

# Uma Proposta para Melhoria da Qualidade de Processos de Software com base em MPS.BR

Miguel Augusto Bauermann Brasil<sup>1</sup>, Lisandra Manzoni Fontoura<sup>1</sup>, Luís Alvaro de Lima Silva<sup>1</sup>

<sup>1</sup>Laboratório de Computação Aplicada - LaCA – Universidade Federal de Santa Maria (UFSM)

Av. Roraima nº 1000 - Cidade Universitária - Bairro Camobi - Santa Maria - RS

{miguel.bauermann, lisandramf, silva.luisalvaro}@gmail.com

**Abstract.** *The improvement of quality in software development processes is a central mechanism to increase the competitiveness of organizations. Maturity models such as MPS.BR, CMM and CMMI along with approaches for the adaptation of processes such as SME offer guidelines so that this activity could be developed. This work describes an approach based on the MPS.BR model directed to the enhancement of quality of software processes. The proposed approach focuses on the reuse of method fragments which are retrieved by using expected results from the MPS.BR model. In order to automate this approach, a system to guide users during the improvement process was developed.*

**Resumo.** *A melhoria da qualidade em processos de software é um importante mecanismo para aumentar a competitividade das organizações. Modelos de maturidade como o MPS.BR, CMM e o CMMI aliados a abordagens de adaptação de processos como a SME oferecem orientações para que esta atividade venha a ser desenvolvida. Este trabalho descreve uma abordagem baseada no modelo MPS.BR para a melhoria da qualidade de processos de software. A abordagem proposta é centrada na reutilização de fragmentos de método, os quais são recuperados a partir dos resultados esperados extraídos do modelo MPS.BR. Visando automatizar a abordagem, um sistema para guiar usuários durante o processo de melhoria foi desenvolvido.*

## 1.Introdução

A qualidade de processos de software é um fator decisivo para a qualidade do produto. A implantação de um programa de qualidade inicia-se com a definição e adoção de um processo de software. Um processo de software definido possibilita, além de melhorar a qualidade dos produtos gerados a partir deste, diminuir o retrabalho, aumentar a produtividade da equipe, proporcionar uma maior competitividade da empresa e permitir uma maior precisão nas estimativas, tais como tempo e custo.

Após a definição e implantação dos processos de desenvolvimento e gerência de software, por exemplo, é necessário que tais processos sejam continuamente avaliados e

melhorados para que a empresa mantenha-se competitiva. Vários modelos e normas internacionais, tais como: ISO/IEC 12207 [ISO/IEC12207 2008], ISO/IEC 15504 [ISO/IEC 15504 2003], MPS.BR [SOFTEX 2011] e CMMI [SEI 2010], foram criados com o objetivo de definir diretrizes para auxiliar as organizações na definição e melhoria contínua de processos de software.

A definição de processos que atendam esses modelos de qualidade não é uma tarefa trivial. Um processo de software pode ser definido a partir de “fragmentos de processo”, que podem ser vistos como “blocos para a construção”. Os fragmentos podem ser definidos a partir de modelos de processos como também a partir de boas práticas de desenvolvimento. Neste caso, fragmentos de processo podem ser reusados na elaboração de diferentes modelos de processos. Neste contexto, o problema abordado por este trabalho é facilitar a melhoria de processos de desenvolvimento de software durante as atividades de definição e aumento da qualidade de processos de software.

Este trabalho propõe uma abordagem para melhoria da qualidade em processos de software a partir do reuso de fragmentos de métodos, definidos segundo conceitos do *Situational Method Engineering* (SME), e da associação destes fragmentos com o guia para Melhoria de Processo do Software Brasileiro (MPS.BR). Assim como descrito em [CARVALHO et. al. 2011], o reuso de processos previamente definidos pode vir a disseminar conhecimento e boas práticas entre diferentes projetos de uma organização e consequentemente impactar na qualidade do processo construído ou adaptado.

A abordagem SME foca na construção de processos específicos de acordo com o contexto dos projetos [SELLERS 2010]. Neste trabalho, o guia MPS.BR foi escolhido por ser um modelo ajustado à realidade de desenvolvimento de software brasileiro, elaborado para fomentar a qualidade de pequenas e médias empresas [SOFTEX 2011]. Um sistema, na forma de um *Wizard* de configuração, foi criado com a finalidade de automatizar o processo de melhoria e de guiar o responsável pela definição do processo facilitando assim seu.

O artigo está organizado como segue: a seção 2 descreve conceitos sobre a abordagem SME. A seção 3 aborda o guia MPS.BR para melhoria de processos de software. A seção 4 discute a abordagem proposta para melhoria de processos de software adaptados. A seção 5 apresenta uma ilustração para abordagem para melhoria de processos e a ferramenta construída para automatizar a abordagem proposta. A seção 6 apresenta os trabalhos relacionados que motivaram esta pesquisa. Finalmente, a seção 7 apresenta considerações finais e trabalhos futuros.

## **2. *Situational Method Engineering***

*Situational Method Engineering* (SME) visa à construção de processos de desenvolvimento de software de acordo com características específicas de cada projeto. Identificar as características que descrevem cada situação em um projeto e os fragmentos mais adequados de acordo com estas características são atividades cruciais em SME [SELLERS 2010].

O principal elemento utilizado para construir processos adaptados em SME denomina-se *method fragment*. Este fragmento pode ser gerado a partir de um metamodelo segundo um processo de instanciação. Um fragmento pode ser do tipo *process-fragment* (uma atividade, por exemplo) e *product-fragment* (um artefato, por exemplo), sendo que estes tipos de fragmentos são definidos separadamente. O conceito de *method fragment* permite a descrição de relacionamentos muitos-para-muitos (n...n) entre as partes processo e produto. Essa característica tende a ser mais adequada para

situações reais onde tarefas podem ter muitos artefatos envolvidos, por exemplo. Tal como descrito em [SELLERS e RALYTE 2010], uma associação entre *process fragment* e *product fragments* deve ser usada para capturar dependências apropriadas entre eles, uma vez que cada fragmento de processo, quando instanciado, instancia um ou mais fragmentos de produto para construção do processo adaptado.

