

Uma Abordagem Baseada em Técnicas de Reutilização para a Definição de Processos de Software

Ahilton Silva Barreto

ahilton@cos.ufrj.br

*Orientadores: Ana Regina Cavalcanti da Rocha e Leonardo Gresta Paulino Murta
{darocha, murta}@cos.ufrj.br*

COPPE/Universidade Federal do Rio de Janeiro
Caixa Postal 68511 – CEP 21945-970 – Rio de Janeiro, Brasil

Tese de Doutorado

Curso iniciado em Março de 2006

Previsão de conclusão em Março de 2010

Proposta de Tese aprovada em: Qualificação ainda não realizada (prevista para Agosto/2007)

Resumo

Definir processos de software não é uma tarefa simples, exigindo grande quantidade de conhecimento e experiência por parte dos profissionais que a realizam. Assim sendo, explicitar o conhecimento necessário para a definição de processos, considerando diversos fatores (características da organização, melhores práticas existentes, modelos de maturidade, etc.) e formalizando esse conhecimento em partes reutilizáveis de processos pode trazer grandes benefícios e apoiar de forma significativa a realização dessa atividade. Dessa forma, o conhecimento poderá ser reutilizado por profissionais de diferentes níveis de experiência, aumentando a qualidade dos processos gerados e evitando retrabalho desnecessário em definições similares. Neste contexto, este trabalho apresenta uma proposta de abordagem para a definição de processos de software fortemente baseada em reutilização e em gerência de conhecimento, que visa facilitar a definição de processos, buscando fornecer conhecimento relevante, evitar retrabalho e melhorar a qualidade dos processos gerados. Serão considerados componentes, frameworks e padrões de processo. Além disso, será considerado também o conceito de linha de processos, uma adaptação das linhas de produtos de software para o contexto de processos, tratando questões de gerência de configuração e gerência de conhecimento envolvidas. Rastreabilidade para requisitos de processos e modelos de maturidade também serão considerados.

Palavras-chave: Definição de Processos, Reutilização de Processos, Linhas de Processos de Software, Gerência de Conhecimento.

Uma Abordagem Baseada em Técnicas de Reutilização para a Definição de Processos de Software

1. Caracterização do Problema

A definição de processos de software é uma atividade bastante complexa, que precisa levar em conta uma série de fatores, tais como: necessidades e características da organização, competências das pessoas que irão executar os processos, técnicas e métodos que serão utilizados, características dos projetos realizados pela organização, e a conformidade com padrões ou modelos de referência de processos, entre outros. Assim, definir processos de software requer uma grande quantidade de conhecimento, o que exige um profissional bastante especializado e que consiga harmonizar todos esses fatores.

No entanto, é de se esperar que caso esse conhecimento de posse dos engenheiros de processo experientes seja explicitado, formalizado e disponibilizado para outros profissionais, se torne possível que profissionais menos experientes sejam capazes de definir processos, reutilizando o conhecimento dos mais experientes.

Portanto, é importante o desenvolvimento de um apoio à definição de processos que reutilize o conhecimento de engenheiros de processos experientes. Esse apoio, para ser efetivo, deve considerar muitos dos aspectos normalmente considerados por um engenheiro de processos no momento de definir processos, tais como: (i) rastreabilidade entre partes do processo e normas, modelos de maturidade, objetivos organizacionais e requisitos do processo; (ii) outros processos anteriormente definidos e sua adequação ou não a situações específicas; (iii) consistência entre as diversas partes do processo, garantindo que o que é requerido por uma parte do processo seja produzido por outra, e que todos os pré-requisitos sejam satisfeitos; (iv) conformidade do processo definido a padrões, normas e modelos de maturidade; e (v) adequação do processo à organização que o utilizará. Além disso, esse apoio precisa permitir que a definição de processos seja fácil, e que o conhecimento a ser reutilizado esteja disponível. Esse tipo de apoio pode auxiliar não apenas os profissionais menos experientes, mas também os próprios engenheiros de processo, que poderão reutilizar de forma mais efetiva processos definidos anteriormente. Com isso, espera-se atingir um aumento de produtividade, com a diminuição do esforço necessário para realizar as atividades. Além disso, espera-se um aumento na qualidade e adequação dos processos gerados, devido à reutilização do conhecimento de especialistas.

