de vista dos* pesquisadores *no contexto dos* estudos primários que descrevem tais abordagens.

A revisão sistemática foi inicialmente executada durante o mês de dezembro de 2004, e re-executada, durante o mês de dezembro de 2005. A decisão para re-executar a revisão sistemática deveu-se a dois motivos: (a) identificar novas publicações e abordagens arquiteturais que eventualmente tenham surgido durante esse intervalo de tempo, e (b) verificar a viabilidade de se identificar publicações na máquina de busca da ACM [ACM 2006], que sofreu evoluções após a primeira revisão ter sido executada.

Como resultado, foram encontrados 119 artigos, sendo 38 selecionados. Tais artigos representavam 27 diferentes abordagens de inspeção de documentos arquiteturais. Maiores detalhes sobre as revisões conduzidas, incluindo o protocolo de revisão utilizado e a análise dos resultados dos artigos identificados, podem ser

encontrados em Barcelos e Travassos (2006b) e Barcelos (2006).

5.1.1.1. Contribuição do Estudo Secundário para a Definição de ArqCheck

Os resultados obtidos pela revisão sistemática chamaram a atenção sobre a relativa imaturidade da área de arquitetura de software. Como exemplo de tal imaturidade pode ser citado a falta de consenso na comunidade acadêmica sobre a definição de conceitos básicos e sobre a forma como uma arquitetura de software deve ser representada.

Outro fato que chamou a atenção foi a falta de evidência experimental sobre o assunto. Nenhum dos artigos obtidos descrevia estudos experimentais com o objetivo de avaliar a aplicação das abordagens propostas. Essa falta de evidência experimental limitou as contribuições que a condução de uma revisão sistemática poderia fornecer para a definição de ArqCheck.

Entretanto, a condução de um estudo secundário possibilitou principalmente a caracterização das abordagens de avaliação arquitetural identificadas. Essa caracterização permitiu o entendimento das abordagens identificadas, assim como uma comparação entre elas. A partir dessa caracterização, foi possível identificar limitações que dificultavam a aplicação dessas abordagens em um contexto industrial.

Nesse sentido, a principal contribuição da revisão sistemática conduzida foi a identificação dessas limitações que influenciaram significativamente a definição de ArqCheck. A Tabela 1 resume as decisões de projetos tomadas visando a minimizar as limitações identificadas na aplicação de ArqCheck.

Tabela 1 - Decisões tomadas para minimizar as limitações identificadas

Limitação das Abordagens Identificadas	Solução para Minimizar as Limitações
Subjetividade na avaliação	Uso de um <i>checklist</i> composto por itens de avaliação que orientam o inspetor sobre o que ser avaliado, não deixando essa decisão somente a cargo de sua experiência.
Custo de aplicação	Uso de inspeção de software por não exigir a realização de tarefas complexas, como a definição de cenários. Além disso, ArqCheck não exige uma grande quantidade de profissionais ou especialistas em arquitetura para que a avaliação seja realizada.
Avaliação de múltiplas características de qualidade	Definição de itens no <i>checklist</i> para avaliar as características de qualidade de acordo com os requisitos de qualidade especificados pelo cliente e que possuem relevância em um contexto arquitetural
Contexto de aplicação limitado	Definição de um processo de configuração que permite adaptar o <i>checklist</i> às características do documento arquitetural a ser avaliado e aos objetivos da avaliação.

Na próxima subseção são descritos os estudos primários realizados com o objetivo de refinar ArqCheck.

5.1.2. Estudos Primários: Avaliação Experimental de ArqCheck

A abordagem ArqCheck ainda encontra-se em fase de definição, tendo sido avaliada em dois estudos experimentais. No momento (Abril de 2006), estudos de caso no contexto de um ciclo de vida e na indústria estão em fase de planejamento.

Estudo de viabilidade. O objetivo foi avaliar se os itens que compõem ArqCheck estavam claros para os participantes e se auxiliavam na detecção de defeitos [Barcelos e

Travassos 2006a]. O estudo foi executado em outubro de 2005 com quatro participantes escolhidos por conveniência. Os participantes eram alunos de pós-graduação com conhecimento sobre o domínio do problema e sobre técnicas de inspeção.