Um fragmento de processo pode ser representado conforme o modelo descrito na Tabela 1 (adaptado de [PEREIRA e FONTOURA 2011]).

**Tabela 1. Modelo para definição de fragmentos de processo.**

<b>Fragmento</b>	<i>Nome do fragmento.</i>
<b>Propósito</b>	<i>Descrição dos propósitos do fragmento, as metas do fragmento.</i>
<b>Fonte</b>	<i>Fonte do fragmento (RUP, Scrum e XP, por exemplo).</i>
<b>Disciplina</b>	<i>Disciplina relacionada com projetos de desenvolvimento de software.</i>
<b>Fase(s)</b>	<i>Uma ou mais fases de desenvolvimento praticadas na organização.</i>
<b>Tarefas(s)</b>	<i>Tarefas(s) associadas com o fragmento.</i>
<b>Papéis(s)</b>	<i>Papéis envolvidos na implementação do fragmento.</i>
<b>Técnicas de mapeamento</b>	<i>Técnicas e orientações para executar as tarefas do fragmento.</i>
<b>ações(s)</b>	<i>Relações entre fragmentos de processo e fragmentos de produto.</i>
<b>Guia de adaptação</b>	<b>Informações de adaptação:</b> <i>Descrição dos valores dos fatores propostos para definir o contexto do fragmento, como por exemplo, “riscos associados ao projeto”, “tamanho da equipe”, etc...</i>
<b>Critérios de adaptação:</b> <i>Descrição de um ou mais critérios de adaptação.</i>	

Uma vez identificados e documentados, os fragmentos de processo são armazenados em um repositório chamado de *method base* (base de métodos), para posteriormente serem reutilizados. Neste repositório, fragmentos podem ser selecionados a partir das características do projeto [SELLERS 2010], tais como: criticidade, taxa de mudança, custo do projeto e tamanho da equipe [SOMMERVILE 2011].

Em resumo, a abordagem SME é relevante para a melhoria da qualidade em processos de software, pois: i) propõem que “partes de processos”, denominadas fragmentos em SME, possam ser definidas e armazenadas em uma base de métodos facilitando a criação de novos processos por meio do reuso; ii) possibilita a definição de fragmentos a partir de métodos ágeis ou planejados através do *template* apresentado na Tabela 1; e iii) independente de abordagens tradicionais realizadas de forma *ad-hoc*, a abordagem SME leva em consideração as características do projeto e o processo de desenvolvimento de software adotado pela organização para realizar a adaptação de um processo.

### 3. MPS.BR

O MPS.BR é o resultado de um projeto nacional especialmente desenvolvido para buscar melhorar a qualidade de software produzido por empresas no Brasil. Em geral, a proposta do modelo MPS.BR é servir como um guia para melhoria de processos de software adaptado à realidade brasileira. Assim como descrito em [BOAS e ROCHA 2010], o MPS.BR tem a finalidade de tornar pequenas e médias empresas (foco principal), como também grandes organizações públicas e privadas, mais competitivas na área de desenvolvimento de software.

O modelo MPS.BR é cuidadosamente fundamentado em normas internacionais como a ISO/IEC 15504 [ISO/IEC 15504 2003] e ISO/IEC 12207 [ISO/IEC 12207 2008], bem como o modelo de CMMI para desenvolvimento (CMMI-DEV) [SEI 2010]. Como principal vantagem, o modelo permite que a organização selecione um ou mais processos específicos para aplicar melhorias de qualidade e assim busque de forma gradativa agregar eficiência e eficácia em seus processos. Com isso, o modelo busca ofertar produtos e serviços criados para satisfazer padrões nacionais e internacionais de qualidade [SOFTEX 2011].

De acordo com Boas e Rocha (2010), o modelo MPS.BR é composto por três (3) componentes: i) o modelo de referência para melhoria de processo de software (MR-MPS); ii) o método de avaliação para melhoria do processo de software (MA-MPS); iii) o modelo de negócios para melhoria de processo de software (MN-MPS). Cada componente tem a função de implementar, avaliar e dar sustentação e garantia de que o modelo MPS.BR está sendo empregado de forma coerente com suas definições.

O modelo MR-MPS é formado por vinte e um (21) processos distribuídos em níveis de maturidade. Tais níveis caracterizam os estágios de melhoria de processos, os quais uma organização deve concentrar seus esforços de aprimoramento. O MR-MPS define sete (7) níveis de maturidade [SOFTEX 2011]: i) Nível A (Em otimização); ii) Nível B (Gerenciado quantitativamente); iii) Nível C (Definido); iv) Nível D (Largamente definido); v) Nível E (Parcialmente definido); vi) Nível F (Gerenciado); vii) Nível G (Parcialmente Gerenciado).

Cada nível de maturidade é formado por uma combinação entre processos e capacidades de processos, conforme a estrutura apresentada na Figura 1 (extraída de [SOFTEX 2011]).



**Figura 1. Estrutura do MR-MPS**

Na Figura 1, o processo é descrito segundo um propósito que descreve o objetivo a ser atingido (metas) mediante sua execução. Os resultados esperados estabelecem os objetivos que devem ser satisfeitos com a efetiva implementação do processo. Neste caso, um processo é agrupado de acordo com sua natureza e ordenado conforme o ciclo de vida do software. [SOFTEX 2011].

A capacidade do processo é representada por um conjunto de atributos de processo (AP). Atributos de processo descrevem a habilidade que o processo tem de atingir os objetivos de negócio atuais e futuros. É fundamental ressaltar que tais atributos são descritos na forma de resultados esperados. Estes resultados esperados proporcionam o cumprimento dos AP. A capacidade do processo, tal como descrito em [BOAS e ROCHA 2011], expressa o grau de refinamento com que o processo é executado na organização.

No MPS quanto mais um processo evolui, um maior nível de capacidade para executar o processo é exigido. O alcance de determinado nível de maturidade pode ser obtido quando são atendidos: i) os resultados esperados dos processos; e ii) os resultados esperados dos atributos de processos estabelecidos para o nível em questão e nos níveis anteriores [SOFTEX 2011].

