

Um Estudo sobre a Adaptação de Processos de Software Baseado em Características de Projeto

Rodrigo Rattay Andrade

Orientado por: Prof. Me. Abraham Lincoln Rabelo de Sousa

Centro Universitário La Salle de Ensino Superior, Av. Victor Barreto, 2288, Centro,
Canoas/RS - Cep 92010-000

rodrigo_rattay@gmail.com

Abstract: *This article looks forward on establishing a preliminary overview on the state of the art for software process tailoring, stand out its relevance and presenting a support model for this task. As the final objective, we have the process tailoring in a controlled, repeatable fashion, so that the different existent types of software projects can take advantage of the process models.*

Resumo: *Este artigo visa estabelecer uma visão preliminar sobre o estado da arte relacionado à adaptação de processos, ressaltando sua importância e apresentando um modelo de suporte a esta atividade. Como objetivo final, temos a adaptação de processos de forma repetível, controlada e produtiva, para que os diferentes tipos existentes de projetos de software possam receber o melhor benefício dos modelos de processo.*

1. Introdução

A satisfação do cliente tem se tornado a ambição de muitas organizações buscando a sobrevivência e a prosperidade no mundo competitivo de hoje. Ao mesmo tempo em que estas organizações estão focalizando a experiência do cliente, há uma crescente percepção de que qualidade de software é o ponto fraco no desenvolvimento de produtos de alta qualidade. [1]

Apesar do rápido avanço nas tecnologias de software, temos a impressão de que a complexidade dos problemas resolvidos por softwares está crescendo mais rápido que nossa habilidade de desenvolvê-los e mantê-los. [1] Logo percebemos que a maior dificuldade neste processo realmente está em gerenciar consistentemente a capacidade de desenvolver softwares confiáveis e de acordo com o planejamento estabelecido.

Considerando esta dificuldade, a comunidade de software voltou seus olhares mais ao “como” do que ao “o que”, redirecionando suas pesquisas ao processo de software e não somente ao produto final. Começaram a surgir no mercado então, uma série de modelos de qualidade voltados à engenharia de processos de software, que auxiliam as organizações na tarefa de desenvolver e manter um conjunto de ativos de processos que possam melhorar o desempenho dos projetos e trazer benefícios de longo prazo à organização.

Conforme citado em [2]: “*Não se obtém qualidade do produto de forma espontânea. Ela tem de ser construída. Assim, a qualidade do produto depende fortemente da qualidade de seu processo de desenvolvimento*”.

A grande dificuldade na definição de um processo organizacional está em atender todas as particularidades de cada projeto. Dentro da mesma equipe de desenvolvimento podem-se ter projetos de teste de software, projetos de manutenção de aplicações legadas, configuração de aplicações, etc. Se um projeto de testes for obrigado a seguir um processo desenhado para projetos de desenvolvimento, este estará somente desperdiçando tempo da equipe em uma atividade que não vai trazer qualquer benefício. Da mesma forma, projetos de característica técnica semelhante e tamanhos diferentes podem ter necessidades específicas com relação aos processos.

Ginsberg [4] levanta a seguinte pergunta: “*Como estabelecer um conjunto único de processos para uma organização onde podemos ter uma grande variedade de tipos e ambientes de projetos de software, sem que os processos tornem-se muito complexos ou muito vagos?*”. Mas talvez a pergunta de Ginsberg tenha de ser modificada da seguinte forma: “*Como maximizar os benefícios de um processo organizacional em um contexto de projetos altamente diversificados?*”. Esta segunda forma aparenta ser mais adequada uma vez que não restringe a liberdade da sua própria solução, não definindo que o processo utilizado no projeto seja um conjunto único, quando talvez possa ser mais útil em forma de um subconjunto.