Para a realização desse estudo, foi utilizado um documento representando a arquitetura de software desenvolvido em um contexto acadêmico. Esse documento continha 18 defeitos conhecidos, identificados através de uma inspeção *ad hoc*. Além disso, o pesquisador semeou 10 defeitos que permitiam avaliar se os itens de ArqCheck estavam claros e se poderiam auxiliar na identificação de defeitos.

Devido à falta de uma abordagem padrão para representação arquitetural, foi realizado, antes da execução do estudo, um treinamento com os participantes sobre a abordagem de representação que foi utilizada para descrever o documento arquitetural avaliado. O principal objetivo na realização desse treinamento foi o de evitar eventuais problemas de entendimento dos participantes em relação à simbologia utilizada, o que poderia influenciar na tarefa de identificação de defeitos.

Após a realização do estudo, o pesquisador analisou os dados fornecidos por cada participante. Com base nesses dados, foi possível obter indícios de que ArqCheck auxilia na identificação de defeitos, um vez que 75% dos defeitos conhecidos foram encontrados. Além disso, os participantes indicaram que ArqCheck foi útil na identificação dos defeitos. Através desse estudo, foi possível identificar itens de ArqCheck que não foram compreendidos pelos participantes. Com isso, esses itens foram evoluídos, resultando numa nova versão de ArqCheck. Um detalhamento mais completo desse estudo e dos resultados obtidos pode ser encontrado em Barcelos e Travassos (2006a).

Estudo de Observação. O estudo foi conduzido em fevereiro de 2006 em um ambiente acadêmico utilizando estudantes de pós-graduação com conhecimentos sobre arquitetura de software e experiência na condução de inspeções. A condução desse estudo permitiu avaliar a aplicação de ArqCheck durante a inspeção de um documento arquitetural desenvolvido em ambiente industrial. Para permitir essa troca de informações academia-indústria, um acordo foi firmado entre a COPPE/UFRJ e a BENQ-SIEMENS.

A BENQ-SIEMENS é um instituto de pesquisa e de desenvolvimento de aplicações de software voltadas para a área de telefonia celular. O acordo firmado entre essas duas instituições previa o acesso dos pesquisadores a documentos de requisitos e de projeto de alguns projetos da BENQ-SIEMENS visando à realização do estudo.

Portanto, com o acesso a esses dados, foi possível caracterizar ArqCheck em relação a um *baseline*. Para esse estudo, o *baseline* escolhido foi uma abordagem de avaliação *ad hoc* utilizada pela BENQ-SIEMENS na melhoria da qualidade do documento arquitetural utilizado no contexto desse estudo.

Após os participantes terem aplicado ArqCheck, o pesquisador consolidou as discrepâncias encontradas e as enviou para os profissionais da BENQ-SIEMENS que tiveram a responsabilidade de identificar quais discrepâncias representavam defeitos reais. Baseado nas informações fornecidas por esses profissionais, foi possível constatar:

- A viabilidade de ArqCheck no que se refere à identificação de defeitos em documentos arquiteturais (47 discrepâncias representavam defeitos reais);

- A dificuldade de se utilizar ArqCheck na avaliação de documentos arquiteturais grandes devido a quantidade de avaliações realizadas durante sua aplicação.

Como resultado, ArqCheck foi evoluída resultando em nova versão a ser investigada em estudos de caso posteriores, em fase de planejamento. Uma descrição mais detalhada desse estudo pode ser encontrada em Barcelos (2006).

5.2. OO-PBR: Uma Técnica de Leitura de Requisitos de Software

A motivação para a pesquisa sobre técnicas de leitura de documento de requisitos de software foi decorrente da importância exercida por tal documento no contexto de um ciclo de desenvolvimento. Tal importância pode ser explicada pelo fato do documento de requisitos [Mafra e Travassos 2006a]: (a) representar o interesse de diferentes *stakeholders* e (b) servir como base para a construção de diversos artefatos em fases posteriores do desenvolvimento, como o modelo de projeto, por exemplo.