Em resumo, o modelo MPS.BR é relevante para a melhoria da qualidade pois este modelo permite: i) apresentar um custo menos elevado em relação a outros padrões internacionais, assim permitindo a adesão de micro, pequenas e médias empresas; e ii) estar em conformidade com padrões internacionais de melhoria de processos como a ISO 15504, o CMM e o CMMI-DEV, assim permitindo que a organização uma vez qualificada no modelo MPS.BR tente obter outra certificação como a CMMI.

#### **4. Uma proposta para melhoria da qualidade de processos de software com base em MPS.BR**

Neste trabalho, a principal prática explorada é que a adaptação de processos deve ser realizada com base na situação específica de um projeto. Tal situação é indicada a partir de um contexto previamente definido, ora por características do projeto/organização, ora por metas ou resultados a serem alcançados.

Pereira e Fontoura (2011) propõem uma abordagem para prevenção de riscos em processos de software adaptados a partir de conceitos de SME. A abordagem denominada Octopus SME - *Risk Management Approach* (OSRiMA) faz uso de a) características extraídas do processo padrão da organização e de b) um modelo de caracterização de projetos denominado *Octopus Model* [KRUCHTEN 2010]. Neste caso, este modelo é empregado para selecionar fragmentos armazenados em um repositório e assim criar o processo adaptado. Diferente do trabalho apresentado por Pereira e Fontoura (2011), este trabalho não está voltado para a prevenção direta de riscos em um projeto. O objetivo deste trabalho é apresentar uma abordagem para melhorar a qualidade do processo de desenvolvimento construído a partir do uso da abordagem SME e das diretrizes descritas no Guia de melhoria de Processos MPS.BR.

##### **4.1 Definição dos fragmentos de métodos**

Este trabalho utiliza como modelo para representação dos fragmentos o *template* proposto por Pereira e Fontoura (2011) descrito na seção 2 com algumas adequações. Estes autores definiram um metamodelo para representar os conceitos de elementos de processos a partir do metamodelo do *Rational Unified Process* (RUP) [BENCOMO 2005] e da ISO/IEC 24744 [ISO/IEC 2007]. Optou-se neste trabalho por utilizar essa estrutura porque a base de métodos existentes já possui um conjunto de fragmentos definidos, sendo necessário associá-los ao modelo de melhoria de processos MPS.BR, ao invés de se ter que definir todos os fragmentos novamente.

Um fragmento de método é descrito por meio de um conjunto de atributos brevemente descritos a seguir. O atributo “Fragmento” descreve o nome do fragmento. No caso do fragmento ter sido definido a partir do RUP, o nome da atividade do RUP é usado como nome do fragmento. O atributo “Fase” apresenta a fase do ciclo de vida do processo no qual as tarefas descritas pelo fragmento são executadas. O atributo “Disciplina” descreve a qual área de concentração a atividade está relacionada. O objetivo do fragmento é dado pelo item “Propósito”. No atributo “Tarefas” são descritas as tarefas que precisam ser realizadas para que o propósito do fragmento seja atingido.

Os novos atributos propostos por este trabalho são: nível de maturidade mínimo, processo, resultados esperados e atributos de processo. O atributo “Nível de Maturidade Mínimo” indica o nível de maturidade mínimo da organização. Por fim, os atributos “Processo” e “Resultados esperados” provenientes do Guia MPS.BR são responsáveis por definir a “área de processo” em que serão aplicadas as melhorias, e os resultados esperados a partir da efetiva implementação das tarefas mapeadas no fragmento. O atributo denominado “Atributos de processo” representa a capacidade do processo em relação à área de processo e ao nível de maturidade mínimo da organização, ambos representados no fragmento pelas propriedades “Processo” e “Nível de Maturidade Mínimo”. Neste trabalho, optou-se por chamar de “áreas de processo” os processos extraídos do modelo MPS.BR que deverão receber melhorias.

Na Figura 2 é exibido um exemplo de fragmento de processo descrito usando esses atributos. O fragmento exibido na Figura 2 descreve a atividade do RUP “Analisar o problema” que foi mapeada e associada segundo o resultado esperado do modelo MPS.BR.

