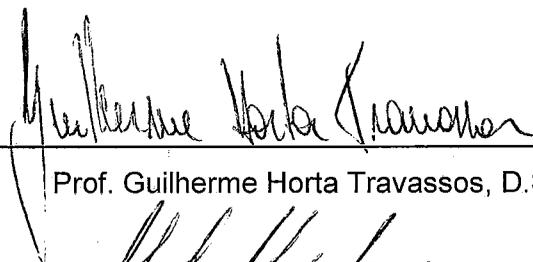


UMA INFRA-ESTRUTURA PARA PLANEJAMENTO, EXECUÇÃO E
EMPACOTAMENTO DE ESTUDOS EXPERIMENTAIS EM ENGENHARIA DE
SOFTWARE

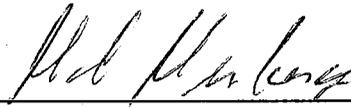
Wladimir Araújo Chapetta

DISSERTAÇÃO SUBMETIDA AO CORPO DOCENTE DA COORDENAÇÃO DOS
PROGRAMAS DE PÓS-GRADUAÇÃO DE ENGENHARIA DA UNIVERSIDADE
FEDERAL DO RIO DE JANEIRO COMO PARTE DOS REQUISITOS NECESSÁRIOS
PARA A OBTENÇÃO DO GRAU DE MESTRE EM CIÊNCIAS EM ENGENHARIA DE
SISTEMAS E COMPUTAÇÃO.

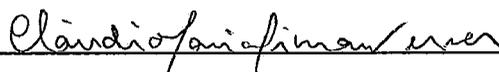
Aprovada por:



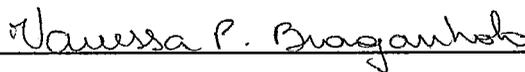
Prof. Guilherme Horta Travassos, D.Sc.



Prof. Manoel Gomes de Mendonça Neto, Ph.D.



Profa. Claudia Maria Lima Werner, D.Sc.



Profa. Vanessa de Paula Braganholo, D.Sc.

RIO DE JANEIRO, RJ – BRASIL

SETEMBRO DE 2006

CHAPETTA, WLADMIR ARAÚJO

Uma Infra-Estrutura para Planejamento
Execução e Empacotamento de Estudos
Experimentais em Engenharia de Software
[Rio de Janeiro] 2006

IX, 142 p., 29,7 cm (COPPE/UFRJ, M.Sc.,
Engenharia de Sistemas e Computação, 2006)

Dissertação - Universidade Federal do
Rio de Janeiro, COPPE

1. Experimentação em Engenharia de
Software

I. COPPE/UFRJ II. Título (série)

A família e amigos.

Agradecimentos

Aos meus pais, Walcir Chapetta e Maria das Graças Araújo Chapetta, por respeitarem minhas decisões, mesmo quando estas desafiavam seus ideais do que acreditavam ser o melhor para mim.

Aos meus irmãos, Tiago Araújo Chapetta e Rômulo Araújo Chapetta, eternos companheiros.

À Nilma Santos Masson, o ser humano mais generoso, companheiro e bom que conheci e conhecerei nesta vida.

À Mariana Vianna Vasconcelos por me trazer a perspectiva e desejo de ser e fazê-la feliz.

Ao André Luiz Carvalho Chapetta, por sua compreensão e carinho em um período muito difícil e conturbado de minha vida.

Aos irmãos da graduação, Claudia, Fábio, Leonardo, Ricardo, Rafael pelo carinho, preocupação, atenção e, principalmente, por todos os momentos que vivemos juntos.

Aos meus amigos do CEFET-RJ, em especial Pedro, Theodor, Michel, Cajú, Arthur, Tatá, Nit, Max, Siri, Cléber e Petra. Devo em parte a vocês toda a alegria e bondade que existem em meu ser. Este agradecimento é uma homenagem do ser humano que vocês ajudaram a construir.

Ao Luís Felipe Santos Silva e esposa, pelo apoio, carinho e amizade.

Ao amigo Guilherme Horta Travassos pela paciência, carinho e generosidade no período mais difícil da minha vida.

Ao meu orientador Guilherme Horta Travassos, cujos estímulos foram essenciais para conclusão deste trabalho.

Ao Prof. Guilherme Horta Travassos, pela sua confiança depositada em todas as empreitadas realizadas juntos.

À Equipe de Engenharia de Software Experimental COPPE/UFRJ, em especial a Paulo Sérgio, Rafael, Rosa e Ana, pela agradável companhia e amizade.

Aos professores Manoel Mendonça, Claudia Werner e Vanessa Braganholo por participarem da banca de defesa deste trabalho, mas, principalmente, pela compreensão e paciência dado os acontecimentos que se seguiram neste ano.

À CAPES pelo apoio financeiro.

Resumo da Dissertação apresentada à COPPE/UFRJ como parte dos requisitos necessários para a obtenção do grau de Mestre em Ciências (M.Sc.)

UMA INFRA-ESTRUTURA PARA PLANEJAMENTO, EXECUÇÃO E
EMPACOTAMENTO DE ESTUDOS EXPERIMENTAIS EM ENGENHARIA DE
SOFTWARE

Wladimir Araújo Chapetta

Setembro / 2006

Orientador: Guilherme Horta Travassos

Programa: Engenharia de Sistemas e Computação

Estudos experimentais vêm sendo utilizados como meio de adquirir conhecimento através de uma metodologia científica baseada na observação e medição de fenômenos em diferentes áreas da Engenharia de Software. Seus benefícios estão relacionados à obtenção de resultados que justifiquem o uso ou não de uma tecnologia baseada em algum indicativo de que esta tecnologia possa contribuir para a melhoria de qualidade de produto ou processo de software. Deste modo, resultados de estudos experimentais executados em diferentes cenários do processo de desenvolvimento podem ser utilizados como pontos de partida para definir critérios e apoiar o processo de tomada de decisão.

Entretanto, executar tais estudos é difícil tanto em ambientes acadêmicos quanto em industriais. Muitas são as variáveis que precisam ser controladas ou monitoradas, e os recursos para executá-los na maioria das vezes são escassos. Além disso, em geral há uma falta de representatividade da população nas amostras utilizadas, o que implica em várias repetições de um mesmo estudo. Para permitir essas repetições, os artefatos produzidos ao longo da preparação, execução e obtenção dos resultados de um estudo experimental devem ser empacotados. De forma a melhorar o monitoramento e controle de alguns desses problemas, esta dissertação apresenta a pesquisa, o desenvolvimento e os estudos experimentais executados na avaliação de uma infra-estrutura de apoio ao planejamento, execução, empacotamento e repetição de estudos experimentais em Engenharia de Software.

Abstract of Dissertation presented to COPPE/UFRJ as a partial fulfillment of the requirements for the degree of Master of Science (M.Sc.)

AN FRAMEWORK FOR SOFTWARE ENGINEERING EXPERIMENTAL STUDIES
PLANNING, OPERATION AND PACKING

Wladimir Araújo Chapetta

September / 2006

Advisor: Guilherme Horta Travassos

Department: Computer and Systems Engineering

Experimental studies have been used as a mechanism to acquire knowledge through a scientific methodology based on measurement of phenomena in different Engineering Software areas. Its benefits are related to attainment of results that justify the use or not of a technology based on some indicative that this technology can contribute for the software product or process quality improvement. Thus, the results of experimental studies executed in the different development process scenarios can be used as starting points to define criteria and support the decision making process.

However, to execute such studies is difficult in academic and industrial environments. In general, there are many variables that need be controlled or monitored, and the resources to execute them are scarce. Moreover, there is a lack of representative population used in the samples, which implies in some study replications. In this way, the produced artifacts throughout the preparation, execution and results of an experimental study must be packaged to allow these replications. As a mechanism to improve the monitoring and controlling of some of these problems, this dissertation presents the research, development and evaluation experimental studies of an infrastructure to support the Software Engineering experimental studies planning, operation, packaging and replication.

Índice:

Agradecimentos	iv
Índice:.....	vii
Capítulo 1 - Introdução.....	10
1.1 – Motivação.....	10
1.2 – Objetivos	11
1.3 – Contexto: <i>experimental Software Engineering Enviroment</i>	12
1.3 – Organização.....	13
Capítulo 2 – Experimentação em Engenharia de Software	14
2.1 - Introdução	14
2.2 - Métodos de Pesquisa aplicados à Experimentação em Engenharia de Software	16
2.3 - Definição dos Conceitos Relacionados à Execução de Estudos Experimentais em Engenharia de Software.....	17
2.4 - Taxonomias aplicadas a estudos experimentais em Engenharia de Software	20
2.5 - O Processo de Experimentação em Engenharia de Software.....	29
2.5.1 – Etapa de Definição	31
2.5.2 – Etapa de Planejamento.....	33
2.5.3 – Etapa de Operação	38
2.5.4 – Etapa de Análise e Interpretação.....	40
2.5.5 – Etapa de Empacotamento.....	40
2.6 - Abordagens Relacionadas à Experimentação.....	41
2.7 – Conclusão	47
Capítulo 3 – Infra-Estrutura para Planejamento, Execução e Empacotamento de Estudos Experimentais em Engenharia de Software.....	49
3.1 – Motivação.....	49
3.2 – Escopo do Trabalho	50
3.3 – Descrição do Processo de Apoio ao Planejamento e Execução de Estudos Experimentais em Engenharia de Software.....	51
3.3.1 – Definir Tipo de Execução	52
3.3.2 – Registrar Documentos de Definição e Planejamento.....	53
3.3.3 – Registrar Instrumentos.....	54
3.3.4 – Definir Tratamentos.....	55

3.3.5 – Avaliar Estudo	56
3.3.6 – Obter Consentimento e Caracterização dos Participantes	60
3.3.7 – Disponibilizar Treinamento.....	61
3.3.8 – Definir Distribuição	62
3.3.9 – Distribuir Material de Execução.....	63
3.3.10 – Coletar Dados	64
3.3.11 – Registrar Documentos da Análise.....	65
3.3.12 – Diagrama de Atividades do Processo Proposto.....	67
3.4 – Requisitos da Infra-Estrutura para Apoiar o Planejamento Execução e Repetição de Estudos Experimentais em Engenharia de Software.....	68
3.5 – Arquitetura.....	70
3.6 – Protótipo da Infra-estrutura	71
3.6.1 – Descrição Geral.....	71
3.6.2 – Protótipo na Perspectiva do Engenheiro de Software.....	72
3.6.3 – Protótipo na Perspectiva do Pesquisador	74
3.6.4 – Protótipo na Perspectiva do Revisor Externo.....	85
3.6.5 – Protótipo na Perspectiva do Participante	86
3.6.6 – Mecanismos de Gerência de Conhecimento na Infra-estrutura....	89
3.7 - Conclusão	90
Capítulo 4 – Estudos Experimentais	92
4.1 – Introdução	92
4.2 – 1º Estudo Experimental: Viabilidade de Empacotamento.....	93
4.2.1 – Definição dos Objetivos.....	93
4.2.2 – Planejamento do Estudo	94
4.2.3 – Execução do Estudo	96
4.2.4 – Análise de Resultados.....	97
4.2.5 – Lições Aprendidas.....	98
4.3 – 2º Estudo de Viabilidade: Execução de um Estudo Real	98
4.3.1 – Definição dos Objetivos.....	98
4.3.2 – Planejamento do Estudo	99
4.3.3 – Execução do Estudo	101
4.3.4 – Análise de Resultados.....	102
4.3.5 – Lições Aprendidas.....	102
4.4 – Lições aprendidas dos Estudos Executados	103
4.5 – Conclusão	104
Capítulo 5 – Considerações finais	105
5.1 – Conclusões	105

5.2 – Contribuições	105
5.3 – Limitações	106
5.4 – Perspectivas futuras.....	106
Referências Bibliográficas.....	109
ANEXO I.....	118
ANEXO II.....	127

Capítulo 1 - Introdução

Neste capítulo são apresentadas as principais motivações e os objetivos que levaram à realização deste trabalho, juntamente com a forma como está organizado o texto desta dissertação.

1.1 – Motivação

O método (ou processo) científico é fundamental na investigação e aquisição de conhecimento baseado em evidências, onde cientistas buscam propor explicações para determinados fenômenos naturais através de observações (medições). Estas explicações são chamadas hipóteses. A partir destas hipóteses, modelos de predição para a ocorrência destes fenômenos são gerados e testados através de estudos experimentais. O progresso em qualquer disciplina científica envolve a construção de modelos que possam ser testados através de estudos experimentais (BASILI *et al.*, 1999). Deste modo, realizar estudos experimentais provê um meio de testar, refinar e/ou refutar a compreensão sobre um determinado fenômeno. A partir desta compreensão, este fenômeno deve poder ser descrito e/ou repetido sob determinadas condições.

Na Engenharia de Software, observa-se ao longo dos últimos anos o crescente interesse em executar estudos experimentais para avaliar o uso de novas tecnologias (métodos, técnicas, linguagens ferramentas, etc) nos diferentes processos de software. Segundo PFLEEGER (1999), a execução de estudos experimentais ajudaria a entender as chances que, sob certas condições, uma tecnologia conduzirá o aprimoramento de alguma característica do software. A execução de estudos experimentais também poderia auxiliar no entendimento ou identificação da influência da interferência humana na execução de tarefas. De acordo com AMARAL (2003), a Experimentação oferece um modo sistemático, disciplinado, computável e controlado para avaliação de novas tecnologias, já que intuição, opinião e especulação não podem ser consideradas fontes confiáveis de conhecimento. Deste modo tecnologias não deveriam ser apenas sugeridas, publicadas ou apresentadas sem experimentação e avaliação.

Entretanto, a pesquisa experimental é uma atividade complexa e consome tempo, especialmente em Engenharia de Software. Já que, diferentemente de outros meios de produção, desenvolvedores de software não produzem sempre os mesmos

produtos. O software é produzido e cada produto é diferente do anterior (BASILI *et al.*, 1999).

A execução de estudos experimentais em Engenharia de Software gera uma grande quantidade de informações e artefatos. Além disso, por ser a Engenharia de Software altamente impactada pelo fator humano, existem outras inúmeras variáveis que tornam mais crítica a execução de estudos tanto no ambiente acadêmico quanto no industrial. Por exemplo, do ponto de vista experimental podemos ressaltar a representatividade do número de participantes em uma amostra da população. Alguns estudos executados não possuem um número de participantes que garanta a representatividade de uma população de desenvolvedores de software com um intervalo de confiança estatística adequado. Há também a questão da aleatoriedade, pois tanto na academia quanto na indústria, os pesquisadores não contam com a possibilidade de selecionar indivíduos aleatoriamente dentro de uma população. Estes participantes são geralmente selecionados por conveniência, mais especificamente, por estarem participando de algum treinamento, curso, projeto, etc. Do ponto de vista organizacional, fica difícil alinhar a execução de um estudo experimental aos objetivos e tarefas cotidianas da academia ou indústria. Por exemplo, é um risco para um projeto, acadêmico ou industrial, alocar desenvolvedores, tempo e/ou equipamentos para executar um estudo se este está fora de seu escopo. Ratificando essa idéia, SILVA & TRAVASSOS (2004) atentam para a necessidade de alinhar os objetivos de um estudo ao conteúdo, o objetivo e cronograma de uma disciplina acadêmica, para que se possa utilizar estudantes como participantes. A ausência de informações sobre esses e outros fatores dificultam a execução de um estudo e se não forem bem gerenciados podem constituir um risco à sua repetibilidade e, conseqüentemente, à validade de seus resultados.

A partir da experiência na execução de estudos *in-vitro* (de laboratório) (TRAVASSOS & BARROS, 2003) junto a Equipe de Engenharia de Software Experimental da COPPE/UFRJ e na tentativa de tornar alguns dos riscos que envolvem a execução destes estudos mais gerenciáveis, este trabalho explicita parte do conhecimento adquirido, explorando a infra-estrutura *Web* como uma abordagem para apoiar a execução e o empacotamento de estudos experimentais em Engenharia de Software.

1.2 – Objetivos

O objetivo deste trabalho é explorar a infra-estrutura *Web* para apoiar o planejamento, execução, empacotamento e repetição de estudos *in-vitro*, através da definição de um *workflow* de execução de estudos experimentais em Engenharia de

Software. Ao utilizarmos a *Web*, espera-se que esta infra-estrutura possa: (1) atuar como apoio para uma base de conhecimento de estudos experimentais que permite seu acesso em diferentes localidades; (2) permitir maior visibilidade dos estudos executados e maior colaboração entre pesquisadores, e; (3) permitir pesquisa inter-institucional.

SHULL *et al.* (2004) destacam a necessidade e as dificuldades em se repetir estudos experimentais e argumentam que o uso de pacotes de laboratório constitui uma medida facilitadora. Além disso, definem alguns objetivos que estes pacotes devem apoiar, visando contribuir para um repositório de conhecimento experimental, dentre eles:

- (1) Facilitar o acesso aos artefatos experimentais;
- (2) Explicitar o processo de execução do experimento, e;
- (3) Permitir sua própria evolução (ou seja, dos artefatos que constituem o pacote).

Em AMARAL (2003), são definidos um pacote de laboratório (ou Pacote de Experimento) e um Processo de Experimentação (WOHLIN *et al.*, 2000) para estudos sobre Inspeção de Software. O processo proposto por AMARAL, (2003), está fortemente focado em guiar o pesquisador no registro das informações e artefatos, ou seja, no empacotamento de um estudo experimental. Entretanto, o processo proposto não dá apoio ou guia o pesquisador nas tarefas do Processo de Experimentação. Deste modo, a proposta deste trabalho é de certa forma formalizar e disponibilizar para pesquisadores da comunidade de Engenharia de Software o conhecimento sobre a execução de estudos experimentais obtido pela Equipe de Engenharia de Software Experimental da COPPE/UFRJ através da definição de um processo, implementado numa infra-estrutura baseada na *Web* através da tecnologia de *workflow*.

1.3 – Contexto: experimental Software Engineering Environment

Este trabalho está dentro do contexto de um ambiente de apoio ao Processo de Experimentação, denominado eSEE (*experimental Software Engineering Environment*), um ambiente que têm por objetivo fornecer apoio aos diferentes tipos de estudos experimentais em Engenharia de Software através da instanciação de Processos de Experimentação (CHAPETTA *et al.*, 2004; DIAS NETO *et al.*, 2004; MIAN *et al.*, 2004; CHAPETTA *et al.*, 2005; MIAN *et al.*, 2005a; MIAN *et al.*, 2005b). A infra-estrutura proposta poderá ser utilizada de forma integrada ao eSEE apoiando a execução de estudos experimentais, ou de forma independente como uma aplicação *Web* que pode ser acessada via *browser*.

No ANEXO I é apresentada a arquitetura básica do Meta-Ambiente eSEE, os conceitos pertinentes a ela, bem como a abordagem de construção que vêm sendo utilizada no eSEE.

Neste trabalho, será explorada uma aplicação *Web* baseada em *workflow* e encapsulada em um *Web service* para apoiar o planejamento de Processo de Experimentação em Engenharia de Software proposto por AMARAL (2003). Esta aplicação segue a definição de um processo específico e terá algumas de suas funcionalidades disponíveis para serem utilizadas como uma ferramenta do Meta-Ambiente eSEE. Com isso espera-se ter uma prova de conceito próxima de uma situação real do que seria executar um Processo de Experimentação apoiado pela infra-estrutura disponibilizada pelo eSEE.

Acreditamos que a elaboração desta infra-estrutura permita observar aspectos iniciais que levem, num futuro próximo, a organização de ambientes de Engenharia de Software para apoiar a Experimentação em Larga Escala (MATTOSO & TRAVASSOS, 2006).

1.3 – Organização

Além desta introdução, esta dissertação possui mais 4 capítulos.

No próximo capítulo será discutido Experimentação em Engenharia de Software, enfatizando seus principais conceitos e aspectos do Processo de Experimentação. No capítulo 3, é apresentada a definição de um processo, os requisitos, a arquitetura e um protótipo construído de uma infra-estrutura de apoio ao planejamento, execução e empacotamento de estudos experimentais em Engenharia de Software. No capítulo 4, são apresentadas as avaliações experimentais utilizadas para verificar as idéias apresentadas nos capítulos 3, apresentando detalhes do planejamento, resultados e observações pertinentes. E por fim uma conclusão, com contribuições e perspectivas futuras.

Capítulo 2 – Experimentação em Engenharia de Software

Neste capítulo são apresentados os principais conceitos relativos à Experimentação na Engenharia de Software.

2.1 - Introdução

Pesquisadores de Engenharia de Software buscam melhores formas de desenvolver e avaliar software. Eles são motivados pelos problemas práticos, e seus objetivos chave de pesquisa são qualidade, custos e prazos relacionados aos produtos de software (SHAW, 2002). Geralmente Engenheiros de Software apresentam demonstrações de suas tecnologias como uma forma de avaliação. Demonstrações podem prover provas de conceitos ou incentivos para se estudar mais de uma questão. Entretanto, demonstrações meramente ilustram um potencial, pois dependem da imaginação e voluntariedade dos observadores. Raramente demonstrações produzem uma evidência sólida (TICHY, 1998).

Assim, para contornar essa falta de evidências e solidificar um corpo de conhecimento sobre as tecnologias propostas não somente baseado em intuições, julgamentos ou opiniões, Engenheiros de Software deveriam observar fenômenos, formular teorias e testá-las. Experimentos são testes de uma determinada teoria. Entretanto, muitos pesquisadores e profissionais da prática da Engenharia de Software alegam não utilizarem experimentos para avaliar suas tecnologias, argumentando que realizar um estudo experimental é inapropriado, muito difícil, sem utilidade, caro e até mesmo prejudicial.

Estudos experimentais realmente são caros e difíceis de executar, pois impõem o uso de recursos adicionais nem sempre facilmente disponíveis em uma organização, tais como desenvolvedores, tempo e recursos de hardware/software. Mas executá-los é complexo em qualquer disciplina científica. Logo, pesquisadores da Comunidade de Engenharia de Software não possuem justificativa para avaliar uma percentagem de suas propostas muito menor do que em outras áreas, como Física, Medicina, Administração e Psicologia. Em artigos selecionados em um estudo apresentado por ZELKOWITZ & WALLACE (1998) demonstra-se que algo entre 50% deles não utilizavam nenhum método de avaliação criteriosa das tecnologia propostas. Em SJØBERG *et al.* (2005) é realizado um *survey* sobre os experimentos executados

entre 1993 e 2002 em 9 periódicos e 3 anais de conferências com relativa importância na Comunidade de pesquisadores da Engenharia de Software. Neste estudo foram avaliados 5.453 artigos e somente 1.9% dos artigos haviam relatado o uso de experimentos para avaliar suas propostas. A principal diferença entre os resultados obtidos em SJØBERG *et al.* (2005) e ZELKOWITZ & WALLACE (1998) é que o primeiro e mais recente observa somente o uso de experimentos controlados, enquanto o segundo utiliza uma definição mais ampla de estudos experimentais na seleção de artigos, diferindo em propósitos, critérios de seleção e taxonomias.

Experimentação representa o centro do método científico. Se o assunto dos questionamentos são os processos de transformação da informação (no caso da Engenharia de Software) ou da energia (como na Física) não faz nenhuma diferença quanto à aplicabilidade do método científico (TICHY, 1998). Se outras disciplinas podem se beneficiar com a execução de estudos experimentais, por que a Engenharia de Software não pode?

De acordo com PFLEEGER (1999), nenhuma ciência pode avançar sem experimentação e medição. Experimentação em Engenharia de Software é necessária. Experiência, intuição, especulação e provas de conceitos não são fontes confiáveis de conhecimento. Pelo contrário, progresso em qualquer disciplina científica envolve a construção de modelos que podem ser testados, através de um estudo experimental, para verificar se a atual compreensão de um campo é correto (BASILI *et al.*, 1999).

Deste modo, observa-se uma crescente compreensão na Comunidade de Engenharia de Software que estudos experimentais são necessários para se desenvolver e aprimorar processos, métodos e ferramentas para o desenvolvimento e manutenção de software (SJØBERG *et al.*, 2005). Outro aspecto importante é que estudos experimentais têm sido utilizados para avaliar o comportamento humano em áreas multidisciplinares como Ciências Sociais. Neste contexto, podemos justificar o uso de Experimentação como meio de avaliar a influência do fator humano nos processos de software, por ser a Engenharia de Software intensamente influenciada pelo fator humano (WOOD *et al.*, 1999; WOHLIN *et al.*, 2000).

Neste capítulo são discutidos os métodos de pesquisa aplicados à Experimentação e apresentadas algumas taxonomias de classificação de estudos experimentais com seus respectivos exemplos. Também são apresentados o Processo de Experimentação (WOHLIN *et al.*, 2000; AMARAL, 2003) e conceitos relacionados à execução de experimentos em Engenharia de Software. Por fim, são mostrados outros tópicos relacionados à pesquisa experimental na Engenharia de Software, tais como o

uso de estudantes como participantes, transferência de tecnologia da academia para indústria, dentre outros.

2.2 - Métodos de Pesquisa aplicados à Experimentação em Engenharia de Software

Dentre os métodos de pesquisa aplicados em estudos experimentais em Engenharia de Software temos a pesquisa qualitativa e quantitativa.

Pesquisa qualitativa tem diferentes significados em diferentes campos, nas Ciências Sociais está o mais bem conhecido. Neste contexto, é um termo amplo que descreve pesquisa que foca em como indivíduos e grupos visualizam e compreendem o mundo e constroem significado a partir de suas experiências; é essencialmente orientada à narrativa. Em estatística, pesquisa qualitativa consiste em procedimentos que usam somente dados dicotômicos - isto é, dados que possuem valores 0 (zero) e 1 (um), utilizados quando eventos ou entidades podem ser somente contados ou classificados, e não medidas (TAYLOR & BOGDAN, 1998).

Pesquisa qualitativa na Engenharia de Software geralmente possui o mesmo significado das Ciências Sociais, buscando focar na compreensão do fenômeno de pesquisa em seu contexto de natural ocorrência. De um modo geral, as principais técnicas aplicadas na Engenharia de Software são a observação de participantes, entrevistas e análise de artefatos.

A **Pesquisa quantitativa** é baseada na representação numérica de observações com propósito de descrever e explicar um fenômeno. É utilizada em ciências naturais e sociais. A pesquisa quantitativa inicia-se com a coleta de dados, seguida da aplicação de vários métodos estatísticos descritivos e de inferência (TAYLOR & BOGDAN, 1998).

A Pesquisa qualitativa pode ser utilizada junto com a pesquisa quantitativa para se alcançar uma compreensão mais profunda sobre as causas de um fenômeno social ou ajudar a criar novas questões de pesquisas. É possível investigar o mesmo tópico com pesquisa quantitativa e qualitativa, porém cada uma delas irá abordar um tipo diferente de questão. Por exemplo, uma investigação quantitativa poderia ser realizada com o propósito de investigar o quanto um novo método de inspeção diminui o número de falhas encontradas no teste. A fim de responder as questões sobre as fontes de variação entre diferentes grupos de inspeção, uma pesquisa qualitativa deveria ser realizada. Ou seja, a pesquisa quantitativa é apropriada quando se deseja testar os efeitos de variações em uma variável, enquanto uma pesquisa qualitativa é apropriada

para descobrir porque os resultados de uma pesquisa quantitativa são como eles se apresentam (AMARAL, 2003).

Estes dois tipos de pesquisa devem ser considerados como complementares (WOHLIN *et al.*, 2000). Pesquisa qualitativa é geralmente considerada exploratória e indutiva, enquanto pesquisa quantitativa é geralmente confirmatória e dedutiva.

O uso de uma abordagem multi-método (isto é, utilizar o método de pesquisa quantitativo e qualitativo conjuntamente,) provê benefícios em termos de conclusões mais robustas, desenvolvimento e investigação de hipóteses de pesquisa de um modo evolutivo, e uma maior compreensão dos resultados encontrados (WOOD *et al.*, 1999).

Em WOOD *et al.* (1999) demonstra-se a aplicação de uma abordagem multi-método na investigação sobre os benefícios da orientação a objetos, combinando entrevistas estruturadas com desenvolvedores, a aplicação de uma pesquisa de opinião e execução de experimentos controlados.

2.3 - Definição dos Conceitos Relacionados à Execução de Estudos Experimentais em Engenharia de Software.

Pesquisa experimental em Engenharia de Software deveria almejar adquirir conhecimento sobre **que tecnologia é útil para quem realizar que tarefas em quais ambientes** (SJØBERG *et al.*, 2005). Como pesquisadores, devemos aplicar métodos investigativos para obter uma melhor compreensão do que torna bom e como construir bem um software (PFLEEGER, 1999).

Segundo BASILI *et al.* (1999), um **estudo experimental** representa um ato ou operação para se descobrir algo desconhecido ou testar uma hipótese, envolvendo a coleta e análise de dados pelo pesquisador, para que este dê significado aos dados obtidos.

Em grande parte da literatura, os termos estudos experimentais e experimentos são utilizados intercambiavelmente como tendo o mesmo significado. No contexto dessa dissertação nem todo estudo experimental é um experimento. Como veremos detalhadamente na próxima seção, são denominados experimentos os experimentos controlados e os *quasi*-experimentos (DALY *et al.*, 1997; WOHLIN *et al.*, 2000; AMARAL, 2003; SJØBERG *et al.*, 2005).

Um experimento é um teste de uma teoria sobre o funcionamento de um sistema ou processo, onde há variáveis de entrada e saída. O propósito de um experimento é provocar alterações controladas e propositalmente nas variáveis de entrada para se verificar as alterações nas variáveis de saídas (Figura 2.1).

Uma **teoria** é uma possível explicação de algum fenômeno. Qualquer teoria é composta por um conjunto de hipóteses.

De acordo com PFLIEGER (2001a appud AMARAL,2003), uma **hipótese** é uma tentativa teórica ou uma suposição do que se pensa, e que explica o comportamento do que se pretende explorar. Segundo PERRY *et al.* (2000 appud AMARAL,2003) as **hipóteses** definem as questões de pesquisa que esta procura responder. As **hipóteses abstratas** são declarações em linguagem natural geralmente utilizadas nos termos diários, como por exemplo: “reunião é uma parte indispensável no processo de inspeção”. As **hipóteses concretas** são utilizadas em termos do projeto de estudo, como por exemplo: “equipes que aplicam inspeções com reuniões encontram mais defeitos do que equipes que aplicam inspeção sem reuniões”.

A hipótese principal de um experimento se chama **hipótese nula** e declara que não há nenhum relacionamento estatisticamente significante entre a causa e o efeito. O objetivo principal do experimento é, então, rejeitar a hipótese nula.

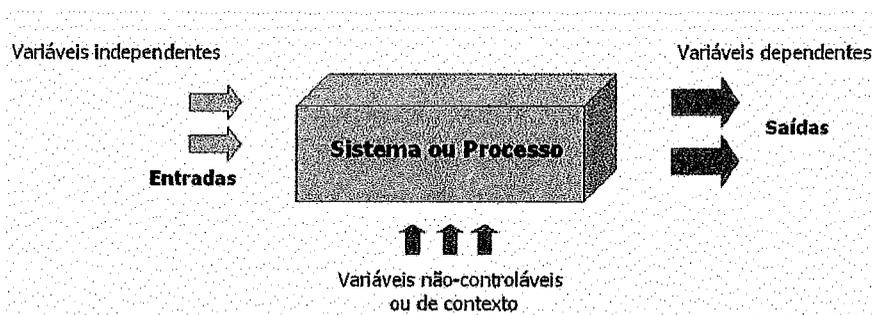


Figura 2.1- Percepção geral sobre um experimento

Em um experimento, as variáveis de entrada são denominadas **variáveis independentes** ou **fatores**. São estas as variáveis sobre as quais o pesquisador tem controle, ou seja, as variáveis utilizadas para provocar as perturbações no sistema ou processo.

As variáveis de saída são denominadas **variáveis dependentes** ou **resultados**. Estas são as variáveis que o experimento irá medir para verificar as variações provocadas no processo ou sistema através das alterações provocadas nas variáveis independentes.

Há ainda as **variáveis de contexto** ou **variáveis não-controláveis** (BASILI *et al.*, 1999). Estas variáveis são aquelas que, de algum modo, perturbam o processo ou sistema, mas sobre as quais não se tem controle adequado. Sobre essas variáveis busca-se amenizar seus efeitos através da homogeneização de amostras ou grupos.

Para cada **fator** é definida uma faixa de valores a ser atribuída durante o experimento com intuito de se verificar a saída resultante no sistema. A cada

combinação de valores dos fatores a serem testados no experimento denomina-se **tratamento**.

Por exemplo, desejamos comparar duas técnicas, X e Y, para inspeção de requisitos. E queremos utilizar os documentos Req1 e Req2 para verificar os resultados. Neste contexto, temos:

- Dois fatores: (1) técnicas de inspeção de requisitos, e; (2) diferentes documentos de requisitos (relativo ao domínio do problema).
- Que para o fator técnicas de inspeção de requisitos temos os valores: A e B;
- Que para o fator documentos de requisitos temos os valores: Req1 e Req2;
- Que os seguintes tratamentos são aplicáveis:
 - Tratamento A1: Técnica X e Documento Req1,
 - Tratamento A2: Técnica X e Documento Req2,
 - Tratamento B1: Técnica Y e Documento Req1,
 - Tratamento B2: Técnica Y e Documento Req2.
- Como variável dependente se poderia definir o número de defeitos encontrados no documento inspecionado e o tempo total de inspeção do documento utilizando uma técnica.
- Como variável de contexto se poderia indicar a experiência do inspetor em inspeção de requisitos.

Além disso, alguns conceitos ainda precisam estar definidos, são eles;

- **Objetos:** são as ferramentas utilizadas para verificar a relação entre causa e efeito no experimento. Compõem os chamados instrumentos ou a instrumentação do experimento.
- **Participantes:** são os indivíduos selecionados para composição de uma amostra representativa de uma população-alvo para qual se deseja generalizar os resultados do experimento. Quanto maior a diversidade dos indivíduos na população-alvo maior o número de indivíduos necessários para compor uma amostra representativa. São geralmente selecionados por conveniência em Engenharia de Software e divididos em grupos.
- **Trial ou rodada experimental:** é, de acordo com os tratamentos, a combinação de objetos e participantes (ou grupos). Também é denominado em (JURISTO & MORENO, 2001) como **repetição interna**.

De acordo com exemplo anterior e assumindo que temos 4 participantes escolhidos por conveniência, 2 com experiência e 2 sem nenhuma experiência em inspeção de requisitos, temos que:

- Os objetos são as instâncias físicas dos documentos A e B (papel ou arquivo), as instâncias físicas das técnicas A e B (as instruções da respectiva técnica impressas, por exemplo).
- Como a experiência é uma variável não-controlável, queremos atenuar sua influência formando 2 grupos (Grupo1 e Grupo2): cada grupo com 1 indivíduo experiente e 1 sem experiência;
 - se a experiência fosse uma variável independente, se poderia formar 2 grupos um com indivíduos com experiência e outro com indivíduos sem experiência em inspeção de requisitos.
- As possíveis rodadas experimentais são:
 - Tratamento A1, Grupo1 e Tratamento B2, Grupo2
 - Tratamento A1, Grupo2 e Tratamento B2, Grupo1
 - Tratamento A2, Grupo1 e Tratamento B1, Grupo2
 - Tratamento A2, Grupo2 e Tratamento B1, Grupo1

2.4 - Taxonomias aplicadas a estudos experimentais em Engenharia de Software

A escolha de um método de pesquisa experimental depende dos pré-requisitos da investigação, do seu propósito, da disponibilidade de recursos e de como se pretende analisar os dados coletados (WOHLIN *et al.*, 2000).

Quando pensamos em um experimento, pensamos em uma sala cheia de indivíduos executando alguma tarefa. Esta tarefa é geralmente seguida de uma coleta e análise de dados. Isto certamente é um tipo de Experimentação. Mas há outras abordagens que podem ser agrupadas em 4 categorias gerais (ZELKOWITZ & WALLACE, 1998; ZELKOWITZ *et al.*, 2003):

- **Método científico:** Cientistas desenvolvem uma teoria para explicar um fenômeno; propõem uma hipótese e então testam variações alternativas dessas hipóteses. A partir disso, coletam dados para verificar ou refutar esta hipótese.
- **Método de Engenharia:** Engenheiros desenvolvem e testam uma solução para uma hipótese. Baseado nos resultados deste teste melhora-se a solução até que não seja requisitada mais nenhuma melhoria.
- **Método experimental:** um método estatístico é proposto como meio de testar uma hipótese. Diferentemente do método científico, não há um

modelo formal ou teoria que descreva uma hipótese. Simplesmente os dados são coletados para verificar a hipótese.

- **Método analítico:** Uma teoria formal é desenvolvida, e os resultados derivados daquela teoria podem ser comparados com observações empíricas.

Segundo AMARAL, (2003), o método da engenharia e o experimental podem ser vistos como variações do método científico. Tradicionalmente, o método analítico é utilizado em áreas formais da Engenharia Elétrica e da Ciência da Computação. O método da engenharia é provavelmente o método dominante na indústria. Embora cada um desses métodos possa ser utilizado na Engenharia de Software com diferentes propósitos, todos possuem em comum a coleta e análise de dados.

Alguns autores propõem o que chamam de métodos de pesquisa experimental ou modelos de validação de tecnologias aplicados à Engenharia de Software. Sejam métodos ou modelos, tais propostas buscam definir uma taxonomia para categorização dos estudos executados na Engenharia de Software. Assim, será apresentado uma extensão ao levantamento apresentado por AMARAL (2003), onde serão brevemente relatadas as taxonomias discutidas em DALY *et al.* (1997), WOHLIN *et al.* (2000), TRAVASSOS & BARROS (2003) e ZELKOWITZ *et al.* (2003).

DALY *et al.* (1997) classifica os métodos experimentais em:

- **Estudo de observação ou observacional:** nestes estudos o investigador não expõe objetos ou participantes a nenhum tratamento. O propósito de um estudo de observação pode ser de caracterizar, definir um *baseline* e/ou identificar relacionamentos. Os dados podem ser coletados através de observação visual, entrevistas e formulários de coleta de dados, ou pode ainda utilizar dados históricos de um repositório.
- **Estudo pré-experimental:** Em um estudo pré-experimental, um tratamento é usado e aplicado a somente um caso, por exemplo, a uma empresa, a um projeto, a um único grupo de participantes, a um único indivíduo, dentre outros. Em seguida uma comparação é conduzida em relação a um *baseline*. O termo pré-experimental serve para denotar que este tipo de estudo não atinge as características de um experimento e não para caracterizar como sendo necessariamente um estudo que deve ser conduzido antes de um experimento. Geralmente um estudo pré-experimental é conduzido como investigação de uma nova tecnologia introduzida em um ambiente de uma organização para ver o quão útil é quando comparada à prática corrente. Por exemplo, a

avaliação da introdução da tecnologia de orientação a objetos em um projeto piloto através da comparação em relação ao *baseline* da empresa.