Neste contexto, o objetivo deste trabalho é desenvolver uma abordagem para definição de processos baseada em técnicas de reutilização, que permita a fácil reutilização de conhecimento sobre processos. Para explicitar esse conhecimento serão utilizadas diversas técnicas, principalmente algumas comumente aplicadas no desenvolvimento de produtos de software tradicional. Acredita-se que essas técnicas possam ser adaptadas para o uso não apenas no desenvolvimento de software, mas também na definição de processos.

2. Fundamentação Teórica e Trabalhos Relacionados

Há diversos trabalhos na literatura que defendem que processos têm semelhanças com software, e que é possível aplicar métodos e técnicas de desenvolvimento de software na definição de processos. Osterweil [1] já argumentava que processos de software são software também, e assim como software, podem ter requisitos especificados, podem ser modelados, desenvolvidos e testados. O autor também argumenta que conhecimento de processo de software é um ativo valioso que deve ser preservado e disseminado, e que a explicitação desse conhecimento permitiria sua reutilização.

Da mesma forma que há muitas analogias entre processos de software e produtos de software, podem ser feitas também analogias interessantes entre reutilização de processo de software e reutilização de produtos de software. Kellner [2] destaca que o conhecimento de técnicas de reutilização de produtos de software, tais como: arquiteturas povoadas com componentes reusáveis; gerenciamento de repositórios para armazenar, catalogar, procurar, acessar, etc. os ativos reusáveis; gerência de configuração de ativos reusáveis; entre outros; poderiam ser aplicados a processos de software.

2.1 – Componentes

Podem-se considerar componentes de software, no contexto do desenvolvimento baseado em componentes, como elementos fundamentais, que servem como unidade de encapsulamento com interfaces bem definidas e podem ser reutilizados ou substituídos, alavancando a produtividade e a qualidade [3]. De forma semelhante, um componente de processo pode ser visto como um encapsulamento de informações e comportamentos de processo em um dado nível de granularidade [4]. Assim, um processo pode ser tratado como a integração de um conjunto de componentes de processo em diferentes níveis de granularidade [5]. Componentes de processo reutilizáveis podem fornecer apoio eficiente aos engenheiros de processo e equipes de projetos para a construção de modelos de processo e realização de ajustes dinâmicos no processo, o que contribui para a otimização e controle do processo [6].

Há muitos trabalhos sobre componentes de processo para formar processos maiores. Fusaro *et al.* [5] focam no uso de componentes de processo para inovações, ou melhorias no processo. Gary e Lindquist [4] apresentam uma abordagem para encapsular fragmentos de processos como sendo componentes de processo interoperáveis e reusáveis. Ru-Zhi *et al.* [6] apresentam um framework para representação de componentes de processo, baseado em reutilização, para a classificação e descrição de componentes de processo.

No entanto, a maioria dos trabalhos existentes não considera que o apoio ferramental para a execução do componente seja parte do próprio componente. Também há muitas questões referentes a como tratar a gerência de configuração de componentes de processos ainda em aberto, incluindo a rastreabilidade (componentes, requisitos, processos instanciados) e a evolução dos componentes de processo. As interfaces e os conectores de componentes de processos também foram pouco abordados na literatura [2]. Além disso, apesar de o uso de componentes para composição de processos aparentar fornecer muitas vantagens, partir de unidades tão pequenas para compor grandes processos parece ainda ser insuficiente. Se analisarmos novamente a reutilização no contexto de produtos de software, podemos observar que uma das lições aprendidas com os esforços para se alcançar a reutilização nas últimas décadas foi a de que a reutilização bottom-up, ou seja, a composição de componentes arbitrários para construir sistemas, não funciona na prática; programas de reutilização bem sucedidos devem empregar, também, uma abordagem top-down, ou seja, componentes são desenvolvidos de forma a se encaixarem em uma estrutura de alto nível definida por uma arquitetura de software [7]. Assim, combinar as duas abordagens parece ser mais adequado, ou seja, iniciar a partir de um nível mais alto (top-down) e ao longo da composição considerar também componentes individuais (bottom-up).