Nesse sentido, observou-se que a definição de uma técnica de leitura que contemplasse aspectos OO poderia ser útil ao fornecer ao engenheiro de software orientação sobre como obter um entendimento satisfatório sobre o problema e os conceitos representados nos requisitos. Através da obtenção desse entendimento, o engenheiro de software poderia ser orientado a: (a) revisar o documento de requisitos, e (b) construir um modelo de projetos OO com base nos requisitos. Na próxima subseção, a revisão sistemática sobre técnicas de leitura de software é descrita.

5.2.1. Estudo Secundário: Revisão Sistemática sobre Técnicas de Leitura de Requisitos de Software

No momento da definição do escopo do trabalho, não era do conhecimento dos pesquisadores a existência de técnicas de leitura que explorassem a construção de projetos OO durante a sua aplicação. A perspectiva de projetista utilizada pela técnica de leitura PBR [Shull 1998], por exemplo, explora a construção de diagramas de fluxo de dados, utilizando técnicas de análise estruturada. Além do mais, não sabia-se também da existência de todos os estudos experimentais dessa natureza. Nem sobre o grau de generalização das evidências geradas por eles, caso esses estudos existissem, e caso houvesse evidências significativas de seus resultados. Portanto, com o propósito de se obter um entendimento sobre a pesquisa experimental realizada na área, optou-se pela condução de uma revisão sistemática.

Porém, apesar do foco inicial da pesquisa estar na identificação de técnicas de leitura para documentos de requisitos de software que explorassem a construção de modelos de projeto OO, decidiu-se por não restringir a busca apenas a esse tipo de técnicas. Como justificativa, caso os resultados da revisão sistemática conduzida demandassem, uma nova técnica de leitura seria proposta. Nesse sentido, os resultados experimentais, as lições aprendidas, e os procedimentos envolvidos na condução dos estudos sobre todas as técnicas de leitura obtidas poderiam auxiliar de forma considerável uma eventual avaliação experimental da nova técnica de leitura proposta. Com isso, para a condução da revisão a seguinte estratégia foi seguida:

- Obter os diversos estudos experimentais realizados sobre técnicas de leitura de documento de requisitos de software descritos em linguagem natural;
- Avaliar os resultados experimentais obtidos, as lições aprendidas, e os procedimentos utilizados na condução desses estudos;

- Identificar, no universo desses estudos, a existência de estudos sobre técnicas de leitura que explorassem a construção de modelos de projeto orientado a objetos;
- Caso a revisão retornasse resultados positivos, esses estudos seriam analisados no sentido de avaliar o grau de evidência a ser extraído de seus resultados, visando a obter benefícios e limitações do uso das técnicas investigadas;
- Caso contrário, pretendia-se propor uma nova técnica de leitura e conduzir estudos no sentido de tentar responder eventuais hipóteses de interesse.

A revisão sistemática foi conduzida entre o final de 2004 e o início de 2005, durante um período de três meses. Ao todo foram analisados 278 artigos dos quais 38 foram selecionados e tiveram os seus dados extraídos e analisados, de acordo com as recomendações previstas no protocolo da revisão. O protocolo de revisão utilizado e os principais resultados experimentais obtidos por essa revisão sistemática podem ser encontrados em Mafra e Travassos (2005) e Mafra (2006).

5.2.1.1. Contribuição do Estudo Secundário para a Definição de OO-PBR

Através da análise dos resultados da revisão, concluiu-se que a definição de uma nova técnica PBR que explorasse a construção de modelos de projeto OO minimizaria potenciais riscos referentes à definição de uma tecnologia inócua ou inviável.

Estudos experimentais demonstraram que a construção de modelos durante a aplicação de PBR permitiu aos revisores uma compreensão refinada sobre o problema descrito nos requisitos [Denger *et al.* 2004; Belgamo e Fabbri 2005]. Como consequência a aplicação de PBR permitiu a identificação de defeitos críticos, muitas vezes não capturados por outras abordagens utilizadas na indústria, como *checklist* e leitura *ad hoc* [Biffi e Halling 2003; Berling e Thelin 2004]. Além disso, resultados obtidos da avaliação experimental das OORT's apontaram para a viabilidade de se explorar conceitos OO durante o processo de inspeção [Travassos *et al.* 1999].