- **Experimento controlado:** Em um experimento controlado é realizada a atribuição aleatória dos participantes, há um projeto experimental, um ou mais tratamentos para dois ou mais participantes/grupos e a comparação direta entre participantes/grupos é executada. Na maior parte das vezes experimentos controlados são conduzidos em laboratório. Todas as variáveis que possam vir a confundir os resultados devem ser mantidas constantes através dos participantes para que um alto nível de controle possa ser atingido. Diferentes efeitos são comparados com participantes assinalados aleatoriamente aos diferentes níveis de tratamento das variáveis independentes.
- **Quasi-experimento:** nestes estudos não há atribuição aleatória dos tratamentos aos participantes. Um ou mais tratamentos podem ser aplicados a dois ou mais grupos, e os grupos são comparados diretamente. A maior restrição é que os investigadores geralmente são incapazes de selecionar e atribuir de forma aleatória os participantes ou projetos às condições. Conseqüentemente, amostras representativas da população sob estudo não são prováveis de serem conseguidas e comparações não podem ser feitas nas bases de grupos equivalentes sem alguma forma de manipulação de estudo. Dependendo das restrições, um *quasi-experimento* pode ser conduzido em condições representativas, onde o nível de controle disponível é mínimo, ou ainda nas condições mais controladas de um laboratório.
- **Pesquisa de Opinião ou Survey:** algum tipo de amostra estruturado é empregado. Uma pesquisa de opinião é conduzida através de entrevistas, questionários, ou ambos. É usada para identificar conhecimento de um conjunto particular de pessoas sobre um objeto de estudo, por exemplo, suas opiniões sobre uma tecnologia em particular ou suas crenças sobre as práticas de engenharia de software em suas organizações. Podem ainda ser utilizadas para descrever uma população a partir de uma amostra de dados ou para identificar relacionamentos gerais em uma população;

Para ZELKOWITZ *et al.* (2003), a classificação dos estudos executados em Engenharia de Software se dá através de modelos de validação de tecnologias para a academia e para indústria que se diferem em objetivos. Modelos de pesquisa

aplicados na academia têm caráter exploratório, ou seja, tentam avaliar resultados de pesquisa. Enquanto modelos de pesquisa aplicados na indústria têm caráter confirmatório, ou seja, representam os esforços industriais na adoção de uma tecnologia. Enquanto pesquisadores buscam comparar eficiência entre tecnologias, a indústria busca tornar-se mais competitiva (com a redução de custos, prazos, retrabalho, ou melhorando atributos de qualidade de processo/produto), nem sempre utilizando a tecnologia mais eficiente.

Em ZELKOWITZ & WALLACE, (1998); ZELKOWITZ *et al.* (2003) os métodos experimentais são classificados em:

- **Métodos de observação:** nestes métodos os dados são coletados ao longo do desenvolvimento de um projeto. Há relativamente pouco controle sobre o processo de desenvolvimento a não ser sobre o uso da nova tecnologia que está sendo introduzida.
- **Métodos históricos:** A coleta de dados é realizada em projetos que já foram finalizados, ou seja, os dados já existem. Somente torna-se necessário analisar o que já foi coletado.
- **Métodos controlados:** Nestes métodos se provê validade estatística dos resultados para múltiplas instâncias de uma observação e é o clássico método de projeto experimental em outras disciplinas científicas.
- **Métodos formais:** Visa utilizar um modelo formal para descrever um processo e depende do uso de outro método para determinar se o modelo proposto condiz com a realidade. O único tipo de método formal seria a análise teórica.
- **Métodos informais:** seriam métodos *ad-hoc*, não provendo resultados significantes que uma tecnologia traz algum tipo de vantagem.

Para cada método há um sub-conjunto de tipos de estudo tanto para pesquisa acadêmica quanto para a industrial, mas sua discussão está fora do escopo deste trabalho. A Tabela 2.1 mostra os tipos de estudos para cada um dos quatro métodos segundo ZELKOWITZ *et al.* (2003).

Tabela 2.1- Métodos de validação segundo ZELKOWITZ *et al.* (2003)

Método	Acadêmica	Industrial
Observacional	Estudo de Caso	Estudo de Caso
	Monitoramento de Projeto	Monitoramento de Projeto
	Estudo de Campo	Estudo de Campo
Histórico	Dados Legado	Dados Legado
	Busca na Literatura	Busca na Literatura
	Lições Aprendidas	Educação
	Análise Estática	<i>Benchmark</i> ou Comparativo de Características
		Opinião de Especialistas
Controlado	Análise Dinâmica	Estudo Piloto
	Experimento Repetido ou Replicado	Projetos de Demonstração
	Simulação	Projetos Repetidos ou Replicados
	Ambiente Sintético	Benchmark sintético
Formal	Análise Teórica	Análise Teórica
Informal	Assertiva	Opinião do Vendedor
	Sem validação	Externo

Segundo WOHLIN *et al.* (2000) os métodos experimentais são classificados em:

- **Pesquisa de campo:** como já definido, é geralmente uma investigação executada, por exemplo, quando uma ferramenta ou técnica tem sido utilizada por um determinado tempo em uma empresa e pretende-se avaliá-la sob algum aspecto. As formas primárias de coletar dados qualitativos ou quantitativos são através de entrevistas ou questionários, utilizando uma amostra representativa da população a ser estudada. Os resultados da pesquisa de campo são então analisados a fim de derivar conclusões explanatórias e descritivas, e então são generalizados para a população a partir da qual a amostra foi tirada. Pesquisas de campo são muito parecidas com o censo, diferindo primariamente no fato de que a pesquisa de campo tipicamente examina uma amostra de uma

população, enquanto um censo geralmente implica em uma enumeração de toda a população.

- **Estudos de caso** são usados para monitorar projetos, atividades ou exercícios. Dados são coletados para um propósito específico do estudo. Baseado na coleção de dados, análises estatísticas podem ser executadas. O estudo de caso normalmente tem como objetivo rastrear um atributo específico ou estabelecer relacionamentos entre diferentes atributos. O nível de controle é mais baixo em um estudo de caso do que em um experimento. Um estudo de caso é um estudo de observação enquanto que o experimento é um estudo controlado. Estudo de caso é a estratégia preferida quando as questões "como" e "porque" são colocadas, quando o investigador tem pouco controle sobre os eventos e quando o foco é em um fenômeno contemporâneo dentro de um contexto real. Os estudos de caso podem ser complementados por dois outros tipos: exploratório e descritivo. Independente do tipo de estudo, os pesquisadores devem ter grande cuidado ao projetar e executar o estudo de caso a fim de superar a crítica tradicional ao método (YIN, 1994).
- **Experimentos** geralmente são realizados em um ambiente de laboratório (*in vitro*) o que oferece um maior nível de controle, mas podem ser realizados sob condições normais (*in vivo*). São apropriados para confirmar as teorias, confirmar o conhecimento convencional, explorar os relacionamentos, avaliar a predição dos modelos ou validar as medidas. A maior força dos experimentos encontra-se no controle sobre o processo e as variáveis e na possibilidade de ser repetido. Quando se experimenta, participantes são assinalados a diferentes tratamentos de uma forma aleatória. O objetivo é manipular uma ou mais variáveis e controlar todas as outras variáveis em níveis fixos. O efeito da manipulação é medido e, baseado nisto, uma análise estatística pode ser realizada. Em alguns casos pode ser impossível usar verdadeiros experimentos, nestes casos usa-se a *quasi-experimentação*. *Quasi-experimentos* são realizados quando é impossível realizar assinalamentos aleatórios dos participantes aos diferentes tratamentos.

Segundo a taxonomia proposta por TRAVASSOS & BARROS, (2003), os estudos experimentais que podem ser executados na Engenharia de Software podem ser classificados em:

- Estudos *in-vitro*: compõem a maior parte dos estudos executados em Engenharia de Software, os chamados estudos de laboratório. Nestes estudos, pesquisadores montam situações próximas às reais em ambientes preparados ("laboratórios"), para observarem como os desenvolvedores executam determinada tarefa. Neste contexto, um ambiente preparado é um ambiente em que algumas de suas variáveis podem ser controladas ou pelo menos observadas. Geralmente estudos *in-vitro* envolvem riscos gerenciáveis, por não estarem afetando nenhum processo produtivo dentro de uma organização. Porém são também os estudos mais difíceis, já que é difícil para as organizações, independente de sua natureza acadêmica ou industrial, alinhar a execução desses estudos com os seus objetivos organizacionais. Por exemplo, para executar um estudo em uma disciplina acadêmica, a tecnologia estudada deve estar inserida no contexto do conteúdo programático da disciplina. Além disso, são necessários a alocação de recursos físicos (tais como, salas, laboratórios, mesas) e de hardware e software para composição do ambiente onde o estudo será executado. Quanto à seleção de indivíduos, é difícil, principalmente em ambientes industriais, retirar desenvolvedores de suas tarefas e alocá-los para executar um estudo. Há problemas que vão desde cronograma e alocação de atividades a imprevistos como atrasos em projetos.
- Estudos *in-vivo*: são os estudos executados ao longo de uma cadeia produtiva, ou seja, de um processo de software. São estudos com alta influência na cadeia produtiva de uma organização e por isso são considerados os de maiores riscos. Um fracasso na execução do estudo pode refletir nos produtos intermediários de um processo e acarretar na queda de qualidade do produto final ou a identificação de erros, gerando re-trabalho, custos e prazos adicionais, etc. Ao contrário de um estudo *in-vitro* em que o ambiente é preparado, nestes estudos o ambiente é o próprio processo onde as tarefas serão executadas.
- Estudos *in-virtuo*: são estudos em que o ambiente de execução do estudo é virtual, ou seja, o ambiente será definido por algum modelo descrito computacionalmente. Os riscos associados a este tipo de estudo são parecidos com os de um estudo *in-vitro*. Mas como o ambiente trata-se de um modelo, seu comportamento é capaz de ser previsto e customizável. Nestes estudos os desenvolvedores executam suas tarefas e serão somente influenciados pelo comportamento descrito no modelo do ambiente, característica essa quase nunca obtida em estudos *in-vitro*. Um

risco associado a este tipo de estudo que pode comprometer a validade desses estudos está relacionado à completude e corretude do modelo utilizado para descrever o ambiente.

- Estudos *in-silico*: são estudos em que o comportamento do ambiente e dos desenvolvedores é descrito através de modelos computacionais. Os riscos associados a este tipo de estudo se aproximam ao que observamos nos estudos *in-virtuo*. Nestes estudos, são eliminados os riscos associados à seleção e participação dos desenvolvedores, mas são introduzidos riscos relativos à corretude e completude dos modelos utilizados para a descrição do comportamento humano executando determinadas tarefas, na maioria das vezes muito mais complexas que a dos modelos para descrever os ambientes em que os estudos serão executados.

Na Figura 2.2 é ilustrada a relação entre o aumento do controle no ambiente de execução e aumento dos riscos que afetam o processo produtivo. Podemos observar que nos estudos *in-vivo* temos os maiores riscos relativos ao processo de produção com o menor controle sobre as variáveis do ambiente. Enquanto que no outro extremo temos os estudos *in-silico* com o maior controle do ambiente e com menor impacto no processo produtivo. Já os estudos *in-vitro* e *in-virtuo* se encontram numa faixa intermediária, diferindo-se principalmente no controle do ambiente.

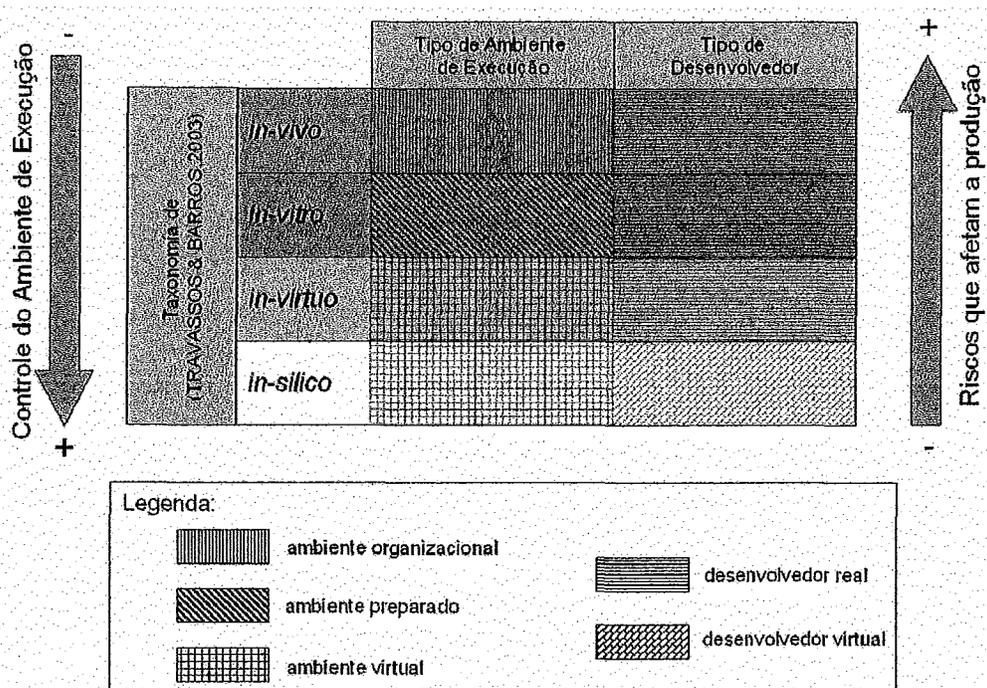


Figura 2.2- Controles e riscos associados a estudos experimentais segundo a taxonomia de TRAVASSOS & BARROS (2003)

As taxonomias apresentadas acima não são as únicas formas de classificação existentes (ver BASILI (1996) e KITCHENHAM (1996)). Cada taxonomia define uma perspectiva para classificação e, conseqüentemente, uma abordagem para a execução de estudos experimentais e Engenharia de Software.

A Tabela 2.2 mostra alguns exemplos de estudos encontrados na literatura para cada uma das taxonomias discutidas anteriormente.

Tabela 2.2- Taxonomia de Métodos Experimentais e seus exemplos

Métodos	Classificação dos Autores				Exemplos
	DALY et al.	TRAVASSOS & BARROS	ZELKOWITZ et al.	WOHLIN Et al.	
	1997	2003	2003	2000	
Experimento		in-vitro		X	(WOHLIN <i>et al.</i> , 2000) (SILVA, 2004)
Experimento Repetido		in-vitro in-vivo	X		(BASILI <i>et al.</i> , 1999) (SHULL <i>et al.</i> , 2002)
Experimento Controlado	X	in-vitro			(PRECHELT <i>et al.</i> , 2001) (SALLIE & HUMPHREY, 1990)
Experimento em Ambiente Sintético		in-vitro in-virtuo	X		(KERLINGER, 1973) (BARROS <i>et al.</i> , 2001)
Quasi-Experimento	X	in-vitro			(CAMPBELL & STANLEY, 1979) (SILVA, 2004)
Análise Dinâmica			X		(VERGILIO <i>et al.</i> , 1995)
Simulação		in-silico	X		(BARROS <i>et al.</i> , 2002) (COSTA <i>et al.</i> , 2005)
Estudo Observacional	X	in-vitro in-vivo			(ROSENBAUM, 1995)
Monitoramento de Projeto		in-vivo	X		(BASILI, 1985)
Pesquisa de Campo	X	in-vivo		X	(BABBIE, 1990) (KERLINGER, 1973)
Assertiva					(SHNEIDERMAN <i>et al.</i> , 1977)
Estudo de Caso		in-vitro in-vivo	X	X	(YIN, 1994)
Pré-Experimento	X	in-vitro			(KITCHENHAM <i>et al.</i> , 1995) (TRAVASSOS <i>et al.</i> , 1999)
Estudo de Campo		in-vivo	X		(KERLINGER, 1973)
Análise Teórica			X		(CARVER, 2003) (MOOKERJEE & CHIANG, 2002)
Pesquisa na Literatura			X		(SJØBERG <i>et al.</i> , 2005) (CONTE <i>et al.</i> , 2005) (MAFRA, 2005)
Dados Legados		in-vivo	X		(RAMIL & LEHMAN, 2000)
Lições Aprendidas			X		(SHULL <i>et al.</i> , 2002) (BASILI <i>et al.</i> , 2002)

Análise Estática		in-vitro	X		(FENTON & PFLEEGER, 1997)
Benchmark sintético		in-vitro in-vivo	X		(WOHLIN <i>et al.</i> , 2002)

2.5 - O Processo de Experimentação em Engenharia de Software

Uma premissa da Gerência de Processos é que “a qualidade de um produto é fortemente influenciada pela qualidade do processo usado para adquiri-lo, desenvolvê-lo ou mantê-lo” (CMMI Overview, 2005). Baseado nesta premissa, processos vêm sendo amplamente utilizados na Engenharia de Software e em outras áreas como Engenharia Química e Manufatura. E a partir desse raciocínio, podemos assumir que a qualidade dos resultados de um estudo experimental está relacionada à qualidade do processo utilizado para executá-lo.

Segundo SHULL *et al.*, (2001), um processo bem definido pode ser observado e medido; e desta forma melhorado. Processos podem ser usados para capturar as melhores práticas para se tratar um determinado problema. Além disso, a utilização de um processo permite que práticas de trabalho sejam disseminadas mais rapidamente do que a construção de experiência pessoal. Uma ênfase no processo também ajuda o desenvolvimento de software se tornar mais engenharia, com as restrições de tempo e recursos já prognosticadas, e menos arte.

Levando-se em conta tais considerações, observa-se a crescente preocupação na adoção de um processo para execução de estudos experimentais em Engenharia de Software (WOHLIN *et al.*, 2000; JURISTO & MORENO, 2001; AMARAL, 2003). Nesta seção, abordaremos sucintamente os Processos de Experimentação propostos por WOHLIN *et al.* (2000) e AMARAL (2003).

WOHLIN *et al.*, (2000) definem um processo em cascata com o objetivo executar um estudos experimentais. O processo de Experimentação proposto por WOHLIN *et al.* (2000) é composto por cinco macro-atividades ou etapas: (1) Definição; (2) Planejamento; (3) Operação; (4) Interpretação e Análise,e; Empacotamento (Figura 2.3).

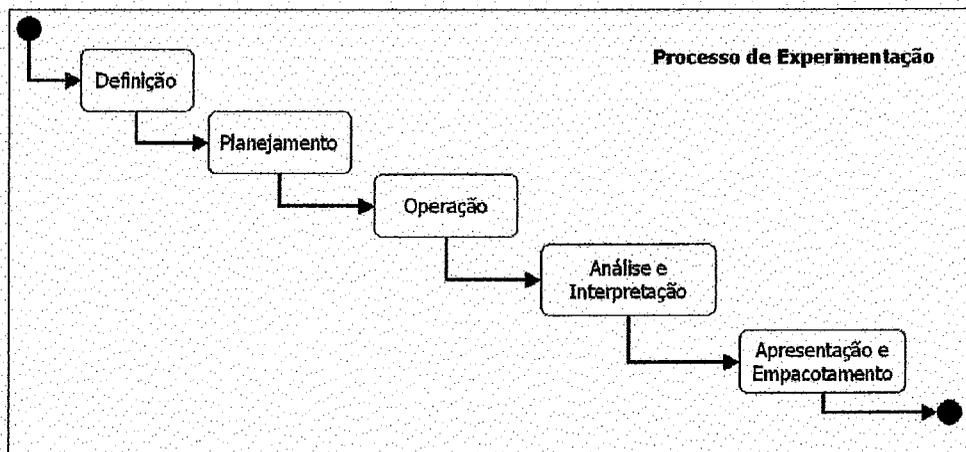


Figura 2.3 - Processo de Experimentação proposto por (WOHLIN *et al.*, 2000)

Porém, este processo não é descrito detalhadamente em termos das atividades, recursos, produtos, papéis envolvidos na execução de um estudo experimental, deixando estas questões ainda por serem definidas. Deste modo, AMARAL (2003) utiliza a proposta de WOHLIN *et al.* (2000) como ponto de partida para definição e descrição detalhada de um Processo de Experimentação.

AMARAL (2003) propõe e formaliza (em termos de atividades, recursos, produtos e papéis envolvidos) um Processo de Experimentação baseado nas mesmas etapas do processo citado anteriormente, diferindo no fato de após a etapa de Planejamento o pesquisador poder retornar à etapa de Definição para corrigir ou refinar as informações registradas e que o empacotamento do estudo é realizado ao longo da execução das outras etapas, a partir do qual as informações e artefatos empacotados são armazenados numa base de conhecimento de estudos experimentais (Figura 2.4).

Além disso, na proposta de AMARAL (2003) é definida uma taxonomia para descrição dos artefatos produzidos ao longo do Processo de Experimentação. A esse conjunto de artefatos denomina-se Pacote do Experimento.

Nas seções subseqüentes serão discutidas as etapas do processo proposto por AMARAL (2003), por possuir essencialmente as mesmas etapas do processo proposto por WOHLIN *et al.* (2000) e por ter mais riqueza de detalhes na formalização do processo.

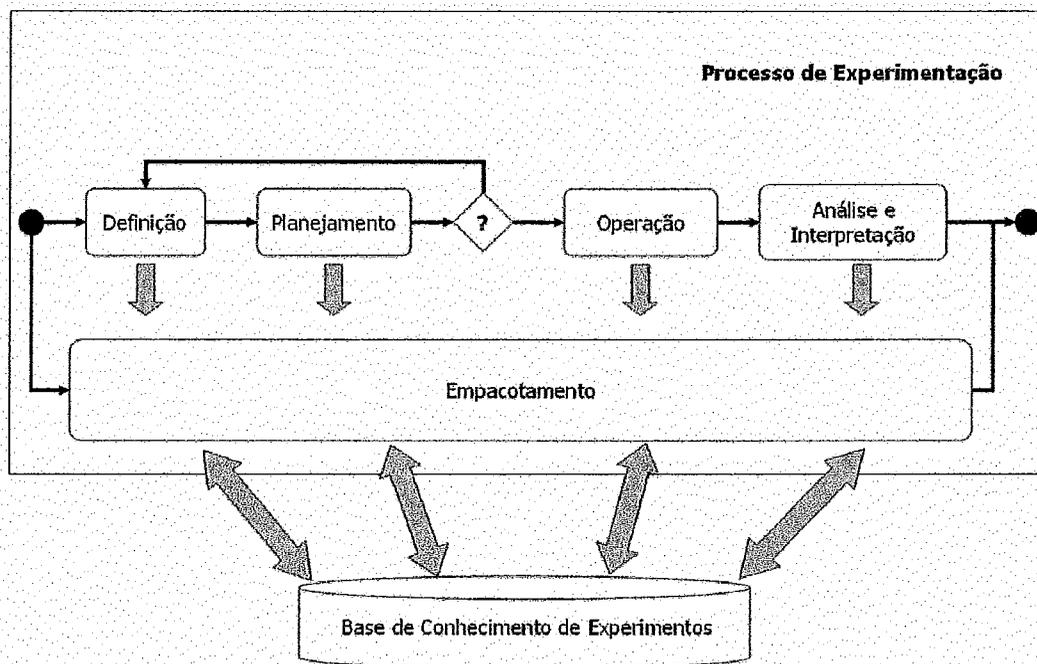


Figura 2.4 - Processo de Experimentação proposto por AMARAL (2003)

2.5.1 – Etapa de Definição

A primeira etapa do Processo de Experimentação é a de Definição. Seu propósito é definir os objetivos de um estudo experimental, justificando o porquê da sua realização. Se estes objetivos não forem apropriadamente definidos pode haver muito re-trabalho ou o estudo experimental pode não ser usado para o que foi inicialmente definido. Tal etapa tem como meta explicitar as informações iniciais (razão, objetivo, organização que está promovendo a pesquisa, dentre outros) sobre o estudo experimental. Segundo AMARAL (2003), também é importante manter o objetivo geral em mente. Existem muitos possíveis objetivos de um estudo experimental, incluindo confirmação, descoberta, dentre outros.

Nesta atividade, as hipóteses e metas do experimento devem ser bem definidas. Além disso, os objetivos devem estar definidos e formulados a partir do problema a ser resolvido.

A fim de capturar as informações necessárias nesta etapa, muitos pesquisadores (WOHLIN *et al.*, 2000; JURISTO & MORENO, 2001; KITCHENHAM *et al.*, 2002; AMARAL, 2003; SJØBERG *et al.*, 2005) sugerem o uso da abordagem GQM (Goal/Question/Metric) definido em (BASILI, 1992), cujo template é:

Analisar <objeto de estudo>;
Com o propósito de <objetivo>;
Com respeito a <o foco de qualidade>;
Do ponto de vista <perspectiva>;
No contexto de <contexto>.

No contexto da definição de um estudo experimental, o objeto de estudo é a entidade estudada. Por exemplo, produtos, processos, recursos, modelos, métricas ou teorias. O objetivo define qual é a intenção do estudo experimental, que pode ser, por exemplo, o de avaliar o impacto de duas diferentes técnicas ou de caracterizar a curva de aprendizado de uma organização. O foco de qualidade é(são) o(s) efeito(s) primário(s) sob estudo, tais como custo, confiabilidade, dentre outros. A perspectiva mostra o ponto de vista sob o qual os resultados do estudo experimental são interpretados. Exemplos de perspectivas são o desenvolvedor, o gerente do projeto, cliente, pesquisador, ou seja, a perspectiva assumida para a qual se quer avaliar o experimento. E o contexto é o ambiente no qual o estudo experimental é executado. O contexto brevemente define que pessoas estão envolvidas no estudo experimental (participantes) e que artefatos de software (objetos) estão sendo utilizados no experimento. Os participantes podem ser caracterizados pela experiência, tamanho da equipe, dentre outros. Os objetos podem ser caracterizados pelo tamanho, complexidade, prioridade, domínio de aplicação, por exemplo.

De acordo com WOHLIN *et al.* (2000), o contexto de um estudo experimental é composto das condições em que este está sendo executado. O contexto pode ser caracterizado de acordo com as seguintes dimensões:

In-vivo, in-vitro, in-virtuo, in-silico (TRAVASSOS & BARROS, 2003);

Alunos ou profissionais: caracteriza os participantes do estudo experimental;

Problema de sala de aula ou problema real: define o estudo experimental que está sendo estudado;

Específico ou geral: caracteriza se os resultados do estudo experimental são válidos para um contexto específico ou para o domínio geral da Engenharia de Software.

Ao término desta etapa o contexto, as hipóteses, os objetivos e propósitos do estudo experimental devem estar bem claros para quem está projetando o estudo experimental.

2.5.2 – Etapa de Planejamento

Esta etapa descreve como o estudo experimental vai ser executado. Como em outras atividades da Engenharia de Software, os estudos experimentais devem ser planejados e seus planos devem ser seguidos para se obter o controle desejado durante a execução do estudo experimental. O resultado do estudo experimental pode ser afetado ou até mesmo invalidado se não for planejado devidamente.

Além disso, é nesta etapa que a decisão sobre a atribuição dos objetos e dos participantes ocorre. Sendo ela composta de vários componentes, como as variáveis dependentes e independentes, um plano para sistematicamente manipular as variáveis independentes e observar a variável dependente, e o contexto operacional do estudo, ou seja, uma descrição física, intelectual e cultural do meio onde o estudo está será executado.

A etapa de Planejamento é responsável por descrever os produtos, recursos e processos envolvidos no estudo, incluindo: a população a ser estudada, a técnica utilizada para obter amostras da população de estudo, o processo para alocar e administrar os tratamentos e os métodos usados para reduzir o viés e determinar o tamanho da amostra (KITCHENHAM *et al.*, 2002).

A partir da hipótese formulada na etapa de Definição, dois tipos de hipóteses devem ser formulados: (1) a hipótese nula, e; (2) a(s) hipótese(s) alternativa(s).

A hipótese nula (H_0) assume que não há diferença entre os tratamentos com relação às variáveis dependentes;

A(s) hipótese(s) alternativa(s) (H_a , H_1 , H_2 , etc.) sugere(m) que há uma diferença significativa entre os dois tratamentos.

Exemplo: supondo que o objetivo do estudo experimental é comparar duas técnicas de inspeção de requisitos A e B.

Então:

H_0 : Não há diferença entre as técnicas A e B.

H_a : Há diferença entre as técnicas A e B.

H_1 : A técnica A é mais eficiente que a técnica B.

H_2 : A técnica B é mais eficiente que a técnica A.

Ao se definir as hipóteses, é necessário atentar para os riscos associados ao método utilizado no teste de hipóteses. Estes riscos tentam traduzir os possíveis erros que podem ocorrer durante a verificação das hipóteses. Dentre estes erros geralmente são considerados o erro do primeiro tipo (*type-I-error*) e do segundo tipo (*type-II-error*).

O erro do primeiro tipo (*type-I-error*) acontece quando o teste estatístico rejeita a hipótese nula e esta não poderia ser rejeitada. Esse tipo de erro indica a presença de um relacionamento, quando este relacionamento não existe. Um meio de avaliar este erro é através de sua probabilidade de ocorrência, definida como (WOHLIN *et al.*, 2000):

- $P(\text{type-I-error}) = P(H_0 \text{ é rejeitada} | H_0 \text{ é verdadeira})$

O erro do segundo tipo (*type-II-error*) acontece quando o teste estatístico não rejeita a hipótese nula e esta deveria ser rejeitada. Esse tipo de erro indica a ausência de um relacionamento, quando este relacionamento poderia ser efetivamente demonstrado. Como no erro de primeiro tipo, um meio de avaliar este erro através de sua probabilidade de ocorrência, definida como:

- $P(\text{type-II-error}) = P(H_0 \text{ não é rejeitada} | H_0 \text{ é falsa})$

Um meio de avaliar o erro cometido durante a verificação das hipóteses é através da potência do teste estatístico a ser aplicado. A potência do teste estatístico é definida como a probabilidade de que o teste corretamente irá rejeitar a hipótese nula. A potência pode ser avaliada como (WOHLIN *et al.*, 2000):

- $\text{Potência} = P(H_0 \text{ é rejeitada} | H_0 \text{ é falsa}) = 1 - P(\text{type-II-error})$.

Deste modo o teste estatístico com a maior potência deve ser escolhido. O conhecimento da potência do teste estatístico pode influenciar o planejamento, execução e resultados da pesquisa experimental. Em (DYBA° *et al.*, 2005a) discute-se a relevância da potência estatística aplicados nos estudos experimentais executados na Engenharia de Software e é apresentada uma revisão sistemática (KITCHENHAM, 2004; KITCHENHAM *et al.*, 2004; BIOLCHINI *et al.*, 2005; DYBA° *et al.*, 2005b) sobre a potência estatística de estudos executados pela Comunidade de Engenharia de Software entre os anos de 1993 e 2002.

O próximo passo nesta atividade consiste em definir as variáveis independentes e dependentes a serem utilizadas no estudo experimental, devendo ser derivada(s) diretamente da(s) hipótese(s). As variáveis independentes são aquelas que se pode controlar em um estudo experimental, sendo usadas para limitar o escopo de um estudo experimental e também para diferenciar estudos experimentais. Já as variáveis dependentes são os fatores que se espera que mudem ou que apresentem uma diferença como consequência da aplicação de alterações nas variáveis independentes.

Outra tarefa realizada nesta etapa é a seleção dos participantes. Para generalizar os resultados, a seleção deve ser representativa para esta população. Para se obter uma amostra, deve-se começar definindo a população alvo, representada pelo grupo ou participantes a quem o estudo se aplica. Idealmente, uma população alvo deveria ser representada como uma lista finita de todos os seus membros. É muito importante que se defina uma amostra representativa, pois caso não se consiga esta amostra representativa, não será possível reivindicar que os resultados obtidos sejam generalizados para a população alvo.

As técnicas utilizadas para definir a amostra de uma população são definidas como técnicas probabilísticas e não probabilísticas. A técnica probabilística é aquela na qual cada membro de uma população alvo tem uma probabilidade conhecida e diferente de zero de ser incluído na amostra. O objetivo desta técnica é o de eliminar a subjetividade e obter uma amostra que não permita o viés e seja representativo da população alvo. É importante lembrar que não se pode fazer nenhuma inferência estatística nos dados a menos que se utilize uma técnica probabilística. A técnica não-probabilística é utilizada quando os participantes são escolhidos pelo fato de serem facilmente acessíveis ou porque os pesquisadores têm alguma justificativa para acreditar que eles sejam representativos. Este tipo de técnica apresenta o risco de apresentar viés, sendo, portanto, difícil estabelecer qualquer inferência forte com a sua utilização. Alguns métodos que compõem as técnicas probabilísticas e não-probabilísticas são mostrados em KITCHENHAM & PFLEEGER (2002).

Em KITCHENHAM *et al.* (2002) são definidas algumas diretrizes para a atividade de seleção de participantes:

- Identificar a população a partir da qual os participantes são selecionados;
- Definir o processo pelo qual participantes e objetivos foram selecionados;
- Definir o processo pelo qual participantes são atribuídos aos tratamentos.

Baseado na suposição estatística, isto é, nas escalas de medição e em quais objetos e participantes estão disponíveis para serem utilizados, determina-se o projeto experimental. Durante o projeto determina-se quanto teste o experimentador deve executar para ter certeza que o efeito do tratamento é visível.

Para tornar os resultados mais conclusivos e significativos, a abordagem padrão de projetos experimentais em Engenharia de Software se dá através de comparação de formação de grupos. Entretanto, esta estratégia requer um número significativo de participantes, envolvendo custo, prazos e aplicabilidade. Embora pouco

comuns, HARRISON (2000) discute o uso de projetos experimentais utilizando um único indivíduo, argumentando sobre facilidades e riscos associados a este tipo de projeto experimental. Um projeto bem feito forma a base para permitir a repetição do estudo experimental (WOHLIN *et al.*, 2000).

Os princípios gerais a serem considerados para projetar um estudo experimental são: aleatoriedade, agrupamento e balanceamento.

A aleatoriedade é um dos princípios mais importantes. Todos os métodos estatísticos usados para analisar os dados requerem que as observações sejam de variáveis aleatórias independentes. Este princípio aplica-se na alocação dos objetos, participantes e em que ordem os testes são executados. É também utilizada para colocar na média o efeito de um fator que pode de outra forma estar presente e para selecionar participantes que sejam representativos da população de interesse.

Às vezes se tem um fator que provavelmente tem um efeito na resposta, mas não se está interessado neste efeito. Se o efeito do fator é conhecido e controlável, pode-se usar uma técnica conhecida como agrupamento. Esta técnica é usada sistematicamente para eliminar o efeito indesejado através da comparação entre os tratamentos. Com a utilização desta técnica aumenta-se, a precisão do estudo experimental (WOHLIN *et al.*, 2000).

A técnica de balanceamento consiste em definir os tratamentos de tal forma que cada tratamento tenha a mesma quantidade de participantes. Esta técnica serve para simplificar e fortalecer a análise estatística dos dados, porém ela não é totalmente necessária (WOHLIN *et al.*, 2000).

O próximo passo é projetar os instrumentos. A instrumentação provê meios de executar o estudo experimental e monitorá-lo, sem afetar seu controle. Os resultados devem ser os mesmos independentemente de como o estudo experimental é instrumentado. Se a instrumentação afeta a saída do estudo experimental, então os resultados são inválidos (AMARAL, 2003).

Os instrumentos para um estudo experimental são de três tipos: objetos, diretrizes e artefatos de medição (JURISTO & MORENO, 2001). Os objetos podem ser, por exemplo, documentos de especificação ou de código. As diretrizes são necessárias para guiar os participantes no estudo experimental. Elas podem ser, por exemplo, descrições de processos e *checklists*. Os artefatos de medição são utilizados através da coleta de dados. Geralmente esta coleta é realizada através de formulários ou de entrevistas. A tarefa de medição a ser usada é a de preparar formulários e questões de entrevista e de validar estes formulários e questões com algumas pessoas, tendo habilidades similares aos dos participantes do estudo experimental.

AMARAL (2003) caracteriza os instrumentos em artefatos do estudo experimental e artefatos de software. Artefatos do Experimento seriam, basicamente, as diretrizes e os artefatos de software os objetos.

Por fim, é necessário verificar a adequação de validade dos resultados. Segundo AMARAL (2003), os conceitos de validade referem-se à melhor aproximação disponível da verdade ou falsidade das proposições, incluindo proposições sobre causa.

Uma validade adequada refere-se aos resultados que devem ser válidos para a população de interesse. Primeiro, os resultados devem ser válidos para a população de onde a amostra foi retirada. Segundo pode ser de interesse generalizar os resultados para uma população maior (WOHLIN *et al.*, 2000). A validação inclui examinar cada formulário procurando completude e consistência. E quando necessário conduzir entrevistas com os participantes para se obter dados perdidos e corrigir erros (BASILI & WEIS, 1981 appud AMARAL,2003).

Ameaças à validade são influências que podem limitar a habilidade de interpretar ou de escrever resultados dos dados de estudo (PERRY *et al.*, 2000 appud AMARAL,2003). Pesquisadores têm a responsabilidade de discutir quaisquer limitações de seus estudos (KITCHENHAM *et al.*, 2002).

Existem pelo menos quatro tipos de validade que podem ser utilizadas para se proteger de tais ameaças: conclusão, interna, construção e externa.

A validade de conclusão é relativa ao relacionamento entre os tratamentos e a saída. Deve-se assegurar que existe um relacionamento estatístico com um significado.

A validade interna diz respeito a como se infere que um relacionamento entre duas variáveis é casual ou que a falta de um relacionamento implica na falta de uma causa. Ou seja, é aquele mínimo básico sem o qual qualquer estudo experimental não seria interpretável. Responder à pergunta a seguir é uma forma de se avaliar a validade interna: os tratamentos experimentais fizeram, de fato, diferença nesta instância particular deste estudo experimental? (CAMPBELL e STANLEY, 1979a appud AMARAL, 2003).

A validade de construção significa que as variáveis independentes e dependentes modelam exatamente as hipóteses abstratas.

A validade externa implica que se pode inferir se o relacionamento causal presumido pode ser generalizado para e através de medidas alternativas de causa e efeito e através de diferentes tipos de vezes, pessoas e cenários (CAMPBELL e COOK, 1979b appud AMARAL, 2003). Ela levanta a questão da generalidade: a que

populações, conjuntos, variáveis de fragmento e medidas pode esse efeito ser generalizado (AMARAL, 2003)?

Segundo CARVER *et al.* (2004), a compreensão sobre as ameaças de cada tipo de validade é influenciada por três fatores na Engenharia de Software: (1) processo; (2) pessoas, e; (3) produto. E ao projetar um estudo experimental, as ameaças à validade devem levar em conta cada um desses fatores.

Um tema polêmico sobre as ameaças à validade é a utilização de estudantes como participantes equivalentes a indivíduos profissionais, uma ameaça à validade externa. É conveniente utilizar estudantes como participantes de estudos experimentais. Entretanto, há um ceticismo comum sobre a capacidade de se generalizar resultados destes estudos para ambientes industriais (HARRISON, 2000).

Segundo HOST *et al.* (2000), estudantes podem ser utilizados como profissionais sob determinadas condições. Em BERANDER (2004) também se demonstra que estudantes tiveram resultados próximo à indivíduos profissionais quando lidavam com clientes industriais. Além disso, CARVER *et al.* (2003) atentam que o uso de estudantes como participantes vêm ajudando no desenvolvimento de novas tecnologias e na depuração de protocolos experimentais antes de aplicá-los na indústria, e sugerem uma abordagem baseada em estudos de observação e treinamento para acelerar a aquisição de experiência dos estudantes e diminuir a diferença entre estudantes e indivíduos profissionais ao realizar estudos de sala de aula.