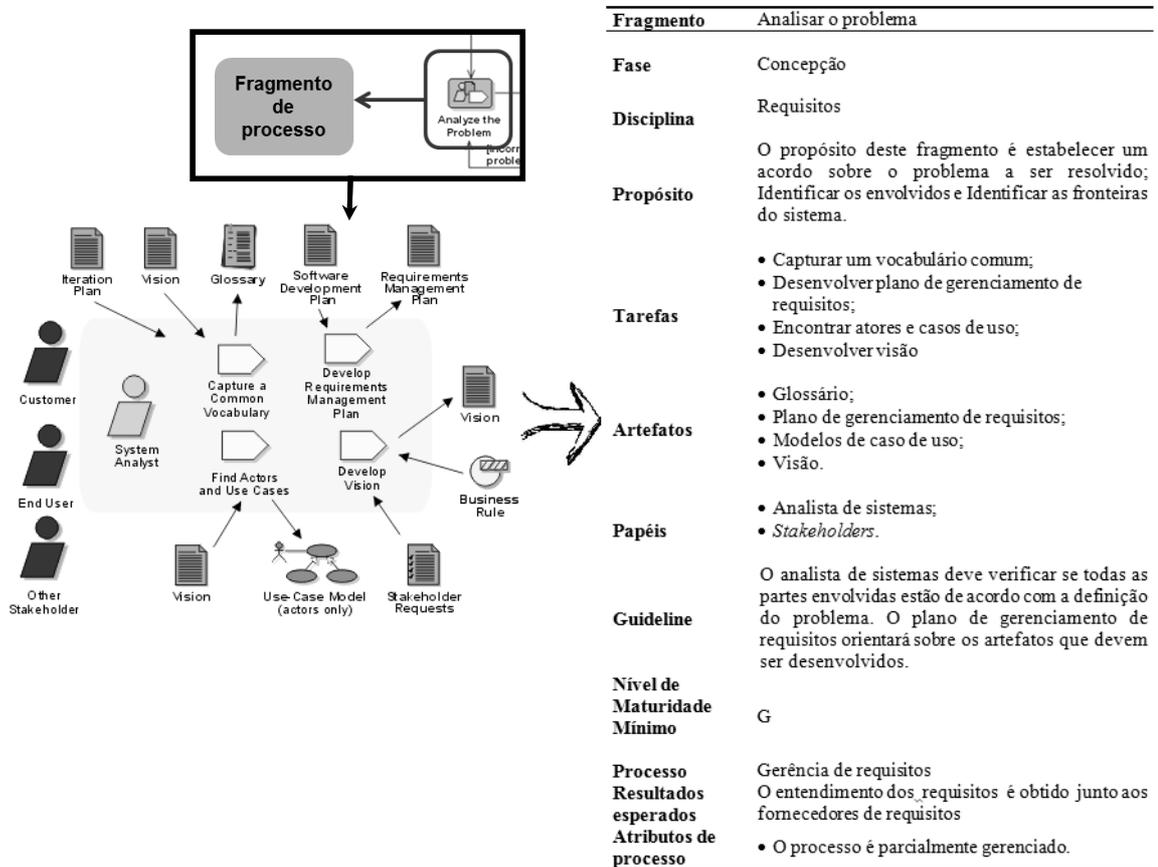


Figura 2. Fragmento “Analisar o problema”

Na abordagem proposta, é importante notar que os fragmentos de métodos são associados aos resultados esperados do Guia MPS.BR. Tal associação indica que o fragmento de método pode, através das tarefas contidas nele, satisfazer um ou mais resultados esperados associados ao modelo. A construção dos fragmentos de método é baseada nos seguintes passos:

- Para cada resultado esperado no Guia MPS.BR, fragmentos da base de métodos de Pereira e Fontoura (2011) foram analisados para determinar se os fragmentos

descrevem elementos (atividades, tarefas, papéis, artefatos) que implementam o resultado;

- Quando o resultado é suportado pelos fragmentos, os elementos descritos pelo fragmento (fase, disciplina, atividades, tarefas, papéis e artefatos) são associados ao resultado;
- Quando o resultado não é satisfeito pelos fragmentos provenientes da base de métodos proposta por Pereira e Fontoura (2011), buscaram-se padrões de processo e padrões organizacionais, bem como a literatura referente à engenharia de software para definição e criação de novos fragmentos.

Ressalta-se que os fragmentos criados e as associações realizadas para elaboração deste trabalho são, na verdade, sugestões, as quais podem ser alteradas de acordo com as experiências da organização.

#### 4.2 Recuperação dos fragmentos na base de métodos

Na abordagem proposta, os fragmentos recuperados da base de métodos podem ser usados para definir novos processos ou adaptar processos já existentes na organização. A criação de novos processos ou a adaptação dos processos existentes será realizada a partir das características esperadas para o projeto, definidas a partir dos resultados esperados do modelo MPS.BR.

O processo de recuperação ocorre a partir da escolha das áreas de processo que serão usadas para melhorar o processo. Em seguida, resultados esperados (metas de melhoria de processo) são escolhidos e fragmentos oriundos da base de métodos são recuperados. A Figura 3 apresenta os passos para recuperação dos fragmentos na base de métodos.

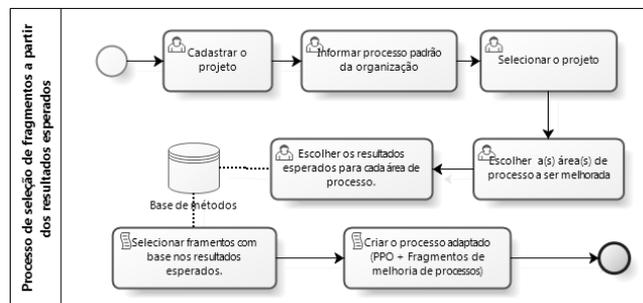


Figura 3. Passos para recuperação dos fragmentos

Na Figura 3, ações descritas pela imagem de uma “pessoa” representam as atividades que o responsável por definir o processo da organização, neste artigo denominado engenheiro de processos, deverá realizar para construir o processo adaptado. As atividades com símbolo de script representam algoritmos de execução que formarão o supracitado processo. A Tabela 2 apresenta um exemplo da recuperação de fragmentos seguindo o fluxo descrito. Desta forma, caso o engenheiro de processos venha a selecionar a área de “Gerência de requisitos” e logo em seguida o resultado esperado “GRE02 - Os requisitos são avaliados com base em critérios objetivos e um comprometimento da equipe técnica com estes requisitos é obtido.”, o algoritmo de recuperação selecionará da base de métodos o fragmento de processo “Definir o sistema” indicando que as tarefas, artefatos e papéis incluídos neste fragmento satisfazem o resultado esperado GRE02 extraído do Guia MPS.BR.

**Tabela 2. Exemplo de fragmento recuperado**

<b>Fragmento</b>	<b>Definir o sistema</b>
<b>Fase</b>	Concepção
<b>Disciplina</b>	Requisitos
<b>Propósito</b>	O propósito deste fragmento é criar uma compreensão comum do sistema dentro da equipe do projeto. Uma análise de alto nível deverá ser realizada e a visão deverá ser refinada para que contenha as características a serem incluídas no sistema.
<b>Tarefas</b>	<ul style="list-style-type: none"> <li>• Desenvolver visão;</li> <li>• Gerenciar dependências;</li> <li>• Capturar vocabulário comum;</li> <li>• Localizar atores e casos de uso.</li> </ul>
<b>Artefatos</b>	<ul style="list-style-type: none"> <li>• Visão (refinada);</li> <li>• Especificações suplementares;</li> <li>• Atributos de requisitos refinados;</li> <li>• Modelo de casos de uso refinados;</li> <li>• Glossário.</li> </ul>
<b>Papeis</b>	<ul style="list-style-type: none"> <li>• Analista de sistemas;</li> </ul>
<b>Guideline</b>	Na Definição do Sistema o Analista de sistemas deverá se concentrar em identificar atores e casos de uso inteiramente, e expandir os requisitos não funcionais globais, conforme definidos nas Especificações Suplementares.
<b>Nível de Maturidade Mínimo</b>	G
<b>Processo</b>	Gerência de requisitos
<b>Resultados esperados</b>	Os requisitos são avaliados com base em critérios objetivos e um comprometimento da equipe técnica com estes requisitos é obtido.
<b>Atributos de processo</b>	<ul style="list-style-type: none"> <li>• O processo é parcialmente gerenciado.</li> </ul>

Como na base de métodos existem muitos fragmentos cadastrados com objetivos semelhantes, cabe ao engenheiro de processos selecionar os fragmentos que melhor satisfazem os resultados esperados para o projeto. Esta seleção é realizada de forma manual por um profissional e em trabalhos futuros deverá ser automatizada, permitindo assim a seleção otimizada dos “melhores fragmentos”.