A condução da revisão sistemática possibilitou também a caracterização dos procedimentos experimentais utilizados nos estudos executados sobre técnicas de leitura. A caracterização desses procedimentos permitiu o planejamento dos estudos sobre OO-PBR com relativa segurança, ao levar em consideração as lições aprendidas e as recomendações propostas. Como benefícios podem ser citados a identificação das hipóteses investigadas, o material de treinamento disponibilizado aos participantes, os questionários de avaliação de OO-PBR utilizados nos estudos, o projeto dos estudos experimentais, entre outros. A definição desses mecanismos exclusivamente a partir da própria experiência dos pesquisadores certamente consumiria tempo e esforço considerável de pesquisa.

5.2.2. Estudos Primários: Avaliação Experimental de OO-PBR

A técnica OO-PBR foi avaliada em dois estudos de observação precedidos por um estudo de viabilidade [Mafra 2006], todos executados em ambiente acadêmico. OO-PBR ainda encontra-se em fase de avaliação, necessitando de estudos na indústria.

Estudo de viabilidade. O estudo não teve como objetivo o teste de hipóteses; a formulação de quaisquer hipóteses durante essa etapa inicial de elaboração da técnica poderia ser vista como mera especulação, dado o grau de imaturidade de uma tecnologia recém-definida. O estudo de viabilidade de OO-PBR teve como objetivo proporcionar aos pesquisadores um entendimento sobre OO-PBR decorrente da aplicação prática da

técnica. Pretendia-se durante a condução do estudo avaliar principalmente a usabilidade e a utilidade de OO-PBR. Acreditava-se que a falha no atendimento satisfatório desses dois fatores poderia comprometer uma futura avaliação experimental de OO-PBR.

Para esse estudo foram utilizados dois participantes com larga experiência na condução de inspeções de software na indústria, e na elaboração de modelos de projeto OO. Além disso, os participantes utilizados possuíam experiência também na aplicação de outras técnicas de leitura, entre elas PBR perspectiva do usuário e OORT's. Os participantes foram estimulados a tecerem quaisquer comentários que julgassem pertinentes acerca dos materiais envolvidos no estudo. Nesse sentido, o relato de críticas (positivas ou negativas) e sugestões poderia fornecer um indicativo sobre os potenciais pontos fortes e fracos de OO-PBR.

Como principal resultado do estudo de viabilidade, verificou-se um efeito colateral de OO-PBR referente à orientação sobre a construção de um modelo de projeto. Revisores tendiam a investir a maior parte do tempo de inspeção na construção do modelo ao invés de revisarem o documento em busca de defeitos. Tal efeito pôde ser explicado devido ao alto nível de detalhamento dos passos para a construção do modelo. Nesse sentido, resolveu-se alterar OO-PBR para aumentar o foco na detecção de defeitos.

Primeiro Estudo de Observação. O primeiro estudo de observação foi executado no contexto de uma disciplina de Engenharia de Software OO da UFRJ. O estudo utilizou 15 alunos de graduação com baixa experiência industrial de desenvolvimento (média 2,5 anos) como participantes. Os participantes foram divididos em cinco equipes, sendo cada equipe responsável pelo desenvolvimento de um sistema seguindo um determinado ciclo de desenvolvimento. Nesse sentido, OO-PBR teve sua aplicação avaliada após a fase de definição dos requisitos e antes da fase de projeto de alto nível.

Como principal resultado do estudo, foi observado a inadequação do processo de leitura definido em OO-PBR. Tal processo exigia que o revisor lesse o documento diversas vezes, de forma a identificar defeitos. Como consequência, observou-se que a fadiga causada pela leitura em excesso prejudicou o desempenho dos revisores, ocasionando um baixo nível de conformidade com o processo. OO-PBR teve seu processo de leitura revisado, de forma a: (a) minimizar a quantidade de leitura necessária, e (b) aumentar o foco sobre a detecção de defeitos.

Segundo Estudo de Observação. O segundo estudo foi realizado no contexto de uma disciplina de pós-graduação da COPPE/UFRJ. O estudo utilizou quatro estudantes como participantes, sendo que dois deles possuíam mais de 15 anos de experiência industrial.

A análise dos resultados do estudo permitiu avaliar de forma positiva as alterações no processo de leitura de OO-PBR. A modificação no processo de leitura permitiu um aumento na frequência de acesso às questões para identificação de defeitos, ocasionando num aumento de defeitos detectados. Além disso, a aplicação da técnica demandou um menor número de leituras do documento, conforme esperado.