É de se esperar que para a definição de processos, apesar de os componentes de processo serem fundamentais, uma abordagem de definição que se inicie top-down, onde componentes de processo sejam encaixados em estruturas reusáveis maiores, tais como frameworks para modelos de ciclos de vida, ou templates de processos de um determinado nível de maturidade, facilite e potencialize a definição de processos. Essas estruturas reutilizáveis maiores poderiam ser usadas tanto para compor um componente de mais alto nível (ex.: macro-atividades) quanto para compor estruturas maiores do processo (ex.: sub-processos, fases de processos).

2.2 – Frameworks

Um framework pode ser visto como um “esqueleto” de uma aplicação, que pode ser adaptado por um desenvolvedor de aplicações [8]. Acredita-se que o conceito de framework possa também ser aplicado a processos. Neste caso, poderiam ser definidos “esqueletos” de processo, baseados, por exemplo, em modelos de ciclo de vida ou em níveis de maturidade, e essas estruturas genéricas seriam preenchidas por componentes de processo que atendessem aos requisitos de cada parte do framework. Uma abordagem de framework de processos existente na indústria é o “Eclipse Process Framework” (EPF) [9], um framework gratuito que permite a definição de processos a partir de componentes menores, em diversos níveis de granularidade. Apesar de bastante completo, o EPF não considera diversos fatores como rastreabilidade para requisitos de processos e modelos de referência e foca na definição do processo, não em sua execução.

2.3 – Linhas de Produtos e Linhas de Processos

Quando comparado com o desenvolvimento baseado em componentes, o uso de linha de produtos representa uma mudança de paradigma: da reutilização de componentes para a reutilização de arquiteturas. Enquanto uma arquitetura de software “normal” define apenas a estrutura de um único sistema de software, uma arquitetura de linha de produtos define a estrutura arquitetural para um conjunto de produtos relacionados [10].

Uma linha de produtos fornece um conjunto de características (*features*) que podem ser instanciadas em produtos específicos gerados a partir da linha de produtos. Uma característica é uma unidade lógica de comportamento especificada por um conjunto de requisitos funcionais e de qualidade. Características normalmente não são independentes e muitos tipos de relações podem existir entre elas, tais como: (i) “depende de”: uma característica requer que uma outra também esteja presente para que seja útil; (ii) “exclusão mútua”: duas características relacionadas não podem nunca estar presentes simultaneamente em um produto, o que levaria a inconsistências lógicas; (iii) “conflitante”: características conflitantes não são logicamente incompatíveis, mas exibem requisitos de qualidade conflitantes [7].

Para maximizar a reutilização e o entendimento, uma arquitetura de linha de produtos distingue elementos obrigatórios (*core elements*) que estão presentes em todas as arquiteturas de produtos de “pontos de variação” que capturam diferenças entre arquiteturas de produtos específicas. Três tipos de pontos de variação são usados para diferenciar uma arquitetura da outra [10]: (i) elementos opcionais, que descrevem elementos arquiteturais que podem ou não estar presentes em uma arquitetura de produtos particular, (ii) elementos variantes, que definem elementos que sempre devem estar presentes, mas podem ser configurados para uma ou várias alternativas, e (iii) elementos variantes opcionais, que especificam elementos variantes que podem ou não existir.

Uma linha de produtos funciona como uma fábrica, que instancia produtos parecidos, cada um com um conjunto de características, através da composição de componentes existentes. Da mesma forma, é possível imaginar que se possam definir processos dessa maneira, ou seja, seria possível instanciar processos a partir de componentes de processos pré-existentes, sendo que cada instância teria um conjunto de características.

Surge então a idéia de linha de processos de software, que são linhas de produtos cujos produtos são processos de software. Conceitos como variantes, opcionalidade e elementos obrigatórios também estão presentes. A idéia de linha de processos de software é bastante recente na literatura e há pouquíssimos trabalhos publicados sobre o assunto. Rombach [11] sugere pela primeira vez, em 2005, o uso de linhas de processos, apresentando os benefícios esperados e uma abordagem de integração entre os conceitos de linha de processos e linhas de produtos de software. Jauffman e Munch [12] apresentam uma abordagem para adaptação de

processos padrão para projetos específicos, baseando-se em linhas de processos. Washizaki [13] propõe uma técnica para estabelecer linhas de processos e arquiteturas de linhas de processos que incorporam semelhanças e variabilidade.