## **5. Ilustração da abordagem para melhoria de processos**

A abordagem para melhoria da qualidade de processos de software com base em MPS.BR pode ser demonstrada por um estudo de caso em uma empresa. A referida empresa utiliza como processo de software padrão da organização o RUP.

### **5.1. Descrição do Contexto**

Os sistemas produzidos pela empresa são desenvolvidos localmente por uma pequena equipe considerada de média habilidade. Um alto grau de esforço é dispensado pela equipe na correção de erros, refatoração de código e nas tentativas de adaptar o processo de software padrão da organização para os diferentes projetos executados impactando de forma direta no tempo e no cronograma destes projetos.

Fatores adicionais que afetam diretamente o processo de desenvolvimento da empresa são: i) a falta de documentação das não-conformidades; ii) a falta de *feedback* da equipe para com a gerência e vice-versa; e iii) a falta de avaliação dos processos e produtos construídos.

O processo de software padrão da empresa é baseado em um pequeno conjunto de atividades do RUP, como mostra a Tabela 3.

**Tabela 3. Processo de software padrão da empresa**

<b>Disciplina</b>	<b>Atividades</b>
Requisitos	Entender as necessidades dos envolvidos
Análise e Design	Definir uma arquitetura candidata
Implementação	Implementar components
Teste	Verificar abordagem de teste Testar e Avaliar
Implantação	Desenvolver material de suporte Produzir unidade de implantação
Gerência de mudanças e configuração	Gerenciar <i>baselines</i> e <i>releases</i>
Gerenciamento de projetos	Planejar a próxima interação Iniciar projeto

## 5.2 Um sistema para melhoria de processos

Um sistema foi construído para automatizar a abordagem proposta. Este sistema tem o objetivo de guiar o usuário no processo de adaptação com base nos processos e resultados esperados oriundos do modelo MPS.BR e assim buscar melhorias no processo.

O sistema foi desenvolvido utilizando-se a linguagem Java para especificação das regras de negócio e acesso a dados; Java Server Faces (JSF), HTML, CSS3 e JQuery para compor a camada de apresentação e banco de dados MySQL para armazenar a base de métodos, esta adaptada de Pereira (2011).

Dessa forma, de posse do processo padrão da empresa e com objetivo de ilustrar a abordagem para melhoria da qualidade em processos de software apresentada na Seção 4, deseja-se aplicar melhorias de processo na área de “Gerência de Projetos” da organização. Neste caso, o objetivo é construir um processo adaptado aplicando melhorias significativas extraídas do guia MPS.BR. Para tal, os seguintes resultados esperados oriundos do modelo MPS.BR foram selecionados na ferramenta:

- GPR01 - O escopo do trabalho para o projeto é definido;
- GPR02 - As tarefas e os produtos de trabalho do projeto são dimensionados utilizando métodos apropriados;
- GPR06 - Os riscos do projeto são identificados e o seu impacto, probabilidade de ocorrência e prioridade de tratamento são determinados e documentados.

A Figura 4 apresenta os passos para melhoria do processo na forma de um *Wizard* de configuração. Neste conjunto de passos, o engenheiro de processos pode: i) selecionar um projeto já cadastrado na base de dados (Passo 01); ii) elencar processos, na forma de área de processos, para melhoria (Passo 02); iii) selecionar os resultados esperados para o projeto (Passo 03); e iv) gerar o processo adaptado com base no Guia MPS.BR (Passo 04).

Passo 01. Engenheiro de processos seleciona projeto para vincular o processo adaptado.

Universidade Federal de Santa Maria  
Programa de Pós Graduação em Informática - LaCA

Esta ferramenta apresenta quatro passos para adaptar o processo de sua organização.

**Cadastrar**  
Organização  
Projeto  
Gerente de processos

**Operações**  
Adaptar processo

**Passo 01**  
Escolha um projeto para adaptar seu processo.

Projeto TE001  
 Projeto TE002  
 Projeto TE003

**Passo 02**  
Escolha que áreas deseja aplicar melhorias de processo.

**Nível G**  
 Gerência de Requisitos (GRE)  
 Gerência de Projetos (GPR)

**Nível F**  
 Medição (MED)  
 Garantia da qualidade (GQA)  
 Gerência de portfólio de Projetos (GPP)  
 Gerência de configuração (GCO)  
 Aquisição (AQU)

**Nível E**  
 Gerência de projetos (GPR-Evolução)  
 Gerência de Reutilização (GRU)  
 Gerência de Recursos Humanos (GRH)  
 Definição do processo organizacional (DFP)  
 Avaliação e Melhoria do Processo Organizacional (AMP)

**Nível D**  
 Verificação (VER)  
 Validação (VAL)  
 Projeto e construção do produto (PCP)  
 Integração do produto (ITP)  
 Desenvolvimento de requisitos (DRE)

**Nível C**  
 Gerência de Riscos (GRI)  
 Desenvolvimento para reutilização (DRU)  
 Gerência de decisões (GDE)

**Nível B**  
 Gerência de Projetos (GPR-Evolução)

**Nível A**  
 Este nível não possui processos resultados esperados.

Passo 02. É realizada a escolha das áreas onde se deseja aplicar melhorias.