HOST *et al.* (2000) argumentam que é essencial a compreensão sobre os fatores que diferenciam estudantes de profissionais e assim obter uma melhor compreensão sobre a validade dos resultados que utilizam estudantes como participantes, já que não é tão clara o quão bem os resultados de estudantes se generalizam para engenheiros de software profissionais (HARRISON, 2000).

2.5.3 – Etapa de Operação

Chegar nesta etapa significa que após um estudo experimental ser projetado e planejado então ele deve ser executado com o propósito de coletar os dados que posteriormente serão analisados.

Quando se executa um estudo experimental é vital monitorar o processo cuidadosamente, para ter certeza de que tudo seja feito de acordo com o planejado. Erros de procedimentos neste estágio irão usualmente destruir a validade do estudo experimental. Sugere-se ainda que antes de realizar o estudo experimental sejam realizados estudos pilotos ou *trials*, os quais provêm informações sobre a consistência do material experimental, uma verificação no sistema de medida, uma boa idéia dos

erros experimentais, e uma chance de praticar as técnicas experimentais (MONTGOMERY, 2001 appud AMARAL, 2003).

É nesta etapa que os tratamentos são aplicados aos participantes. Isto significa que esta parte do estudo experimental é onde o experimentador realmente se encontra com os participantes. Na maior partes dos estudos experimentais em Engenharia de Software, existem poucas outras ocasiões em que os participantes estejam realmente envolvidos (AMARAL, 2003).

Ao lidar com pessoas, pesquisadores precisam motivá-las a participar de estudos experimentais, pois, por melhor que seja o projeto experimental e a análise dos dados coletados, os resultados serão inválidos caso os participantes não tenham participado seriamente do estudo experimental. Alguns aspectos devem ser levados em consideração com o propósito de convencê-los a participar de um estudo experimental como participantes de um estudo: obtenção de seu consentimento, apresentar resultados que os sensibilizem, preparar elementos atrativos para participarem do estudo experimental, e evitar que se decepcionem com o estudo experimental (WOHLIN *et al.*, 2000).

Outro aspecto a se levar em consideração é se os participantes devem coletar os dados por conta própria, na maior parte dos casos, que alguns tipos de formulários devem ser distribuídos aos participantes.

A coleta de dados a partir de entrevistas (HOVE & ANDA, 2005) e de questionários (KITCHENHAM & PFLIEGER, 2002; PUNTER *et al.*, 2003) são as mais usuais. A vantagem de usar questionários é que não requer muito esforço do experimentador, pois ele não tem que tomar parte ativamente nesta atividade de coleta. Uma desvantagem é que não há a possibilidade para o experimentador descobrir diretamente inconsistências, incertezas, falhas nos questionários, entre outros.

Em SILVA & TRAVASSOS (2004) e KARAHASANOVIC *et al.* (2005) são apresentados métodos de coleta de dados de forma não-intrusiva, ou seja, sem a intervenção dos participantes e dos pesquisadores no processo de obtenção dos dados. Além disso, (SILVA & TRAVASSOS, 2004) discutem as vantagens e a questões éticas sobre tais metodologias.

Um ponto a se determinar quando os formulários são construídos é se eles devem ser identificados ou os participantes devem preenchê-los anonimamente. Caso não haja estudos adicionais e, portanto não existe a necessidade real para o experimentador distinguir entre diferentes participantes, então pode ser apropriado usar formulários anônimos. Porém isto quer dizer que não há a possibilidade de entrar em contato com o participante se algo não for preenchido de uma maneira clara. Em

muitos casos é apropriado preparar um conjunto pessoal de instrumentos para cada participante, pois muitos projetos lidam com aleatoriedade e repetição de testes, de tal forma que diferentes tratamentos devam ser atribuídos a eles. Todavia isto também pode ser feito quando os participantes são anônimos (WOHLIN *et al.*, 2000).

Após a coleta de dados, é importante verificar se os dados foram devidamente coletados e se são razoáveis. Isto quer dizer se os participantes entenderam os questionários e desta forma os preencheram de maneira correta. Outra forma de erro ocorre se os participantes não tenham participado do estudo experimental de maneira séria e, portanto, alguns dados devem ser removidos antes da análise. Uma forma de verificar se os participantes entenderam as intenções do experimentador é oferecer, por exemplo, um seminário para apresentar os resultados (WOHLIN *et al.*, 2000).

2.5.4 – Etapa de Análise e Interpretação

Após a coleta dos Dados na etapa de Operação, deve-se estar pronto para escrever conclusões baseadas nestes dados coletados. Para escrever conclusões válidas, estes dados devem ser interpretados (WOHLIN *et al.*, 2000). A interpretação dos dados nos leva de volta às nossas questões originais. Neste momento deve-se entender e explicar os limites do estudo: que conclusões podem-se definir? Onde está o limite destas conclusões? O que pode ter influenciado os resultados? É importante também tentar explicar que questões foram respondidas e não simplesmente apresentar os dados. Deve-se também discutir o significado prático dos resultados (PERRY *et al.*, 2000 appud AMARAL, 2003).

Dois abordagens para analisar os dados são: análise quantitativa e análise qualitativa.

A Análise Quantitativa trata principalmente com a comparação de dados numéricos. As comparações tipicamente objetivam rejeitar ou não a hipótese nula. Dois métodos utilizados nesta análise são o Teste de Hipóteses e a Análise de Potência. A Análise Qualitativa tende a usar os dados que sejam menos quantificáveis, como observações, entrevistas, dentre outros. Estas técnicas tendem a ser utilizadas quando se quer entender as perspectivas das pessoas sobre uma situação (PERRY *et al.*, 2000 appud AMARAL,2003).

2.5.5 – Etapa de Empacotamento

Nesta etapa são registradas as informações, artefatos e conclusões sobre o estudo experimental e são definidas as políticas de como estes estarão disponíveis para a Comunidade de Engenharia de Software. Por isto, pacotes de experimentos, contendo todos os artefatos e documentos produzidos e utilizados ao longo da

execução de estudos experimentais, precisam ser construídos (AMARAL & TRAVASSOS, 2001).

A questão do empacotamento permite a repetição e comprovação dos resultados por outros pesquisadores (AMARAL & TRAVASSOS, 2001). Para estudos isolados é difícil entender como generalizar os resultados e, desta forma, como avaliar a sua verdadeira contribuição para a Engenharia de Software.

Infelizmente muitos estudos tendem a ser isolados e não são repetidos, ou pelo(s) mesmo(s) pesquisador(es) ou por outros (BASILI *et al.*, 1999). Para que tenham validade científica, estudos experimentais devem permitir sua repetição e comprovação.

2.6 - Abordagens Relacionadas à Experimentação

Um das principais expectativas em relação à Experimentação em Engenharia de Software é sua atuação como agente facilitador de melhoria de processo (OTT *et al.*, 1999) e da transferência de tecnologia da academia para indústria (SHULL *et al.*, 2001; SJØBERG *et al.*, 2002).

Sob essa perspectiva, algumas abordagens propostas na Engenharia de Software estão relacionadas à aplicação de uma metodologia de aquisição de conhecimento através da coleta e análise de dados. De um modo geral, essas abordagens buscam aplicar, em sua essência, um ciclo contínuo, os seguintes passos:

- Caracterizar o processo, o ambiente e os objetivos básicos de uma tecnologia, processo ou organização;
- Definir a partir dos objetivos, o que será medido no produto ou processo;
- Obter a coleta dos dados ou feedback do desenvolvedor, projeto ou processo;
- Comparar as práticas atuais com as novas ou detectar deficiências;
- Analisar os dados coletados;
- Propor uma melhoria futura para ser incorporada através de novas práticas ou aquisição de nova tecnologia.

A respeito do foco em melhoria de processo, podemos destacar abordagens como o Quality Improvement Paradigm – QIP (BASILI, 1985), a Fábrica de Experiência (BASILI *et al.*, 1994), *Knowledge Dust and Pearls* (BASILI *et al.*, 2001) e GQM (BASILI, 1992).

O QIP está relacionado ao conceito da Fábrica de Experiência através da definição de um ciclo completo para melhoria do processo. Fábrica de Experiência (BASILI *et al.*, 1994) é uma infra-estrutura que tem como objetivo armazenar e

reutilizar experiências no uso de ciclos de vida e produtos. É uma organização física e lógica, e suas atividades independem das atividades de desenvolvimento de uma empresa. Ela pode, além do armazenamento passivo dos dados experimentais, processar os pedidos do projeto atual oferecendo a informação relevante de projetos semelhantes (BASILI *et al.*, 1994).

Knowledge Dust and Pearls (Poeira e Pérolas do Conhecimento) é uma abordagem para a gerência de conhecimento de organizações de pesquisa de Engenharia de Software. Esta seria uma variação da abordagem da Fábrica de Experiência que endereça questões de aquisição de conhecimento a curto prazo para construção de memória organizacional.

O GQM é geralmente utilizado como abordagem de apoio ao programa de medição em organizações de software e pode ser utilizado como uma ferramenta para apoiar as demais abordagens.

Em relação à transferência de tecnologia para indústria, atribui-se o ceticismo da indústria em se convencer dos reais benefícios de uma tecnologia à distância entre a realidade das condições dos estudos experimentais e a realidade de um ambiente industrial. Neste sentido pesquisadores de Engenharia de Software possuem 3 desafios a enfrentar, para conduzir estudos experimentais mais realistas (HARRISON, 2000; SJØBERG *et al.*, 2002):

1. Tratar de tarefas de tamanho, complexidade, duração mais próximas à realidade encontrada nos processos de software reais (HOST *et al.*, 2000).
2. Selecionar participantes que representem a população-alvo, levando-se em conta que alguns estudos mostram que sob certas condições estudantes e profissionais não apresentam diferenças significativas (HOST *et al.*, 2000; CARVER *et al.*, 2003).
3. Realizar estudos experimentais em ambientes mais realísticos, mesmo sendo com tarefas mais realistas e participantes representando a população-alvo. Grande parte dos estudos não reproduz um ambiente experimental com uma configuração de ambiente industrial com tecnologia de apoio (processos, técnicas, ferramentas, etc.). Dificilmente exercícios com lápis e papel usados em salas de aula representarão a realidade para lidar com problemas relevantes de tamanho e complexidade da maioria dos sistemas contemporâneos (SJØBERG *et al.*, 2002).

No entanto, essa diferença entre os estudos experimentais realizados na academia e a realidade industrial brasileira vem sendo tratada também em processos

iterativos de transferência de tecnologias baseados na execução e pacotes de estudos experimentais (FREITAS *et al.*, 2004; HÖHN *et al.*, 2004; SILVA, 2004).

Uma metodologia de transferência de tecnologia iterativa baseada na execução de estudos experimentais é proposta por SHULL *et al.*, (2001). Segundo os autores, há muitos fatores que influenciam a transferência de tecnologia e que não a tornam uma tarefa trivial. Não importa o quão boa seja uma idéia, são esses fatores que influenciam sua utilidade. Muito desses fatores não podem ser determinados e controlados em ambientes de laboratório, tornando a inserção de processos industriais uma atividade de alto risco.

Por isso, SHULL *et al.*, (2001) partem da premissa que a definição de processos de transferência de tecnologia requer iteração para separar esses fatores e testá-los em grupos menores. Desta forma, se tem uma chance maior que os resultados gerem uma real compreensão da influência de variáveis do ambiente organizacional no uso de uma tecnologia. Uma segunda razão para a adoção de uma abordagem iterativa é garantir que questões fundamentais tenham sido tratadas em etapas anteriores antes de implantarmos uma tecnologia em um ambiente particular.

Os estudos de cada etapa da metodologia proposta estão intimamente relacionados aos riscos que podem ser assumidos e ao controle desejado de variáveis do ambiente em um determinado momento, e são (Figura 2.4): (1) Estudos de viabilidade; (2) Estudos de Observação; (3) Estudos de Caso – Ciclo de Vida Real,e; (4) Estudos de Caso – Indústria.

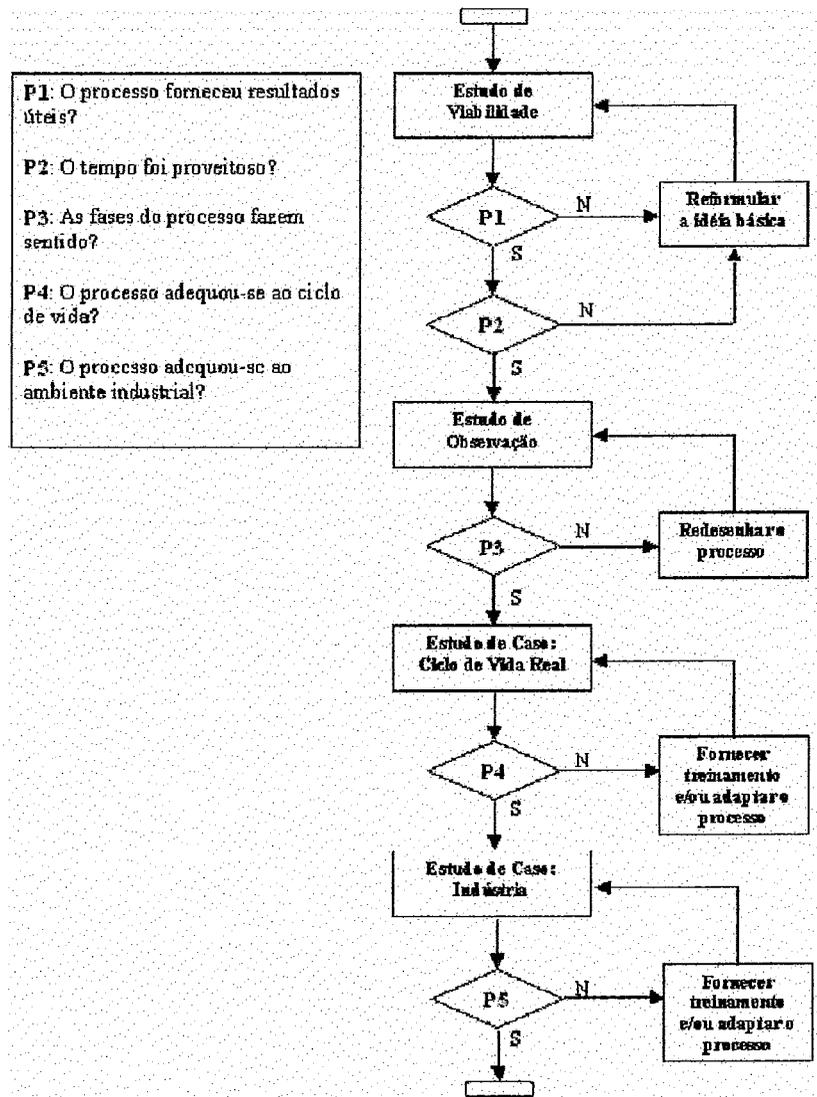


Figura 2.4 – Metodologia de introdução de tecnologias (SHULL *et al.*, 2001)

Os estudos de viabilidade, também conhecidos como pré-experimentos, possuem um planejamento de execução e coleta de dados, embora neste momento seja difícil o controle das variáveis envolvidas. Neste momento, busca-se explorar as questões fundamentais que motivaram a criação da tecnologia e não os seus detalhes. A preocupação é com a geração de hipóteses que possam ser testadas para a avaliação da tecnologia a ser transferida. Embora o controle das variáveis neste momento possa ser difícil, os riscos assumidos neste tipo de estudos são baixos em relação aos outros tipos de estudos por serem executados em ambientes fora dos meios de produção (laboratórios). A preocupação nessa etapa é avaliar se vale à pena empregar recursos para desenvolver a tecnologia estudada.

Nos estudos de observação, os participantes submetidos são observados enquanto desempenham uma determinada tarefa. Neste momento, a coleta de dados busca nos fornecer informação sobre como uma determinada tarefa é executada com a tecnologia sendo estudada. Desta forma, os pesquisadores podem obter uma melhor compreensão sobre como uma nova tecnologia é aplicada e quais as eventuais dificuldades enfrentadas na prática. Questões sobre necessidades de treinamento podem ser levadas em conta e avaliadas a partir dos resultados dos estudos de viabilidade. Embora estes estudos também sejam executados em laboratórios, há agora o controle de variáveis na tentativa de isolar e testar os fatores que influenciam o uso da tecnologia em questão. Estes estudos possuem maiores riscos associados do que os de viabilidade exatamente pela dificuldade de se reproduzir um ambiente com variáveis controladas.

Nos estudos de caso - Ciclo de Vida Real, podemos pré-supor que já há indícios suficientes que a tecnologia investigada é eficaz. Entretanto, o seu uso foi apenas feito em ambientes controlados, tornando difícil predizer o comportamento humano no uso da tecnologia na execução de tarefas no processo produtivo de uma organização. Estudos de casos são dispendiosos, pois os indivíduos precisam ser treinados. Deste modo, os riscos associados a esse tipo de estudo estão principalmente relacionados ao custo, à falta de controle do ambiente e do comportamento humano. Embora estes estudos sejam executados no ambiente organizacional, ainda há o isolamento dos processos produtivos organizacionais. Tais estudos funcionam como casos de teste de utilização da tecnologia dentro do ambiente organizacional. Da interação entre a tecnologia e o ambiente organizacional, problemas e deficiências podem ser identificados e tratados, levando-se em conta todo o conhecimento construído na análise dos dados das etapas anteriores. Neste momento, pode-se chegar à conclusão que a adoção de uma tecnologia pode ser inviável para determinados ambientes organizacionais.

Em estudos de caso - Indústria, o estudo será executado no processo produtivo de uma organização. Os riscos desse tipo de estudo são os mais altos, pois vão de custo de treinamento, à falta de controle das variáveis e do comportamento humano até os impactos negativos sobre a execução de tarefas na cadeia produtiva de uma organização. Todas as etapas anteriores buscaram minimizar esses riscos associados à introdução da nova tecnologia. As últimas avaliações podem ser feitas com o propósito de identificar eventuais pontos ainda não cobertos e se os resultados observáveis encontram-se dentro do esperado, para que medidas possam ser tomadas se necessário.

A Figura 2.5 ilustra o aumento dos riscos associados à execução dos vários tipos de estudos no decorrer tempo. Observa-se também o fato de que, em geral, à medida que os riscos aumentam, o controle sobre as variáveis do ambiente diminui.

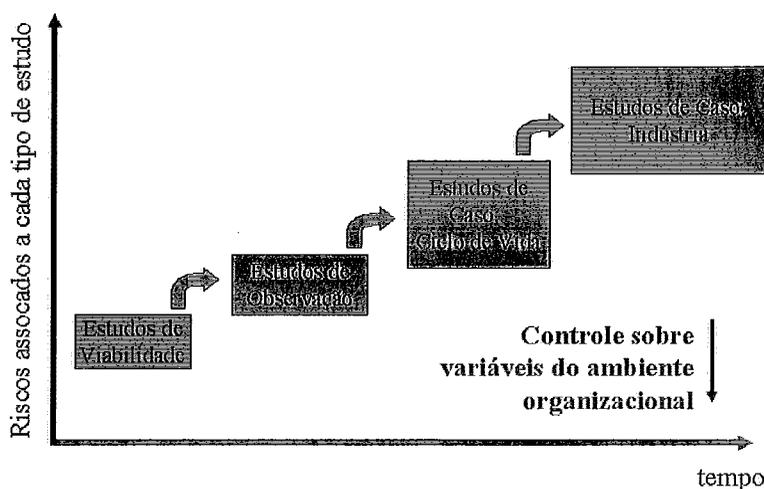


Figura 2.5 – Riscos associados em função do tempo decorrido

Esta metodologia foi utilizada como parte do processo de pesquisa para apoio ferramental para técnica de leitura PBR em SILVA (2004).

Entretanto, a utilização de abordagens iterativas traz uma série de outros novos desafios para os pesquisadores. Já que as iterações implicam em várias repetições de um mesmo estudo experimental.

Quando se utiliza o termo repetição, geralmente assume-se que isso implicará em repetir o estudo sem quaisquer mudanças. No escopo deste trabalho, uma repetição é um estudo executado baseado no mesmo projeto do estudo experimental e resultados de um estudo anterior, onde o objetivo é verificar ou ampliar a aplicabilidade dos resultados do estudo inicial. Se o pesquisador deseja verificar a aplicabilidade dos resultados em outro contexto, então, o projeto do estudo deve ser ligeiramente modificado, mas ainda será considerado uma repetição (SHULL *et al.*, 2002).

Baseado na experiência adquirida no Projeto Readers (SHULL *et al.*, 2002; SHULL *et al.*, 2004), um projeto de cooperação financiado por agências de fomento a pesquisa brasileira (Conselho Nacional de Desenvolvimento Científico e Tecnológico - CNPq) e americana (National Science Foundation - NSF), foram investigados desafios relacionados à repetição e transferência de conhecimento na execução de estudos experimentais.

Segundo SHULL *et al.* (2002), mesmo quando pacotes do experimento são efetivamente especificados e existem, pesquisadores sentem dificuldade ao

compreender e selecionar um estudo apropriado para repeti-lo. A principal dificuldade é compreender o conhecimento necessário sobre o objeto de estudo e como explicitar o conhecimento sobre a execução de um estudo experimental que não foi ou não pode ser registrado no pacote do experimento, para que este possa ser utilizado pelos pesquisadores que repetirem um estudo.

Além disso, dificilmente na repetição de um estudo experimental será possível reproduzir todas as variáveis do estudo original. O ambiente e a população de amostra serão diferentes, e talvez seja necessária a adequação dos instrumentos para estender a aplicabilidade do estudo original.

Deste modo, SHULL *et al.* (2002) atentam para a necessidade de se desenvolver mecanismos para compreensão e definição do processo de transferência de conhecimento experimental e evolução dos artefatos de um pacote de experimento, através de estruturas de colaboração para facilitar a interação entre pesquisadores.

Na Engenharia de Software Experimental, o desafio central é compartilhar eficientemente conhecimento experimental entre pesquisadores de modo a facilitar a comparação entre repetições. A absorção do conhecimento tácito foi a maior dificuldade durante as replicações realizadas no Projeto Readers (SHULL *et al.*, 2004).

Em SHULL *et al.* (2004), é realizada uma adaptação do modelo em espiral de evolução do conhecimento de Nonaka-Takeuchi (NONAKA & TAKEUCHI, 1995) para o contexto de experimentação em Engenharia de Software. A esta adaptação denomina-se Experimentation Knowledge-Sharing Model (EKSM).

Em síntese, o EKSM age de forma iterativa na questão do compartilhamento de conhecimento entre pesquisadores em cada uma de suas fases. Na primeira fase, o conhecimento tácito é compartilhado através da socialização entre pesquisadores. A seguir esse conhecimento tácito, quando possível, é explicitado e incorporado ao pacote de experimento. Por fim o conhecimento explícito pode ser melhorado e internalizado, e, assim, utilizado na evolução do pacote de experimento.

2.7 – Conclusão

Neste capítulo foram apresentados tópicos relacionados à Experimentação em Engenharia de Software, dentre eles:

A definição dos principais conceitos e termos (hipótese, fator, tratamento, etc.) utilizados na execução de estudos experimentais em Engenharia de Software;

Uma visão geral do Processo de Experimentação proposto por AMARAL (2003), mostrando suas principais tarefas e questões, tais como, a ameaça à validade, formulação de hipóteses, o realismo em estudos experimentais

e a seleção de participantes. Além da introdução do conceito de pacote de experimento.

Uma visão geral sobre métodos de pesquisa qualitativa e quantitativa, utilizados na pesquisa experimental em Engenharia de Software, elucidando seus principais objetivos e diferenças;

A discussão sobre algumas taxonomias para classificação dos estudos experimentais na Engenharia de Software encontradas na literatura, com respectivos exemplos, e;

A citação de algumas abordagens relacionadas à Experimentação em Engenharia de Software, utilizadas como agentes facilitadores da melhoria de processo e transferência de tecnologia.

Por fim, vale ressaltar que devido às características específicas da Experimentação aplicada na Engenharia de Software, tais como recursos de hardware/software e disponibilidade de participantes, a etapa de Execução é a atividade mais crítica do Processo de Experimentação.

Após o Planejamento, uma falha na etapa de Execução compromete todo o resultado do estudo e, via de regra, os invalida. Os principais riscos estão relacionados à escassez de participantes para realização do estudo experimental. Deste modo, algumas lições aprendidas no Projeto Readers têm sido aplicadas nos estudos experimentais executados pela Equipe de Engenharia de Software Experimental da COPPE/UFRJ, dentre elas: (1) a execução de estudos pilotos para antecipar desvios em relação ao plano experimental e adequá-lo a esses desvios; (2) treinar os pesquisadores tanto no objeto de estudo quanto no conhecimento experimental necessário para repetirem um determinado estudo; (3) selecionar revisores externos (indivíduos não envolvidos no planejamento e na execução do estudo) para revisarem os artefatos produzidos até então, antes de efetivamente se executar o estudo e, assim, antecipar desvios e adequar o plano experimental.

Capítulo 3 – Infra-Estrutura para Planejamento, Execução e Empacotamento de Estudos Experimentais em Engenharia de Software

Neste capítulo são apresentados as principais motivações e o escopo que nortearam este trabalho, juntamente com as principais características de um processo e aspectos de desenvolvimento de um protótipo que utiliza a infra-estrutura Web para apoiar o Processo de Experimentação.

3.1 – Motivação

Como observado nos capítulos anteriores, executar estudos experimentais em Engenharia de Software é uma tarefa complexa. Muitos são os fatores complicadores, tais como: as restrições quanto à representatividade das amostras utilizadas e aplicabilidade dos resultados em diferentes contextos.

A repetição dos estudos executados vem sendo utilizada como uma possível abordagem para minimizar tais problemas. Mas executar estudos experimentais em Engenharia de Software gera muitos artefatos, que são produzidos ao longo do Processo de Experimentação e que precisam ser empacotados para viabilizar futuras repetições.

Deste modo, destacamos a necessidade de se fornecer mecanismos de colaboração entre pesquisadores permitindo o acesso aos artefatos e ao conhecimento relativos a um estudo experimental (SHULL *et al.*, 2002; SHULL *et al.*, 2004). Assim, construir uma base de estudos experimentais ajudaria a consolidar um corpo de conhecimento sobre tecnologias utilizadas nos processos de software (PFLEEGER, 1999; AMARAL, 2003).

Neste contexto, podemos citar iniciativas na tentativa de se consolidar conhecimento baseado na execução de estudos experimentais, tais como o CeBASE (<http://www.cebase.org>) e AMARAL (2003).

Em AMARAL (2003) é definido um Processo de Experimentação e Empacotamento de Experimentos, e também foi construído um protótipo sobre um portal *Web* para que os pesquisadores pudessem fazer o acompanhamento e empacotamento dos artefatos produzidos em cada uma das atividades do processo. Entretanto, este protótipo, apesar de levar em conta o processo proposto, não fornece

nenhum apoio à execução das tarefas relativas a cada atividade. Nesta infra-estrutura é apenas fornecido um guia sobre que informações e artefatos devem ser produzidos e empacotados em uma determinada atividade. Além disso, para executar a repetição de um estudo experimental é necessário re-executar todo o processo e re-empacotar todos os artefatos como se fosse um novo estudo.

O **CeBASE** é um portal disponível na *Web* com uma base de conhecimento, onde o material relativo a estudos experimentais executados pode ser disponibilizado e estar acessível para que a Comunidade de Engenharia de Software possa acessá-lo e, se houver interesse, repetí-lo.

Já no apoio às atividades do Processo de Experimentação foi encontrada uma iniciativa de uma aplicação para permitir e controlar a execução de experimentos controlados através da seleção de participantes geograficamente distribuídos na *Web*, O **SESE - Simula Experiment Support Environment** (ARISHOLM *et al.*, 2002a; ARISHOLM *et al.*, 2002b; KARAHASANOVIC *et al.*, 2005). Em tal aplicação os pesquisadores registram as informações definidas no planejamento, e a seguir organizam, controlam e monitoram a execução de um experimento controlado bem como os recursos que vão ser utilizados, distribuídos e recolhidos dos participantes através da *Web*, através da utilização de um portal. Entretanto, esta proposta não leva em consideração as questões do uso de uma taxonomia de artefatos para o empacotamento de estudos e da repetição de estudos.

Nesta dissertação é proposta uma infra-estrutura *Web* que permite múltiplas instanciações de um processo para apoiar a execução e repetição de estudos experimentais, levando-se em consideração a taxonomia de artefatos para empacotamento de experimentos proposto por AMARAL (2003).

3.2 – Escopo do Trabalho

Segundo ARISHOLM *et al.* (2002b), típicos procedimentos na execução de estudos experimentais são:

1. Definir o estudo experimental: Projetar o estudo experimental com os questionários para coletar informações colhidas dos participantes, recursos de software e hardware, descrições de tarefas, e arquivos.
2. Definir, caracterizar e atribuir participantes: Definir o tipo e número de participantes e recrutá-los. Tipicamente um estudo experimental controlado consiste de dois ou mais tratamentos. Os tratamentos apropriados devem ser atribuídos aos respectivos grupos de participantes.

3. Cada participante executa o experimento: Os artefatos e recursos definidos no primeiro passo são distribuídos aos participantes de modo que comecem o estudo experimental.
4. Monitorar o experimento: Para garantir que os participantes executam corretamente a(s) tarefa(s) atribuída(s) e que os dados apropriados sejam coletados. O pesquisador monitora o progresso de cada participante.
5. Coleta de Resultados: Quando um participante termina a(s) tarefa(s) os seus resultados são coletados e armazenados em um repositório seguro. Quando todos os participantes terminam, o pesquisador pode começar a análise desses resultados agregados.

De modo bem simplificado, esses passos caracterizam em um alto nível de abstração algumas das tarefas que impactam todo processo de execução de um estudo experimental.

Como estratégia de limitação de escopo para o desenvolvimento deste trabalho, buscou-se a definição de uma infra-estrutura que pudesse apoiar de alguma forma todos os 5 passos citados acima. Entretanto, foram os passos 2, 3 e 4, aqueles intrinsecamente relativos à Etapa de Execução do Processo de Experimentação proposto por (WOHLIN *et al.*, 2000), os mais explorados e detalhados. Os passos 1 e 5 são contemplados a nível de gerenciamento e organização dos artefatos de modo a garantir o devido empacotamento e repetição dos experimentos.

Nas seções seguintes, será discutido o processo proposto para apoiar a execução e repetição de estudos experimentais, detalhando-o a nível de atividades, recursos (insumos e produtos), e papéis.

3.3 – Descrição do Processo de Apoio ao Planejamento e Execução de Estudos Experimentais em Engenharia de Software

Segundo ARISHOLM *et al.* (2002b), a utilização de recursos da *Web* para apoiar a execução de estudos experimentais pode ajudar a torná-las mais próximas das situações encontradas na indústria e a contornar vários problemas de logística, tais como a execução de tarefas por equipes geograficamente distribuídas, ou em seu próprio ambiente de trabalho. Isso facilitaria também o recrutamento, seleção e participação de participantes, permitindo o que os autores denominam de experimentos em larga escala (*large-scale experiments*), vislumbrando diminuir o problema de representatividade de amostras e aumentar a confiabilidade dos resultados observados.

Deste modo, como parte integrante deste trabalho, foi proposto um processo que utilizasse os recursos disponíveis na *Web* (notificação via e-mail, páginas e portais com controle de acesso, e *download* e *upload* de materiais, etc) para apoiar a execução de experimentos. Foram levados em consideração na definição desse processo fontes como AMARAL (2003), JURISTO & MORENO (2001) e WOHLIN *et al.* (2000). Além da experiência dos integrantes da Equipe de Engenharia de Software Experimental COPPE/UFRJ e pesquisadores de Engenharia de Software do Projeto Readers, onde tal processo foi debatido em reuniões de pesquisa e *workshops*.

Este processo é composto pelas seguintes macro-atividades: Definir tipo de Execução, Registrar Documentos de Definição e Planejamento, Registrar Instrumentos, Definir Tratamentos, Avaliar Estudo, Obter Consentimento e Caracterização dos Participantes, Definir Distribuição, Disponibilizar Treinamento, Distribuir Material de Execução, Coletar Dados, Registrar Documentos da Análise.

A seguir serão descritas e formalizadas em termos dos papéis desempenhados, artefatos produzidos e consumidos (insumos e produtos), entre outras características, as atividades do processo proposto. Na formalização das atividades foi utilizado um *template* de descrição de atividades de processos de software utilizada pela linha de Engenharia de Software no Programa de Engenharia de Sistemas e Computação da COPPE/UFRJ. Estas informações foram utilizadas para nortear as etapas seguintes do desenvolvimento da infra-estrutura.

3.3.1 – Definir Tipo de Execução

Esta consiste na primeira atividade do processo. Nela o pesquisador tem acesso aos experimentos empacotados já executados na infra-estrutura. O objetivo dessa atividade é, basicamente, definir se o processo de apoio à execução será utilizado na execução de um novo experimento ou se trata de uma repetição a ser executada. A única diferença é que se tratar-se de uma repetição, vários artefatos do Pacote do Experimento serão reaproveitados para as atividades posteriores e disponibilizados para o pesquisador para as devidas evoluções, se necessárias, conforme veremos na descrição das próximas atividades. Nesta atividade é que ocorre a instanciação de todo arcabouço estrutural para que o devido empacotamento dos artefatos seja feito ao longo da execução, como no processo proposto por AMARAL (2003). A descrição desta atividade pode ser vista na Tabela 3.1.

Tabela 3.1- Definir Tipo de Execução

Atividade	Definir Tipo de Execução
Descrição	O pesquisador do estudo experimental escolhe por repetir ou executar um novo estudo experimental.
Obrigatoriedade	Esta é a primeira atividade e obrigatória.
Super-atividade	-
Insumos	-
Produtos	Arcabouço que irá gerenciar os artefatos durante a execução do estudo experimental. Se for uma repetição, todos os documentos, tratamentos e instrumentos do estudo original serão copiados para este arcabouço.
Papeis	Pesquisador do estudo experimental.
Pré-condições	Caso seja uma repetição, a indicação do pacote do experimento do estudo original.
Pré-atividades	-
Sub-atividades	-
Outros comentários	-

3.3.2 – Registrar Documentos de Definição e Planejamento

Esta atividade tem o intuito de registrar na infra-estrutura os documentos com as informações sobre a definição e planejamento de um experimento. De modo que o principal documento a ser registrado é o plano do estudo experimental (AMARAL, 2003). Se o experimento for um estudo ainda não executado, o pesquisador fará o registro dessas informações através de documentos que será realizada através de *uploads* de arquivos, que poderão ser produzidos em qualquer ferramenta e formato de conveniência. Caso seja a repetição de um experimento, a infra-estrutura cria duplicatas dos documentos originais e os disponibiliza ao pesquisador para que este atualize, evolua ou substitua-os, através do *downloads* e/ou subseqüentes *uploads*. A formalização desta atividade pode ser vista na Tabela 3.2.

Tabela 3.2- Registrar Documentos da Definição e Planejamento

Atividade	Registrar Documentos da Definição e Planejamento
Descrição	Se for a execução de um novo estudo, o pesquisador do estudo experimental deve registrar os documentos já produzidos pelas etapas de Definição e Planejamento. Se for uma repetição, os artefatos copiados do estudo original são disponibilizados ao pesquisador para as evoluções necessárias.
Obrigatoriedade	Obrigatória.
Super-atividade	-
Insumos	Documentos da Definição e Planejamento.
Produtos	Documentos da Definição e Planejamento registrados e disponíveis no infra-estrutura.
Papeis	Pesquisador do estudo experimental.
Pré-condições	Caso seja uma repetição, todos os documentos de um experimento executado registrados na infra-estrutura.
Pré-atividades	Definir Tipo de Execução
Sub-atividades	-
Outros comentários	-

3.3.3 – Registrar Instrumentos

Nesta atividade os instrumentos (Artefatos de Software, do Experimento, Formulários de Caracterização e Consentimento) serão registrados na infra-estrutura pelo pesquisador. Artefatos de Software e do Experimento serão registrados na forma de arquivos (com extensões doc, pdf, xml, tif, jpeg, ps, bmp, wmv, mpeg, etc). Já os Formulários de Consentimento e de Caracterização serão registrados na forma de formulários HTML (com extensões html, htm), de modo que as informações que irão ser submetidas durante o seu preenchimento serão utilizadas para apoiar a distribuição do material aos participantes. Estes artefatos compreendem o material utilizado na execução do estudo experimental, tais como formulários, artefatos de projetos, material de treinamento e componentes. Se for a repetição de um estudo, os instrumentos serão duplicados a partir dos instrumentos do estudo original e estarão disponíveis para evoluções necessárias, assim como descrito na atividade anterior (através de *downloads* e *uploads* de arquivos). A formalização desta atividade pode ser vista na Tabela 3.3.

Tabela 3.3- Registrar Instrumentos

Atividade	Registrar Instrumentos
Descrição	Os instrumentos (Artefatos de Software, do Experimento, Formulários de Caracterização e Consentimento) serão registrados na infra-estrutura pelo pesquisador do estudo experimental. Artefatos de Software e do Experimento serão registrados na forma de arquivos (com extensões doc, pdf, xml, tif, jpeg, ps, bmp, wmv, mpeg, etc). Já os Formulários de Consentimento e de Caracterização serão registrados na forma de formulários HTML (com extensões html, htm). Estes artefatos compreendem o material utilizado na execução do estudo experimental, tais como formulários, artefatos de projetos e componentes. Se for a repetição de um estudo, os instrumentos copiados do estudo original estarão disponíveis para evoluções necessárias.
Obrigatoriedade	Obrigatória.
Super-atividade	-
Insumos	Instrumentos planejados e confeccionados.
Produtos	Os instrumentos do estudo experimental registrados na infra-estrutura.
Papeis	Pesquisador do Estudo Experimental.
Pré-condições	Caso seja uma repetição, todos os instrumentos de um experimento executado registrados na infra-estrutura.
Pré-atividades	Registrar Documentos da Definição e Planejamento
Sub-atividades	-
Outros comentários	-

3.3.4 – Definir Tratamentos

Esta atividade tem o intuito de permitir, depois de registrados os instrumentos, que o pesquisador descreva na infra-estrutura os tratamentos a serem aplicados durante a execução do experimento. Lembrando que um tratamento é a organização de como serão distribuídos os instrumentos numa rodada de execução do estudo experimental. As informações sobre os tratamentos a serem aplicados já devem ter sido planejadas no plano do estudo experimental. No caso de ser a execução de uma

repetição, as descrições dos tratamentos do estudo original são duplicadas e disponibilizadas para que o pesquisador opte por seu reaproveitamento ou exclusão; ou a descrição adicional de novos tratamentos. A formalização desta atividade pode ser vista na Tabela 3.4.