O desperdício de tempo do projeto com processos irrelevantes gera um aumento no custo de desenvolvimento e podem afetar o prazo de entrega do produto, estes dois fatores estão diretamente relacionados à diminuição da qualidade do software. Para resolver este problema, alguns modelos de qualidade definem atividades que direcionam à adaptação dos processos, como o Modelo Integrado de Maturidade e Potencialidade (CMMI, do inglês: *Capability Maturity Model Integration*) [5], porém não apresentam uma técnica para a execução desta atividade.

Para preencher esta lacuna, o objetivo deste estudo é desenvolver um Modelo de Suporte à Adaptação de Processos de Software (**SPT-SM** – do inglês: *Software Process Tailoring Support Model*) que conduza o time de projeto na atividade de desmembrar o processo organizacional, avaliando suas potencialidades e identificando suas limitações, e que com consciência das características únicas do projeto, permita que se obtenha um processo objetivo e bem dimensionado.

2. Metodologia

Este artigo apresenta uma pesquisa que teve um objetivo exploratório, uma vez que já existem estudos avançados relacionados à área, tornando-se oportuna a sugestão do SPT-SM em uma proposta de natureza aplicada. Os procedimentos utilizados serão uma *survey* neste primeiro momento, e futuramente um estudo de caso, onde poderemos aprofundar o estudo e avaliá-lo.

Para possibilitar a criação deste estudo, apresenta-se nas subseções abaixo um protocolo de revisão sistemática (PRS), conforme descrito por Biolchini em [6]. O PRS foi escolhido por conduzir a pesquisa de forma estruturada, facilitando a localização dos artigos mais relevantes e permitindo que a mesma pesquisa possa ser reproduzida ou até mesmo complementada.

2.1. Etapas da Pesquisa

O plano de execução desta pesquisa divide-se em quatro etapas, conforme modelo PDCA (“Planejar, Implementar, Verificar, Implantar Melhorias”, do inglês: “Plan, Do, Check, Act”) [13]. Em primeiro lugar será efetuado o **Planejamento**, baseado em um aprofundamento da revisão bibliográfica, onde com a ajuda do protocolo de revisão sistemática será possível identificar novos estudos e idéias que venham a contribuir com o modelo. Logo após a **Implementação**, pois o levantamento de novas pesquisas pode necessitar que o modelo receba algum ajuste e também a sua formatação final. Na fase de **Verificação**, serão selecionadas as variáveis de controle e criados os diferentes cenários para a aplicação do modelo, permitindo que se faça uma análise de seus resultados. A última fase será a de **Implantação de Melhorias**, resultado das análises efetuadas no modelo.

2.2. Protocolo de Revisão Sistemática

Formulação da Questão

Foco: Encontrar uma maneira padronizada de auxiliar as organizações na tarefa de adaptar o conjunto de processos de software aos diferentes tipos de projetos, buscando manter apenas os processos mais relevantes ao tipo de projeto executado.

Qualidade e amplitude da questão:

a) Problema: Os modelos de qualidade de software (para este estudo vamos utilizar o como base o CMMI), apesar de proverem suporte à adaptação, não propõem técnicas que orientem a atividade de adaptação.

b) Questões:

- Pesquisar as principais características dos projetos de software, que podem influenciar na diferente utilização do processo estabelecido pela organização (p.ex., como é possível caracterizar um projeto de software?);

- Determinar uma forma de avaliação da aderência do processo em relação ao projeto (p.ex., como é possível avaliar a aderência de um processo a determinado tipo de projeto);

- Deduzir uma forma de apontar as práticas de processo mais relevantes para determinados tipos de projeto de software (p.ex., como obter o melhor conjunto de processos para a realidade de um determinado projeto de software?).

c) Palavras-chaves e sinônimos: Qualidade de Software; Adaptação de Processos; *Tailoring*; CMMI e CMM; Processo Definido; Engenharia de Processos de Software.

d) Intervenção: Definição de um processo de software em nível de projeto a partir de um processo de software em nível organizacional.

e) Controle: Leituras.

f) Efeito: Maximização dos benefícios do processo de software dentro da realidade de cada projeto.