Passo 03. Resultados esperados de acordo com a área de processo são selecionados para o projeto.

Passo 04. Fragmentos de processo são recuperados e posteriormente priorizados pelo engenheiro de processos.

**Passo 03**  
Escolha os resultados esperados para o processo.

Gerência de Projetos (GPR)

O escopo do trabalho para o projeto é definido.

As tarefas e os produtos de trabalho do projeto são dimensionados utilizando métodos apropriados.

O modelo e as fases do ciclo de vida do projeto são definidos

O esforço e o custo para a execução das tarefas e dos produtos de trabalho são estimados com base em dados históricos ou referências técnicas (até nível F).

O planejamento e as estimativas das tarefas do projeto são feitos baseados no repositório de estimativas e no conjunto de ativos de processo organizacional (a partir do nível E).

O orçamento e o cronograma do projeto, incluindo a definição de marcos e pontos de controle, são estabelecidos e mantidos.

Os riscos do projeto são identificados e o seu impacto, probabilidade de ocorrência e prioridade de tratamento são determinados e documentados.

Os recursos humanos para o projeto são planejados considerando o perfil e o conhecimento necessários para executá-lo.

**Passo 04**  
Clique sobre o link "Adaptar processo" para adaptar seu processo.

Adaptar processo do projeto  
Visualizar processo adaptado

Figura 4. Sistema que implementa a abordagem proposta

Com base no contexto da empresa e nos passos de melhoria automatizados pelo sistema, três (3) fragmentos foram recuperados da base de métodos: “Gerenciar escopo do sistema” (Figura 5), “Elaborar plano de desenvolvimento de software” (Figura 6) e “Avaliar escopo e risco do projeto” (Figura 7). É importante notar que as tarefas, artefatos e papéis destes fragmentos satisfazem os resultados esperados para o projeto.

Fragmento	Gerenciar escopo do sistema
Fase	Concepção
Disciplina	Requisitos
Propósito	O propósito deste fragmento é priorizar e refinar as informações fornecidas para selecionar as características e os requisitos que serão incluídos na iteração atual.
Tarefas	<ul style="list-style-type: none"> <li>• Priorizar casos de uso;</li> <li>• Desenvolver visão;</li> <li>• Gerenciar dependências;</li> <li>• Criar WBS;</li> </ul>
Artefatos	<ul style="list-style-type: none"> <li>• Documento de arquitetura de software;</li> <li>• Atributos de requisito (refinados);</li> <li>• Visão (refinada);</li> <li>• WBS.</li> </ul>
Papéis	<ul style="list-style-type: none"> <li>• Arquiteto de software;</li> <li>• Analista de sistemas.</li> </ul>
Guideline	O escopo de um projeto é definido pelo conjunto de requisitos alocados para ele.
Nível de Maturidade Mínimo	G
Processo	Gerência de projetos
Resultados esperados	O escopo do trabalho para o projeto é definido.
Atributos de processo	<ul style="list-style-type: none"> <li>• O processo é executado;</li> <li>• O processo é gerenciado.</li> </ul>

**Figura 5. Fragmento “Gerenciar escopo do sistema”**

Fragmento	Elaborar Plano de Desenvolvimento de Software
Fase	Concepção
Disciplina	Gerenciamento de projeto
Propósito	O propósito deste fragmento é estimar o escopo, o esforço e o custo totais do projeto.
Tarefas	<ul style="list-style-type: none"> <li>• Desenvolver plano de métricas;</li> <li>• Desenvolver plano de gerenciamento de riscos;</li> <li>• Desenvolver plano de garantia da qualidade;</li> <li>• Definir a equipe e a organização do projeto;</li> <li>• Planejar fases e interações;</li> <li>• Compilar plano de desenvolvimento de software;</li> <li>• Revisar o planejamento do projeto.</li> </ul>
Artefatos	<ul style="list-style-type: none"> <li>• Plano de métricas;</li> <li>• Plano de gerenciamento de riscos;</li> <li>• Plano de garantia da qualidade;</li> <li>• Plano de desenvolvimento de software.</li> </ul>
Papéis	<ul style="list-style-type: none"> <li>• Gerente de projeto;</li> <li>• Revisor do projeto.</li> </ul>
Guideline	A estimativa deve ser baseada idealmente na própria experiência da organização. Se o Gerente de Projeto estiver começando do zero, usando os valores padrão dos coeficientes de modelo, será importante usar outros métodos para validar as estimativas. Também é importante obter a equipe e outro acordo dos envolvidos para que as estimativas sejam realistas e alcançáveis.
Nível de Maturidade Mínimo	G
Processo	Gerência de projetos
Resultados esperados	As tarefas e os produtos de trabalho do projeto são dimensionados utilizando métodos apropriados.
Atributos de processo	<ul style="list-style-type: none"> <li>• O processo é executado;</li> <li>• O processo é gerenciado.</li> </ul>

**Figura 6. Fragmento “Elaborar plano de desenvolvimento de software”**

Fragmento	Avaliar Escopo e Risco do Projeto
Fase	Concepção
Disciplina	Gerenciamento de projeto
Propósito	O propósito deste fragmento é reavaliar as capacidades pretendidas e as características do projeto, e os riscos associados a alcançá-las.
Tarefas	<ul style="list-style-type: none"> <li>• Identificar e avaliar riscos;</li> <li>• Desenvolver casos de negócio.</li> </ul>
Artefatos	<ul style="list-style-type: none"> <li>• Lista de riscos;</li> <li>• Casos de negócio.</li> </ul>
Papéis	<ul style="list-style-type: none"> <li>• Gerente de projeto.</li> </ul>
Guideline	A Lista de Riscos e os Casos de Negócio devem ser submetidos a inspeções técnicas internas e revisões para garantir que haja um consenso geral, antes que a próxima rodada de planejamento detalhado comece.
Nível de Maturidade Mínimo	G
Processo	Gerência de projeto
Resultados esperados	Os riscos do projeto são identificados e o seu impacto, probabilidade de ocorrência e prioridade de tratamento são determinados e documentados.
Atributos de processo	<ul style="list-style-type: none"> <li>• O processo é executado;</li> <li>• O processo é gerenciado.</li> </ul>