Tabela 3.4- Definir Tratamentos

Atividade	Definir Tratamentos
Descrição	Depois de registrados os instrumentos, o pesquisador do estudo experimental deve descrever os tratamentos a serem aplicados na execução do estudo experimental. Um tratamento é a organização de como serão distribuídos os instrumentos numa rodada de execução do estudo experimental. As informações sobre os tratamentos a serem aplicados já devem estar planejadas. No caso de uma repetição, os tratamentos do estudo original devem poder ser reaproveitados.
Obrigatoriedade	Obrigatória.
Super-atividade	-
Insumos	Instrumentos registrados do Estudo Experimental.
Produtos	Tratamentos do estudo experimental.
Papeis	Pesquisador do estudo experimental
Pré-condições	Caso seja uma repetição, todos os tratamentos de um experimento executado registrados na infra-estrutura.
Pré-atividades	Registrar Instrumentos
Sub-atividades	-
Outros comentários	-

3.3.5 – Avaliar Estudo

O intuito desta atividade é permitir ao pesquisador avaliar o experimento através da execução de um estudo piloto e/ou através da seleção de revisores externos. Cada uma dessas avaliações produz um laudo. Através desses laudos, o pesquisador decide se alterações nos artefatos são necessárias. Se forem necessárias alterações nos artefatos, retornamos a atividade “Registrar Documentos da Definição e Planejamento do Processo”. Caso contrário o processo segue para as atividades “Obter Consentimento e Caracterização dos Participantes” e “Disponibilizar Treinamento”. Esta atividade é composta de três sub-atividades: Executar Piloto,

Revisar Estudo e Avaliar Resultados. A formalização desta atividade pode ser vista na Tabela 3.5.1.

Tabela 3.5.1- Avaliar Estudo

Atividade	Avaliar Estudo
Descrição	O pesquisador do estudo experimental pode avaliar a execução do estudo experimental através da execução de um estudo piloto e/ou a avaliação de um revisor externo. Cada uma dessas avaliações produz um laudo. Através desses laudos o pesquisador do estudo experimental decide se alterações nos artefatos são necessárias. Se forem necessárias alterações nos artefatos, retornamos a atividade "Registrar Documentos da Definição e Planejamento do Processo". Caso contrário, o processo continua.
Obrigatoriedade	Opcional.
Super-atividade	-
Insumos	Documentos da Definição e Planejamento, e Tratamentos do Estudo Experimental descritos.
Produtos	Laudo de Avaliação do Estudo Experimental.
Papeis	Pesquisador do Estudo Experimental e/ou Revisor Externo.
Pré-condições	-
Pré-atividades	Definir Tratamentos
Sub-atividades	Executar Piloto Revisar Estudo Avaliar Resultados
Outros comentários	-

Na atividade Executar Piloto, pode-se avaliar a execução do estudo experimental através da execução de um estudo piloto. Nela, o pesquisador escolhe participantes fora da população de sua amostra e lhes atribui o(s) tratamento(s) para que estes executem uma determinada tarefa. Antes, o pesquisador pode optar por obter ou não a caracterização e consentimento desses participantes através de formulários *on-line*. Os resultados e respostas provenientes da caracterização, consentimento e rodada experimental desta execução serão registrados via *browser* pelos próprios participantes que são sempre notificados por *e-mail* com as instruções necessárias em cada uma das tarefas a serem realizadas. Após a execução do piloto,

o pesquisador registrará um Laudo da Execução do Piloto e emitirá explicitamente sua opinião se o estudo apresenta problemas e se pode ser executado como planejado. A formalização desta atividade pode ser vista na Tabela 3.5.2.

Tabela 3.5.2- Executar Piloto

Atividade	Executar piloto
Descrição	O pesquisador do estudo experimental pode avaliar a execução do estudo experimental através da execução de um estudo piloto. O pesquisador do estudo experimental escolhe participantes fora da população de sua amostra e lhes atribui o(s) tratamento(s) para que estes executem uma determinada tarefa. Antes, o pesquisador pode optar por obter ou não a caracterização e consentimento desses participantes. Os resultados e repostas provenientes da execução serão registrados pelos próprios participantes via <i>browser</i> . Após a execução do piloto, o pesquisador registrará um Laudo da Execução do Piloto e emitirá explicitamente sua opinião se o estudo apresenta problemas ou se pode ser executado como planejado.
Obrigatoriedade	Opcional.
Super-atividade	Avaliar Estudo
Insumos	Documentos da Definição e Planejamento, e Tratamentos do Estudo Experimental descritos.
Produtos	Laudo da Avaliação do Estudo Piloto.
Papeis	Pesquisador do Estudo Experimental.
Pré-condições	-
Pré-atividades	Definir Tratamentos
Sub-atividades	-
Outros comentários	-

Na atividade Revisar Estudo, o pesquisador pode avaliar a execução do estudo experimental através de uma solicitação revisor(es) externo(s). Um revisor externo recebe uma notificação e tem acesso ao material do estudo experimental para realizar a avaliação via *browser*. O revisor deve avaliar a conformidade entre o planejamento do experimento e os artefatos que serão utilizados para executá-lo, e então registrar

um Laudo de Revisão do Estudo Experimental, onde emitirá explicitamente sua opinião se o estudo apresenta problemas e se pode ser executado como planejado. A formalização desta atividade pode ser vista na Tabela 3.5.3.

Tabela 3.5.3- Revisar Estudo

Atividade	Revisar Estudo
Descrição	O pesquisador do estudo experimental pode avaliar a execução do estudo experimental através de uma solicitação a um revisor externo. O revisor externo recebe uma notificação e tem acesso ao material do estudo experimental para realizar a avaliação. O revisor deve registrar um Laudo de Revisão do Estudo Experimental e emitirá explicitamente sua opinião se o estudo apresenta problemas ou se pode ser executado como planejado. Assim que o avaliador externo efetuar o registro da avaliação, o pesquisador do estudo experimental deve ser notificado.
Obrigatoriedade	Opcional.
Super-atividade	Avaliar Estudo
Insumos	Documentos da Definição e Planejamento, e Tratamentos do Estudo Experimental descritos.
Produtos	Laudo da Revisão do Estudo Experimental.
Papeis	Pesquisador do Estudo Experimental e/ou Revisor Externo.
Pré-condições	-
Pré-atividades	Descrever Tratamentos
Sub-atividades	-
Outros comentários	-

A formalização destas sub-atividades está fortemente relacionada às lições aprendidas durante o projeto Readers CNPq/NSF, onde detalhes podem ser obtidos em (SHULL *et al.*, 2002).

Encerradas as duas atividades anteriores, o pesquisador irá para a atividade Avaliar Resultados. Nela o pesquisador avalia os resultados do Laudo de Execução do Piloto e/ou o Laudo da Revisão do Estudo Experimental e registra um Laudo de Avaliação do Estudo Experimental e emite explicitamente sua opinião se o estudo apresenta problemas ou se pode ser executado como planejado. Sendo nesta

atividade o pesquisador decide se o processo continua preparando-se para a execução experimental ou se os artefatos devem ser corrigidos. A formalização desta atividade pode ser vista na Tabela 3.5.4.

Tabela 3.5.4- Avaliar Resultados

Atividade	Avaliar Resultados
Descrição	O executor do estudo experimental avalia o Laudo de Execução do Piloto e/ou o Laudo da Revisão do Estudo e registra um Laudo de Avaliação do Estudo Experimental e emite explicitamente sua opinião se o estudo apresenta problemas ou se pode ser executado como planejado.
Obrigatoriedade	Opcional.
Super-atividade	Avaliar Estudo
Insumos	Laudo da Avaliação da Execução do Piloto e Laudo da Revisão do Estudo Experimental.
Produtos	Laudo da Avaliação do Estudo Experimental.
Papeis	Executor do Estudo Experimental e/ou Revisor Externo.
Pré-condições	-
Pré-atividades	Executar Piloto e Revisar Estudo.
Sub-atividades	-
Outros comentários	-

3.3.6 – Obter Consentimento e Caracterização dos Participantes

Esta atividade tem o intuito de permitir que o pesquisador indique o momento em que os formulários de consentimento e de caracterização serão disponibilizados na *Web* e acessados pelos participantes via *browser*. Cada participante recebe uma notificação por *e-mail* com os endereços eletrônicos na *Web* (URLs) e instruções específicas a seguirem. É neste momento em que ocorre o recrutamento dos participantes. Deste modo, a notificação pode ser enviada para indivíduos específicos ou listas de discussões via *e-mail*. Além disso, o pesquisador, nesta atividade, passa a ser capaz de monitorar as respostas e visualizar os dados da caracterização e consentimento de cada um dos participantes. A formalização desta atividade pode ser vista na Tabela 3.6.

Tabela 3.6- Obter Consentimento e Caracterização dos Participantes

Atividade	Obter Consentimento e Caracterização dos Participantes
Descrição	O pesquisador do estudo experimental indica o momento em que os formulários de consentimento e de caracterização serão disponibilizados via <i>browser</i> . A infra-estrutura notifica ao pesquisador do estudo os endereços das páginas na <i>Web</i> que devem ser acessadas pelos participantes via <i>browser</i> . A infra-estrutura deve ser capaz de notificar os participantes por email se o pesquisador assim desejar. A caracterização e o consentimento dos participantes são empacotados em um artefato de medição onde os dados submetidos são extraídos.
Obrigatoriedade	Obrigatória.
Super-atividade	-
Insumos	Formulários de Consentimento e de Caracterização dos Participantes.
Produtos	Caracterização dos Participantes e Consentimento dos Participantes.
Papeis	Pesquisador do estudo experimental.
Pré-condições	-
Pré-atividades	Avaliar Estudo.
Sub-atividades	-
Outros comentários	-

3.3.7 – Disponibilizar Treinamento

Esta atividade é referente à disponibilização do material de treinamento aos participantes. Nela o pesquisador tem acesso ao material de treinamento via *browser* para acessá-lo quando for realizar o treinamento dos participantes, que foi registrado e empacotado na atividade Registrar Instrumentos. É dada também ao pesquisador a opção de enviar este material através de *e-mail's* a todos os participantes. Esta opção foi influenciada pelos estudos executados por (LIMA, 2005), onde o material de treinamento foi diretamente disponibilizado aos participantes sem a intervenção do pesquisador antes da execução das tarefas do experimento, onde ocorreu o denominado auto-treinamento dos participantes. Esta atividade ocorre em paralelo à seqüência de atividades Obter Consentimento e Caracterização dos Participantes e

Definir Distribuição para permitir que o pesquisador indique o momento em que deverá ser disponibilizado o material de treinamento, podendo isto ocorrer antes, após ou ao longo destas outras atividades conforme planejado no experimento. A formalização desta atividade pode ser vista na Tabela 3.7.

Tabela 3.7- Disponibilizar Treinamento

Atividade	Disponibilizar Treinamento
Descrição	Refere-se ao treinamento necessário dado aos participantes. Nesta atividade o pesquisador do estudo pode acessar o material via browser e/ou enviar uma notificação via e-mail para que os participantes acessem o material de treinamento via <i>browser</i> .
Obrigatoriedade	Opcional.
Super-atividade	-
Insumos	Instrumentos empacotados que compõem o material de Treinamento.
Produtos	Nenhum
Papeis	Pesquisador do Estudo Experimental.
Pré-condições	-
Pré-atividades	Avaliar Estudo.
Sub-atividades	-
Outros comentários	-

3.3.8 – Definir Distribuição

A partir dos dados da caracterização dos participantes e definição dos tratamentos produzidos em atividades anteriores, o pesquisador do estudo experimental tem que descrever como ocorrerá a distribuição do material aos participantes. Nesta atividade, o pesquisador indica os tratamentos que cada participante receberá em cada rodada do experimento. Nesta atividade não é indicada a ordem em que ocorrerá as distribuições dos artefatos relativos a cada rodada do experimento. A formalização desta atividade pode ser vista na Tabela 3.8.

Tabela 3.8- Definir distribuição

Atividade	Definir Distribuição
Descrição	A partir da caracterização dos participantes e dos tratamentos definidos, o pesquisador do estudo experimental tem que descrever como ocorrerá a distribuição do material aos participantes. Essas informações serão posteriormente utilizadas na atividade "Distribuir Material de Execução". Cada repetição interna do estudo experimental deve ser descrita.
Obrigatoriedade	Obrigatória.
Super-atividade	-
Insumos	Tratamentos e Caracterização dos Participantes.
Produtos	Relação de distribuição de tratamentos aos participantes para cada repetição interna do experimento.
Papeis	Pesquisador do estudo experimental
Pré-condições	-
Pré-atividades	Obter Consentimento e Caracterização dos Participantes
Sub-atividades	-
Outros comentários	-

3.3.9 – Distribuir Material de Execução

Neste momento os participantes já foram recrutados, caracterizados e treinados, e a definição da distribuição de cada rodada do experimento já foram definidos. Então o pesquisador indica o momento e ordem que cada rodada de distribuição do material deve ser aplicada. Uma indicação desta acarreta na notificação a cada participante para que acesse e baixe via *browser* o material para execução, segundo o tratamento que lhe foi atribuído numa determinada rodada de execução. Nesta notificação também são enviadas instruções para os participantes de como proceder para enviar os artefatos com os resultados do experimento, relativas à coleta de dados. A formalização desta atividade pode ser vista na Tabela 3.9.

Tabela 3.9- Distribuir Material de Execução

Atividade	Distribuir Material de Execução
Descrição	O pesquisador indica o momento e ordem que cada relação de distribuição deve ser aplicada. Uma indicação desta acarreta na notificação a cada participante para que acesse e baixe via <i>browser</i> o material para execução segundo o tratamento que lhe foi atribuído numa determinada rodada de execução.
Obrigatoriedade	Obrigatória.
Super-atividade	-
Insumos	Relações de Distribuições do Material.
Produtos	Nenhum
Papeis	Pesquisador do Estudo Experimental.
Pré-condições	-
Pré-atividades	Disponibilizar Treinamento e Definir Distribuição
Sub-atividades	-
Outros comentários	-

3.3.10 – Coletar Dados

Nesta atividade o pesquisador deve monitorar os artefatos com resultados do experimento que são submetidos e estão disponíveis para a verificação. Os artefatos retornados são submetidos pelos participantes através de formulários disponíveis na *Web* e acessados via *browser*. As instruções de como realizar essa submissão dos resultados é dada a cada rodada do experimento aos participantes na notificação de *e-mail* da atividade descrita anteriormente. A formalização desta atividade pode ser vista na Tabela 3.10.

Tabela 3.10- Coletar Dados

Atividade	Coletar Dados
Descrição	Consiste na coleta dos dados crus do experimento. Os participantes farão o registro dos artefatos com os dados em um formulário via <i>browser</i> .
Obrigatoriedade	Obrigatória.
Super-atividade	-
Insumos	Nenhum
Produtos	Artefatos de Medição e Dados Cru do Experimento
Papeis	Participantes do Estudo Experimental.
Pré-condições	-
Pré-atividades	Disponibilizar Treinamento e Definir Distribuição
Sub-atividades	-
Outros comentários	-

3.3.11 – Registrar Documentos da Análise

Esta atividade tem o intuito de permitir que o pesquisador registre na infraestrutura os documentos com as informações sobre a análise dos resultados do experimento. Se o experimento for um estudo ainda não executado, o pesquisador fará o registro dessas informações através de documentos, produzidos em qualquer ferramenta e formato de conveniência, através de *uploads* de arquivos na infraestrutura. Caso seja a repetição de um experimento, a infraestrutura cria duplicatas dos documentos originais e os disponibiliza ao pesquisador para que os utilize como padrão ou substitua por outros, através do *downloads* e/ou subseqüentes *uploads*. A formalização desta atividade pode vista ser na Tabela 3.11.

Tabela 3.11- Registrar Documentos da Análise

Atividade	Registrar Documentos da Análise
Descrição	Se for a execução de um novo estudo, o pesquisador do estudo experimental deve registrar os documentos já produzidos pela Etapa de Análise. Se for uma repetição, os documentos registrados do estudo original são copiados e disponibilizados ao pesquisador para evoluções necessárias.
Obrigatoriedade	Essa atividade é obrigatória na execução do workflow.
Super-atividade	-
Insumos	Documentos da Análise.
Produtos	Documentos da Análise registrados e disponíveis no infraestrutura.
Papeis	Pesquisador do estudo experimental.
Pré-condições	-
Pré-atividades	Coletar Dados e Distribuir Material de Execução.
Sub-atividades	-
Outros comentários	-

3.3.12 – Diagrama de Atividades do Processo Proposto

A figura 3.1 mostra o diagrama de atividades que descreve as relações de dependências entre as atividades do processo proposto.

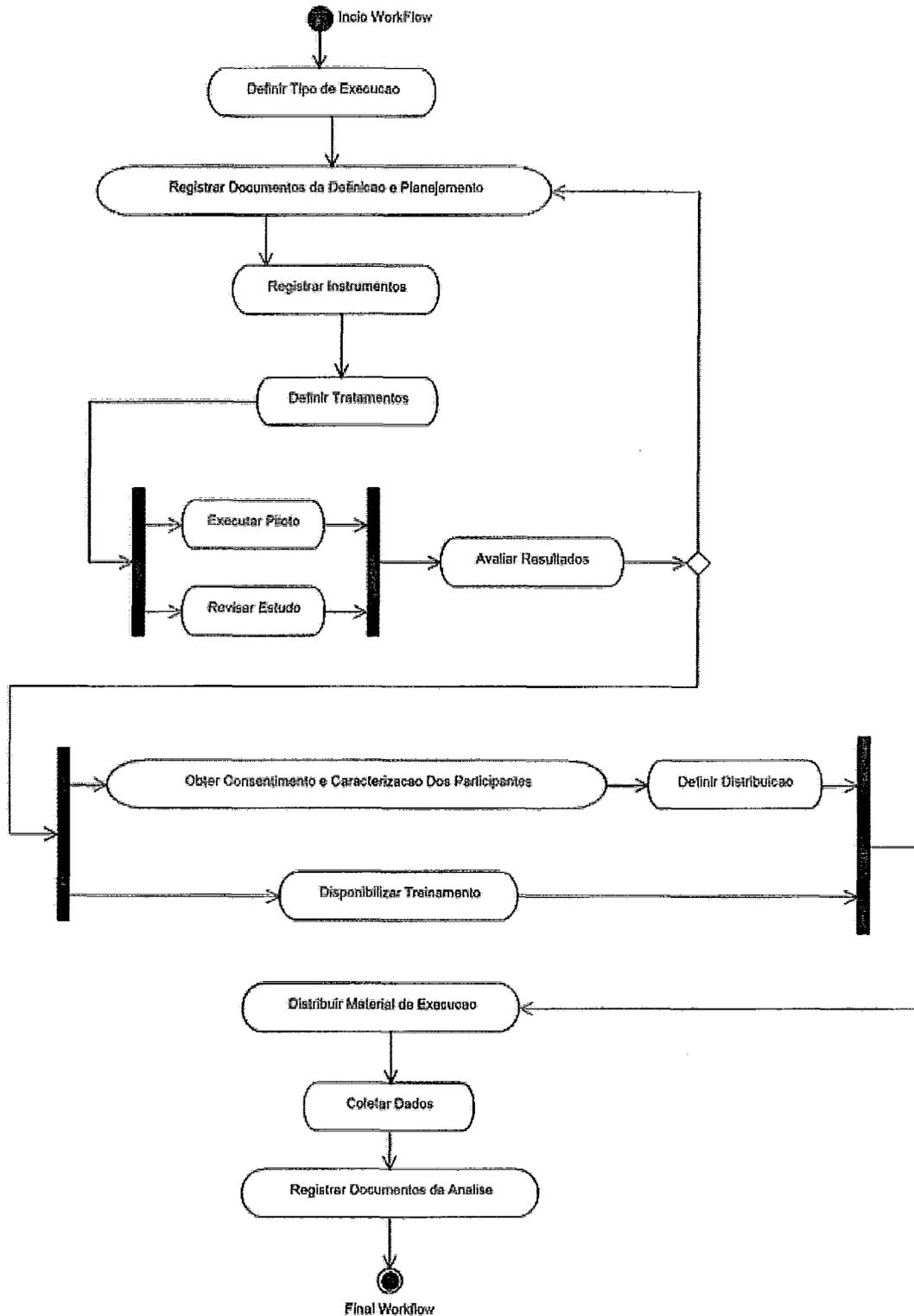


Figura 3.1- Processo de Apoio à Execução de Experimentos

O processo foi descrito utilizando o diagrama de atividades da UML, para estar de acordo com o padrão SPEM - *Software Process Engineering Metamodel* (SPEM Specification, 2006) da OMG (*Object Management Group*) para representação de *workflow* de processos.

3.4 – Requisitos da Infra-Estrutura para Apoiar o Planejamento Execução e Repetição de Estudos Experimentais em Engenharia de Software

Nesta seção é apresentada uma descrição sucinta dos requisitos da infra-estrutura proposta.

Foram levados em conta na definição destes requisitos o diagrama de atividade, as descrições e formalizações das atividades do processo proposto, bem como os seguintes aspectos:

- Funcionalidades oferecidas por outras abordagens;
- Experiência dos integrantes da Equipe de Engenharia de Software Experimental na execução de estudos;
- Discussões com pesquisadores do Projeto Readers;
- A proposta de taxonomia de Pacote de Experimento e Processo de Experimentação proposto por AMARAL (2003);
- Proposta de aquisição de conhecimento em processos de software proposto por MONTONI (2003);
- Fontes como JEDLITSCHKA & PFAHL (2005), SHULL *et al.* (2004) e SHULL *et al.* (2002), que relatam lições aprendidas na execução de estudos experimentais, e;
- Utilizar a tecnologia de *workflow* para implementar processos de software automatizados.
- Integração com eSEE (ANEXO I).

Em linhas gerais, os requisitos da infra-estrutura visam:

- Dar apoio na construção de uma base de conhecimento de estudos experimentais em Engenharia de Software.
- Utilizar a proposta de processo de empacotamento de AMARAL (2003).
- Gerenciar o conteúdo para apoiar a repetição e algumas das tarefas da execução de um estudo experimental, utilizando para isto o processo proposto.

Um documento de requisitos foi produzido. Este documento foi debatido com a Equipe de Engenharia de Software Experimental da COPPE/UFRJ e inspecionado por

alunos da pós-graduação do Programa de Engenharia de Sistemas e Computação da COPPE/UFRJ. Sua última versão foi utilizada para apoiar a construção de um protótipo e pode ser visto no ANEXO II desta dissertação.

O diagrama de classes de projeto de alto nível relativo aos artefatos manipulados pela infra-estrutura pode ser visto na figura 3.2 .

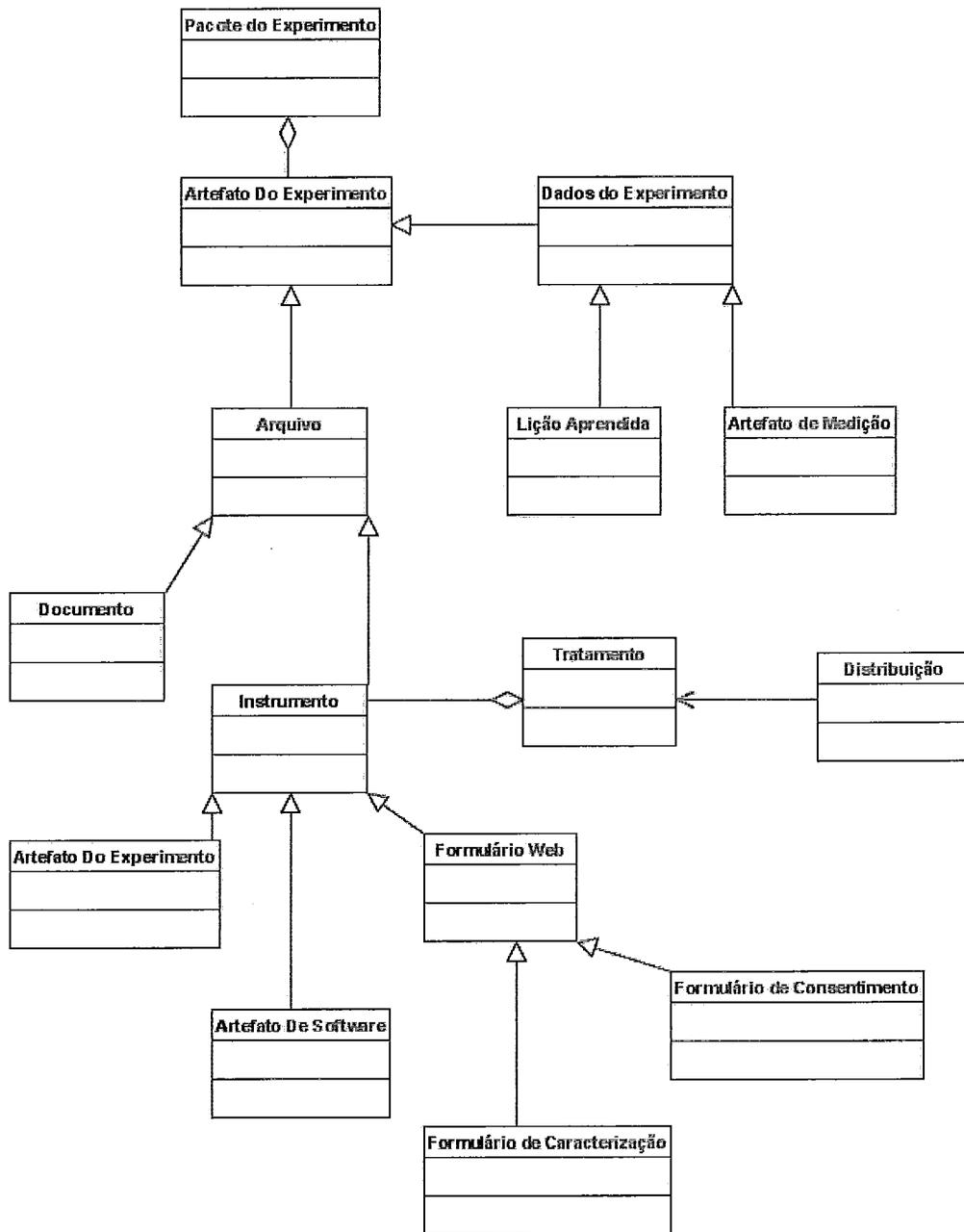


Figura 3.2- Modelo de Alto Nível

Os modelos conceituais são os mesmos utilizados na proposta de AMARAL (2003), onde os artefatos de um pacote de estudo experimental são organizados em três pacotes além de definir os papéis desempenhados pelos indivíduos no Processo de Experimentação. Os pacotes organizam os artefatos segundo sua natureza dentro do Processo de Experimentação que são:

- Organização do Experimento: que basicamente constitui um artefato com a definição e o plano do estudo experimental;
- Instrumentos: os artefatos que compõem a instrumentação do estudo experimental;
- Formulários de Resultados: que são os dados coletados bem como os resultados da análise desses dados obtidos a partir da execução do estudo experimental.

3.5 – Arquitetura

Em relação à arquitetura, o principal aspecto levado em consideração é que eSEE prevê a utilização de ferramentas externas disponibilizadas como e-services. Aspecto esse levado em conta, como pôde ser visto, nos requisitos da infra-estrutura. Como vimos, estes e-services vêm sendo desenvolvidos no eSEE como *Web services*.

No contexto deste trabalho, o próximo passo foi tentar explorar possíveis interações entre *Web services* para disponibilizar serviços mais complexos para apoiar atividades do processo proposto no capítulo 3.

Deste modo, a arquitetura proposta sugere a disponibilização de algumas funcionalidades através de um *Web service*. A idéia é que os serviços estejam disponíveis para a construção de sub-aplicações que possam ser utilizadas nas atividades do processo proposto. O acesso a estas sub-aplicações também é realizado por meio deste serviço. Um serviço intermediário é disponibilizado e utilizado para implementar e instanciar *workflows* do processo proposto. Este último serviço é acessado por um cliente SOA, que provê aos usuários todas as funcionalidade da infra-estrutura de acordo com seus papéis no processo proposto. A figura 3.3 abaixo ilustra esta arquitetura.

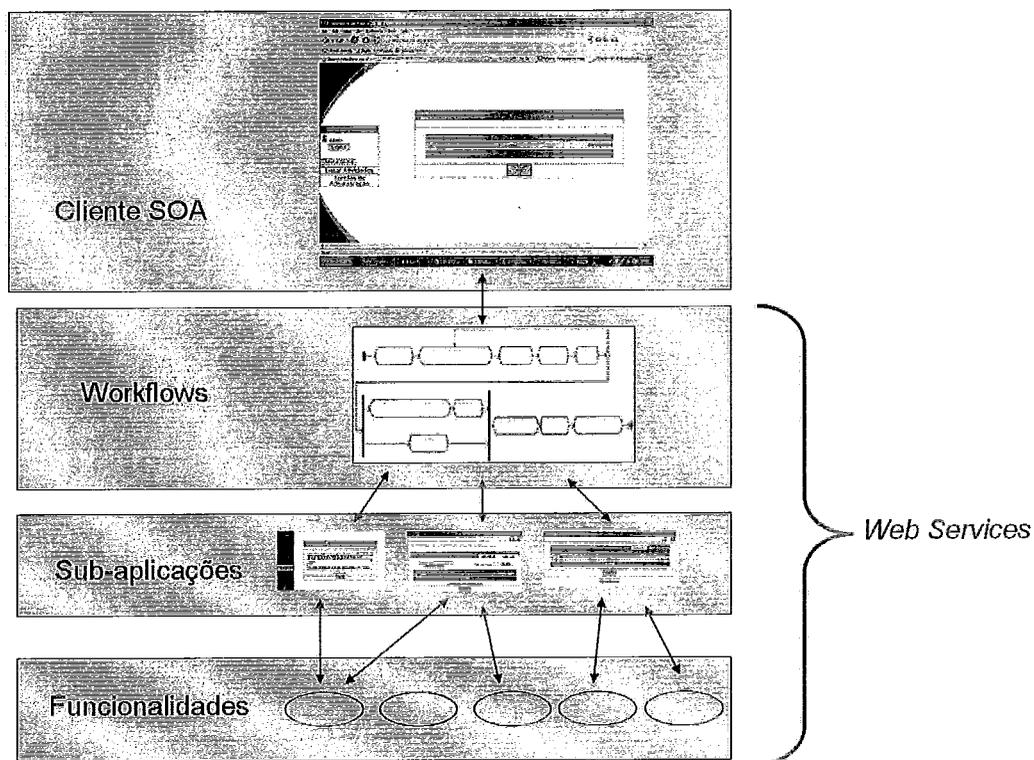


Figura 3.3- Arquitetura da Infra-estrutura

3.6 – Protótipo da Infra-estrutura

3.6.1 – Descrição Geral

Com o propósito de avaliarmos a viabilidade e esforço de construção desta infra-estrutura um protótipo foi construído.

Um dos serviços, as páginas e o repositório de estudos experimentais foram construídos em Zope (<http://www.zope.org>), um servidor de aplicação com um banco de dados OO utilizando a linguagem de programação Python (<http://www.python.org>). O serviço que implementa o gerenciamento de *workflows* foi primeiramente construído em Python sobre o Zope, mas substituído por um outro utilizando a linguagem de *workflows* de Web services, BPEL – *Business Process Execution Language* (<http://www-128.ibm.com/developerworks/library/specification/ws-bpel/>).

De um modo geral, os envolvidos na execução do experimento podem acessar as funcionalidades desde que sejam usuários devidamente cadastrados. Cada usuário usará no *login* um *e-mail* válido e será autenticado por uma senha. Este cadastro é controlado, como medida de controle da COPPE/UFRJ. Na figura 3.4 pode ser vista a página inicial da infra-estrutura.

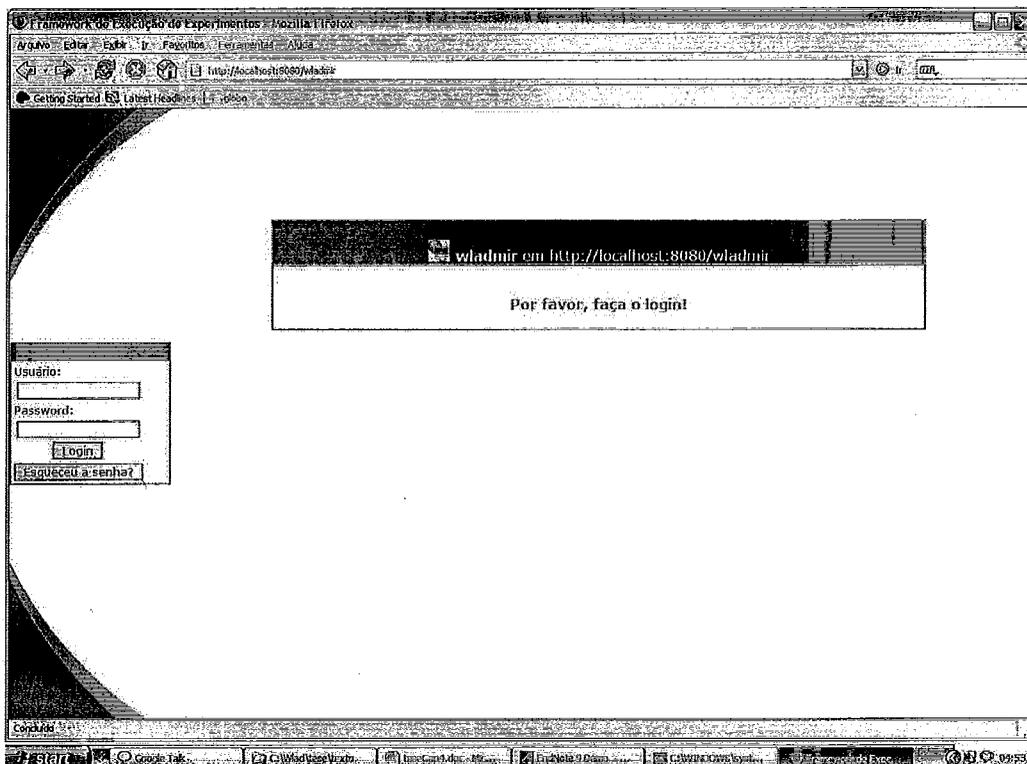


Figura 3.4- Página Inicial da Infra-estrutura

Nas seções seguintes serão apresentados os detalhes do protótipo da infra-estrutura de acordo com a perspectiva de cada um dos papéis identificados no processo proposto: Engenheiro de Software, Pesquisador, Revisor Externo, Participante.

3.6.2 – Protótipo na Perspectiva do Engenheiro de Software

A principal função do Engenheiro de Software na infra-estrutura é administrar o uso da infra-estrutura. Ele é o responsável pela instanciação e monitoramento de processos, estando responsável pelas tarefas administrativas.

O cadastro de Pesquisadores é realizado pelo Engenheiro de Software via Zope. Na infra-estrutura o Engenheiro de Software tem o poder de instanciar os processos de apoio à execução de experimentos, visualizar em que atividades estão os processos em execução através do menu Tarefas de Administração, e navegar pelos pacotes de experimentos já executados pela infra-estrutura. O acesso de um Engenheiro de Software à infra-estrutura pode ser exemplificado na figura 3.5.

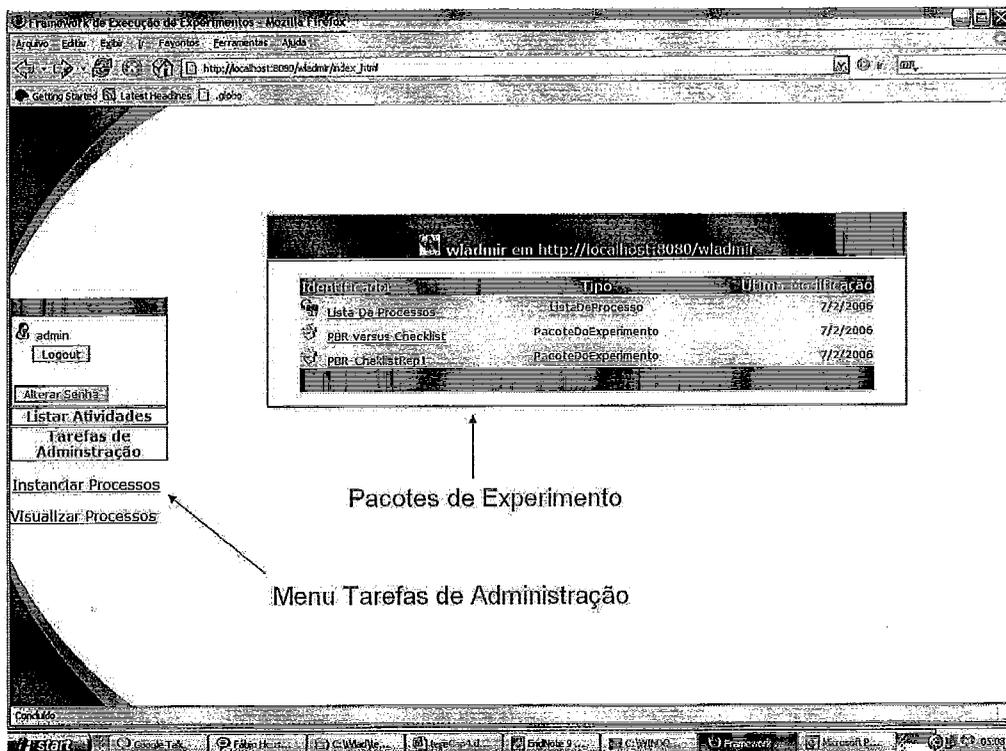


Figura 3.5- Acesso de um Engenheiro de Software

Na figura 3.6, vemos o preenchimento do formulário para instanciação de um processo.

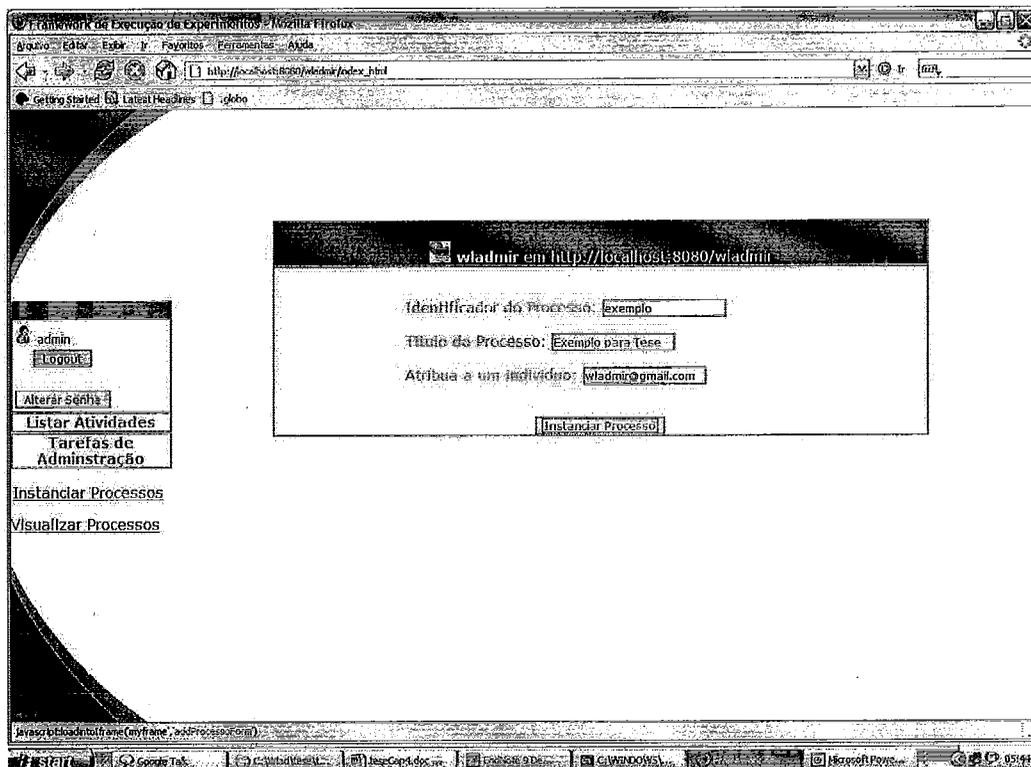


Figura 3.6- Formulário de Instanciação de Processo

3.6.3 – Protótipo na Perspectiva do Pesquisador

Este é papel mais complexo na infra-estrutura. Em um processo instanciado ele participa de todas as atividades. Quando ele efetua o *login* tem acesso à navegação pelos pacotes do experimento como o Engenheiro de Software. Entretanto não tem acesso ao Menu de Tarefas de Administração. Por poder estar participando de mais de uma execução de um estudo, ele tem acesso através do menu Lista de Atividades a todas as atividades dos processos que interage.