6. Lições Aprendidas do Uso da Metodologia Estendida

A utilização de uma metodologia baseada em evidência representada por revisões sistemáticas e por estudos experimentais contribui satisfatoriamente para a definição de uma nova tecnologia.

A condução de uma revisão sistemática como etapa do passo de definição inicial de uma tecnologia possibilita: (a) minimizar riscos e (b) acelerar o processo de definição. Por sua vez, a execução de estudos experimentais permite avaliar a aplicação da tecnologia sendo definida, evitando a introdução de uma tecnologia relativamente imatura no contexto industrial.

A identificação de evidências na literatura é o foco de uma revisão sistemática. Essa evidência pode ser caracterizada tanto pela presença de tópicos que reconhecidamente necessitam de maior investigação assim como por tópicos sobre os quais já se observa um grau de consenso na comunidade acadêmica e na indústria. Nesse sentido, a identificação de oportunidades de pesquisa decorrentes da caracterização da evidência na literatura minimiza o risco relacionado à definição de uma tecnologia inadequada, ou seja, que não represente solução para problemas vivenciados na indústria.

Além disso, a caracterização dos resultados experimentais obtidos por uma revisão sistemática pode acelerar o processo de definição de uma nova tecnologia. A caracterização desses resultados provê o pesquisador com lições aprendidas decorrentes de experiências anteriores. Tais lições são fundamentais para dissipar eventuais dúvidas e incertezas que caracterizam a definição de uma nova tecnologia. Além disso, os pesquisadores podem fazer uso de mecanismos comprovadamente eficazes, evitando o desperdício de tempo na elaboração de tais mecanismos.

7. Conclusões

A Engenharia de Software ainda faz pouco uso de métodos científicos na definição de novas tecnologias. Tecnologias de software são frequentemente definidas em laboratórios sem que um processo adequado de transferência para a indústria seja conduzido. A falta de caracterização da tecnologia em uso, decorrente da falta de utilização de uma abordagem baseada em evidência durante sua etapa de definição, torna a Engenharia de Software palco para especulações sobre a eficácia e a qualidade de novas tecnologias.

De forma a tratar esses problemas, Shull *et al.* (2001) propuseram uma metodologia para transferência de tecnologias de software para a indústria. Tal metodologia prevê etapas que contemplam a avaliação experimental da tecnologia sendo definida, visando a caracterizar o seu grau de maturidade, antes de sua transferência para a indústria. Entretanto, a etapa de definição inicial de uma tecnologia representada pela revisão de literatura não é contemplada por tal metodologia.

Nesse sentido, o presente trabalho propôs como a metodologia de Shull *et al.* (2001) poderia ser estendida para contemplar a fase de concepção de uma nova tecnologia. Através da condução de revisões sistemáticas, seria possível identificar e caracterizar o grau de evidência experimental existente na área. Como consequência, eventuais dificuldades e incertezas que povoam o processo de definição de uma nova tecnologia poderiam ser minimizadas.

Com a adoção de uma abordagem baseada em evidência, caracterizada pela condução de estudos primários e secundários, a Engenharia de Software poderia atingir um elevado grau de maturidade. Dessa forma, conforme apontado por Juristo e Moreno (2002), evoluir-se-ia do desenvolvimento de software baseado em especulação para o desenvolvimento de software baseado em fatos. Essa evolução permitiria transformar o

processo de construção de software de qualidade em um processo de predição.

Agradecimentos. Os autores agradecem o apoio financeiro do CNPq e da FAPEAM para a realização deste trabalho e a colaboração da BENQ-SIEMENS durante os estudos de ArqCheck.