- g) Medida de saída: Qualidade dos estudos relacionados.
- h) População: Publicações orientadas à adaptação de processos, CMMI/CMM nível 3, Adaptação.
- i) Aplicação: Gerentes de projetos, projetistas e equipes de desenvolvimento de software.
- j) Projeto experimental: Nenhum método estatístico é utilizado.

Seleção de Fontes

Definição do critério de seleção: Consulta de artigos pela *web*; consulta de livros.

Linguagem de estudo: Inglês e Português.

Identificação de fontes:

- a) Métodos de pesquisa de Fontes: Máquinas de pesquisa pela *web* foram utilizadas para busca (i.e., *Google Scholar*, *Portal ACM*, *Portal IEEE* e *Microsoft Live Academic*).
- b) Argumentos de pesquisa: (“Qualidade de Software” or “Adaptação de Processos” or “*Tailoring*” or “CMM” or “Processo Definido” or “Engenharia de Processos de Software” or “Custo da Qualidade de Software” or “Caracterização de Projetos de Software”).

Checagem das referências: Os estudos mais abrangentes sobre o assunto foram os de Coelho [7] e Borges [10].

Seleção de Estudos

Definição dos estudos

- a) Definição dos critérios de inclusão de estudos: Os artigos devem tratar de processos e projetos de desenvolvimento de software.
- b) Definição dos tipos de estudos: Artigos, publicações e trabalhos acadêmicos relacionados com o objetivo deste trabalho.
- c) Procedimentos para seleção de estudos: O resumo de cada referência será lido inicialmente. Os aprovados serão lidos completamente.

3. Estudos sobre Adaptação de Processos

Realizada uma pesquisa conforme os critérios mencionados no protocolo de revisão sistemática descrito anteriormente, foi possível coletar uma série de referências que resumem o estado da arte relacionado à adaptação de processos. A seguir são apresentados alguns dos trabalhos mais significantes que contribuíram à realização desta pesquisa.

3.1. Estudos Relacionados

Comprovando a motivação desta pesquisa, Johnson [8] descreveu em seu trabalho que processos pré-definidos, e até mesmo padronizados, devem ser adaptados às necessidades locais como, por exemplo, os contextos organizacionais, tamanhos de projeto, requisitos sobre regulamentos, práticas da indústria e questões culturais da corporação. Menciona ainda, que padrões como o IEEE/EIA 12207 contém mecanismos e recomendações para a obtenção dos processos adaptados.

A dissertação de mestrado apresentada por Coelho [7], traz um modelo compatível com o CMM e que auxilia a adaptação de um processo padrão para projetos específicos, além de promover o reuso e melhoria de processos de software. Sua pesquisa

destaca um modelo (chamado MAPS) para adaptação de processos de software a partir de um processo padrão. Logo após, um modelo de caracterização de projetos, que permite comparar projetos de software a partir de suas características. Outro aspecto de sua pesquisa é uma análise das principais características dos projetos de desenvolvimento de software que influenciam o planejamento e o gerenciamento desses projetos. Por fim, apresenta um relacionamento entre os artefatos da disciplina “Planejamento e Gerenciamento” do RUP e as características que influenciam a disciplina, identificando, caso a caso, a necessidade de produzir, ou não, cada artefato.

Duarte [2] contribui criando uma ontologia de qualidade de software, onde faz uma revisão sobre os conceitos que circulam o tema “Qualidade de Software”.

O estudo de caso descrito por Sakamoto [9] relata o esforço de implantação de SPI (do inglês: *Software Process Improvement* – Melhoria de Processo de *Software*) com o objetivo de identificar problemas e desafios da implantação de SPI, e desenvolver um modelo prático passo-a-passo que possa ajudar a resolver este problema. Conclui que os principais problemas encontrados incluem a divergência de objetivos entre os *stakeholders*; falta de acompanhamento do progresso do SPI e subutilização da complexa informação gerada durante o processo de SPI. Sakamoto também sugere uma atenção especial ao plano de implementação de SPI e a clara definição dos papéis envolvidos neste processo.