Sua primeira tarefa é, na atividade Definir Tipo de Execução, optar por definir um novo experimento ou repetir um já executado na infra-estrutura. A figura 3.7 ilustra essa tarefa.

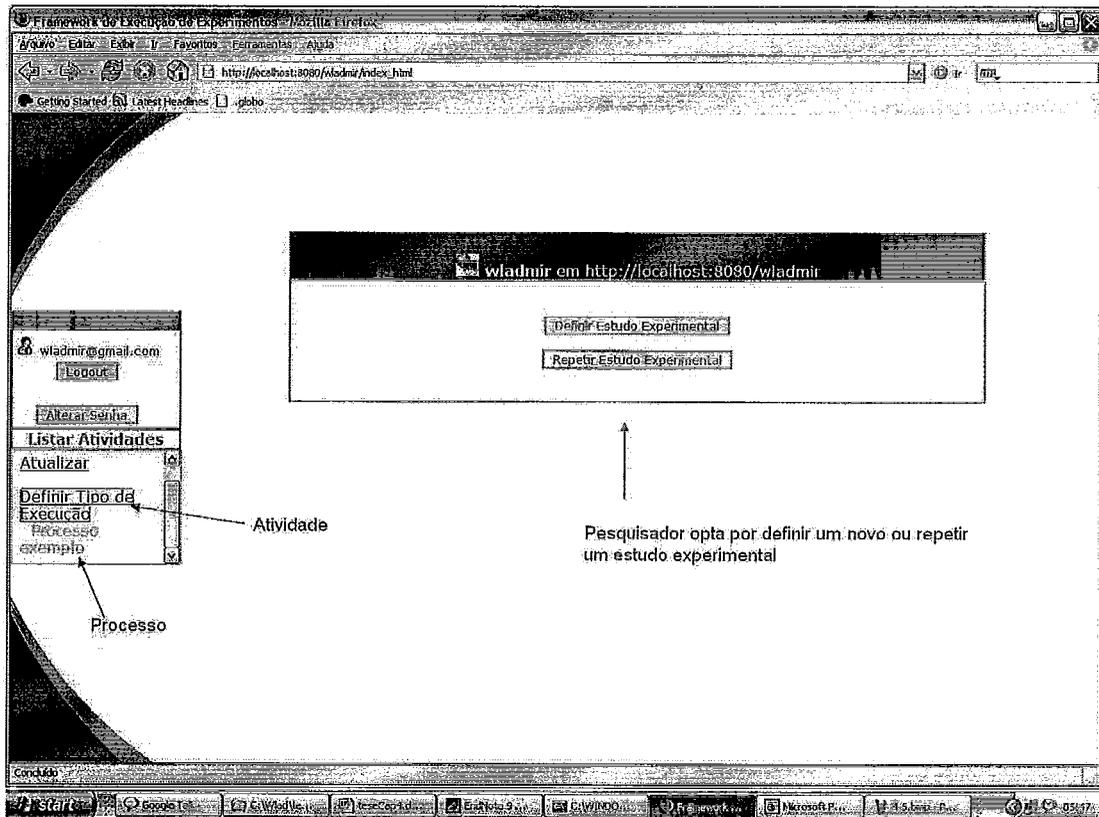


Figura 3.8- Opção do Pesquisador

Ao selecionar uma das opções a infra-estrutura prepara o arcabouço para o empacotamento dos artefatos ao longo da execução do processo. A única diferença entre as duas opções é que desejando o pesquisador repetir um estudo ele deve indicar qual o estudo a ser repetido. A figura 3.9 ilustra o preenchimento do formulário de preparação para repetição de um estudo.

Figura 3.9- Formulário de Repetição de Experimento

Na atividade Registrar Documentos de Definição e Planejamento, o pesquisador armazena e torna acessível em outras atividades o plano experimental e outros documentos relativos à Definição e Planejamento do Processo de Experimentação, como, por exemplo, um GQM do experimento. A figura 3.10 ilustra a tela na atividade de Registrar Documentos de Definição e Planejamento.

Figura 3.10- Registro de Documentos de Definição e Planejamento

Na atividade seguinte, Registrar Instrumentos, o Pesquisador registra para o devido empacotamento e uso em atividades futuras os instrumentos a serem utilizados no processo. Para o devido registro na infra-estrutura, são exibidos os quatro tipos de instrumentos segundo a taxonomia de AMARAL (2003): Artefato de Software, Artefato do Experimento, Formulário de Consentimento e Formulário de Caracterização. A figura 3.11 exibe a tela nesta atividade.

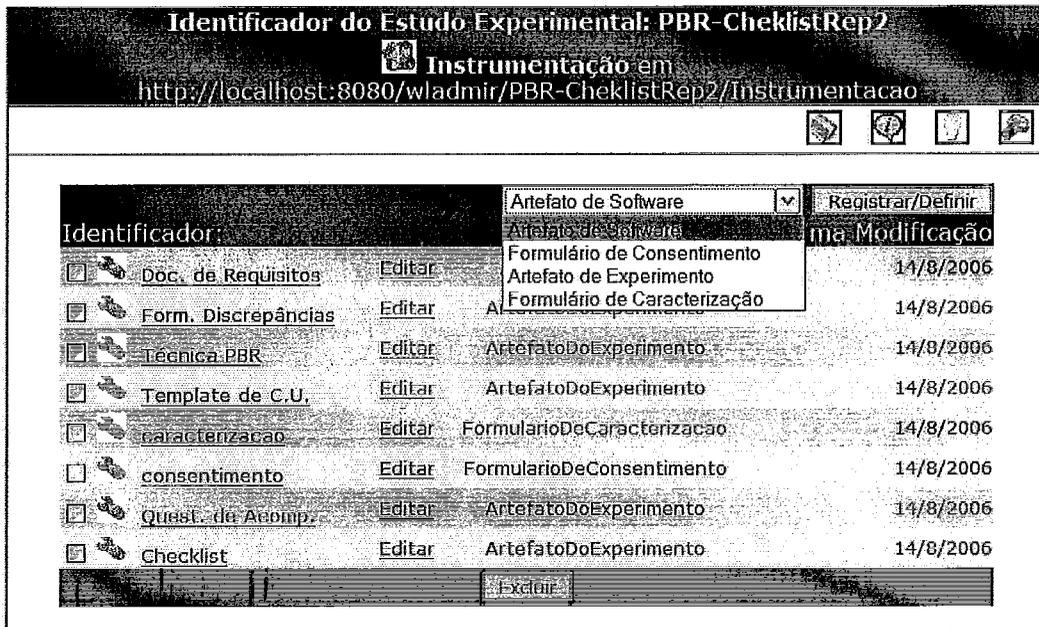


Figura 3.11- Registro de Instrumentos

Para exemplificar o registro de um instrumento a figura 3.12 mostra o formulário de registro de um Artefato de Software. Como pode ser visto na figura, além de fornecer o artefato em arquivo, o Pesquisador tem de preencher informações adicionais como previstas nos requisitos e de acordo com AMARAL (2003).

Identificador(opcional):

Título (opcional):

Arquivo:

Uso:

Contexto de Uso:

Criador:

Este artefato faz parte do material de treinamento? Sim Não

Tipo:

Figura 3.12- Formulário de Registro de Artefatos de Software

Na próxima atividade, Definir Tratamentos, o Pesquisador define e registra os tratamentos a serem aplicados aos participantes. Por questão de consistência de interface gráfica, a visualização dos tratamentos obedece a mesma metáfora de interface das duas atividades anteriores, como pode ser visto na figura 4.15.

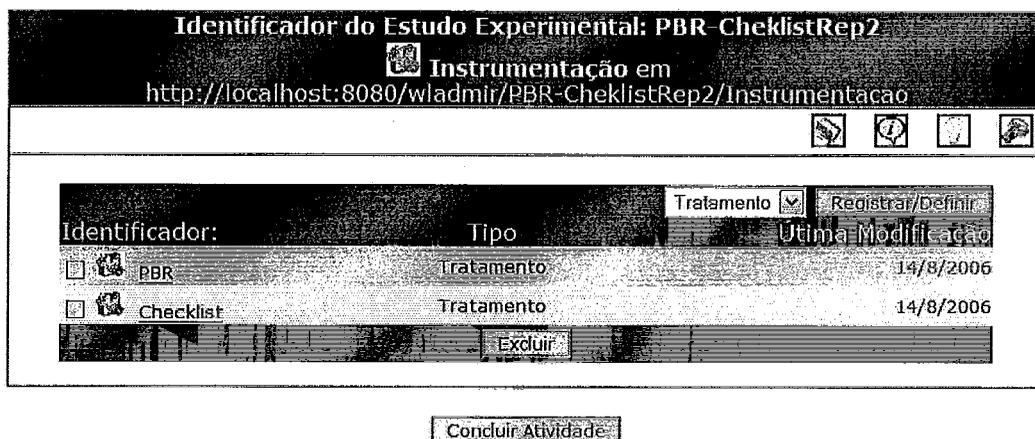


Figura 3.13- Definição de Tratamentos

Na figura 3.14 é exibido o formulário de definição de tratamentos, onde o pesquisador seleciona os artefatos que compõe cada um dos tratamentos do experimento.

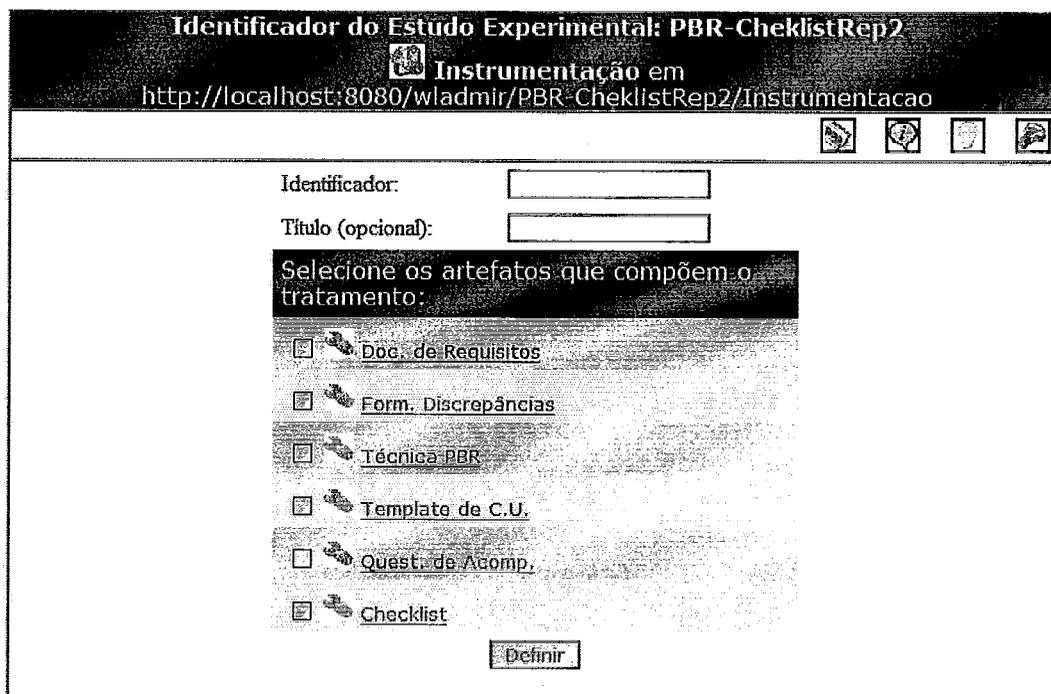


Figura 3.14- Formulário de Definição de Tratamentos

Na atividade Executar Piloto, um menu lateral permite que o pesquisador acesse todos os artefatos já empacotados recrute, obtenha o consentimento e caracterização, atribua os tratamentos aos participantes exatamente como ocorrerá nas atividades seguintes quando o experimento for efetivamente executado. Além disso, o pesquisador deve registrar o devido laudo, como definido no processo proposto. A figura 3.15 exibe o formulário de registro do laudo, bem como o menu de acesso às funcionalidades para execução do piloto.

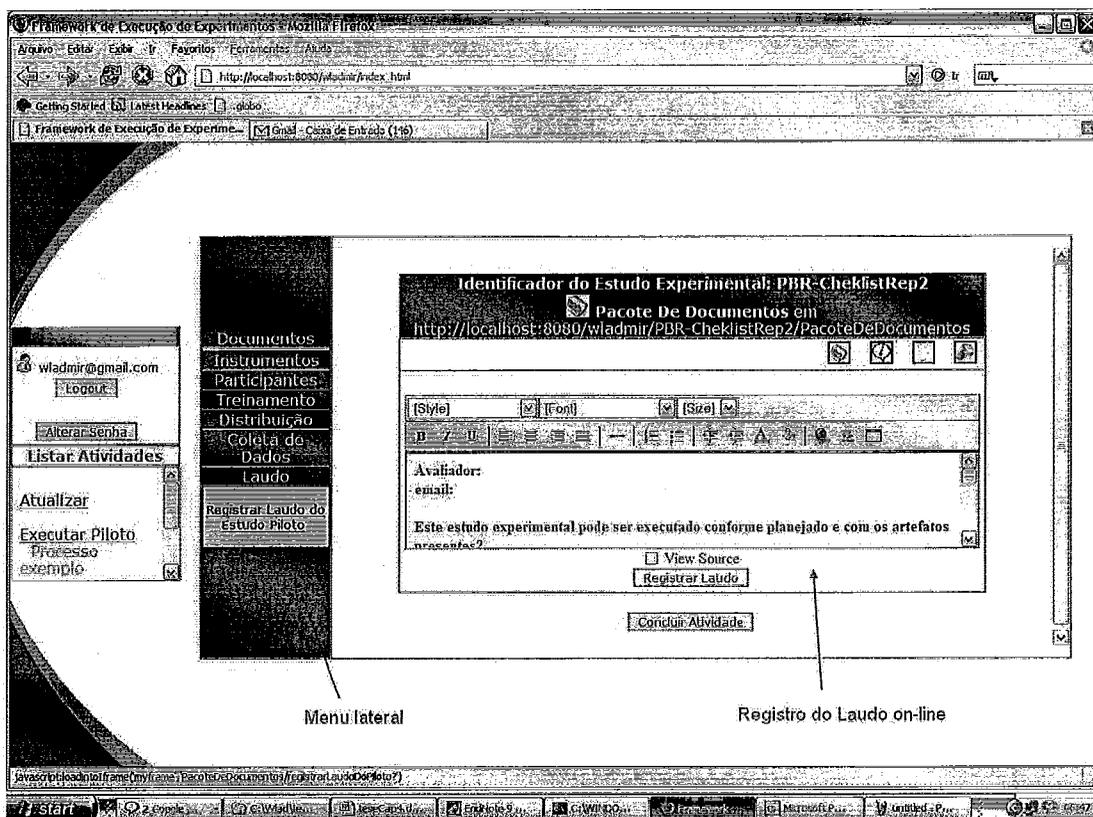


Figura 3.15- Execução do Estudo Piloto

Na atividade Revisar Estudo o Pesquisador pode selecionar um ou mais revisores externos e os notifica por *e-mail*, os indivíduos selecionados são cadastrados, se necessário, como usuários na infra-estrutura com acesso restrito aos artefatos do experimento corrente durante esta atividade. Além disso, a infra-estrutura permite que o Pesquisador acompanhe o registro dos laudos dos revisores e determine quando esta atividade se encerrará. Tanto a seleção dos revisores como o acompanhamento do registro dos laudos pode ser visto na figura 3.16.

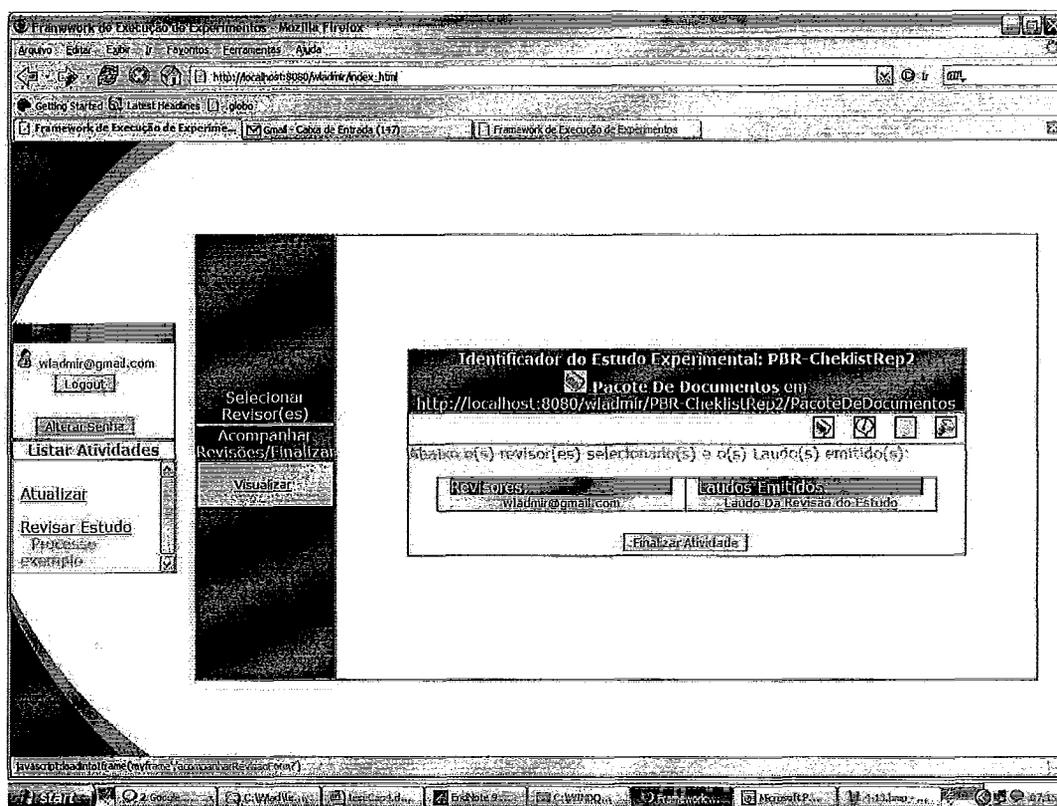
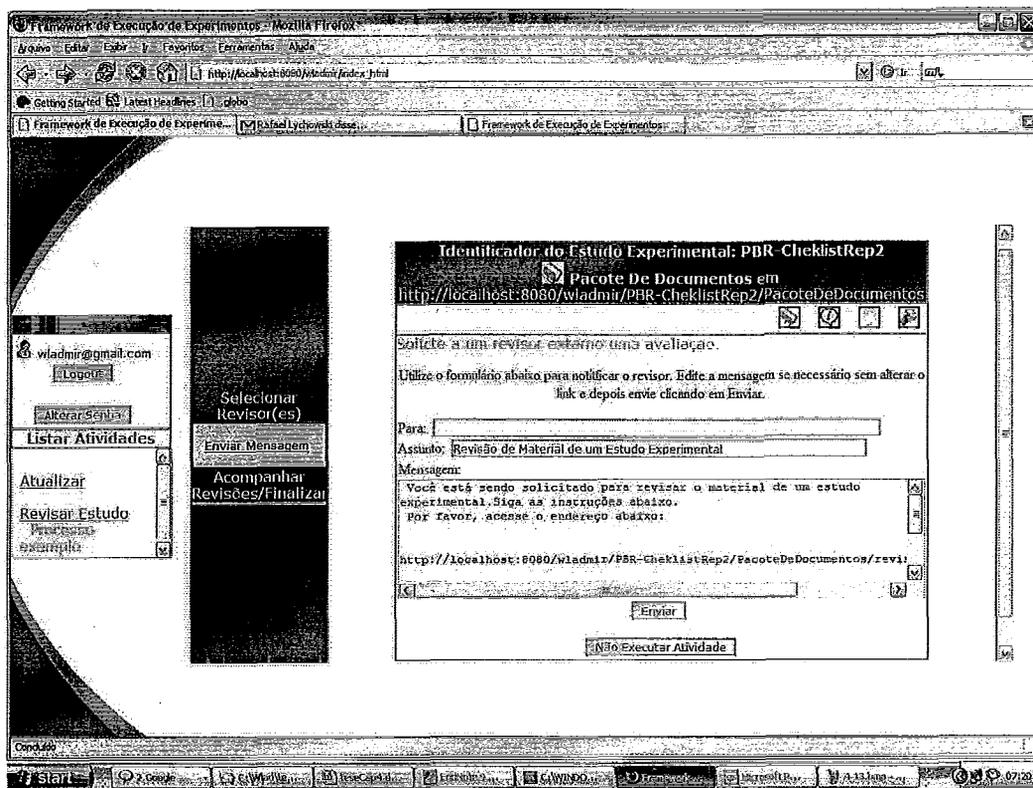


Figura 3.16- Revisão do Experimento

Na atividade Avaliar Resultados, o Pesquisador tem acesso aos laudos de execução do piloto e de revisão do estudo e deve decidir se o experimento pode ser executado

ou não, conforme definido no processo. Além disso, ele deve registrar o laudo de avaliação do estudo, como pode ser visto na figura 3.17.

Figura 3.17- Avaliação do Experimento

Seguindo pelo processo proposto, uma próxima atividade seria a Obter Consentimento e Caracterização dos Participantes. Nela o Pesquisador faz o recrutamento de participantes por uma notificação de *e-mail* (ver figura 3.18). Além disso, é possível acompanhar as respostas dos indivíduos e visualizar os dados submetidos por cada um deles.

Figura 3.18- Recrutamento dos Participantes via *e-mail*

A figura 3.19 ilustra como todos os dados de consentimentos e caracterizações submetidos podem ser exibidos para o acompanhamento do Pesquisador, embora se possa ter também o acesso por indivíduo dos dados.

Identificador do Estudo Experimental: PBR-ChecklistRep2

Pacote De Dados em: <http://localhost:8080/wladmir/PBR-ChecklistRep2/PacoteDeDados>

Consentimentos					
Identificador:	Última Modificação	redirect	aceito	email	
FormularioDeConsentimento_1155566168920	14/8/2006	./caracterizacao	true	wladmir@cos.ufrj.br	
FormularioDeConsentimento_1155566852894	14/8/2006	./caracterizacao	true	wladmir@gmail.com	

Caracterizações						
Identificador:	Última Modificação	Nível	email	trabalharIngles	leituraIngles	select
FormularioDeCaracterizacao_1155566827968	14/8/2006	1	wladmir@cos.ufrj.br	3	2	1
FormularioDeCaracterizacao_1155566858292	14/8/2006	1	wladmir@gmail.com	1	1	0

Figura 3.19- Visualizando Consentimentos e Caracterizações submetidas

Com os dados das caracterizações dos participantes, o Pesquisador pode partir para a atividade de Definir Distribuição, onde ele descreverá as rodadas de execução do experimento. Nesta atividade ele tem acesso aos dados de caracterizações dos indivíduos que aceitaram participar do experimento e às definições dos tratamentos. Deste modo ele define cada rodada de distribuição experimental, atribuindo a cada participante um tratamento, conforme pode ser observado na figura 3.20.

Pacote De Dados em
<http://localhost:8080/wladmir/PBR-ChecklistRep2/PacoteDeDados>

Nesta atividade você deve definir como os tratamentos serão atribuídos aos participantes.

Caracterização do(s) Participante(s)	Tratamento(s)	
	PBR	Checklist
FormularioDeCaracterizacao_1155566827968	<input checked="" type="radio"/>	<input type="radio"/>
FormularioDeCaracterizacao_1155566858292	<input type="radio"/>	<input checked="" type="radio"/>

Defina um rótulo:

Selecionar Dados:

Identificador:	Tipo	Última Modificação
<input type="checkbox"/> Rodada 1	Distribuição	14/8/2006

Visualizar Consentimentos e Caracterizações

Figura 3.20- Definindo Distribuição dos Artefatos

Ainda nesta atividade, a infra-estrutura permite que o Pesquisador selecione uma ou mais variáveis quantificáveis da caracterização dos participantes e veja como ficou sua distribuição por tratamento. Na figura 3.21 podemos ver como a variável trabalharIngles (cujo significado na caracterização significa a facilidade de um participante trabalhar com o material na língua inglesa) está distribuído entre dois tratamentos (PBR versus Checklist). Esta funcionalidade serve para auxiliar o Pesquisador a formar grupos de participantes entre os tratamentos, tentando reduzir determinados efeitos sobre algumas variáveis que ele não pode controlar.

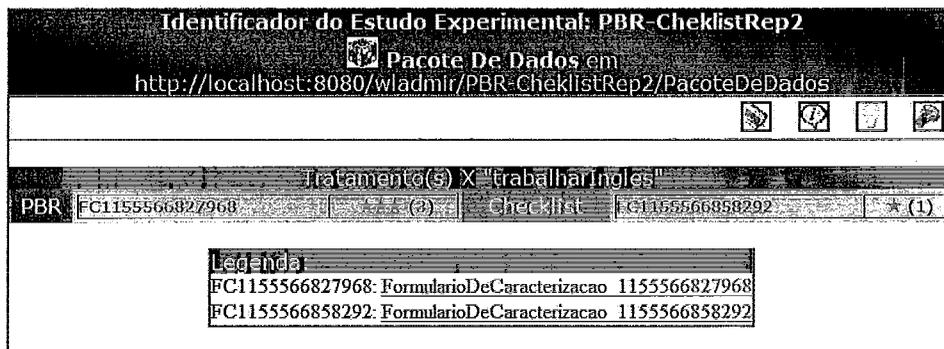


Figura 3.21- Visualizando a formação de Grupos segundo a perspectiva de uma variável

Na atividade Disponibilizar Treinamento, o Pesquisador pode acessar o material de treinamento ou enviar uma notificação de *e-mail* aos participantes para que estes acessem e baixem o material de treinamento via *browser*. A figura 3.22 mostra o formulário de envio de e-mail do material de treinamento.

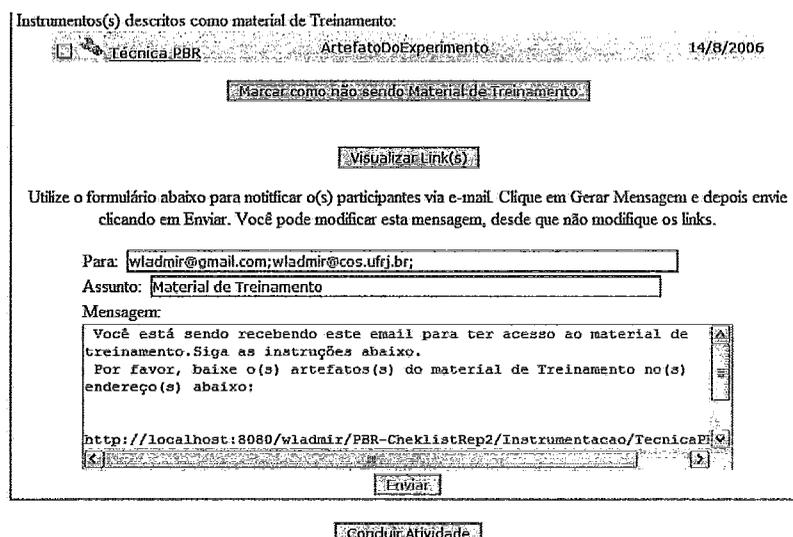


Figura 3.22- Enviando o Material de Treinamento

Assumindo que todas as atividades foram executadas, o próximo passo é realizar a distribuição do material aos participantes. Estando na atividade Distribuir Material de Execução o Pesquisador, seleciona a ordem em que ocorrerá a distribuição de cada rodada experimental. Os participantes são notificados e recebem instruções para baixar o material segundo o tratamento que lhe foi atribuído em cada rodada. A figura 3.23 exibe o formulário onde o Pesquisador seleciona quando ocorrerá uma rodada.

Selecione a distribuição que deve ser aplicada.

Distribuições que não ocorreram:

Distribuições que não ocorreram:	Distribuição	14/8/2006
<input checked="" type="checkbox"/>	Rodada 1	

Distribuições que já ocorreram:

Não há itens a serem exibidos.

Utilize o formulário abaixo para notificar o(s) participantes via e-mail. Edite a mensagem se necessário e depois envie clicando em Enviar.

Assunto:

Mensagem:

Você está sendo recebendo este e-mail para ter acesso ao material para execução do estudo experimental. Siga as instruções abaixo.
Por favor, baixe o(s) artefatos(s) do material para execução do estudo experimental no endereço abaixo:

Figura 3.23- Aplicando a Distribuição

À medida que o material de cada rodada é distribuído a cada participante, o Pesquisador na atividade Coletar Dados é capaz de monitorar a submissão dos artefatos com os dados do experimento realizada por cada participante. Nesta atividade também é possível enviar questionários de acompanhamento após a realização de uma rodada aos participantes, e aguardar e monitorar a submissão destas respostas. A figura 3.24 exemplifica um Pesquisador na infra-estrutura monitorando a coleta de dados.

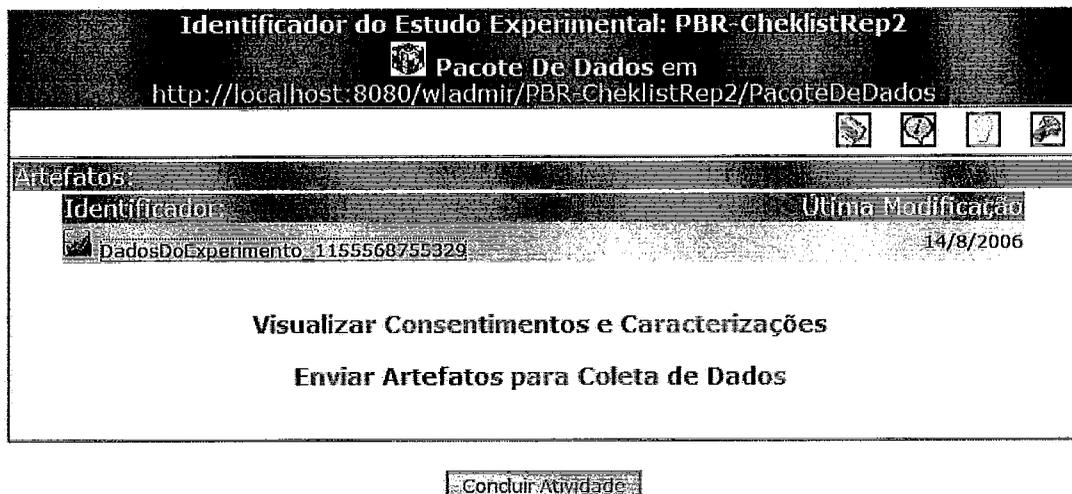


Figura 3.24- Monitorando a Coleta de Dados

Enquanto a figura 3.25 mostra a relação de instrumentos para que o Pesquisador envie artefatos para a coleta adicional de dados após uma rodada.

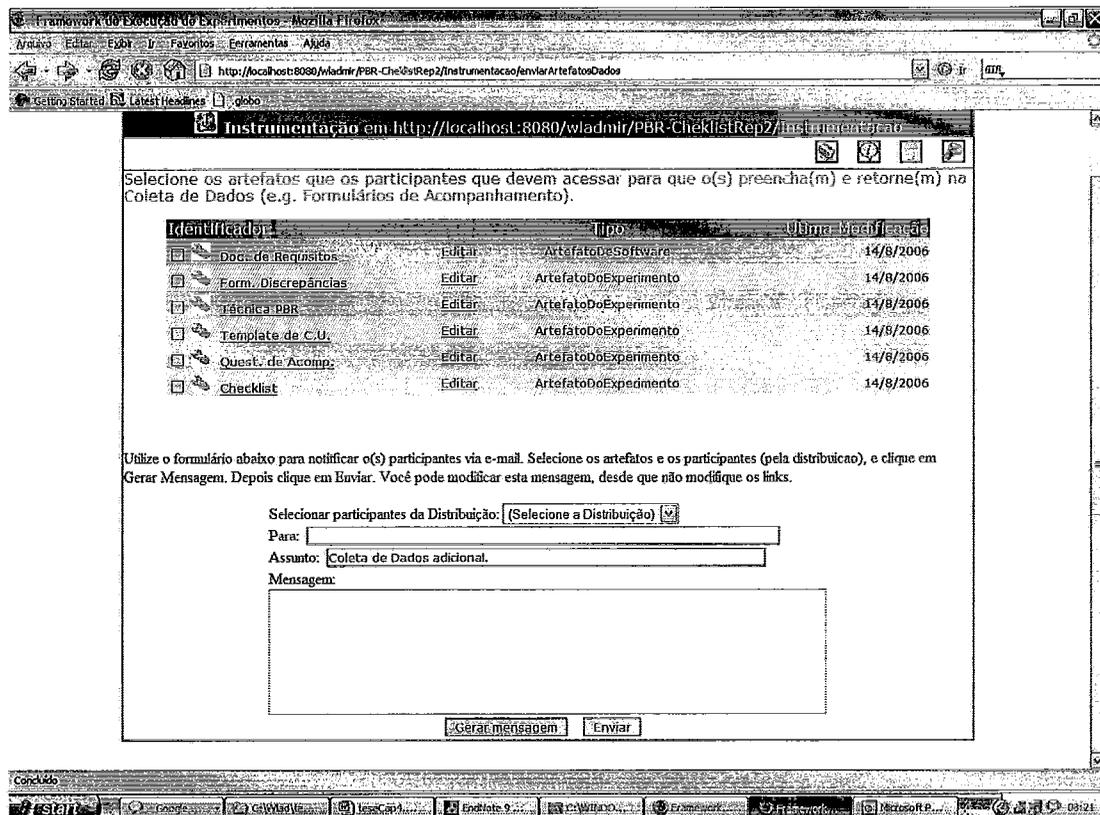


Figura 3.25- Selecionando Artefatos para envio após uma rodada

A última atividade de um Pesquisador é, após a análise dos dados, registrar os documentos com os resultados desta análise. A interface desta atividade segue a

mesma metáfora das atividades anteriores de registro ou definição de artefatos, conforme pode ser vista na figura 3.26.

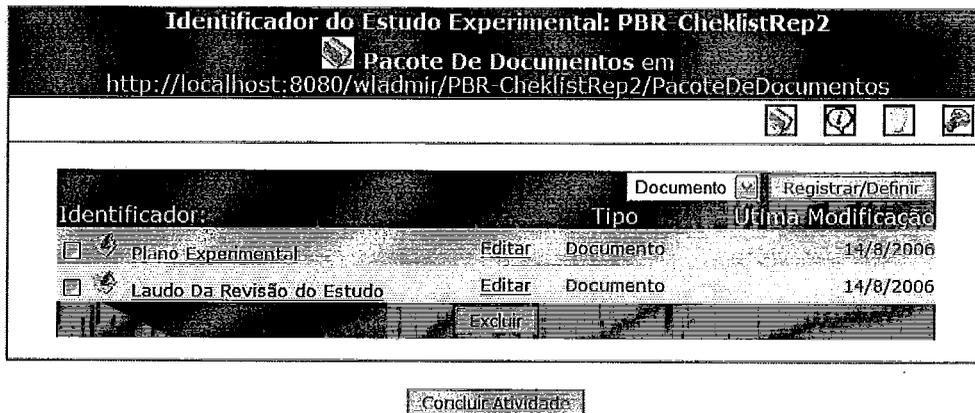


Figura 3.26- Registrando Documentos de Análise

3.6.4 – Protótipo na Perspectiva do Revisor Externo

A única atividade em que um Revisor Externo interage é Revisar Estudo, nela o revisor faz seu *login*, tem acesso a todos os documentos de Definição e Planejamento, e a instrumentação do experimento para livremente baixar e revisá-los. Ao final da revisão ele registra um laudo com as possíveis discrepâncias no experimento, mas quem decide quando esta atividade é encerrada é o Pesquisador. Graças ao controle de acesso sobre os artefatos, um revisor tem acesso e pode alterar seu laudo enquanto a atividade não tiver sido encerrada pelo Pesquisador. A figura 3.27 ilustra a utilização da infra-estrutura por um Revisor Externo.

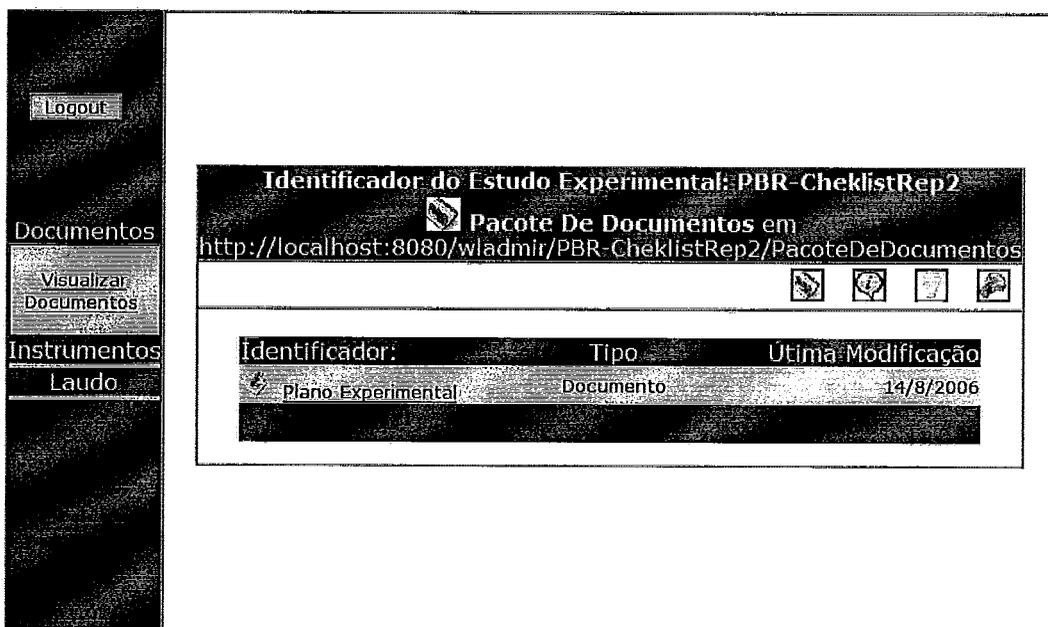


Figura 3.27- Revisando os Artefatos

3.6.5 – Protótipo na Perspectiva do Participante

A primeira atividade que o Participante interage no processo proposto é na atividade Obter Consentimento e Caracterização dos Participantes. Nela o Participante indica se aceita participar do experimento. Somente se a indicação for positiva o Participante é direcionado para o formulário de caracterização para seu preenchimento. Ao final, o Participante visualiza uma mensagem de agradecimento. A figura 3.28 ilustra a visualização de um formulário de consentimento sendo preenchido.