Referências

- ACM (2006). "Association for Computing Machinery (ACM) Digital Library". In: <http://www.acm.org>, acessado em 01/12/2005.
- Antman E., Lau, J., Kupelnick, B., Mosteller, F., Chalmers, T. (1992) "A comparison of results of meta-analysis of randomized controlled trials and recommendations of clinical experts", JAMA, 268(2):240-248, July 1992.
- Barcelos, R. (2006) "Uma Abordagem para Inspeção de Documentos Arquiteturais Baseada em Checklist". Dissertação de Mestrado, Programa de Engenharia de Sistemas e Computação, COPPE/UFRJ, Rio de Janeiro, RJ, Brasil.
- Barcelos, R., Travassos, G. (2006a) "ArqCheck: uma Abordagem para Inspeção de Documentos Arquiteturais Baseada em Checklist". In: V Simpósio Brasileiro de Qualidade de Software, Vila Velha, ES, Brasil.
- Barcelos, R., Travassos, G. (2006b) "Evaluation Approaches for Software Architectural Documents: a Systematic Review". In: Proceedings of IDEAS'06, La Plata, Argentina.
- Belgamo, A., Fabbri, S. (2005) "GUCCRA: Técnica de Leitura para apoiar a Construção Modelos de Casos de Uso e a Análise de Documentos de Requisitos" In: XIX SBES, Uberlândia, MG, Brasil.
- Berling, T., Thelin, T. (2004) "A Case Study of Reading Techniques in a Software Company", 2004 International Symposium on Empirical Software Engineering (ISESE'04), August 19 - 20, 2004, Redondo Beach, California, pp. 229-238.
- Biffi, S., Halling, M. (2003) "Investigating the defect detection effectiveness and cost benefit of nominal inspection teams", IEEE Trans. on Soft. Eng., Vol.: 29:5, May Pages: 385-397.
- Biolchini, J., Mian, P.G., Natali, A.C., Travassos, G.H. (2005) "Systematic Review in Software Engineering: Relevance and Utility", Technical Report ES-679/05, PESC - COPPE/UFRJ, Rio de Janeiro, RJ.
- Chrissis, M., Konrad, M., Shrum, S. (2003) "CMMI: Guidelines for Process Integration and Product Improvement", Addison Wesley.
- Cochrane, Al. (1989) In Chalmers I, Enkin M, Keirse MJNC, eds. "Effective care in pregnancy and childbirth". Oxford University Press, Oxford, 1989.
- Cochrane Collaboration (2003), Cochrane Reviewers' Handbook. Version 4.2.1. <http://www.cochrane.dk/cochrane/handbook/hbook.htm>, acessado em 04/01/2006.
- Conradi, R., Mohagheghi, P., Arif, T. (2003) "Object-Oriented Reading Techniques for Inspection of UML Models – An Industrial Experiment". In: Proc. of the European Conference on Object-Oriented Programming (ECOOP 03), pp. 483-500, July.
- Denger, C., Ciolkowski, M., Lanubile, F. (2004) "Investigating the Active Guidance Factor in Reading Techniques for Defect Detection", Proc. of ISESE'04, Redondo Beach, California.
- Juristo, N., Moreno, A. (2002) "Reliable Knowledge for Software Development", IEEE Software, pp. 98-99, sep-oct, 2002.
- Kaindl, H., Brinkkemper, S., Bubenko, J., Farbey, B., Greenspan, S., Heitmeyer, C., Leite, J.C.S.P., Mead, N., Mylopoulos, J., Siddiqi, J. (2002) "Requirements Engineering and Technology Transfer: Obstacles, Incentives and Improvement Agenda". Requirements Engineering, vol.7, pp. 113-123, 2002.
- Kalinowski, M, Travassos, G. (2004) "A Computational Framework for Supporting Software Inspections". In: Proc. of the 19th IEEE Conference on Automated Software Engineering - ASE'04, pp. 46-55, Linz, Austria.
- Kalinowski, M, Travassos, G. (2005) "Software Technologies: The Use of Experimentation to Introduce ISPIS – a Software Inspection Framework – Into the Industry". In: 2nd ESELAW, Uberlândia, MG, Brazil.
- Kitchenham, B., Dybå, T., Jorgensen, M. (2004) "Evidence-based Software Engineering", Proceedings of the 26th International Conference on Software Engineering (ICSE'04).