Borges [10], em sua dissertação de mestrado, seleciona aspectos quantitativos do desenvolvimento de software (como cronograma, defeitos, esforço e tamanho, entre outros) e cria um modelo flexível que se propõe a ajudar na mensuração do processo de desenvolvimento de software. Seu modelo é compatível com o CMMI; possui como pré-requisito a existência de um modelo que permita o cruzamento das fases e iterações com as atividades do processo.

Hartmann [11] definiu uma metodologia de adaptação de processos que se baseia em padrões. Este modelo visa auxiliar na adaptação de metodologias de desenvolvimento considerando padrões e critérios de risco. O modelo depende de forte dedicação do papel do “projetista” do processo, mas com o auxílio da ferramenta PMT-Tool, conduz a seleção das atividades de forma orientada rumo à melhor utilização do conjunto de processos da organização.

3.2. Modelo Proposto

O modelo SPT-SM é composto por três etapas básicas: a caracterização dos tipos de projeto, a avaliação dos processos de software e a seleção dos itens de processo.

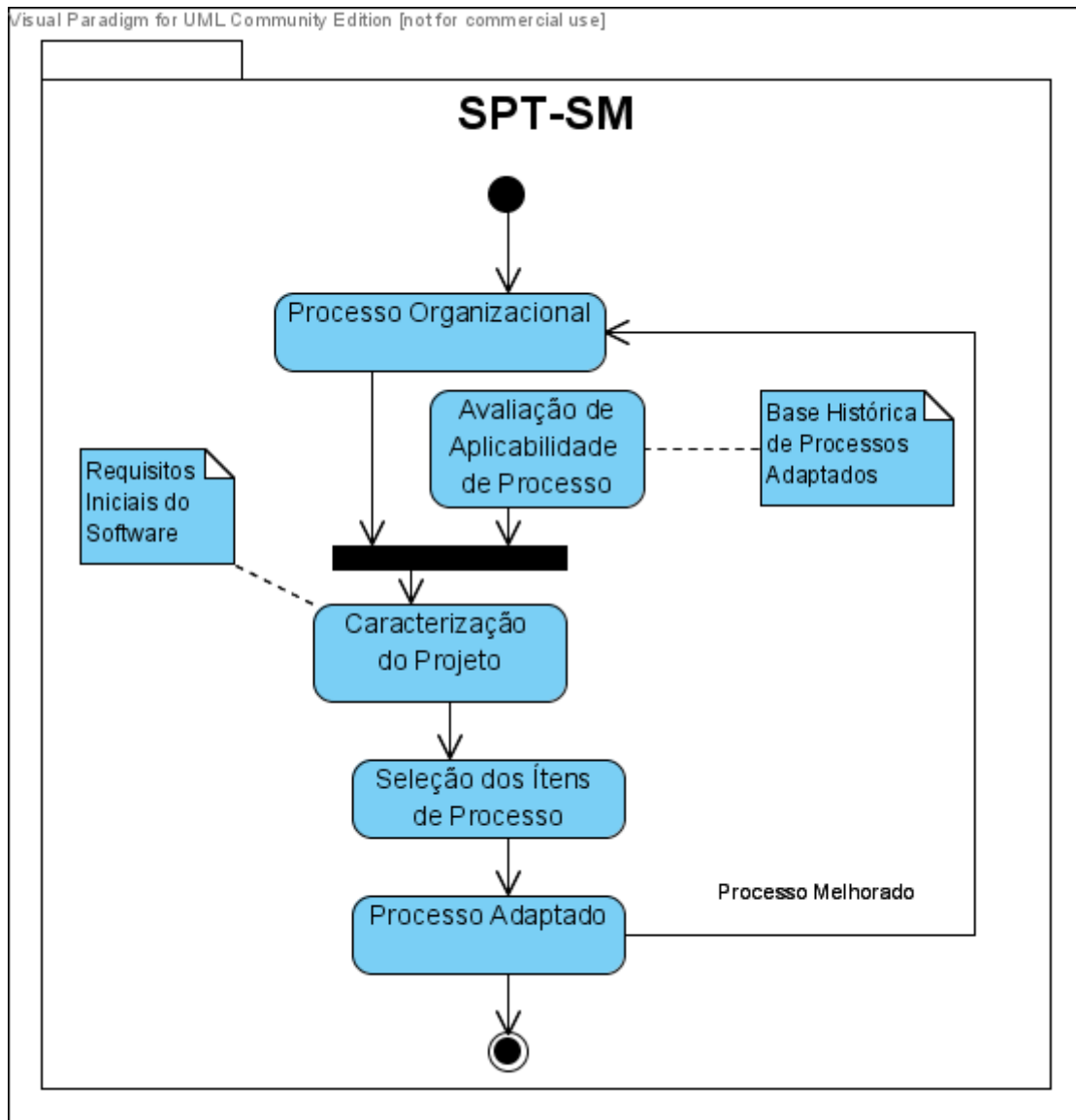


Figura 2. Modelo de Suporte à Adaptação de Processos de Software SPT-SM.

A Caracterização do Projeto, que ocorre como parte da fase inicial de levantamento de requisitos geralmente conduzida pelo gerente de projeto, será responsável por informar as características de mais alto risco dentro do projeto de software.

Outro componente do modelo é a Avaliação de Aplicabilidade dos Processos, onde o grupo de SEPG (Grupo de Engenharia de Processos de Software, do inglês: “*Software Engineering Process Group*”) fará uma análise de cada uma das atividades e tarefas descritas no processo organizacional, com vistas a classificar a relevância desta atividade ou tarefa contra cada aspecto do projeto.

Finalmente o gerente de projetos chega à fase de seleção dos itens de processo que irão conduzir a execução do seu projeto. Nesta etapa, o modelo oferece uma tabela contendo uma proposta de adaptação baseada no histórico de processos adaptados. O gerente tem a possibilidade de aceitar ou não as sugestões do modelo, ou indicar novas opções.