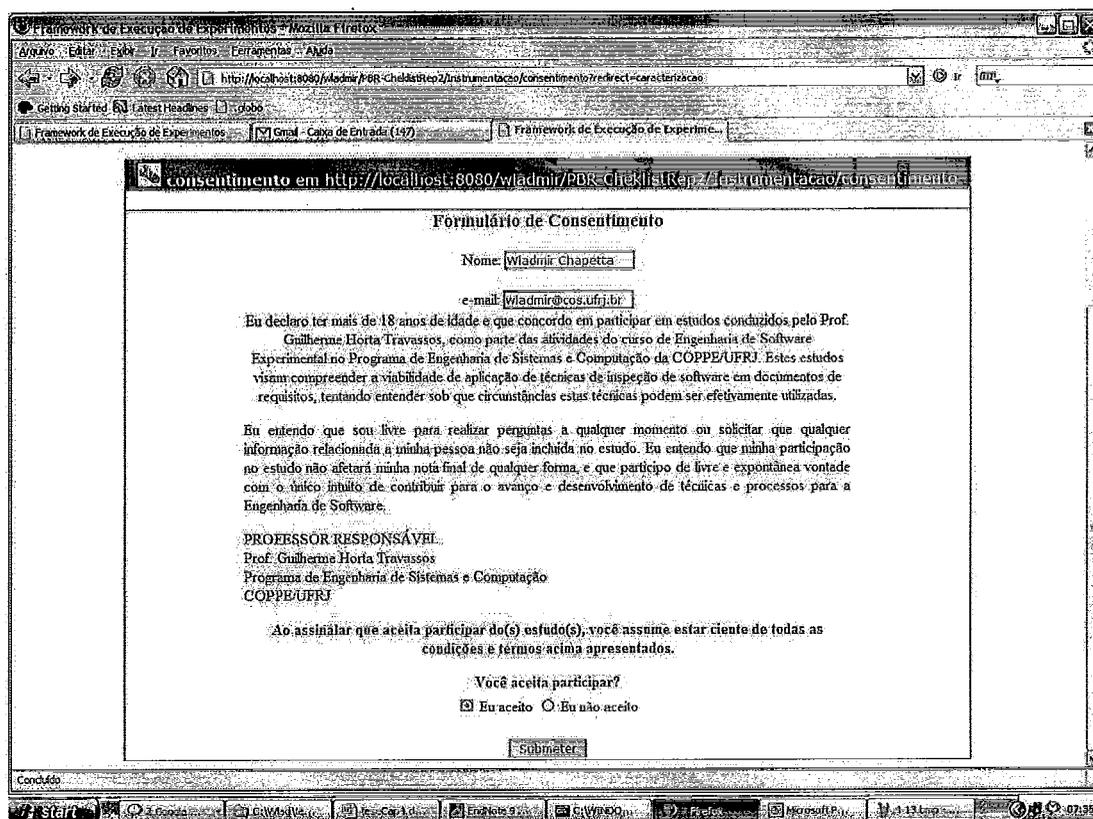


Figura 3.28- Preenchendo o Consentimento

A figura 3.29 ilustra a visualização de um formulário de consentimento sendo preenchido.

caracterizacao.em http://localhost:8080/wladmir/PBR-CheklisRep2/Instrumentacao/caracterizacao

Formulário de Caracterização

Nome: Wladmir Chapetta
e-mail: wladmir@cos.ufrj.br

Nivel: ▼

Formação Geral

Por favor, estime sua habilidade em utilizar material de trabalho em Inglês:

Eu falo, leio e escrevo fluentemente: ▼

Considero o Inglês como sendo uma linguagem onde (Por favor, complete):

Minhas habilidades de leitura e compreensão de textos: ▼

Minha capacidade de trabalhar/seguir instruções escritas em Inglês: ▼

Figura 3.29- Preenchendo a Caracterização

Depois de ter aceito participar e ter sido caracterizado, o participante recebe notificações com instruções para acessar e baixar o material de treinamento e o material de execução das rodadas experimentais. A figura 3.30 ilustra uma notificação via *e-mail* de um participante com instruções para acessar o material de treinamento.

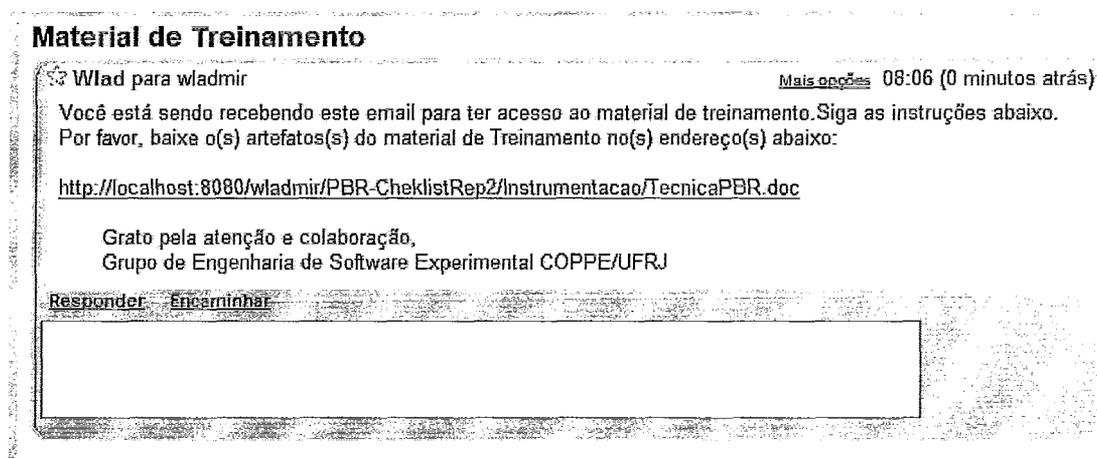


Figura 3.30- Exemplo de Notificação via *e-mail*

A última interação do Participante é na coleta de dados. Nela, o Participante acessa um formulário para fazer o *upload* com os artefatos com os dados do experimento (ver figura 3.31).

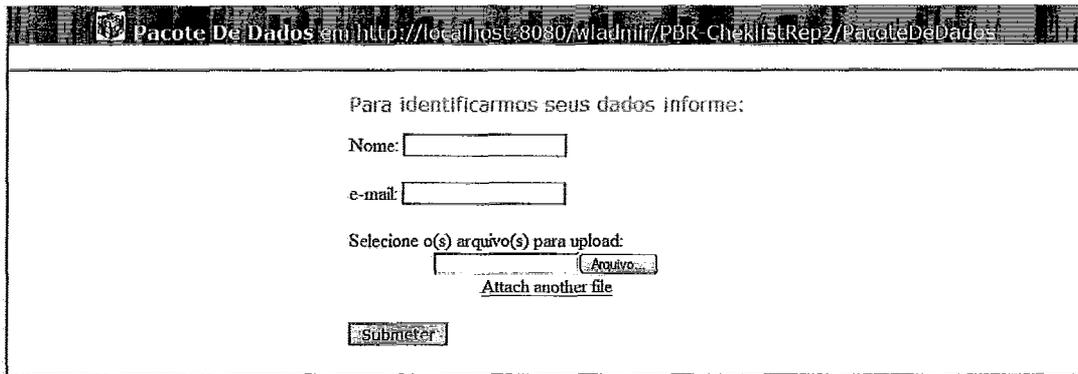


Figura 3.31- Recebendo os Dados do Experimento

Submetendo os arquivos com os dados, o Participante visualiza uma mensagem de agradecimento e uma lista dos arquivos que foram registrados corretamente. Conforme pode ser visto na figura 3.32.

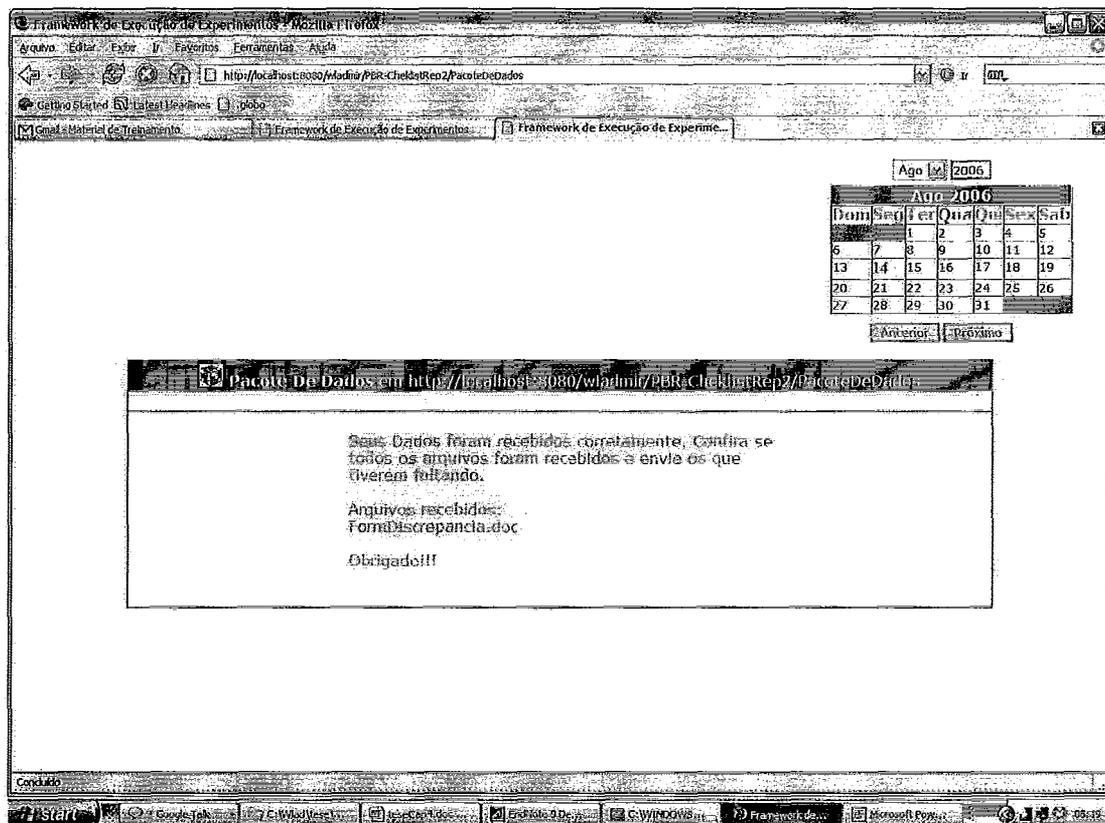


Figura 3.32- Mensagem ao Participante

6.6.6 – Mecanismos de Gerência de Conhecimento na Infra-estrutura

O conhecimento pode estar embutido não somente em documentos e repositórios, mas também em rotinas, processos, práticas e normas organizacionais. O conhecimento é o recurso chave em tomadas de decisão inteligentes, previsão, projeto, planejamento, diagnose, análise, avaliação e julgamento intuitivo. O conhecimento não cresce em bases de dados e sim envolve experiência, sucessos, fracassos e aprendizado ao longo do tempo.

O conhecimento é classificado como de dois tipos: tácito e explícito. Conhecimento tácito é pessoal, específico em um contexto e difícil de formalizar e comunicar. Conhecimento explícito é transmissível em linguagem formal e sistemática. Os conhecimentos tácito e explícito são mutuamente complementares (NONAKA & TAKEUCHI, 1995).

A mera existência de conhecimento em algum lugar na organização é de pouco benefício, e ele torna-se um bem corporativo de valor somente se está acessível. Seu valor aumenta com o nível de acessibilidade. Assim, organizações de software precisam determinar e verificar como o conhecimento organizacional pode prover benefícios na realização de tarefas complexas e não rotineiras.

Neste contexto, a Gerência de Conhecimento é a área de estudo que se preocupa com essa aquisição, criação, compartilhamento/difusão, utilização de conhecimento tácito e explícito recorrentes dos vários ambientes de trabalho organizacionais e é essencial para ambientes de conhecimento complexos. A Gerência de Conhecimento eficiente pode facilitar o rápido acesso a conhecimento preciso que é essencial para executar tarefas, apoiar decisões e permitir a difusão de informações e boas práticas através de uma organização.

Somente os papéis de Pesquisador e Revisor Externo têm acesso aos mecanismos de aquisição de conhecimento da infra-estrutura. Baseando-se na proposta de MONTONI (2003), o protótipo da infra-estrutura permite a captura as lições aprendidas ao longo do processo através de um formulário. As lições aprendidas registradas podem ser consultadas ao longo do processo proposto através de um formulário de busca, onde o usuário pode utilizar alguns filtros na tentativa de especificar o conhecimento que está buscando. Segundo JEDLITSCHKA & PFAHL (2005), desvios em relação ao planejamento durante a execução do estudo experimental deveriam ser registrados sob a forma de lições aprendidas.

Ainda como conhecimento disponibilizado na infra-estrutura, o Pesquisador e o Revisor Externo têm acesso aos *templates* apresentados na seção 3.3 com as

descrições das atividades correntes que eles podem desempenhar em um dado momento do processo.

A disponibilização destes mecanismos se dá através de ícones no canto superior direito na página de apresentação das atividades sendo executadas, dando acesso a paginas específicas como pode ser visto na figura 3.33.

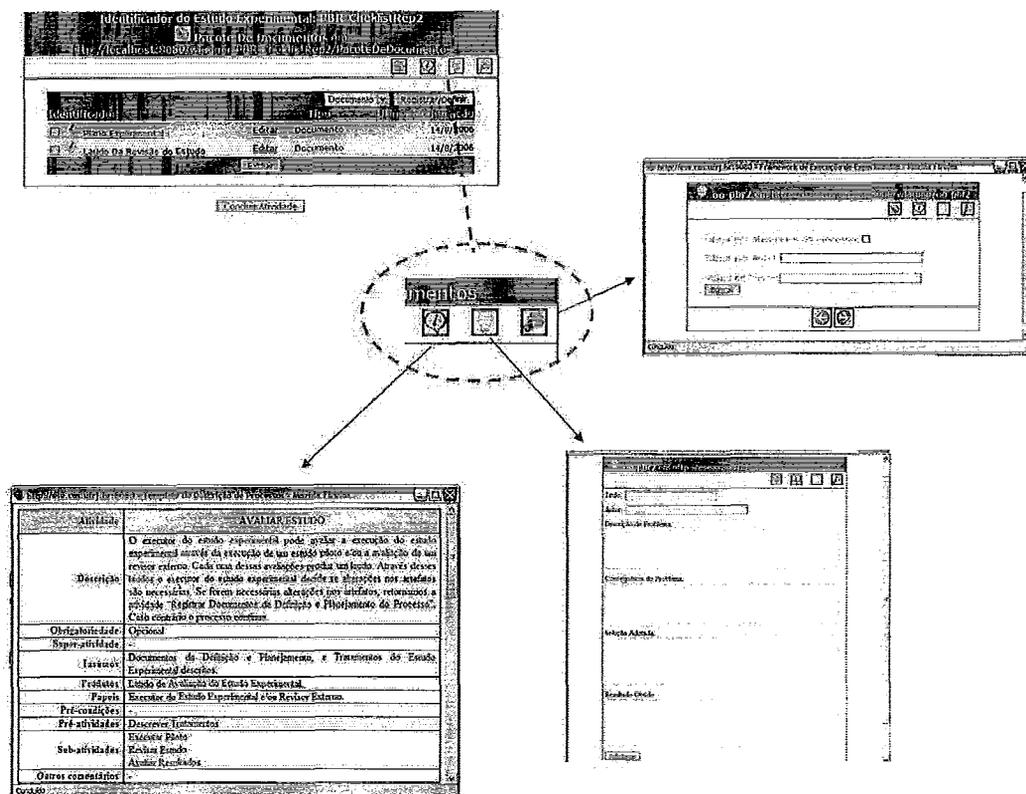


Figura 3.33- Mecanismos de Gerência de Conhecimento

3.7 - Conclusão

Neste capítulo foi apresentado o processo proposto e o protótipo construído de uma infra-estrutura *Web* para apoiar o Processo de Experimentação.

Com a construção do protótipo pudemos verificar a viabilidade de construção relativa a todas as idéias apresentadas ao longo deste capítulo, o processo, os requisitos e arquitetura da infra-estrutura, etc.

No capítulo seguinte serão discutidos os requisitos, a arquitetura e a implementação de uma aplicação *Web* para disponibilizar tal infra-estrutura para comunidade científica de Engenharia de Software. Vale salientar que a ordem de apresentação das seções do Capítulo 3 tenta refletir o encadeamento lógico das atividades de desenvolvimento da infra-estrutura proposta.

No próximo capítulo, iremos apresentar os estudos de viabilidade executados para observar se a infra-estrutura é capaz de alguma forma auxiliar no empacotamento e execução de estudos experimentais utilizando a infra-estrutura *Web* como instrumento facilitador. Foram executados 2 estudos. No primeiro, um estudo já executado foi somente empacotado, onde o pesquisador simula os indivíduos executando todos os papéis do processo. A partir deste, um segundo estudo é executado na infra-estrutura utilizando o processo proposto para apoiar a execução de um estudo experimental.

Capítulo 4 – Estudos Experimentais

Neste capítulo são apresentados os estudos experimentais realizados para se verificar a viabilidade do empacotamento e apoio à execução de experimentos oferecidos pelo protótipo da infra-estrutura proposta.

4.1 – Introdução

São apresentados a seguir os dois estudos experimentais planejados e executados com o objetivo de observar a viabilidade de utilização da infra-estrutura proposta. O primeiro estudo buscou identificar o quão adequado será utilizar a infra-estrutura para empacotar um estudo. No segundo foi observado somente a viabilidade de executar um estudo real com as funcionalidades providas pela infra-estrutura.

Os dois estudos experimentais objetivaram identificar a viabilidade da proposta, já que no estágio corrente desta dissertação seria muito difícil definir, controlar e medir efetivamente todas as variáveis envolvidas. Neste momento buscou-se somente explorar as questões que motivaram a criação da tecnologia, e procurar indícios de que é possível executar estudos experimentais através do apoio proposto, para posteriormente se tentar estabelecer hipóteses concretas que possam ser colocadas à prova em uma avaliação mais precisa e significativa desta tecnologia. Estudos de viabilidade são essencialmente estudos de laboratório, o que diminui os riscos assumidos em relação aos outros tipos de estudo (SHULL *et al.*, 2001).

Nas seções seguintes serão apresentados os planejamentos, resultados e lições aprendidas dos dois estudos, bem como conclusões gerais. Cada um dos estudos mostrou que alguns itens influenciavam no apoio à execução de estudos experimentais. Embora esses itens fossem partes do conhecimento tácito sobre os estudos experimentais executados pela Equipe de Engenharia de Software Experimental da COPPE/UFRJ, eles passaram despercebidos na definição do processo e nos requisitos da infra-estrutura proposta, mas foram levados em conta no desenvolvimento do protótipo apresentado no capítulo anterior. O que fortalece a idéia de que Experimentação em Engenharia de Software como ferramenta poderosa a ser explorada para melhoria de processos e produtos de software, e tecnologias.

As observações gerais sobre os resultados dos estudos, embora não-conclusivas, são otimistas.

4.2 – 1º Estudo Experimental: Viabilidade de Empacotamento

4.2.1 – Definição dos Objetivos

4.2.1.1 – Objetivo Global

Caracterizar a viabilidade do empacotamento de experimentos utilizando a infra-estrutura proposta. Para tal será utilizado um pacote de experimento de um estudo de observação já executado sobre inspeção de software, onde foram comparadas a proposta de apoio automatizado à aplicação de PBR com sua aplicação convencional (SILVA, 2004; SILVA & TRAVASSOS, 2004).

Nesta etapa busca-se avaliar a aderência da infra-estrutura no empacotamento de experimentos, utilizando um experimento já empacotado sem nenhum apoio automatizado. Este estudo tem o mesmo intuito do estudo executado por AMARAL (2003).

4.2.1.2 – Objetivo da Medição

Investigar e caracterizar as possíveis características presentes ou ausentes que poderiam impedir o devido empacotamento de um estudo.

4.2.1.3 – Objetivo do Estudo

Analisar a infra-estrutura de apoio ao empacotamento e execução de experimentos

Com o propósito de caracterizar

Com respeito à aderência da infra-estrutura no empacotamento de experimentos

Do ponto de vista do pesquisador

No contexto do estudo de observação executado por SILVA (2004).

4.2.1.4 – Questões

Q1: A infra-estrutura de apoio ao empacotamento e execução de experimentos é capaz de empacotar um estudo experimental?

Métrica1: Número de artefatos que não podem ser empacotados pela infra-estrutura.

4.2.2 – Planejamento do Estudo

4.2.2.1 – Definição das Hipóteses

Hipótese nula (H0): Os artefatos de um estudo experimental, que podem ser empacotados sem nenhum apoio automatizado, não podem ser empacotados pela infra-estrutura de apoio ao empacotamento e execução de estudos experimentais.

A_{infra} : conjunto de artefatos de um estudo experimental empacotados pela infra-estrutura, e;

A_{conv} : conjunto de artefatos de um estudo experimental empacotados sem nenhum apoio (convencional).

H0: $|A_{infra}| - |A_{conv}| \neq 0$

Hipótese alternativa (H1): Os artefatos de um experimento, que podem ser empacotados sem nenhum apoio automatizado, podem ser também empacotados pela infra-estrutura.

H1: $|A_{infra}| - |A_{conv}| = 0$

4.2.2.2 – Seleção do contexto

O contexto é o estudo de observação realizado por SILVA (2004), onde é comparada a eficiência, o tempo e a relação eficiência/tempo da aplicação manual versus a aplicação automatizada da técnica de Leitura PBR (*Perspective Based Reading*) para inspeção de requisitos. Neste pacote constavam: dois documentos de requisitos de domínios diferentes, um sobre um sistema de estacionamento e outro sobre um sistema para vídeo-locadoras; 5 estudantes de pós-graduação do Programa de Engenharia de Sistemas e Computação da COPPE/UFRJ, 4 mestrandos e 1 doutorando, e; a técnica de Leitura para inspeção de requisitos PBR fornecida aos participantes com e sem o apoio ferramental proposto. No estudo realizado por (SILVA, 2004) todos os participantes inspecionaram os dois documentos e utilizaram a técnica PBR com e sem apoio ferramental, de modo que na sua execução foram realizadas duas rodadas experimentais dividindo os participantes em 2 grupos. Para este trabalho foi utilizado o pacote de experimento empacotado deste estudo experimental.

4.2.2.3 – Seleção dos indivíduos

O único participante desse estudo é o próprio pesquisador, devido ao seu conhecimento do processo e infra-estrutura propostos. Neste caso, o pesquisador simula todas as interações dos participantes e do pesquisador do estudo de SILVA (2004) com a infra-estrutura, desempenhando todos os diferentes papéis do processo proposto. As considerações sobre a validade deste estudo serão vistas a seguir.

4.2.2.4 – Variáveis

As variáveis independentes são:

1. A infra-estrutura de apoio ao empacotamento e execução de experimentos proposta.
2. Os artefatos do pacote do estudo de observação executado por SILVA (2004).

As variáveis dependentes são:

1. Os artefatos do pacote original que tentarão ser empacotados pela infra-estrutura proposta.

4.2.2.5 – Validade do Estudo

4.2.2.5.1 – Validade Interna

Nenhum aspecto quanto ao modo da aplicação dos tratamentos e social foi identificado. Nesse estudo o único participante foi o próprio pesquisador. Logo não houve a necessidade de, por exemplo, se evitar que a troca de informações entre os participantes que pudesse influenciar o resultado do estudo. Deste modo, qualquer consideração que leve em conta qualquer um desses dois tipos de aspectos pode comprometer a validade dos resultados e dever ser desconsiderada.

4.2.2.5.1 – Validade Externa

Validade externa mede a capacidade do estudo de refletir o mesmo comportamento em outros grupos de participantes. A infra-estrutura foi utilizada fora do contexto de execução de um experimento real e dentro do contexto de um experimento específico sobre inspeção de software, portanto, não é possível generalizar os resultados obtidos de forma mais ampla para experimentos fora do

contexto de inspeções, no qual se têm algum conhecimento consolidado sobre estudos realizados. Assim, os resultados estão restritos a este tipo estudo.

4.2.2.5.2 – Validade de Construção

A validade de construção considera os relacionamentos entre a teoria e a observação, ou seja, se o tratamento reflete bem a causa e o resultado reflete bem o efeito. Neste estudo o tratamento varia pelo uso ou não da infra-estrutura que reflete no resultado de um mesmo estudo poder ou não ser empacotado de maneiras diferentes. Deste modo, considera-se que os tratamentos e os resultados refletem bem a relação de causa-efeito que se busca observar.

4.2.2.5.4 – Validade de Conclusão

Embora os artefatos empacotados na infra-estrutura sejam os mesmos do estudo original, o pesquisador desempenhou todos os papéis ao longo do processo proposto através da simulação das interações realizadas dos diferentes usuários com a infra-estrutura. Assim, não há como concluir nada a respeito do comportamento dos diferentes tipos de usuários ao utilizarem esta infra-estrutura, já que o pesquisador tinha amplo conhecimento sobre sua utilização.

4.2.3 – Execução do Estudo

- Primeiramente, uma instância do *workflow* com o processo proposto é criada.
- A partir disso, o participante vai executando as atividades do *workflow*, desempenhando os diferentes papéis do processo proposto.
 - Ao desempenhar o papel de Pesquisador, o participante registra todos os artefatos, define os tratamentos. A única alteração nos artefatos necessária foi nos formulários de consentimentos e caracterização que foram convertidos em formulários em HTML.
 - Para simular o recrutamento, as notificações foram direcionadas ao participante. As atividades relativas à avaliação do estudo não foram executadas.
 - O participante preenche os consentimentos e caracterizações relativas a cada um dos participantes do estudo de observação original.

- De acordo com o planejamento do estudo original, o participante divide as caracterizações originais em grupos e lhes atribui os tratamentos.
- O treinamento do estudo original é dado em sala de aula, de modo que o participante só verifica se este estaria acessível para uso no momento do treinamento.
- O participante simula o envio do material de treinamento aos participantes.
- O próximo passo foi distribuir o material para a execução do estudo, o participante dispara o envio e verifica se o recebimento das notificações está correto.
- O participante simula cada participante do estudo original, submetendo os artefatos com os dados na infra-estrutura.
- O participante verifica se os dados foram corretamente submetidos.
- E por último o participante registra os documentos resultantes da análise do estudo original.

4.2.4 – Análise de Resultados

O número de artefatos do estudo de observação empacotados na infra-estrutura é o mesmo que o número de artefatos empacotados sem nenhum apoio automatizado. Logo, pelo conjunto de artefatos produzidos pelo experimento de SILVA, (2004) há o indício de que é possível se empacotar alguma classe de experimentos. Tanto este estudo quanto o estudo de AMARAL (2003) se tratavam de estudos experimentais controlados sobre inspeção de software. Em AMARAL (2003) o estudo empacotado era sobre a eficácia de diferentes técnicas de inspeção. Neste estudo foi empacotado um estudo experimental que tenta explorar uma possível abordagem de apoio ferramental. Embora se tratem de estudos sobre inspeção, são estudos que diferem muito, por exemplo, na instrumentação, onde um conta como uma ferramenta para apoiar a identificação de defeitos e o outro possui somente formulários a serem preenchidos pelos participantes.

Outro aspecto observado na análise, embora não explorado nas hipóteses, é que número total de tipos de artefatos empacotados e não pela infra-estrutura também é o mesmo, o que ratifica o resultado obtido no estudo de AMARAL (2003) quanto à aplicabilidade de sua taxonomia para categorizar artefatos do Processo de Experimentação.

A partir desses resultados, pode-se perceber que a hipótese nula (H0) foi refutada. Por conseqüência, a hipótese alternativa (H1), onde temos que $H1: |A_{\text{infra}}| - |A_{\text{conv}}| \neq 0$, foi confirmada.

4.2.5 – Lições Aprendidas

A infra-estrutura foi avaliada somente no sentido do empacotamento de um estudo. Mas seguindo o plano experimental do estudo original e o processo proposto, foi registrado um desvio em relação ao planejamento.

No estudo original, durante a execução, após a rodada experimental um questionário de acompanhamento é distribuído aos participantes. Já que esta não foi uma situação discutida (embora rotineira nos estudos executados na COPPE/UFRJ), este questionário foi enviado aos participantes junto com o material distribuído para a execução da rodada, onde instruções adicionais tiveram que ser dadas em uma das notificações. Este fato não impactou o empacotamento, mas poderia impactar a execução da rodada, já que este questionário tentava elucidar aspectos de usabilidade do apoio automatizado proposto por SILVA, (2004), podendo influenciar na execução de quem usou a técnica sem o apoio automatizado, e talvez pudesse ameaçar a validade dos resultados do estudo.

4.3 – 2º Estudo de Viabilidade: Execução de um Estudo Real

4.3.1 – Definição dos Objetivos

4.3.1.1 – Objetivo Global

Caracterizar a viabilidade de empacotamento e execução de experimentos utilizando a infra-estrutura proposta. Para tal executou-se um estudo de observação de uma técnica de leitura de inspeção de requisitos proposta por MAFRA (2006).

Nesta etapa busca-se avaliar a aderência da infra-estrutura tanto no empacotamento quanto na própria execução de um experimento, executando-o com o infra-estrutura proposta.

4.3.1.2 – Objetivo da Medição

Investigar e caracterizar as possíveis características presentes ou ausentes que poderiam impedir o devido empacotamento e execução de um estudo.

4.3.1.3 – Objetivo do Estudo

Analisar a infra-estrutura de apoio ao empacotamento e execução de experimentos

Com o propósito de caracterizar

Com respeito à aderência da infra-estrutura no empacotamento e execução de experimentos

Do ponto de vista do pesquisador

No contexto do estudo de observação sobre uma técnica de leitura de requisitos executado por MAFRA (2006).

4.3.1.4 – Questões

Q1: A infra-estrutura de apoio ao empacotamento e execução de experimentos é capaz de empacotar e executar um experimento?

Métrica1: Número de artefatos que não podem e deveriam ser empacotados na infra-estrutura.

4.3.2 – Planejamento do Estudo

4.3.2.1 – Definição das Hipóteses

Hipótese nula (H0): Um experimento não pode ser executado e empacotado na infra-estrutura.

A_{infra}: conjunto de artefatos de um experimento empacotados pela infra-estrutura, e;

A_{total}: conjunto de artefatos de um experimento quem deveriam ser empacotados na infra-estrutura.

H0: $|A_{infra}| - |A_{total}| \neq 0$

Hipótese alternativa (H1): Um experimento pode ser executado e empacotado na infra-estrutura.

H1: $|A_{infra}| - |A_{total}| = 0$

4.3.2.2 – Seleção do contexto

O contexto é o segundo estudo de observação para avaliar o uso de uma técnica de Leitura para inspeção de requisitos realizado por MAFRA (2006). No estudo

de MAFRA (2006) é medida a eficiência, o tempo e a relação eficiência/tempo dos participantes, onde foram utilizados 4 estudantes de pós-graduação (3 mestrandos e 1 doutorando) do Programa de Engenharia de Sistemas e Computação da COPPE/UFRJ. Foram utilizados 4 documentos de requisitos de projetos reais desenvolvidos por alunos da graduação do Departamento de Engenharia Eletrônica e do Departamento de Ciência da Computação da UFRJ. Cada um dos participantes revisou um documento de requisitos diferente utilizando a técnica proposta, denominada OO-PBR. Assim a atribuição dos tratamentos aos participantes foi individual e não pela formação de grupos, havendo uma única rodada experimental.

4.3.2.3 – Seleção dos indivíduos

Os participantes foram os 4 estudantes de pós-graduação, conforme descrito em MAFRA (2006), do Programa de Engenharia de Sistemas e Computação, 3 mestrandos e 1 doutorando, e o pesquisador responsável pelo segundo estudo de observação sobre a técnica de leitura OO-PBR.

4.3.2.4 – Variáveis

As variáveis independentes são:

3. A infra-estrutura de apoio ao empacotamento e execução de experimentos proposta.
4. Os artefatos do estudo experimental planejado por MAFRA, (2006).

As variáveis dependentes são:

2. O empacotamento e a execução realizada ou não pela infra-estrutura proposta.

4.3.2.5 – Validade do Estudo

4.3.2.5.1 – Validade Interna

Nenhum aspecto quanto ao modo da aplicação dos tratamentos e social foi identificado, já que eram de responsabilidades do pesquisador querendo avaliar a técnica de leitura proposta.

4.3.2.5.1 – Validade Externa

Embora a infra-estrutura tenha sido utilizada no contexto de execução de um experimento real, não é possível generalizar os resultados obtidos de forma mais ampla para experimentos fora do contexto de inspeções, no qual se têm algum conhecimento consolidado sobre suas execuções.

4.3.2.5.2 – Validade de Construção

Teoricamente poderemos observar se é possível ou não executar e empacotar um experimento com participantes e pesquisadores reais.

4.3.2.5.4 – Validade de Conclusão

Diferentemente do primeiro estudo, podemos observar, embora ainda não de forma conclusiva, o comportamento dos diferentes tipos de usuários ao utilizarem esta infra-estrutura.

4.3.3 – Execução do Estudo

- Uma instância do *workflow* com o processo proposto é criada.
- A partir disso, o pesquisador da técnica de leitura OO-PBR registra todos os artefatos e define os tratamentos a serem aplicados.
- No recrutamento, as notificações são enviadas aos participantes.
- As atividades relativas à avaliação do estudo não foram executadas.
- Os participantes são alocados em máquinas no Laboratório de Engenharia de Software da COPPE/UFRJ.
- Os participantes acessam a notificação de recrutamento, e em seguida preenchem os consentimentos e caracterizações.
- O treinamento foi dado em sala de aula, sem a necessidade do envio do material de treinamento aos participantes.
- De acordo com o planejamento, o pesquisador da técnica de leitura atribui 1 tratamento por participante.
- O próximo passo do pesquisador da técnica de leitura foi distribuir o material para a execução do estudo.
- A rodada experimental é iniciada e os participantes executam as tarefas no laboratório.

- Ao final da rodada, os participantes submetem os artefatos com os dados das tarefas executadas na infra-estrutura.
- O pesquisador da técnica de leitura verifica se os dados foram corretamente submetidos.
- E por último o pesquisador da técnica de leitura registra os documentos resultantes da análise do estudo original.

4.3.4 – Análise de Resultados

O estudo foi realizado, e devidamente empacotado como no primeiro estudo executado. Logo, pelo conjunto de artefatos produzidos pelo experimento de MAFRA (2006) há o indício de que é possível se empacotar e executar alguma classe de experimentos.

Outro aspecto observado na análise, embora não explorado nas hipóteses, é que não houve nenhum tipo de artefato que não pôde ser devidamente empacotado, o que ratifica o resultado obtido no estudo de AMARAL (2003) quanto à aplicabilidade de sua taxonomia para categorizar artefatos do Processo de Experimentação.

A partir desses resultados, pode-se perceber que a hipótese nula (H_0) foi refutada. Por conseqüência, a hipótese alternativa (H_1), onde temos que $H_1: |A_{infra}| - |A_{conv}| \neq 0$, foi confirmada.

4.3.5 – Lições Aprendidas

A infra-estrutura foi avaliada no sentido do empacotamento e execução de um estudo. Não foram registrados desvios em relação ao plano.

O pesquisador da técnica de leitura levou em torno de 1,67 horas para registrar os artefatos e definir os tratamentos. Nesse tempo está incluído o registro do plano experimental e da instrumentação mais a conversão dos formulários de consentimento e caracterização em formulários HTML. Na definição dos tratamentos, o pesquisador precisou ser auxiliado, pois o mesmo confundiu os conceitos de tratamento e variáveis dependentes. Já nas demais atividades, o pesquisador conseguiu realizar as tarefas sem maiores dificuldades e intervenções, sendo esclarecidos somente alguns aspectos sobre o processo proposto e o apoio às tarefas de cada atividade.

Em relação aos participantes, todas as notificações de *email* enviadas não foram questionadas pelos mesmos. O único questionamento levantado por um dos participantes foi como saber se todos os arquivos submetidos tinham sido recebidos corretamente pela infra-estrutura. Tal participante foi respondido e todos os outros participantes também foram informados, notificados e respondidos sobre a questão durante a coleta de dados. O tempo médio total de uso da infra-estrutura pelos

participantes foi aproximadamente 3 horas e 50 minutos. As interações mais longas dos participantes foram no acesso ao material distribuído durante a rodada experimental. A tabela 5.1 mostra o tempo gasto por cada participante na infraestrutura do preenchimento do consentimento de participação até a submissão dos dados ao final da execução da rodada experimental.

Tabela 5.1- Tempo Gasto pelos Participantes

Participante	Hora do Preenchimento do Consentimento	Hora da Submissão dos Dados	Tempo Total Gasto
P1	14:22	17:11	2:51
P2	14:22	17:54	3:32
P3	13:44	18:31	4:47
P4	14:27	18:33	4:06

Vale salientar que a infra-estrutura, o pesquisador e os participantes estavam alocados em uma rede de computadores do Laboratório de Engenharia de Software da COPPE/UFRJ, que não apresentou nenhuma falha durante a execução do estudo e que pode influenciar o tempo de acesso aos recursos.

Deste modo, as únicas observações relevantes são relativas às interações do pesquisador e dos participantes durante a execução. Do ponto de vista do pesquisador, foi verificado que talvez seja indicado o treinamento dos pesquisadores no uso da infra-estrutura ou fornecimento de uma ajuda *on-line*. Já os participantes demonstraram certa dificuldade em identificar se a coleta dos dados foi efetivamente realizada.

4.4 – Lições aprendidas dos Estudos Executados

A grande diferença entre estes dois estudos diz respeito às interações dos usuários com a infra-estrutura, já que os dois estudos, embora com propósitos de avaliação diferentes, tratavam do domínio de inspeção de software. Assim, o que se pode dizer de um modo generalizado sobre estes estudos é que é possível executar pelo menos alguns tipos de estudos experimentais utilizando a infra-estrutura *Web*.

Em nenhum destes estudos foram utilizados em suas amostras uma grande população de indivíduos e/ou geograficamente distribuídos, logo não se sabe ao certo o quão adequado seria o apoio da infra-estrutura na execução de estudos

experimentos em larga escala e/ou ambientes fora da academia, os que tornariam estudos mais próximos da definição de estudos *in-vivo*.