**Figura 7. Fragmento “Avaliar escopo e risco do projeto”**

O fragmento “Gerenciar escopo do sistema” (Figura 5) é responsável por priorizar e refinar as informações fornecidas para selecionar as características e os requisitos que serão incluídos na iteração atual do processo. Proveniente da disciplina de requisitos satisfaz o resultado esperado GPR01. Já o fragmento “Elaborar plano de desenvolvimento de software” (Figura 6) tem o objetivo de estimar o escopo, o esforço e o custo total do projeto decompondo o projeto em partes menores fáceis de serem gerenciáveis e dimensionadas. E finalmente o fragmento “Avaliar escopo e risco do sistema” (Figura 7) tem a função de reavaliar as capacidades pretendidas de acordo com as características do projeto, e os riscos associados para alcança-las. Proveniente da disciplina de gerenciamento de projeto satisfaz o resultado esperado GPR06 que se preocupa em analisar, identificar, priorizar e registrar os riscos associados ao projeto.

A partir dos fragmentos recuperados da base de métodos e após a seleção dos fragmentos que serão reutilizados pelo engenheiro de processos, os fragmentos foram inseridos no processo padrão da organização gerando assim o processo adaptado. A Tabela 4 apresenta o processo de software adaptado conforme os resultados esperados para o projeto.

**Tabela 4. Processo de software adaptado com base nos resultados esperados para o projeto**

<b>Disciplina</b>	<b>Atividades</b>
<b>Requisitos</b>	Entender as necessidades dos envolvidos Gerenciar escopo do sistema
<b>Análise e Design</b>	Definir uma arquitetura
<b>Implementação</b>	Implementar componentes
<b>Teste</b>	Definir abordagem de teste Testar e Avaliar
<b>Implantação</b>	Desenvolver material de suporte Gerar unidade de implantação
<b>Gerência de mudanças e configuração</b>	Gerenciar baselines e releases
<b>Gerenciamento de projetos</b>	Planejar a próxima iteração Iniciar projeto Avaliar escopo e riscos do projeto Elaborar plano de desenvolvimento de software

## 6. Trabalhos relacionados

A adaptação de processos e a adoção de normas ou guias que auxiliam a melhoria nos padrões de qualidade de software são temas que vêm sendo discutidos isoladamente em muitos trabalhos. Tais trabalhos visam aumentar a produtividade, diminuir riscos ou até mesmo inserir qualidade nos produtos desenvolvidos.

Santana et. al. (2006) utilizam a metodologia *Extreme Programming* (XP) integrada ao modelo para melhoria de qualidade MPS.BR. Este trabalho demonstra que a combinação de uma abordagem de desenvolvimento ágil com o respectivo guia de qualidade é uma solução para diversos ambientes de desenvolvimento de software que adotam metodologias ágeis. Porém, o trabalho está limitado a utilizar a metodologia XP como modelo de processo de software padrão da organização não adaptando o processo a situação do projeto, nem utilizando outras abordagens de desenvolvimento como SCRUM ou RUP.

Hongying e Cheng (2011) apresentam um modelo para melhoria da qualidade em processos de software para auxiliar equipes de pequena e média habilidade a desenvolver sistemas de qualidade a partir de metamodelos ágeis. Entretanto, a abordagem denominada *Agile Quality Assurance Model* (AQAM) torna-se limitada por

somente utilizar o modelo de desenvolvimento ágil. Esse fato dificulta que organizações que adotam modelos de processo controlados, como o RUP, utilizem este modelo com eficácia.

A Tabela 4 mostra um quadro comparativo de trabalhos relacionados em relação a abordagem para melhoria de processos apresentada neste artigo.

**Tabela 5. Quadro comparativo de trabalhos relacionados**

Características	[Santana et al. 2006]	[Hongying et al. 2011]	[Barreto et al 2009]	Abordagem proposta
Aceita metodologias ágeis ou planejadas	ÁGEIS	ÁGEIS	ÁGEIS E PLANEJADAS (processo padrão e processos instanciados nos projetos)	ÁGEIS E PLANEJADAS (apresentada neste artigo)
Adapta processos às características do projeto	Não. Os autores preocupam-se apenas em inserir o modelo MPS.BR em organizações que utilizam o XP como processo de software padrão de maneira <i>ad-hoc</i> .	Não. Os autores preocupam-se apenas em propor um guia para melhoria do processo ágil, desconsiderando qualquer característica do projeto que leve a adaptação do mesmo.	Sim. Os autores utilizam conceitos de reuso de processos, caracterização de componentes de processo e de linhas de processo como forma de ordenar os componentes candidatos selecionados para compor a adaptação do processo legado.	Sim. Independente de outros trabalhos a abordagem proposta busca adaptar o processo padrão da organização levando em consideração as características do projeto e o processo de software padrão da organização.
Visam a melhoria do processo de software	Sim. A preocupação pela melhoria do processo fica evidente no artigo uma vez que os autores fazem uso do Guia MPS.BR.	Sim. Os autores apresentam a abordagem denominada AQAM.	Sim. Através de componentes, arquiteturas, linhas de produtos e padrões buscam a melhoria de processos de software.	Sim. Preocupa-se em melhorar a qualidade do processo e produto seguindo metas e práticas de qualidade apoiadas pelo modelo MPS.BR.
Podem ser implementadas em micro, pequenas e médias empresas estando em conformidade com a realidade brasileira	Sim. A escolha pelo modelo MPS.BR possibilita o uso em pequenas e médias empresas estando em conformidade com a indústria de software no Brasil. Entretanto, o uso fica restrito a organizações que utilizam o XP.	Não. O modelo de melhoria de processos foca exclusivamente em times ágeis e não levou em consideração as características brasileiras para desenvolvimento de software.	Não. O estudo de caso apresentado pelos autores demonstra que a abordagem foca mais em definir e avaliar componentes para posterior reutilização.	Sim. A meta principal a ser atingida nesta pesquisa é a busca da melhoria de processos criando um processo adaptado que esteja em conformidade com o modelo MPS.BR e consequentemente com a realidade brasileira.