A tabela 1 demonstra as entradas e saídas esperadas em cada etapa e também quem são os atores responsáveis por cada atividade.

Etapa	Atores			Interfaces	
	Gerente de Projetos	Equipe de Desenvolv.	SEPG	Entradas	Saídas
Avaliação de Aplicabilidade de Processo	Não Requer	Não Requer	Responsável	Processo Organizacional; Critérios de Avaliação do Risco do Projeto.	Base de atividades e tarefas adaptáveis sugeridas.
Caracterização do Projeto	Responsável	Provê Suporte	Provê Suporte	Critérios de Avaliação do Risco do Projeto; Levantamento de requisitos iniciais de software.	Enquadramento do projeto de software.
Seleção dos Itens de Processo	Responsável	Provê Suporte	Aprova	Projeto enquadrado de acordo com as características; Base de atividades e tarefas adaptáveis sugeridas.	Processo adaptado final.

Tabela 1. Tabela de Atores e Interfaces do SPT-SM.

3.3. Atividades Futuras

De acordo com os estudos analisados e o modelo SPT-SM proposto neste artigo, acredita-se que seu resultado terá boa aceitação na comunidade de software. Os próximos passos para a finalização deste modelo são:

- Definição dos critérios de avaliação de risco do projeto (p.ex.: “Quais são as características que direcionam o sucesso ou fracasso do projeto?”). Com esta informação será possível identificar as atividades e tarefas do processo organizacional que endereçam cada uma das características e então definir um peso de acordo com a influência do processo de software neste quesito observado;
- Definição do processo de retro alimentação da base histórica de processos adaptados;
- Definição das ferramentas de suporte ao modelo (p.ex.: “Pode-se desenvolver um sistema que faça o gerenciamento de todas as etapas do modelo?”);
- Identificação das variáveis de medição do modelo, para que se possa avaliar sua aplicação;
- Criação de cenários de projetos para a aplicação da avaliação do modelo.