Por outro lado, fica bastante evidente que, apesar do apoio fornecido, é necessário conhecimento no Processo de Experimentação e no processo implementado na infra-estrutura para o total entendimento do encadeamento das tarefas e atividades relativo à execução de estudos experimentais em Engenharia de Software.

4.5 – Conclusão

A partir das lições aprendidas dos dois estudos executados, foram propostas melhorias na infra-estrutura já incorporadas na versão apresentada no capítulo anterior. A primeira melhoria é que pesquisadores podem enviar notificações com instruções adicionais liberando o acesso a artefatos durante a coleta de dados entre e/ou final das rodadas experimentais. Na segunda, os participantes são capazes de visualizar o nome dos arquivos submetidos durante a coleta de dados, com o intuito de dar a percepção explícita dos artefatos submetidos e recebidos na infra-estrutura. Novos estudos precisariam ser executados para avaliar se somente essas melhorias efetivamente suprem as necessidades identificadas, ou são necessárias medidas adicionais.

Além disso, foi identificada a necessidade de treinamento dos pesquisadores na infra-estrutura. Acredita-se que tratar aspectos de usabilidade e de interface gráfica, poderiam atenuar parte das dificuldades encontrada pelo pesquisador (participante do segundo estudo), bem como melhorar o entendimento sobre os procedimentos a cerca da execução de experimentos. Com estes dois estudos pretendeu-se avaliar se os resultados obtidos justificam os recursos a serem despendidos com a continuidade do investimento na tecnologia proposta. De um modo geral, acredita-se que sim, graças aos indícios da viabilidade de utilização da infra-estrutura no empacotamento e execução de experimentos em Engenharia de Software.

Capítulo 5 – Considerações finais

Este capítulo apresenta as principais conclusões, contribuições, limitações e perspectivas de trabalhos futuros.

5.1 – Conclusões

Nesta dissertação foi discutida a complexidade a cerca da execução de estudos experimentais em Engenharia de Software e apresentada a proposta de uma infra-estrutura para apoiar o planejamento, execução e empacotamento de experimentos. Dentre os objetivos desta infra-estrutura estão: tornar mais gerenciável a complexidade e, conseqüentemente, os riscos associados à execução de estudos experimentais, e; organizar e tornar disponível o conhecimento relativo aos estudos experimentais executados aos pesquisadores da comunidade de Engenharia de Software.

Para verificar a viabilidade de construção desta infra-estrutura um protótipo foi construído e avaliado experimentalmente por dois estudos. Nestes dois estudos foram identificados alguns aspectos que não foram identificados na definição dos requisitos da infra-estrutura, e a observação de que é necessário o treinamento dos pesquisadores para seu uso. A partir dos resultados dos estudos apresentados, acredita-se que há indícios suficientes de que pode ser proveitoso investir mais recursos no desenvolvimento desta tecnologia.

Realizar pesquisa com rigor científico é difícil e necessário em Engenharia de Software, dado o impacto dos aspectos humanos e sociais nos processos e produtos de software, e no uso de tecnologias. É necessário identificar e conhecer estes impactos, em que circunstâncias ocorrem, identificar se é possível reduzi-los ou como gerenciá-los.

5.2 – Contribuições

As contribuições deste trabalho estão:

1. Na descrição do conhecimento de execução de estudos experimentais adquirido na Equipe de Engenharia de Software Experimental COPPE/UFRJ e com pesquisadores do Projeto Readers CNPq/NSF;
2. A descrição de uma infra-estrutura e um processo de execução de estudos experimentais em Engenharia de Software;

3. A definição de uma abordagem que pode ser utilizada para repetição de estudos experimentais;
4. A construção de um protótipo disponibilizando essa infra-estrutura na *Web*, e;
5. A apresentação de estudos experimentais realizados para avaliar a viabilidade da infra-estrutura e seus resultados, bem como as análises a cerca desses resultados.

Este trabalho está ainda relacionado à definição e construção de um ambiente de experimentação em Engenharia de Software, o eSEE. Onde se pretende usar a infra-estrutura proposta como uma ferramenta disponível para apoiar a execução de estudos experimentais neste ambiente e em larga escala.

5.3 – Limitações

As limitações deste trabalho em relação ao protótipo da infra-estrutura estão principalmente relacionadas às questões de usabilidade e interface gráfica, que estavam fora do escopo deste trabalho e que foram observadas nos estudos de avaliação. Além disso, poucas informações são coletadas sobre as instâncias dos processos executados que possam ser utilizadas a nível de gerenciamento da infra-estrutura.

Outra limitação deste trabalho que pode ser apontada é o fato de não ter havido tempo hábil de planejar e executar estudos para avaliar a proposta de apoio à repetição de estudos experimentais, embora este apoio esteja razoavelmente condizente com o conhecimento tácito dos pesquisadores da Equipe de Engenharia de Software Experimental da COPPE/UFRJ, e tenha sido apresentada a pesquisadores do Projeto Readers.

Deve-se ressaltar que os dois estudos realizados para avaliar a infra-estrutura foram dentro do contexto experimentação relacionada a inspeções de software, o que pode futuramente implicar na necessidade de evoluções para adequação da infra-estrutura para execução de estudos experimentais em outros contextos.

5.4 – Perspectivas futuras

As perspectivas futuras deste trabalho são que outros estudos experimentais sejam planejados e executados para avaliar:

- A usabilidade e interface com usuário da infra-estrutura;
- A abordagem de repetição de estudos experimentais;

- A utilização da infra-estrutura em experimentos em contextos diferentes de inspeções de software;
- O treinamento necessário para pesquisadores e, talvez, participantes no uso da infra-estrutura,e;
- A utilização da infra-estrutura para apoiar experimentação em larga escala em Engenharia de Software.

Com os resultados destas avaliações, espera-se que seja possível definir novos requisitos, melhorias e evoluções na infra-estrutura para dar alcance à execução de uma classe maior de estudos experimentais da Engenharia de Software.

Um aspecto interessante de ser abordado no contexto desta infra-estrutura seria padronizar a representação dos dados quantitativos e qualitativos coletados durante a execução das rodadas para assim tornar possível a integração de ferramentas, como PBRTool (SILVA, 2004; SILVA & TRAVASSOS, 2004), com a infra-estrutura, permitindo que os dados coletados fossem automaticamente empacotados via *Web Service*, por exemplo. Essa padronização poderia facilitar a agregação e visualização dos dados de um estudo sob várias dimensões e talvez uma maior facilidade na agregação de dados de vários estudos, auxiliando pesquisadores a realizar meta-análise e a tirar conclusões mais gerais sobre classes de estudos executados.

Há alguns pontos não trabalhados, por estarem fora do escopo deste trabalho, que poderiam ser tratados em relação ao pacote do experimento e a abordagem para repetição de estudos experimentais.

Um deles é o fato de não haver mecanismos de controle de versões e de rastro de alterações dos artefatos de um pacote de estudo experimental, já que durante a repetição de um estudo estes artefatos geralmente são modificados e/ou evoluídos. Tais mecanismos, além de permitir um acompanhamento das alterações dos artefatos, poderiam ajudar a entender sob quais circunstâncias um estudo ou uma família de estudos experimentais evolui.

Outro ponto importante neste sentido é a ausência de mecanismos de proteção e controle de autoria dos artefatos, ou seja, criar mecanismos que permitam aos pesquisadores definir políticas que irão reger as alterações dos artefatos durante a criação ou evolução de um pacote experimental. Tais mecanismos visariam a proteção da propriedade intelectual dos autores dos artefatos.

E por fim, poderíamos destacar que alguns estudos são repetidos através da composição de sub-conjuntos de artefatos de diferentes pacotes experimentais quando há algum aspecto comum entre estes estudos que se quer observar. Na abordagem proposta para repetição de estudos experimentais, uma repetição com

essa característica não pode ser realizada. Deste modo, para comportar esse tipo de repetição novas funcionalidades e políticas de acesso e alteração dos artefatos devem ser definidos e anexados à infra-estrutura.

Referências Bibliográficas

- AMARAL, E.A.G.G., 2003, *Empacotamento de Experimentos em Engenharia de Software*, Dissertação de Mestrado, Programa de Engenharia de Sistemas e Computação, COPPE/UFRJ, Universidade Federal do Rio de Janeiro, Rio de Janeiro, RJ, Brasil.
- AMARAL, E.A.G.G., TRAVASSOS, G.H., 2001, "Empacotamento de Experimentos em Engenharia de Software". In: *Workshop de Teses em Engenharia de Software - WTES'2001 (SBES)*, pp. 47-50, Rio de Janeiro, Brasil.
- ARISHOLM, E., SJØBERG, D., CARELIUS, G., LINDSJØRN, Y., 2002a, "SESE – an Experiment Support Environment for Evaluating Software Engineering Technologies". In: *NW-PER2002 (Tenth Nordic Workshop on Programming and Software Development Tools and Techniques)*, pp. 81-98, Copenhagen, Denmark, August.
- ARISHOLM, E., SJØBERG, D.I., CARELIUS, G.J., LINDSJØRN, Y., 2002b, "A Web-based Support Environment for Software Engineering Experiments", *Nordic Journal of Computing*, v. 9, n. 3 (September), pp. 231-247.
- BABBIE, E., 1990, *Survey Research Methods*, 2th ed., Wadsworth, 0-524-12672-3.
- BARROS, M.O., WERNER, C.M.L., TRAVASSOS, G.H., 2001, "From Metamodels to Models: Organizing and Reusing Domain Knowledge in System Dynamics Model Development". In: *19th International Conference of the System Dynamics Society*, Atlanta, July.
- BARROS, M.O., WERNER, C.M.L., TRAVASSOS, G.H., 2002, "Um Estudo Experimental sobre a Utilização de Modelagem e Simulação no Apoio a Gerência de Projetos de Software". In: *XVI Simpósio Brasileiro de Engenharia de Software, Simpósio Brasileiro de Engenharia de Software*, Gramado, Rio Grande do Sul, RS, Brasil.
- BASILI, V., COSTA, P., LINDVALL, M., MENDONCA, M., SEAMAN, C., TESORIERO, R., ZELKOWITZ, M., 2001, "An Experience Management System for a Software Engineering Research Organization". In: *26th Annual NASA Goddard Software Engineering Workshop*, pp. 29 - 35, November.
- BASILI, V.R., 1985, "Quantitative Evaluation of Software Engineering Methodology". In: *First Pan Pacific Computer Conference*, Melbourne, Austrália, September.
- BASILI, V.R., 1992, *Software Modeling and Measurement: The Goal Question Metric Paradigm*, University of Maryland, Computer Science Technical Report Series, CS-TR-2956 (UMIACS-TR-92-96).

- BASILI, V.R., 1996, "The Role of Experimentation: Past, Present, Future". In: *18th International Conference on Software Engineering*, pp. 442-449, Berlin, Germany.
- BASILI, V.R., CALDEIRA, G., ROMBACH, H.D., 1994, "Experience Factory", *Encyclopedia of Software Engineering*, ed. J.J. Marciniak, v. 1, pp. 469-476.
- BASILI, V.R., MCGARRY, F.E., PAJERSKI, R., ZELKOWITZ, M.V., 2002, "Lessons Learned from 25 Years of Process Improvement: The Rise and Fall of The NASA Software Engineering Laboratory". In: *24th International Conference on Software Engineering*, pp. 69 - 79, Orlando, Florida.
- BASILI, V.R., SHULL, F., LANUBILE, F., 1999, "Building Knowledge through Families of Experiments", *IEEE Transactions on Software Engineering*, v. 25, n. 4, pp. 456-473.
- BERANDER, P., 2004, "Using Students as Subjects in Requirements Prioritization". In: *International Symposium on Empirical Software Engineering*, pp. 167-176.
- BIOLCHINI, J., MIAN, P.G., NATALI, A.C.C., TRAVASSOS, G.H., 2005, *Systematic Review in Software Engineering*, Programa de Engenharia de Sistemas e Computação - PESC, COPPE-UFRJ, Technical Report ES-679/05.
- CAMPBELL, D.T., STANLEY, C., 1979, *Delineamentos Experimentais e Quase Experimentais de Pesquisa*, São Paulo, Brasil, EPU - Editora da Universidade de São Paulo.
- CMMI Overview, "Capability Maturity Model® Integration (CMMI®) Overview". In: <http://www.sei.cmu.edu/cmmi/adoption/pdf/cmmi-overview05.pdf>, accessed in December 16th.
- CARVER, J., 2003, *The Impact of Background and Experience on Software Inspections*, Ph.D. Thesis, Department of Computer Science, University of Maryland.
- CARVER, J., SHULL, F., BASILI, V.R., 2003, "Observational Studies to Accelerate Process Experience in Classroom Studies: An Evaluation". In: *2003 International Symposium on Empirical Software Engineering*, pp. 72-80.
- CARVER, J., VANVOORHIS, J., BASILI, V.R., 2004, "Understanding the Impact of Assumptions on Experimental Validity". In: *International Symposium on Empirical Software Engineering*, pp. 251-260.
- CHAPETTA, W.A., CONTE, T.U., TRAVASSOS, G.H., 2004, *Requisitos para um Ambiente de Experimentação em Engenharia de Software*, Programa de Engenharia de Sistemas e Computação - PESC/COPPE/UFRJ, Relatório Técnico da Equipe de Engenharia de Software Experimental.

- CHAPETTA, W.A., SANTOS, P.S.M., TRAVASSOS, G.H., 2005, "Supporting Meta-Description Activities in Experimental Software Engineering Environments". In: *2nd Experimental Software Engineering Latin American Workshop - ESELAW'05*, Uberlândia, Brazil, September.
- CONTE, T., MENDES, E., TRAVASSOS, G.H., 2005, "Processos de Desenvolvimento para Aplicações Web: Uma Revisão Sistemática". In: *XI Simpósio Brasileiro de Sistemas Multimídia e Web*, Poços de Caldas, Brasil, Dezembro.
- COSTA, H.R., BARROS, M.O., TRAVASSOS, G.H., 2005, "Uma Abordagem Econômica Baseada em Riscos para Avaliação de uma Carteira de Projetos de Software". In: *19º Simpósio Brasileiro de Engenharia de Software*, pp. 311-326, Uberlândia, Minas Gerais, Brasil.
- DALY, J., EMAN, K.E., MILLER, J., 1997, "Multi-method research in Software Engineering". In: *IEEE Workshop on Empirical Studies Software Maintenance*, Bari, Italy, October.
- DIAS NETO, A.C., BARCELOS, R., A., C.W., SANTOS, P.S.M., MAFRA, S.N., TRAVASSOS, G.H., 2004, "Infrastructure for SE Experiments Definition and Planning". In: *1st Experimental Software Engineering Latin American Workshop - ESELAW'04*, Brasília, Brazil.
- DYBA°, T., KAMPENES, V.B., SJØBERG, D.I.K., 2005a, "A systematic review of statistical power in software engineering experiments", *Information and Software Technology* (November).
- DYBA°, T., KITCHENHAM, B.A., JØRGENSEN, M., 2005b, "Evidence-Based Software Engineering for Practitioners", *IEEE Software*, v. 22, n. 1, pp. 58-65.
- XML Specification, "Extensible Markup Language (XML) 1.0 (Third Edition)". In: <http://www.w3.org/TR/REC-xml/>, accessed in January 23th.
- FENTON, N.E., PFLEEGER, S.L., 1997, *Software Metrics: A Rigorous and Practical Approach*, 2nd ed., International Press, International Thomson Computer Press.
- FREITAS, M.E.D., PAGLIUSO, P., BOAS, A.L.V., TAMBASCIA, C., MALDONADO, J.C., HÖHN, E., FABBRI, S., 2004, "Inspeção de Documento de Requisitos Baseado em Técnicas de Leitura PBR: Experiência Prática no CPqD". In: *III Simpósio Brasileiro de Qualidade de Software*, pp. 153-160, Brasília, Junho.
- HARRISON, W., 2000, "N = 1: An Alternative for Software Engineering Research?" In: *Beg, Borrow, or Steal: Using Multidisciplinary Approaches in Empirical Software Engineering Research Workshop at 22nd International Conference on Software Engineering (ICSE)*, Limerick, Ireland.
- HÖHN, E., MALDONADO, J.C., MENDONÇA, M., FABBRI, S., BOAS, A.L.V., TAMBASCIA, C., FREITAS, M.E.D., PAGLIUSO, P., 2004, "PBR: Transferência

- de Tecnologia Baseada em Pacotes de Experimentação". In: *III Simpósio Brasileiro de Qualidade de Software*, pp. 161-175, Brasília, Junho.
- HOST, M., REGNELL, B., WOHLIN, C., 2000, " Using Students as Subjects - A Comparative Study of Students and Professionals in Lead-time Impact Assessment ", *Empirical Software Engineering: An International Journal*, v. 5, n. 3 (November), pp. 201-214.
- HOVE, S.E., ANDA, B., 2005, "Experiences from Conducting Semi-Structured Interviews in Empirical Software Engineering Research". In: *11th IEEE International Software Metrics Symposium*, pp. 23-33, September.
- JEDLITSCHKA, A., PFAHL, D., 2005, *Reporting Guidelines for Controlled Experiments in Software Engineering*, Fraunhofer Institute for Experimental Software Engineering, Technical Report IESE-REPORT IESE-035.5/E.
- JURISTO, N., MORENO, A.M., 2001, *Basics of Software Engineering Experimentation*, Hardcover, ISBN: 0-7923-7990-X.
- KARAHASANOVIC, A., ANDA, B., ARISHOLM, E., HOVE, S.E., JØRGENSEN, M., SJØBERG, D.I.K., WELLAND, R., 2005 "Collecting Feedback During Software Engineering Experiments", *Empirical Software Engineering*, v. 10, n. 2 (April), pp. 113 - 147.
- KERLINGER, F.N., 1973, *Foundations of Behavioral Research*, 2th ed., Holt, Rinehart and Winston, Inc.
- KIM, W., GRAUPNER, S., SAHAI, A., LENKOV, D., CHUDASAMA, C., WHEDBEE, S., LUO, Y., DESAI, B., MULLINGS, H., WONG., P., 2002, "Web E-speak: facilitating Web-based e-Services", *Multimedia, IEEE*, v. 9, n. 1 (Jan-Mar), pp. 43-55.
- KITCHENHAM, B.A., 1996, "Evaluating Software Engineering Methods and Tools", *SIGSoft Software Engineering Notes, ACM Press*, pp. 11-15.
- KITCHENHAM, B.A., 2004, "Systematic Reviews". In: *10th International Symposium on Software Metrics*, September.
- KITCHENHAM, B.A., DYBA°, T., JØRGENSEN, M., 2004, "Evidence-based Software Engineering". In: *International Conference on Software Engineering (ICSE'04)*, pp. 273-281, Edinburgh.
- KITCHENHAM, B.A., PFLEEGER, S.L., 2002, "Principles of Survey Research, Part 5: Populations and Samples", *ACM SIGSOFT Software Engineering Notes*, v. 27, n. 5 (September).
- KITCHENHAM, B.A., PFLEEGER, S.L., PICKARD, L.M., JONES, P.W., HOAGLIN, D.C., EMAM, K.E., ROSENBERG, J., 2002, "Preliminary Guidelines for

- Empirical Research in Software Engineering", *IEEE TRANSACTIONS ON SOFTWARE ENGINEERING*, v. 28, n. 8 (AUGUST), pp. 721-734.
- KITCHENHAM, B.A., PICKARD, L., PFLEEGER, S.L., 1995, "Case Studies for Method and Tool Evaluation", pp. 52-62, July.
- LIMA, G.M.P.S., 2005, *Heurísticas para Identificação da Ordem de Integração em Testes de Integração Aplicados a Software Orientado a Objetos*, Tese de Mestrado, PESC/COPPE, Universidade Federal do Rio de Janeiro, Rio de Janeiro, Brasil.
- MAFRA, S.N., 2004, *Infra-estrutura para Automatização de Processos de Software*, Monografia de Projeto Final de Curso, Departamento de Ciência da Computação - DCC/IM, Universidade Federal do Rio do Rio de Janeiro, Rio de Janeiro, RJ, Brazil.
- MAFRA, S.N., 2005, "Técnicas de Leitura: Uma Revisão Sistemática". In: *19º Simpósio Brasileiro de Engenharia de Software*, pp. 72-87, Uberlândia, Minas Gerais, 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, Universidade Federal do Rio de Janeiro, Rio de Janeiro, Brazil.
- MIAN, P., CHAPETTA, W., SANTOS, P.S., MELO JR., C.R., NATALI, A.C.C., BIOLCHINI, J., TRAVASSOS, G.H., G, R., A.C.R., 2005a, "eSEE: an Infrastructure for Supporting Experimental Software Engineering". In: *4th IEEE/ACM International Symposium on Empirical Software Engineering (ISESE2005)*, Australia, November.
- MIAN, P.G., TRAVASSOS, G.H., ROCHA, A.R.C., 2005b, "A Computerized Infrastructure for Supporting Experimentation in Software Engineering". In: *2nd Experimental Software Engineering Latin American Workshop – ESELAW'05*, Uberlândia, Brazil, September.
- MIAN, P.G., TRAVASSOS, G.H., ROCHA, A.R.C., NATALI, A.C.C., 2004, "Towards a Computerized Infrastructure for Experimental Software Engineering Knowledge". In: *Jornadas Iberoamericanas em Ingeniería del Software e Ingeniería del Conocimiento*, Madrid, Spain, November.
- MATTOSO, M., TRAVASSOS, 2006, *Ciência em Larga Escala*, SBC - Grandes Desafios em Computação, Sociedade Brasileira de Computação, In: <http://www.sbc.org/>.
- MONTONI, M.A., 2003, *Aquisição de Conhecimento: Uma Aplicação no Processo de Desenvolvimento de Software*, Tese de Mestrado, Programa de Engenharia de

- Sistemas e Computação, COPPE/UFRJ, Universidade Federal do Rio de Janeiro, Rio de Janeiro, Brazil.
- MOOKERJEE, V.S., CHIANG, I.R., 2002, "A Dynamic Coordination Policy for Software System Construction", *Transactions On Software Engineering*, v. 28, n. 6 (June).
- NONAKA, I., TAKEUCHI, H., 1995, *The Knowledge-Creating Company*, Oxford University Press.
- OTT, L.M., KINNULA, A., SEAMAN, C.B., WOHLIN, C., 1999, "The Role of Empirical Studies in Process Improvement", *Empirical Software Engineering*, v. 4, n. 4, pp. 381-386.
- PFLEEGER, S.L., 1999, "Albert Einstein and Empirical Software Engineering", *IEEE Computer* (October), pp. 32-38.
- PRECHELT, L., UNGER, B., PHILIPPSEN, M., 2001, "A Controlled Experiment on Inheritance Depth as a Cost Factor for Code Maintenance", *Journal of Systems and Software* (November).
- PUNTER, T., CIOLKOWSKI, M., FREIMUT, B., JOHN, I., 2003, "Conducting on-line Surveys in Software Engineering". In: *International Symposium on Empirical Software Engineering*, pp. 80 - 88 Sep./Oct.
- QUADROS, E.R.A., 2005, *Ferramenta para Construção de Modelos de Processos e Documentos, para Apoiar o Planejamento e Execução de Experimentos em Engenharia de Software*, B.Sc. Final Project Dissertation, Departamento de Engenharia Eletrônica - DEL/CT, Universidade Federal do Rio de Janeiro, Rio de Janeiro, RJ, Brazil.
- RAMIL, J.F., LEHMAN, M.M., 2000, "Metrics of software evolution as effort predictors - a case study". In: *International Conference on Software Maintenance*, pp. 163 – 172, October 11-14.
- HTTP/1.1 Specification, "RFC 2616: Hypertext Transfer Protocol -- HTTP/1.1 ". In: <ftp://ftp.isi.edu/in-notes/rfc2616.txt>, accessed in January 23th.
- ROSENBAUM, P., 1995, *Observational Studies*, Springer-Verlag.
- SALLIE, M.H., HUMPHREY, M., 1990, *A controlled experiment to evaluate maintainability of object-oriented software*, Computer Science, Virginia Polytechnic Institute and State University, Technical Report NCSTRL.VATECH_CS/TR-90-30.
- SESSIONS, R., 2005, "Fuzzy Boundaries: Objects, Components, and Services", *ACM Queue Magazine*, v. 2, n. 9 (January), pp. 40-47.

- SHAW, M., 2002, "What Makes Good Research in Software Engineering?" *International Journal of Software Tools for Technology Transfer*, v. 4, n. 1, pp. 1-7.
- SHNEIDERMAN, B., MAYER, R., MCKAY, D., HELLER, P., 1977, "Experimental Investigations of the Utility of Detailed Flowcharts in Programming", *Communications ACM*, v. 20, n. 6 (June), pp. 373-381.
- SHULL, F., BASILI, V.R., CARVER, J., MALDONADO, J.C., TRAVASSOS, G.H., MENDONÇA, M., FABBRI, S., 2002, "Replicating Software Engineering Experiments: Addressing the Tacit Knowledge Problem". In: *International Symposium on Empirical Software Engineering*, pp. 7-16, Nara, Japan, October.
- SHULL, F., CARVER, J., TRAVASSOS, G.H., 2001, "An Empirical Methodology for Introducing Software Processes". In: *8th European Software Engineering Symposium and 9th ACM SIGSOFT Symposium on the Foundations of Software Engineering (FSE-9) and 8th European Software Engineering Conference (ESEC)*, Vienna, Austria, September.
- SHULL, F., MENDONÇA, M.G., BASILI, V., CARVER, J., MALDONADO, J.C., FABBRI, S., TRAVASSOS, G.H., FERREIRA, M.C., 2004, "Knowledge-Sharing Issues in Experimental Software Engineering", *Empirical Software Engineering*, v. 9, n. 1-2, pp. 111 - 137.
- SILVA, L.F.S., 2004, *UMA ABORDAGEM COM APOIO FERRAMENTAL PARA APLICAÇÃO DE TÉCNICAS DE LEITURA BASEADA EM PERSPECTIVA*, Programa de Engenharia de Sistemas e Computação COPPE/UFRJ, Universidade Federal do Rio de Janeiro, Rio de Janeiro.
- SILVA, L.F.S., TRAVASSOS, G.H., 2004, "Tool-Supported Unobtrusive Evaluation of Software Engineering Process Conformance". In: *International Symposium on Empirical Software Engineering (ISESE'04)*, v. 00, pp. 127 - 135, August.
- SOAP, "Simple Object Access Protocol (SOAP) 1.1". In: <http://www.w3.org/TR/2000/NOTE-SOAP-20000508/>, accessed in January 24th.
- SJØBERG, D.I.K., ANDA, B., ARISHOLM, E., DYBA, T., JORGENSEN, M., KARAHASANOVIC, A., KOREN, E.F., VOKAC, M., 2002, "Conducting Realistic Experiments in Software Engineering". In: *International Symposium on Empirical Software Engineering*, pp. 17-26, October.
- SJØBERG, D.I.K., HANNAY, J.E., HANSEN, O., KAMPENES, V.B., KARAHASANOVIC, A., LIBORG, N.-K., REKDAL, A.C., 2005, "A Survey of

- Controlled Experiments in Software Engineering", *IEEE TRANSACTIONS ON SOFTWARE ENGINEERING*, v. 31, n. 9 (SEPTEMBER), pp. 733-753.
- SPEM Specification, "Software Process Engineering Metamodel, version 1.1". In: <http://www.omg.org/cgi-bin/doc?formal/2005-01-06>, accessed in January 23th.
- TAYLOR, S.J., BOGDAN, R., 1998, *Introduction to Qualitative Research Methods*, Wiley, ISBN 0471168688.
- TICHY, W.F., 1998, "Should Computer Scientists Experiment More?" *IEEE Computer*, v. 31, n. 5 (May), pp. 32-40.
- TRAVASSOS, G.H., BARROS, M.D.O., 2003, "Contributions of In Virtuo and In Silico Experiments for the Future of Empirical Studies in Software Engineering". In: *2nd Workshop on Empirical Software Engineering (WSESE'03) - The Future of Empirical Studies in Software Engineering.*, Rome, Italy.
- TRAVASSOS, G.H., SHULL, F., FREDERICKS, M., BASILI, V.R., 1999, "Detecting Defects in Object Oriented Designs: Using Reading Techniques to increase Software Quality", *ACM Sigplan Notices*, v. 34, n. 10, pp. 47-56.
- UDDI, "Universal Description, Discovery and Integration (UDDI)". In: <http://www.uddi.org/specification.html>, accessed in January 24th.
- VERGILIO, S.R., MALDONADO, J.C., JINO, M., 1995, "Um Experimento de Aplicação de Critérios Baseados em Fluxo de Dados no Teste de Programas C". In: *XV Congresso da Sociedade Brasileira de Computação e XXI Conferência Latinoamericana de Informática*, Canela, RS, Brasil, Agosto.
- VILLELA, K., 2004, *Ambientes de Desenvolvimento de Software Orientados à Organização*, D.Sc. Thesis, Programa de Engenharia de Sistemas e Computação -PESC/COPPE, Universidade Federal do Rio de Janeiro -UFRJ, Rio de Janeiro, RJ, Brazil.
- WSDL, "Web Services Description Language (WSDL) 1.1". In: <http://www.w3.org/TR/wsdl>, accessed in January 24th.
- WOHLIN, C., AURUM, A., PETERSSON, H., SHULL, F., CIOLKOWSKI, M., 2002, "Software Inspection Benchmarking - A Qualitative and Quantitative Comparative Opportunity". In: *Eighth IEEE Symposium on Software Metrics*, pp. 118-128.
- WOHLIN, C., RUNESON, P., HÖST, M., OHLSSON, M.C., REGNELL, B., WESSLÉN, A., 2000, *Experimentation in Software Engineering: an Introduction*, Massachusetts, USA, Kluwer Academic Publishers, 0-7923-8682-5.
- WOOD, M., DALY, J., MILLER, J., ROPER, M., 1999, "Multi-Method Research: An Empirical Investigation of Object-Oriented Technology", *The Journal of Systems and Software*, v. 48, n. 1 (August), pp. 13-26.

- YIN, R.K., 1994, *Case Study Research: Design and Methods*, 2th ed., Beverly Hills, California, USA, Sage Publications, Inc.
- ZELKOWITZ, M.V., WALLACE, D.R., 1998, "Experimental Models for Validating Technology", *IEEE Computer*, v. 31, n. 5 (May), pp. 23 - 31.
- ZELKOWITZ, M.V., WALLACE, D.R., BINKLEY, D.W., 2003, "Experimental Validation of New Software Technology", *Lectures Notes on Empirical Software Engineering*, World Scientific.
- Zope, "Zope Application Server". In: <http://www.zope.org/>, accessed in January 23th.

ANEXO I

eSEE: experimental Software Engineering Environment

Um Ambiente de Engenharia de Software deve prover o apoio às diferentes tarefas executadas por diferentes indivíduos desempenhando diferentes papéis ao longo de um processo de software. Em CHAPETTA *et al.* (2004) foram definidas algumas pré-condições para a definição de um ambiente que pudesse apoiar a Experimentação em Engenharia de Software, e são elas:

- Integrar ferramentas de suporte a experimentação, atuando de forma similar a um Ambiente de Desenvolvimento de Software (ADS).
- Ser um sistema Web – para permitir sua utilização em diferentes localidades, incluindo pesquisadores de diferentes instituições.
- Ter arquitetura baseada no paradigma de *e-services*. Segundo (KIM *et al.*, 2002), *e-services* são serviços disponíveis sobre uma rede (Internet ou Intranet), incluindo aplicações, recursos de computação, processos e informação, onde usuários e/ou aplicações devem ser capazes de buscá-los, localizá-los e invocá-los, em uma *infra-estrutura* distribuída. Assim como componentes, *e-services* são usados por proverem um determinado conjunto de funcionalidades, ou seja, são como aplicações auto-contidas e modulares. Mas estes devem poder ser publicados, localizados e invocados sobre uma rede.
- Possuir mecanismos que apoiem a Gestão de Conhecimento – devido a Experimentação em Engenharia de Software ser uma atividade intensa em conhecimento (BASILI *et al.*, 1999; SHULL *et al.*, 2004).

Com base nessas pré-condições, a Equipe de Engenharia de Software Experimental da COPPE/UFRJ (<http://www.cos.ufrj.br/~ese>) partiu para o detalhamento inicial de quais seriam as características desejáveis para o Meta-Ambiente eSEE e, assim, satisfazer requisitos para a construção de uma base de conhecimento de estudos experimentais em Engenharia de Software e capturar os requisitos desejáveis para sistemas Web de Gestão do Conhecimento (CHAPETTA *et al.*, 2004). Para identificar tais características foi realizada uma análise das atividades de acompanhamento, execução e empacotamento de experimentos em Engenharia de

Software conforme as propostas de WOHLIN *et al.* (2000), AMARAL (2003) e MIAN *et al.* (2004).

A partir da definição desses requisitos foi construída uma arquitetura inicial para o eSEE (DIAS NETO *et al.*, 2004). A arquitetura definida é composta por três macro-componentes, de acordo com um agrupamento das características desejáveis na infraestrutura. Esses macro-componentes são o Meta-Configurador, o Ambiente de Instanciação e o Ambiente de Execução (CHAPETTA *et al.*, 2005). A seguir são descritos os macro-componentes que compõe a arquitetura.

O Meta-Configurador possui como funcionalidade definir e configurar todos os elementos básicos que serão utilizados pelos dois outros componentes. Esses elementos básicos são:

- Modelos de documentos: são descrições que servem como base instanciação de artefatos produzidos ao longo do Processo de Experimentação. O Meta-Configurador descreve modelos de documentos a partir de um meta-modelo de documentos pré-estabelecido;
- Modelos de Processo: serve como base para a definição e instanciação de Processos de Experimentação. O Meta-Configurador descreve modelos de processos de software a partir de um meta-modelo de processos pré-estabelecido;
- Serviços Configurados: são informações relativas à utilização de e-services que possam apoiar diferentes atividades do Processo de Experimentação. Esses e-services, ou simplesmente serviços, disponibilizados têm de ser capazes de ter seus uso customizados para serem disponibilizados na infra-estrutura do eSEE.

Note que estes elementos são apenas definidos, mas nenhum relacionamento entre esses conceitos é descrito. Logo nesse componente estamos somente definindo o tipo, organização e granularidade das informações de cada um desses conceitos. Para cada conceito existe um sub-componente que o define: O componente de Definição de Modelos de Processos, componente de Definição de Modelos de Documentos e o componente de Configuração de Serviços. Depois de descritos, os conceitos são armazenados em repositórios específicos do Ambiente de Instanciação.

O segundo componente, o Ambiente de Instanciação, possui como objetivo prover meios para customizar e instanciar um ambiente que permita a execução de um Processo de Experimentação. Baseado em um modelo de processos, um modelo de documentos e os serviços configurados disponíveis definidos no componente anterior, este componente relaciona os artefatos a serem consumidos/produzidos às atividades do Processo de Experimentação bem como os possíveis serviços configurados a

serem disponibilizados na realização de uma determinada tarefa. Para tal, as seguintes ações são realizadas:

- São definidos os relacionamentos entre o modelo do processo a ser seguido, o modelo de documentos que vai ser instanciado e os serviços configurados disponíveis para execução de um estudo experimental. Este conjunto de relacionamentos é denominado Mapa de Associação.
- Baseando-se nos relacionamentos descritos entre os elementos básicos da arquitetura eSEE, este ambiente instancia um Ambiente de Execução. Nesse momento o modelo de processo e o modelo de documento são utilizados para instanciar um processo de Experimentação e um Pacote de Documentos, respectivamente. Enquanto os serviços configurados ficam disponíveis para serem invocados nas atividades do processo instanciado de acordo com os relacionamentos estabelecido em um determinado Mapa de Associação.

Cada mapa de Associação é então armazenado em um repositório específico, permitindo várias instanciações de uma determinada configuração de um Ambiente de Execução.

O Ambiente de Execução é o componente responsável pelo acompanhamento e execução das atividades do Processo de Experimentação. O acompanhamento e controle do estudo serão realizados através do processo definido pelo modelo de processo, de acordo com as restrições de acesso às informações e entre atividades do processo. O acesso às informações de um documento será realizado através dos serviços configurados, onde os relacionamentos descritos no Mapa de Associação são responsáveis pelo link entre o artefato, uma determinada atividade do processo e o serviço configurado para realizar uma determinada tarefa.

A utilização de *e-services* seria a principal maneira de anexar funcionalidades e ferramentas e, assim, customizar um Ambiente de Execução. Na figura 1.1 temos uma visão arquitetural do eSEE e seus macro-componentes.

Vale salientar que um ambiente instanciado eSEE existe sem nenhuma dependência do Ambiente de Instanciação, e que o Ambiente de Instanciação não tem nenhum controle sobre os ambientes eSEE instanciados. A mesma observação pode ser feita entre o Ambiente Meta-Configurador e o Ambiente de Instanciação respectivamente. Deste modo o eSEE apresenta uma arquitetura de modo que seus macro-componentes podem estar geograficamente distribuídos.

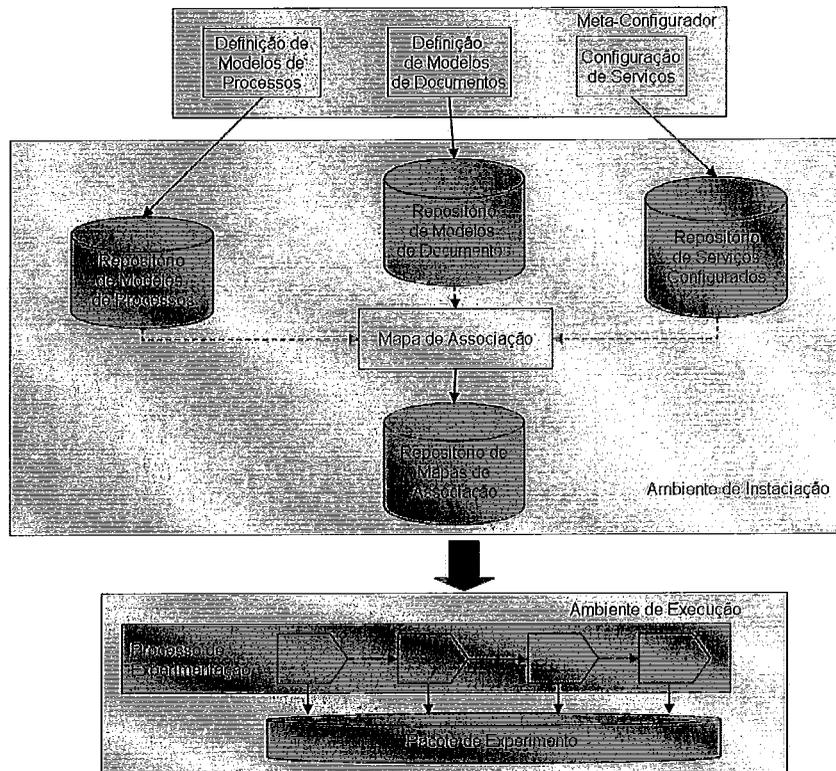


Figura 1.1- Componentes da Arquitetura eSEE

Para maiores detalhes ver (MIAN *et al.*, 2004; MIAN *et al.*, 2005a; MIAN *et al.*, 2005b).