Barreto et. al. (2009) apresentam uma abordagem para guiar a criação de elementos reutilizáveis de processo de software de uma organização a partir da adaptação de processos e dos ativos de processo de software existentes. Tais elementos, denominados componentes, são associados a características de processo, armazenados em um repositório denominado biblioteca de componentes da organização e posteriormente recuperados para formar o processo com base em reuso. Com a criação de componentes extraídos de processos, sua caracterização e posterior ordenação através de linha de processos, os autores demonstram que estratégias baseadas em reuso de componentes podem contribuir para diminuir o esforço necessário para definir processos na organização e conseqüentemente melhorar o processo formado.

## 7. Considerações finais

A adoção de modelos que proporcionam a melhoria contínua dos processos de software juntamente com a adoção de métodos ágeis ou planejados vêm sendo fonte de pesquisa na comunidade acadêmica [SANTANA 2006][SELLERS 2010][BOAS e ROCHA 2010][CARVALHO 2011]. Entretanto, tais metodologias são executadas de maneira *ad-hoc* sem considerar as características particulares dos projetos, apenas a experiência dos engenheiros de processo ou dos responsáveis por definir o processo em uma organização.

Este trabalho apresentou uma abordagem para melhoria da qualidade de processos de software com base no modelo MPS.BR. Conceitos do guia para melhoria de processos foram herdados gerando uma abordagem que visa auxiliar pequenas, médias e grandes empresas a buscar a melhoria contínua em seus processos e conseqüentemente em seus produtos.

Como principais contribuições temos: i) a integração da abordagem SME e do guia para melhoria de processos MPS.BR; ii) a evolução dos fragmentos oriundos da base de métodos de Pereira e Fontoura (2011) para atender os resultados esperados do guia MPS.BR e construir assim o processo adaptado; iii) o desenvolvimento de um sistema na forma de *wizard* de configuração para apoiar o responsável por definir o processo nos passos de melhoria do processo da organização; e iv) a constatação que o uso de fragmentos contendo resultados esperados, áreas de processo e atributos de processo pode ser um importante mecanismo na construção, aumento da qualidade e adaptação de processos de software.

Trabalhos futuros incluem: i) priorizar os fragmentos de forma automatizada a fim de fornecer ao engenheiro de processos ou ao responsável por definir o processo, mecanismos de avaliar qual o melhor fragmento a ser instanciado para a criação do processo adaptado; ii) realizar testes de aceitabilidade por parte dos futuros usuários da metodologia; e iii) integrar fluxos aos fragmentos do processo a fim de proporcionar uma visão geral do processo adaptado na forma de diagramas BPMN. Além disso, pretende-se integrar os resultados da abordagem proposta com resultados que podem ser obtidos pelo emprego de outras ferramentas de modelagem de processos.

## Referências

Barreto, A., Murta, L., Rocha, A. R. .“Componentizando Processos de Software Legados visando a Reutilização de Processos”. In: VIII Simpósio Brasileiro de Qualidade de Software, SBQS’09, Ouro Preto, Brasil.

- Bencomo, A. "Extending the RUP, Part 1: Process modeling". In: IBM, 2005. Disponível em: <<http://www-128.ibm.com/developerworks/rational/library/05/r-3320/>>. Acesso em: Fev 2013.
- Boas, Gisele V.; Rocha, Ana R. C. da. "An approach to implement software process improvement in small and mid-sized organizations". In: Seventh International Conference on the Quality of Information and Communications Technology. 2010.
- Carvalho, D. D.; Costa, A.; Sales, E. O.; Lima, A.; Reis, R. Q. "Apoio à Reutilização de Processos de Software em um Ambiente de Engenharia de Software Centrado em Processo". In: X Simpósio Brasileiro de Qualidade de Software - SBQS, 2011, Curitiba, Paraná. Junho, 2011.
- Hongying, G. ; Cheng, Y. "A Customizable Agile Software Quality Assurance Model". In: College of Computer Science and Technology.IEEE Computer Society, Washington, DC, USA.2011.
- ISO/IEC 15504. "Information Technology. Process Assessment. Part 1: Concepts and Vocabulary", 2003.
- ISO/IEC 24744. "Software Engineering – metamodel for development methodologies", 2007.
- ISO/IEC12207, "Systems and software engineering —Software life cycle processes," 2008.
- Kruchten, P.: "Contextualizing Agile Software Development". In: Proceedings of the EuroS-PI 2010 Conference, pp. 1-12, Grenoble (2010).
- Pereira, G. and Fontoura L., 2011. "An Approach for Risk Prevention Based on Situational Method Engineering". In: Proceedings of the Applied Computing, Rio de Janeiro, Brazil. pp. 491-495.
- Rational Software Corporation. 2003. Rational Unified Process: Version 2003.06.12. Cupertino.
- Santana, Célio A.; Timóteo, Aline L.; Vasconcelos, Alexandre M. L. "Mapeamento do modelo de Melhoria do Processo de Software Brasileiro (MPS.Br) para empresas que utilizam Extreme Programming (XP) como metodologia de desenvolvimento". In: V Simpósio Brasileiro de Qualidade de Software – SBQS '2006.2006.
- SEI . "CMMI-DEV 1.3 comparisons". <http://www.sei.cmu.edu/cmmi/solutions/dev/comparisons.cfm>. Acesso em : Outubro de 2011.
- Sellers, Brian H. and Ralyté, J. "Situational Method Engineering: State-of-the-Art Review". In: Journal of Universal Computer Science, vol. 16, no. 3 (2010), 424-478.
- Sellers, Brian H.; Perez, Cesar G.; Ralyté, Jolita. "Comparison of Method Chunks and Method Fragments for Situational Method Engineering". In: 19th Australian Conference on Software Engineering (aswec 2008). 2008.
- SOFTEX, "MPS.BR – Melhoria de Processo do Software Brasileiro, Guia Geral: 2011," Julho de 2011.
- Sommerville, Ian. Engenharia de Software – 9ª Edição. Editora Pearson Education-BR. 2011.