- Lima, G. (2005) “Heurísticas para Identificação da Ordem de Integração de Classes em Testes Aplicados a Software Orientado a Objetos”. Dissertação de Mestrado, Programa de Engenharia de Sistemas e Computação, COPPE/UFRJ, Rio de Janeiro, RJ, Brasil.
- Mafra, S. N., Travassos, G. H. (2005) “Técnicas de Leitura de Software: Uma Revisão Sistemática”. In: XIX SBES, Uberlândia, MG, Brasil.
- Mafra, S. N. (2006) “Definição de uma Técnica de Leitura Baseada em Perspectiva (OO-PBR) Apoiada por Estudos Experimentais”. Dissertação de Mestrado, PESC, COPPE/UFRJ, Rio de Janeiro, RJ, Brasil.
- Mafra, S., Travassos, G. (2006a) “Leitura Baseada em Perspectiva: A Visão do Projetista Orientada a Objetos”. In: V SBQS, Vila Velha, ES, Brasil.
- Mafra e Travassos (2006b) “Estudos Primários e Secundários Apoiando a Busca por Evidência em Engenharia de Software”. Relatório Técnico ES-687/06, Programa de Engenharia de Sistemas e Computação, COPPE/UFRJ, Rio de Janeiro, RJ, Brasil.
- MCT/SEPIN (2005) “Qualidade e Produtividade no Setor de Software”. In: www.mct.gov.br/sepin/Dsi/Software/Menu_Qualidade.htm Acessado em Fevereiro de 2006.
- Mian, P., Chapetta, W., Santos, P.S., Melo Jr., C., Natali, A.C.C., Biolchini, J., Rocha, A., Travassos, G. (2005) “eSEE: an Infrastructure for Supporting Experimental Software Engineering”. In: Proceedings the 4th IEEE/ACM International Symposium on Empirical Software Engineering (ISESE) - Late Breaking Paper, Australia, November.
- Melo, W., Shull, F., Travassos, G. (2001) “Software Review Guidelines”. Technical Report ES-556/01, PESC - COPPE/UFRJ.
- NHS Centre for Reviews and Dissemination. (2003), “Database of Abstracts of Reviews of Effectiveness”. In: The Cochrane Library, Issue 1. Oxford: Updated quarterly.
- Nunez, L. (2005) “Apoio Automatizado para Aplicação de Técnicas de Leitura Orientada a Objetos (OORTs)”. Dissertação de Mestrado, Programa de Engenharia de Sistemas e Computação, COPPE/UFRJ, Rio de Janeiro, RJ, Brasil.
- PMP (2000) “Project Management Professional: Credential Handbook”. Disponível em http://www.pmi.org/info/PDC_PMP.asp. Acessado em 27/03/2006.
- Shull, F. (1998) “Developing Techniques for Using Software Documents: A Series of Empirical Studies”, PhD Thesis, Depart. of Computer Science, Univ. of Maryland, USA.
- Shull, F., Carver, J., Travassos, G. (2001) “An Empirical Methodology for Introducing Software Processes”, In: Proceedings of the Joint 8th European Software Engineering Conference (ESEC) and 9th ACM SIGSOFT Symposium on the Foundations of Software Engineering (FSE-9), pp. 288-296.
- Shull, F., Mendonça, M., Basili, V., Carver, J., Maldonado, J., Fabbri, S., Travassos, G., Ferreira, M. (2004) “Knowledge-Sharing Issues in Experimental Software Engineering”, Empirical Software Engineering, Volume 9 Issue 1-2, March, 2004.
- Silva, L., F. (2004) “Uma abordagem com apoio ferramental para aplicação de técnicas de leitura baseada em perspectiva”. Dissertação de Mestrado, Programa de Engenharia de Sistemas e Computação, COPPE/UFRJ, Rio de Janeiro, RJ, Brasil.
- Travassos, G., Shull, F., Fredericks, M., Basili, V. (1999) “Detecting defects in object-oriented designs: using reading techniques to increase software quality”. Proc. of the OOPSLA’99, Volume 34 Issue 10, October 1999.
- Weber, K., Rocha, A. R., *et al.* (2004) “Modelo de Referência para Melhoria de Processo de Software: uma abordagem brasileira”. XXX Conf. Latino-americana de Informática, Arequipa - Peru, 2004.
- Wöhlin, C., Runeson, P., Höst, M., Ohlsson, M., Regnell, B., Wesslén, A. (2000) “Experimentation in Software Engineering: An Introduction”, The Kluwer International Series in Software Engineering, Norwell, USA, Kluwer Academic Publishers.