Em uma primeira versão de um protótipo construído do eSEE foram selecionados um sub-conjunto dos requisitos do Meta-Ambiente para limitar o escopo. O intuito desse primeiro protótipo era verificar a viabilidade, obter-se uma noção do esforço e dificuldades na construção da arquitetura proposta, e utilizarmos essas informações identificar e controlar possíveis riscos. A primeira versão do protótipo foi basicamente construída como aplicações em C++ que realizavam troca de dados em XML (XML Specification, 2006) sobre conexões HTTP/1.1 (HTTP/1.1 Specification, 2006). XML é uma linguagem de marcação que permite a representação de dados de forma semi-estruturada. E o HTTP/1.1 é um protocolo de camada de aplicação para troca de dados sobre uma rede, baseado em requisições de objetos (DIAS NETO *et al.*, 2004).

Por restrições de tempo e outros recursos, as funcionalidades relacionadas à disponibilização de serviços no eSEE não foram contempladas. Além disso, no componente Meta-Configurador foram somente disponibilizadas as funcionalidades que permitiam a publicação e disponibilização dos modelos de processos e documentos que serviriam de base para a instaciações de Ambientes de Execução pelo Ambiente de Instaciação. Os modelos de processos e de documentos são

descritos em arquivos XML. A descrição dos modelos de processos seguem o meta-modelo proposto por VILLELA (2004), e a descrição do modelos de documentos seguem um meta-modelo proposto por alunos da Equipe de Engenharia de Software Experimental da COPPE/UFRJ.

Já que as funcionalidades relativas à disponibilização de serviços foram suprimidas, o Ambiente de Instanciação era capaz de definir somente Mapas de Associação relacionando as atividades de um modelo de processo e os artefatos que serão produzidos e consumidos pelo processo instanciado.

Os repositórios dos modelos de Processo, de Documentos e Mapas de associação de foram construídos em servidor de aplicações com um Banco de Dados OO e interface Web, denominado Zope (Zope, 2006). Todas as informações eram registradas/recuperadas nos repositórios através de requisições HTTP cujos conteúdos eram fragmentos de informações em XML.

O Ambiente de Execução foi construído, na primeira versão do protótipo eSEE, se adaptando uma ferramenta para acompanhamento e controle de processos de software denominada enactPro MAFRA (2004), que utiliza modelos de processos descritos em XML segundo o meta-modelo proposto por VILLELA (2004). A adaptação refere-se principalmente ao acesso dos artefatos instanciados para utilização do processo em execução, funcionalidade esta que não existia antes da construção deste protótipo. As informações dos artefatos são também acessadas e armazenadas em um repositório específico sobre a plataforma Zope, com requisições HTTP e fragmentos de arquivos XML.

Em uma segunda fase de construção do eSEE, partiu-se para a definição de algumas decisões de projeto mais concretas. Nesta etapa buscou-se definir como seriam construídas as funcionalidades relativas e a disponibilização dos serviços-configurados, já que estes não foram contemplados na primeira versão do protótipo eSEE.

Web services são e-services e serão utilizados no Meta-Ambiente eSEE como a principal forma de se anexar e customizar funcionalidade e ferramentas em um Ambiente de Execução. Mas são um tipo específico de e-services, pois se comunicam através do protocolo HTTP através da troca de mensagens em SOAP (protocolo de mensagens em XML).

Deste modo, um serviço configurado é a descrição de um *Web Service* disponível para utilização no Meta-Ambiente eSEE (CHAPETTA *et al.*, 2005).

Em uma arquitetura baseada em *e-services*, como a do eSEE, *e-services* constituem os principais blocos de construção de aplicações, geralmente não convencionais, onde cada bloco co-existe em um sistema sem nenhuma dependência

de outro. Como esses blocos estão distribuídos, para serem utilizados, estes devem ser capazes de serem publicados, localizados e invocados por outros *e-services* ou aplicações. Para cada *e-service* encontramos os seguintes elementos neste tipo de arquitetura:

- O Provedor do Serviço que é responsável por prover um conjunto de funcionalidades sobre uma rede.
- O Publicador do Serviço que é responsável por fornecer ao Provedor do Serviço a capacidade de registrar um *e-service*; e publicar (disponibilizar) em uma rede as informações sobre onde, como acessar e as funcionalidades do *e-service* registrado.
- O Requerente do Serviço que é um outro *e-service* ou aplicação que deseja utilizar uma determinada funcionalidade de um *e-service*.

A figura 1.2 ilustra o paradigma de uma arquitetura baseada em *e-services*.

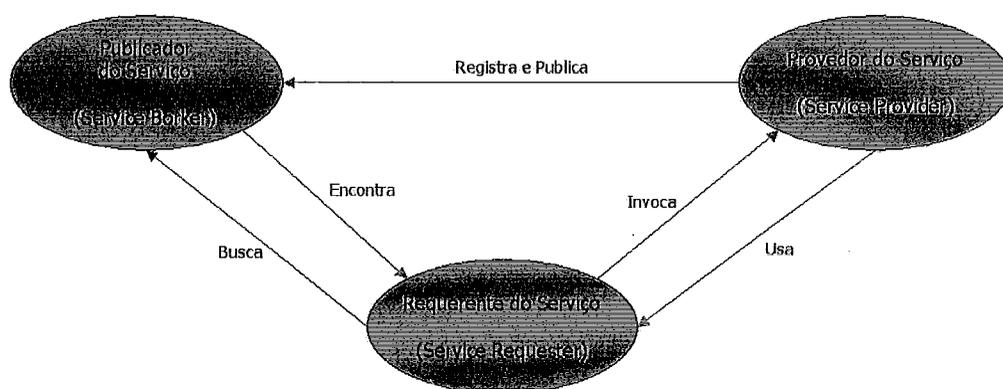


Figura 1.2- Elementos de uma Arquitetura Orientada a *e-services*

Tomando por base um crescente interesse da indústria nos últimos anos, a disponibilidade de apoio de construção em várias plataformas de desenvolvimento, e a utilização de padrões abertos e gratuitos, optou-se por se definir, construir e disponibilizar os serviços configurados, e consequentemente os *e-services*, no eSEE como *Web Services*.

Segundo SESSIONS (2005), objetos, componentes e *Web services* apresentam muitas características em comum (são blocos de código, possuem interfaces, residem em um processo, suportam o conceito de requisição ou invocação, dentre outras). Mas as diferenças entre essas três entidades são principalmente dirigidas por dois fatores: localidade e ambiente. Localidade refere-se ao local relativo dos processos onde entidade (por exemplo, um objeto) e cliente (por exemplo, o núcleo de um sistema)

residem. Ambiente refere-se ao ambiente de execução da entidade e do cliente. Quando o relacionamento entre entidade e cliente é no mesmo processo e consequentemente no mesmo ambiente, então a entidade é um objeto. Se este relacionamento está em processos diferentes e no mesmo ambiente então a entidade é um componente. Logo se o relacionamento entre a entidade e cliente é entre processos e ambientes diferentes, então a entidade é um *Web service*.

Muitas plataformas de desenvolvimento apóiam a construção, publicação ou a utilização de *Web Services*, fornecendo alguma abstração de mais alto nível para trabalhar com suas tecnologias correlatas. Estas tecnologias são a WSDL, UDDI e SOAP (<http://www.oasis.org> e <http://www.w3.org>).

A linguagem WSDL (WSDL, 2006) é a linguagem utilizada para descrever onde e como acessar, bem como a interface das operações disponíveis de um *Web service*. A linguagem é descrita usando uma sintaxe pré-definida em XML. Deste modo, o descritor WSDL de um *Web service* é a informação disponível e referenciada em um repositório UDDI.

Um repositório UDDI é o elemento Publicador de serviços numa arquitetura baseada em e-services. O UDDI (UDDI, 2006) representa um conjunto de especificações que permitem o registro, a publicação e a busca das informações sobre *Web Services* armazenadas em um repositório.

O SOAP (SOAP, 2006) é um protocolo para troca de mensagens em XML entre um *Web Service* e um Requerente de um *Web Service*. Geralmente as mensagens são trocadas através de requisições HTTP.

A figura 4.3 ilustra a como essas tecnologias são empregadas em uma arquitetura baseada em e-services.

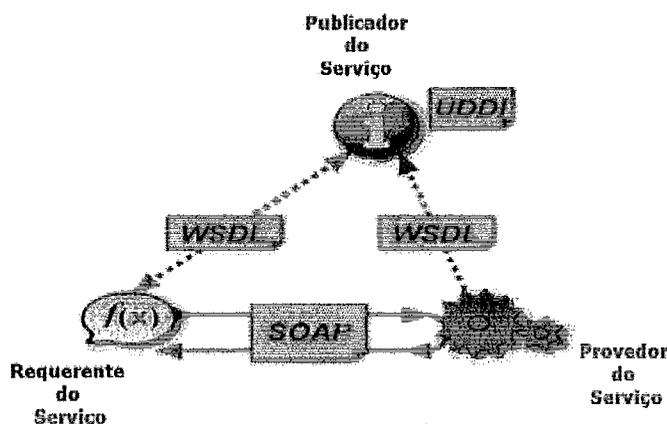


Figura 1.3- Tecnologias relacionadas à *Web services* em uma Arquitetura Orientada a e-services

Como parte de um trabalho de pesquisa de Iniciação Científica, foi construído e anexado à primeira versão do protótipo o componente de Configuração de Serviços do macro-componente Meta-Configurador. Esse componente foi construído como uma ferramenta mesclando recursos do provedor de *Web services* Axis, na plataforma Java, (<http://ws.apache.org/axis/>) e um componente construído em Python (<http://www.python.org>). A esta ferramenta denominou-se Configurador de Serviços.

Lembrando que o intuito deste componente é selecionar, configurar e disponibilizar e-services, o Configurador de Serviços serve para selecionar um *Web service*, definir como este poderá ser utilizado e disponibilizá-lo para um Ambiente de Execução. Deste modo, o que esta ferramenta faz é capturar as definições da interface original do *Web Service* e define novas interfaces para que estas possam ser referenciadas nos Mapas de associação no Ambiente de Instanciação e acessadas pelo Ambiente de Execução. Cada nova interface é uma customização do *Web service* original que é publicada e torna-se disponível para sua utilização (CHAPETTA *et al.*, 2005). A seguir uma ilustração da ferramenta desenvolvida na figura 1.4.

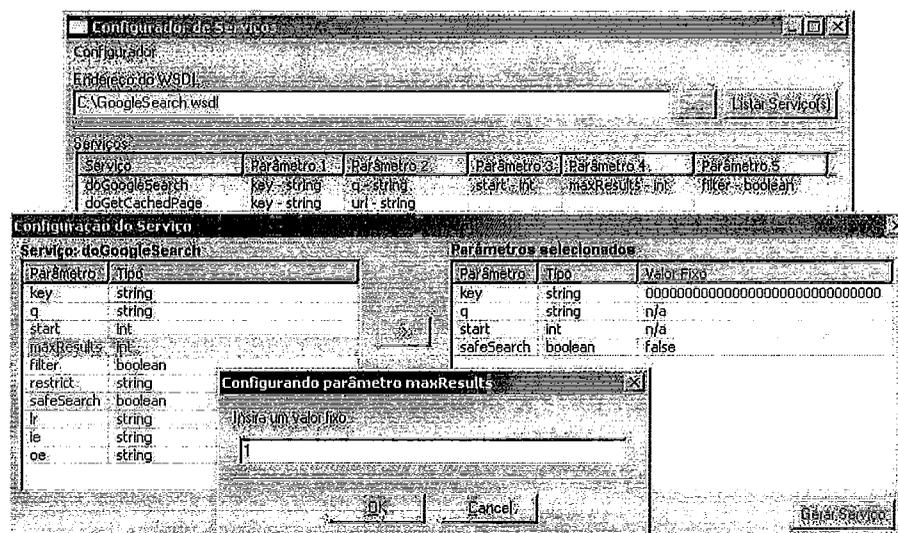


Figura 1.4- Configurador de Serviços do Meta-Ambiente eSEE (CHAPETTA *et al.*, 2005)

Praticamente em paralelo a construção do Configurador de Serviços e como um trabalho de projeto final de curso do Departamento de Engenharia Eletrônica e Computação da Universidade Federal do Rio de Janeiro, foi construída uma ferramenta para implementar os componentes de Definição de Modelos de Processo e de Documentos. Deste modo, esses dois componentes estão encapsulados em uma única aplicação. Esta ferramenta continua gerando os modelos em arquivos XML

segundo seus respectivos meta-modelos, que permaneceram praticamente inalterados. A ferramenta foi desenvolvida na Linguagem C++, gera os arquivos e os armazena em repositórios específicos, definidos na plataforma Zope (QUADROS, 2005). Abaixo na figura 1.5, uma gravura ilustra a ferramenta construída para modelagem de processo e documentos (CHAPETTA *et al.*, 2005; QUADROS, 2005).

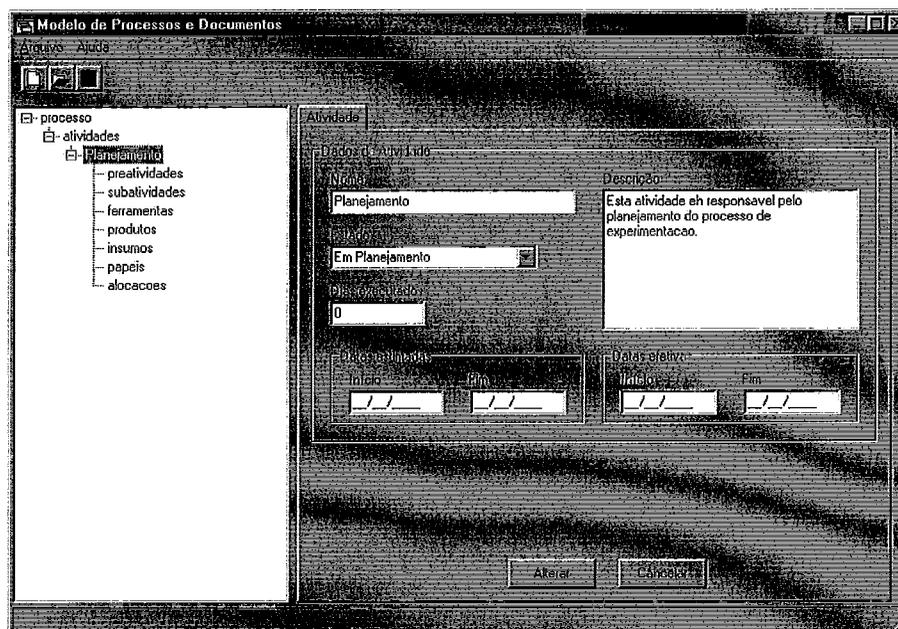


Figura 1.5- Ferramenta de Modelagem de Processos e Documentos do Meta-Ambiente eSEE (CHAPETTA *et al.*, 2005)

Logo em seguida à construção dessas ferramentas, todo o Ambiente de Instanciação foi reprojeto para suportar as funcionalidades relativas à utilização de *Web services*. Este foi um trabalho desenvolvido em uma disciplina de graduação do Departamento de Engenharia Eletrônica da Universidade Federal do Rio de Janeiro, onde um grupo de alunos executou um processo em cascata até produzirem os artefatos de projeto de baixo nível para a redefinição do Ambiente de Instanciação.

Na conjuntura atual deste trabalho, cada sub-componente do Meta-Configurador foi construído e anexado à primeira versão do protótipo eSEE. No Ambiente de Instanciação, foram redefinidos os requisitos e seu projeto para suportarem a utilização de *Web services*. Já o Ambiente de Execução ainda precisa ser adaptado para invocar *Web services* ao invés de acessar as informações via requisições HTTP e fragmentos de dados em XML.

ANEXO II

Documento de Requisitos de uma Infra-Estrutura para Apoiar a Execução, Empacotamento e Repetição de Estudos Experimentais em Engenharia de Software

Objetivo:

Construção de uma infra-estrutura capaz de auxiliar e guiar a etapa de Execução de Estudos Experimentais em Engenharia de Software através da instanciação de workflows.

Glossário:

Workflow: seqüência de atividades que devem ser executadas de forma sistemática com intuito de se alcançar um objetivo.

Formulário Web: é um documento com o corpo de um Formulário HTML. (<http://www.w3.org>)

Etapa de Execução: etapa de Execução do Processo de Experimentação segundo (WOHLIN *et al.*, 2002).

Estudo Experimental ou Experimento: estudos in-vitro e in-vivo segundo (TRAVASSOS & BARROS, 2003).

Engenheiro de Software: responsável pela gerência da *infra-estrutura*.

Pesquisador: responsável pela realização de um estudo experimental e pela gerência de *workflows* instanciados para execução de estudos experimentais.

Participante do Estudo Experimental (ou somente participante): são os indivíduos que proverão os dados a serem observados e/ou medidos para posterior análise dos resultados de um estudo experimental.

Plano do Estudo Experimental ou Plano do Experimento: documento relativo ao planejamento realizado para a execução de um estudo experimental (DIAS NETO *et al.*, 2004; AMARAL, 2003; WHOLIN *et al.*, 2000).

Lição Aprendida: é um relato textual sobre alguma observação relativa a execução, como por exemplo um desvio do Plano do Estudo Experimental, que se quer compartilhar com outros pesquisadores para disseminar melhores práticas ou chamar a atenção de possíveis ameaças relativo a execução de um estudo (JEDLITSCHKA & PFAHL, 2005).

Artefato do Experimento: é um artefato que será utilizado para apoiar a execução de um estudo experimental, geralmente não estando relacionado ao objeto de estudo do experimento. Por exemplo, um questionário de acompanhamento (AMARAL, 2003).

Artefato de Software: é um artefato de software geralmente relacionado ao objeto de estudo do experimento. Por exemplo, um documento de requisitos (SILVA, 2004). Artefatos de software possuem um tipo e estão definidos em (AMARAL, 2003).

Instrumentos: artefatos a serem utilizados pelos participantes para realizarem alguma tarefa na Execução do estudo experimental. São os Artefatos do Experimento, Artefatos de Software, Formulários de Consentimento e de Caracterização, e Materiais de Treinamento (AMARAL, 2003).

Tratamento de um Estudo Experimental: organização de como os artefatos serão distribuídos aos participantes, levando-se em consideração as variáveis que serão observadas (e/ou medidas) no estudo experimental.

Distribuição: relação de como os tratamentos serão distribuídos aos participantes de um estudo experimental. Podem ser levados em conta procedimentos como Balanceamento e Blocking (WOHLIN *et al.*, 2002) no momento de definir uma distribuição.

Artefatos de Medição: segundo (JURISTO & MORENO, 2001), são artefatos que serão medidos para se obter os dados qualitativos e quantitativos da execução do estudo experimental.

Formulário de Caracterização do Participante: um formulário utilizado para obter informações do(s) participante(s) relativa(s) ao contexto do estudo experimental.

Caracterização do Participante: São os dados obtidos do preenchimento do Formulário de Caracterização dos Participantes.

Formulário de Consentimento: um formulário utilizado para selar o comprometimento entre o(s) pesquisador(es) e o(s) participante(s) quanto a confidencialidade, utilização de resultados e outros aspectos da participação do(s) indivíduos relativas ao contexto do estudo experimental.

Consentimento: são os dados obtidos do preenchimento do Formulário de Consentimento.

Material de Treinamento: Conjunto de artefatos que compõem o material a ser utilizado no procedimento de Treinamento dos Participantes de um estudo experimental.

Pacote de Experimento: conjunto de artefatos produzidos ao longo da execução do Processo de Experimentação. Seu armazenamento e organização visam facilitar a repetição de estudos experimentais.

Introdução:

A execução de estudos experimentais em Engenharia de Software tem sido um dos meios pelos quais pesquisadores desta área tentam avaliar novas técnicas, metodologias e ferramentas. Mas como em qualquer disciplina científica, o Processo de Experimentação é uma atividade laboriosa e gera uma quantidade muito grande de informação que precisa gerenciada.

Segundo (WOHLIN et al., 2000), o Processo de Experimentação pode ser descrito pelas seguintes atividades:

- *Definição: onde pesquisadores tentam definir o escopo e o objeto de*

estudo de um determinado experimento.

- *Planejamento: onde pesquisadores planejam como ocorrerá a execução do estudo experimental.*
- *Execução: onde pesquisadores conduzem o estudo experimental segundo as decisões contidas no Plano do Estudo Experimental.*
- *Análise: onde pesquisadores analisam e registram os resultados do estudo experimental a partir dos dados obtidos da execução.*
- *Empacotamento: onde todas as informações, dados e artefatos resultantes do Processo de Experimentação são agregados para formar o Pacote do Experimento.*

Na expectativa de apoiar as atividades bem como gerenciar as informações produzidas ao longo do Processo de Experimentação, a equipe de Engenharia de Software Experimental da COPPE/UFRJ propõe um ambiente de apoio a execução de estudos experimentais. A este se denominou eSEE (experimental Software Engineering Environment) (DIAS NETO et al., 2004).

Os requisitos iniciais do eSEE foram definidos em CHAPETTA et al (2004) e uma arquitetura foi proposta baseado em componentes distribuídos (NETO et al. 2004). A partir desses requisitos e dessa arquitetura alguns trabalhos vêm sendo direcionados na tentativa de avaliar a viabilidade da proposta do eSEE. Pela característica de distribuição dos componentes, a abordagem SOA (Service Oriented Architecture) e Web services vêm sendo utilizados na construção dos módulos do eSEE. As ferramentas de apoio ao Processo de Experimentação serão disponibilizadas através de Web services para utilização no eSEE.

Dentro deste contexto, este documento apresenta os requisitos de uma infraestrutura cujo objetivo é apoiar a execução do Processo de Experimentação. Nesta infraestrutura temos 4 papéis: o Engenheiro de Software, o Pesquisador, o Revisor Externo e o Participante do Estudo Experimental (ou somente participante). Um Engenheiro de Software deve ser capaz de instanciar e atribuir a um pesquisador de um estudo experimental um workflow de execução de estudos experimentais. Além disso, o Engenheiro de Software deve ser capaz de visualizar em que atividades o(s) pesquisadores(s) do(s) estudo(s) experimental(is) se encontra(m) e os artefatos produzidos na execução desse(s) estudo(s). Um pesquisador de um estudo experimental pode acessar as atividades dos workflows instanciados que lhe foram atribuídas e é responsável pelas informações registradas na infraestrutura bem como a execução e notificação de conclusão das atividades relativas à execução de um estudo experimental. Um Participante de um Estudo Experimental deve ser capaz de interagir com a infraestrutura para obter e registrar informações relativas à execução

de um estudo experimental. Um Revisor Externo é um indivíduo selecionado para realizar uma revisão dos artefatos de um estudo experimental. Este indivíduo deve ter acesso aos artefatos de um estudo registrados na infra-estrutura e deve emitir um laudo de sua revisão utilizado na avaliação do estudo experimental.

Em linhas gerais, na execução de um workflow o pesquisador deve ser capaz de:

- *Definir se a execução é relativa a uma repetição ou não;*
- *Registrar os Documentos provenientes da Definição e Planejamento;*
- *Registrar os Instrumentos do Estudo Experimental*
- *Descrever os tratamentos planejados para execução de um estudo;*
- *Obter a Caracterização e Consentimento dos Participantes.*
- *Descrever como ocorrerá a distribuição do material aos participantes do estudo;*
- *Coletar os dados provenientes dos Participantes do estudo experimental;*
- *Executar um estudo piloto;*
- *Avaliar um estudo via um revisor externo;*
- *Ter disponível o material de treinamento e/ou disponibilizá-lo aos participantes do estudo;*
- *Registrar os resultados provenientes da execução obtidos através dos participantes, e;*
- *Registrar os documentos provenientes da Análise (apesar dessa atividade não contemplar a etapa de Execução do Processo de Experimentação, ela está sendo provida para que ao final do Processo de Experimentação o pesquisador possa registrar esses artefatos e desse modo possamos ter um Pacote de Experimento).*

Para ilustrar o workflow de execução proposto temos as figuras 1 abaixo.

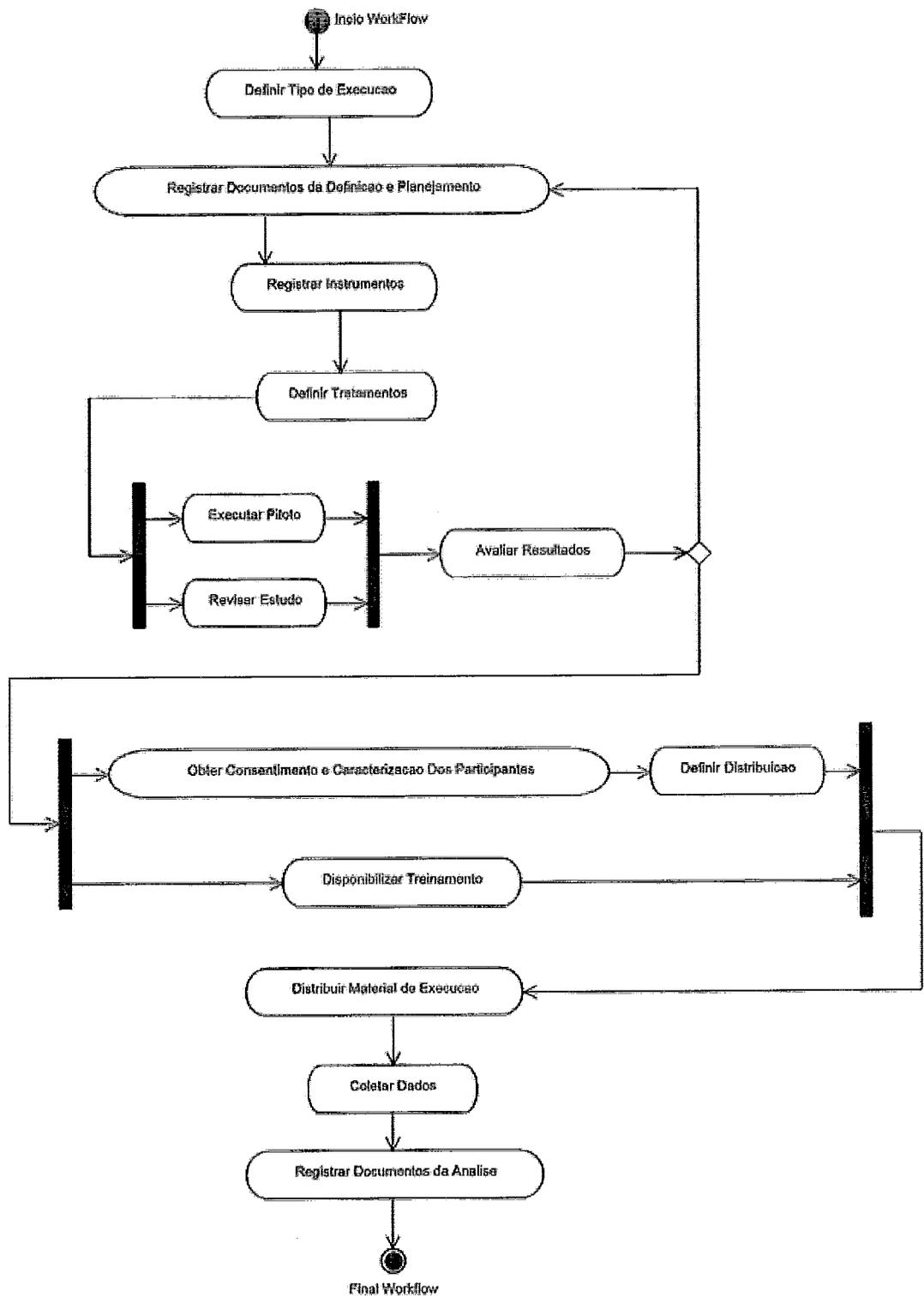


Figura 1- Workflow de Execucao de Estudos Experimentais

Requisitos Funcionais:

Requisito Funcional 1

A infra-estrutura deve permitir que um Engenheiro de software instancie workflows para auxiliar e guiar a etapa de Execução do Processo de Experimentação, atribuindo ao pesquisador do estudo experimental a responsabilidade sobre a gerência do workflow instanciado.

Requisito Funcional 2

A infra-estrutura deve permitir que o pesquisador do estudo experimental anexe os documentos produzidos nas etapas de Definição e Planejamento, como por exemplo, o Plano do Estudo Experimental, sob a forma de arquivos. No caso de uma repetição, os documentos da Definição e Planejamento registrados do estudo original serão copiados e estarão disponíveis para que o Pesquisador do Estudo Experimental os altere.

Requisito Funcional 3

A infra-estrutura deve permitir que o pesquisador do estudo experimental registre os instrumentos que serão utilizados na etapa de Execução do Processo de Experimentação. Artefatos de Software e do Experimento serão registrados sob a forma de arquivos. Os Formulários de Consentimento e de Caracterização serão registrados como corpos de formulários HTML. Entre os instrumentos estão os artefatos que compõem o material de treinamento. No caso de uma repetição, os instrumentos registrados do estudo original serão copiados e estarão disponíveis para que o Pesquisador do Estudo Experimental os altere.

Requisito Funcional 4

A infra-estrutura deve permitir que o pesquisador do estudo experimental descreva os tratamentos planejados a serem utilizados na execução de um estudo experimental. No caso de uma repetição, os

tratamentos descritos do estudo original serão copiados e estarão disponíveis para que o Pesquisador do Estudo Experimental os altere.

Requisito Funcional 5

A infra-estrutura deve permitir que o pesquisador do estudo experimental avalie todo material produzido e anexado através da execução de um estudo piloto. Como resultado desta avaliação, o pesquisador do estudo experimental deve registrar na infra-estrutura um laudo de execução do estudo piloto sob a forma de um arquivo e indicar se o estudo pode ou não ser executado com as informações disponíveis.

Requisito Funcional 6

A infra-estrutura deve permitir que o pesquisador do estudo experimental disponibilize via browser todo material registrado a um revisor externo para que este os revise. Como resultado desta revisão, o revisor externo deve registrar na infra-estrutura um laudo de avaliação da revisão sob a forma de um arquivo e indicar se o estudo pode ou não ser executado com as informações disponíveis.

Requisito Funcional 7

A infra-estrutura deve permitir que o pesquisador do estudo experimental avalie os resultados (laudo e indicação) da execução de um estudo piloto e/ou de uma revisão externa e decida se o estudo experimental poderá ser executado ou se são necessários acertos nos artefatos.

Requisito Funcional 8

A infra-estrutura deve permitir que o pesquisador do estudo experimental publique os Formulários de Caracterização e Consentimento na Web e notifique os participantes da publicação desses formulários via e-mail.

Requisito Funcional 9

A infra-estrutura deve permitir que o(s) participante(s) do estudo experimental preencha(m) os formulários de Consentimento e Caracterização via browser. A Caracterização e Consentimento dos Participantes serão armazenados como Artefatos de Medição.

Requisito Funcional 10

A infra-estrutura deve permitir que o pesquisador do estudo experimental descreva como ocorrerá a distribuição dos tratamentos aos participantes em um determinado estudo, a partir dos Artefatos de Medição que contêm a Caracterização dos Participantes.

Requisito Funcional 11

A infra-estrutura deve permitir que o pesquisador do estudo experimental acesse o material de Treinamento via browser e ainda que este material possa ser disponibilizado via browser ao(s) participantes. O(s) participante(s) deverá(ão) ser notificado(s) via e-mail desta disponibilização.

Requisito Funcional 12

A infra-estrutura deve permitir que o pesquisador do estudo experimental indique o momento em que o material relativo à execução de um estudo estará disponível via browser aos participantes. Os participantes deverão ser notificados via e-mail desta disponibilização.

Requisito Funcional 13

A infra-estrutura deve permitir que o(s) participante(s) do estudo experimental registre(m) os resultados da execução através de um formulário disponível na Web. Os artefatos registrados devem ser armazenados como Artefatos de Medição. O pesquisador do estudo experimental deve indicar quando o registro dos resultados pelos participantes será encerrado.

Requisito Funcional 14

A infra-estrutura deve permitir que o pesquisador do estudo experimental registre desvios em relação ao Plano do Experimento, observações sobre a execução na forma de lições aprendidas ao longo das atividades de execução de um estudo experimental (JEDLITSCHKA & PFAHL, 2005).

Requisito Funcional 15

A infra-estrutura deve permitir que o pesquisador do estudo experimental registre os artefatos resultantes da Etapa de Análise do Processo de Experimentação e em seguida deve disponibilizar os artefatos registrados e os dados produzidos na forma de um Pacote de Experimento. No caso de uma repetição, os documentos da Análise registrados do estudo original serão copiados e estarão disponíveis para que o Pesquisador do Estudo Experimental os altere.

Requisito Funcional 16

A infra-estrutura deve permitir que o Engenheiro de Software visualize o estado de execução de um workflow, bem como os artefatos e os dados registrados durante sua execução.

Requisito Funcional 17

A infra-estrutura deve permitir que o Engenheiro de Software visualize os Pacotes de Experimento presentes na infra-estrutura, bem como os seus artefatos e dados registrados.

Requisito Funcional 18

A infra-estrutura deve permitir que o Pesquisador do Estudo Experimental defina se a execução de um estudo é relativa a uma repetição ou não. Se for uma repetição, a infra-estrutura deve copiar e permitir que Pesquisador do altere, caso necessário, os documentos da Definição, Planejamento e Análise, os Instrumentos e Tratamentos de um Pacote de Experimento para repetição deste estudo.

Requisitos de Dados:

Requisito de Dados 1

Os documentos provenientes das etapas de Definição, Planejamento e Análise devem ser registrados sob a forma de arquivos.

Descrição de Dados:

Arquivo

```
{  
    Nome:      String com 100 caracteres alfanuméricos;  
    Dado: Fluxo de dados do arquivo com no máximo 20  
           Mbytes;  
}
```

Documento: Arquivo

```
{  
    Origem:      String com caracteres alfabéticos:  
                {Definição,Planejamento, Análise}  
}
```

Requisito de Dados 2

Os instrumentos a serem utilizados na execução do estudo experimental devem ser registrados sob a forma de arquivos. Segundo (AMARAL,2003), instrumentos são caracterizados pelo seu contexto de uso, seu uso e seu criador. Um instrumento pode fazer parte ou não do material de treinamento.

Descrição de Dados:

Instrumento: Arquivo

```
{  
    contextoDeUso:      String com 1000 caracteres  
                        alfanuméricos;
```

```

    uso: String com 500 caracteres alfanuméricos;
    criador: String com 100 caracteres
            alfanuméricos;
    EMaterialDeTreinamento: booleano;
}

```

Requisito de Dados 3

Segundo a taxonomia de (AMARAL,2003), os artefatos do experimento são instrumentos que serão utilizados na execução do estudo experimental para auxiliar a coleta de dados dos resultados de um participante.

Descrição de Dados:

ArtefatoDoExperimento:Instrumento;

Requisito de Dados 4

Os artefatos de software, segundo (AMARAL,2003), são instrumentos, tais como os documentos, códigos e componentes, utilizados pelos participantes para executarem uma determinada tarefa na execução do estudo experimental. Segundo (AMARAL,2003), artefatos de software possuem um tipo.

Descrição de Dados:

```

ArtefatoDeSoftware:Instrumento
{
    tipo: String com 100 caracteres ASCII: {Documento de
        Requisitos, Documento de Análise, Modelo do
        Domínio, Documentação do Sistema, Modelo de
        Processo de Software, Documentação do usuário,
        Documento de Projeto, Caso de Teste,
        Componente, Código};
}

```

Requisito de Dados 5

Um Formulário Web é um instrumento que deve ser produzido e publicado na Web pelo pesquisador do estudo experimental, e preenchido pelos participantes do estudo via browser. O conteúdo do formulário deve ser um corpo de formulário em HTML.

Descrição de Dados:

FormularioWeb:Instrumento

```
{  
    corpo:      String com 5000 caracteres ASCII;  
}
```

Requisito de Dados 6

Um Formulário de Consentimento é um Formulário Web utilizado para selar o compromisso entre pesquisadores e participantes do estudo experimental.

Descrição de Dados:

FormulárioDeConsentimento:FormularioWeb;

Requisito de Dados 7

Um Formulário de Caracterização é um Formulário Web utilizado para obter informações sobre os participantes do estudo experimental.

Descrição de Dados:

FormulárioDeCaracterizacao:FormularioWeb;

Requisito de Dados 8

Um Tratamento é composto por uma relação de Instrumentos. A relação destes artefatos descreve como será organizada a distribuição do material aos participantes do estudo experimental.

Descrição de Dados:

```
Tratamento
{
    ListaDeInstrumentos: Instrumentos[];
}
```

Requisito de Dados 9

Artefato de Medição é um artefato que contém os dados resultantes da execução de um estudo experimental e que de alguma forma serão medidos para apoiar alguma atividade do Processo de Experimentação. Um Artefato de Medição possui uma lista de propriedades (tuplas dado:valor) relativos aos dados que foram obtidos durante a execução de uma alguma tarefa, tais como, dados da caracterização de um participante, obtenção de consentimento e lista de discrepâncias. Um artefato de Medição pode conter ainda um conteúdo na forma de um Arquivo;

Descrição de Dados:

```
ArtefatoDeMedicao
{
    Propriedades: (dado:valor)[ ];
    Arquivo: Arquivo;
}
```

Requisito de Dados 10

Uma Distribuição descreve como os tratamentos serão atribuídos aos participantes de um estudo experimental numa rodada ou repetição interna do

experimento (JURISTO & MORENO, 2004). Por haver a possibilidade de um estudo ter mais de uma rodada, um experimento pode ter 1 ou mais distribuições. Uma distribuição possui uma lista de tuplas Participante:Tratamento.

Descrição de Dados:

Distribuição

```
{  
    Lista: (Participante:Tratamento)[,];  
}
```

Requisito de Dados 11

Uma Lição Aprendida é uma descrição de uma situação relativa à execução de um estudo experimental que pode de alguma forma auxiliar a execução de estudos futuros. Tais situações servem para descrever soluções para contornar desvios em relação ao Plano do Estudo Experimental com informações que apoiem a execução de alguma atividade. Uma lição aprendida possui um título e um autor e é caracterizada por informações que indicam a descrição do problema, a consequência do problema, a solução adotada e o resultado obtido. Baseado em (MONTONI, 2003).

Descrição de Dados:

LicaoAprendida

```
{  
    Título: String com100 caracteres ASCII;  
    Autor: String com100 caracteres ASCII;  
    DescricaoDoProblema: String com 5000 caracteres ASCII;  
    ConsequenciaDoProblema: String com 1000 caracteres ASCII;  
    SolucaoAdotada: String com 5000 caracteres ASCII;  
    ResultadoObtido: String com100 caracteres ASCII;  
}
```

Restrições:

Restrição 1

A infra-estrutura deve ser disponibilizada como um Webservice (<http://www.oasis-open.org>) e como um software Web para instanciação de workflows de execução de estudos experimentais para o Engenheiro de Software.

Restrição 2

A infra-estrutura deve ser disponibilizada para o pesquisador do estudo experimental como um software Web.

Restrição 3

As informações sobre os artefatos e dados relativos à execução de um estudo experimental devem poder ser todos exportados e visualizados no formato XML.