4. Considerações Finais

Executada a primeira etapa deste estudo, nota-se que ainda não existe um consenso na comunidade de qualidade de software sobre como obter de forma mais efetiva os benefícios do processo definido para a organização de forma geral.

Apesar disto, foi possível neste artigo referenciar uma coleção dos trabalhos mais relevantes, e ressaltar as principais idéias que surgiram nos últimos anos. Também se apresentou um modelo, o SPT-SM, que com base nestas idéias demonstra como é possível atingir a questão final deste estudo. No entanto, este modelo ainda carece de maior detalhamento das suas atividades, e também é possível que dando continuidade ao PRS proposto anteriormente, novos estudos possam contribuir para a sua melhoria.

Acredita-se que todo este esforço de pesquisa pode agregar um grande valor ao tema da Qualidade de Software, visto que a adaptação de processos influencia diretamente a produtividade das equipes de projeto, e que o custo e o prazo decorrentes ainda são os fatores mais críticos de sucesso na história do desenvolvimento de software.

5. Referências Bibliográficas

- [1] Paulk, Mark C.; Webber, Charles V.; Curtis, Bill; Chrissis, Mary Beth. **The Capability Maturity Model – Guidelines for Improving the Software Process**. Carnegie Mellon University - Software Engineering Institute, 1995.
- [2] Duarte, Katia C.; Falbo, Ricardo de Almeida. **Uma Ontologia de Qualidade de Software**. Workshop de Qualidade de Software, João Pessoa, 2000.
- [3] Rocha, A.R.C., et al. **Uma Experiência na Definição do Processo de Desenvolvimento e Avaliação de Software segundo as Normas ISO**, Relatório Técnico ES-302/94, COPPE/UFRJ, Junho 1994.
- [4] Ginsberg, Mark P. **Process Tailoring and the Software Capability Maturity Model** Technical Report CMU/SEI-94-TR-024, 1995.
- [5] Chrissis, Mary Beth; Konrad, Mike; Shrum, Sandy. **CMMI – Guidelines for Process Integration and Product Improvement**, SEI Series in Software Engineering, 2003.
- [6] Biolchini, J.; Mian, P. G.; Natali, A.C.; Travassos, G. **Systematic Review in Software Engineering**. Rio de Janeiro, 2005, COPPE/UFRJ, <http://cronos.cos.ufrj.br/publicacoes/reltec/es67905.pdf>, acessado em maio de 2006.
- [7] Coelho, Ciro C. **MAPS: Um Modelo de Adaptação de Processos de Software**. Dissertação de mestrado apresentada ao Centro de Informática da Universidade Federal de Pernambuco. Recife, Abril de 2003.

- [8] Johnson, D.; Brodman, J. **Tailoring the CMM for Small Businesses, Small Organizations, and Small Projects.** *Elements of Software Process Assessment and Improvement*, 1999.
- [9] Sakamoto, Keishi; Nakakoji, Kumiyo; Takagi, Yasunari; Niihara, Naoki. **Toward Computational Support for Software Process Improvement Activities.** *icse*, p. 22, *20th International Conference on Software Engineering (ICSE'98)*, 1998.
- [10] Borges, Eduardo P. **Um modelo de medição para processos de desenvolvimento de software.** Departamento de Ciência da Computação do Instituto de Ciências Exatas da Universidade Federal de Minas Gerais. Belo Horizonte, Março de 2003.
- [11] Hartmann, Júlio; Fontoura, Lisandra M.; Price; Roberto T. **Using Risk Analysis and Patterns to Tailor Software Processes.** Instituto de Informática – Universidade Federal do Rio Grande do Sul (UFRGS); 19º Simpósio Brasileiro de Engenharia de Software, Uberlândia, MG, 2005.
- [12] ISD Brasil. **Seminário Capability Maturity Model.** *Integrated Systems Diagnostics Brasil*, São Paulo, 2002.
- [13] Tague, Nancy R. **The Quality Toolbox, Second Edition.** ASQ Quality Press, 2004.