Assim, as linhas de processos de software ainda estão em estágio bastante inicial. Há pouquíssima informação de uso e os conceitos encontram-se ainda de certa forma imaturos. Portanto, ainda há muitas questões em aberto em relação a sua aplicação, incluindo a definição de como representar e automatizar essas linhas de processos, como tratar questões de gerência de configuração (rastreabilidade, evolução, etc), como avaliar seu uso, etc.

2.4 – Padrões

Um padrão de processo descreve uma abordagem ou série de ações bem sucedidas para o desenvolvimento de software [14]. Permite o armazenamento e organização sistemáticos de conhecimento sobre soluções, além de fornecer os meios para reutilização de conhecimento de especialistas. Existem também os anti-padrões que representam as abordagens que não deram certo, explicitando uma situação que não deve ser repetida [14].

Segundo Ambler [14] existem três tipos básicos de padrões de processos: (i) Padrões de tarefa: mostram os passos detalhados para executar uma tarefa específica; (ii) Padrões de estágio: mostram os passos que normalmente são executados de forma iterativa, focando em um único estágio do desenvolvimento. (iii) Padrões de fase: este tipo mostra as interações entre os estágios nas fases do projeto de software. Já Huang e Zhang [15] definem três tipos de padrões: (i) de ciclo de vida (incremental, cascata, etc.); (ii) de atividades (o que deve ser feito, entradas e saídas, etc.); (iii) de fluxo de trabalho (descrevem uma execução de um conjunto de atividades).

De maneira geral, os padrões de processo fornecem conhecimento extra para auxiliar na definição e execução dos processos. No entanto, muitas vezes o conceito se confunde com o conceito de framework de processos, principalmente em situações em que os padrões envolvem agrupamentos de atividades ou tarefas. Assim, um padrão de processo pode variar de simples informação adicional sobre como usar um dado componente, até uma estrutura de componentes relacionados para formar uma fase do processo.

Apesar de existirem vários trabalhos que tratam de padrões de processos, também há questões em aberto, incluindo como detectar padrões e anti-padrões em um dado processo, como automatizar essa detecção, como incluir padrões nos processos instanciados, etc.

3. Resultados Esperados

O principal resultado esperado deste trabalho será a definição de uma abordagem de definição de processos baseada em reutilização, que permita a fácil reutilização de conhecimento sobre processos. Para atingir esse objetivo, uma série de outros resultados é esperada, e serão detalhados a seguir.

Espera-se definir uma infra-estrutura para componentes de processo, de forma que a definição de processos possa ser realizada com base em um conjunto de componentes de processo pré-existent. A definição de cada componente deve ser cuidadosa, de forma a permitir: rastreabilidade para normas, modelos e requisitos de processo; definição clara de interfaces, para permitir a integração com outros componentes; registro de informação de uso e adequação a diferentes cenários; apoio ferramental que auxilia na execução do componente; apoio à análise de desempenho dos processos; entre outros. Merece destaque o fato de que o apoio ferramental poderá ser parte do componente de processo, ou seja, o próprio componente fornecerá o apoio à sua execução. Além disso, componentes podem ser compostos por outros

componentes mais básicos, ou seja, o nível de granularidade de um componente de processo pode variar (ex.: tarefa, atividade, macro-atividade, etc).

Também pretendemos desenvolver uma infra-estrutura para criação e uso de frameworks de processo para definição de processos. Assim, deverá ser possível a definição de um processo aderente a um framework através da escolha de componentes de processo que atendam a seus requisitos. Por exemplo, pode-se definir um framework para processos que utilizem o RUP, ou que sejam aderentes ao nível C do MPS.BR.

O desenvolvimento de uma infra-estrutura para criação e uso de linhas de processos para definição de processos também está no escopo deste trabalho. Assim, deverá ser possível a definição de um processo através da simples configuração de uma linha de processos.

Uma linha de processos terá elementos obrigatórios e pontos de variação. Alguns elementos serão opcionais (por exemplo, implementar o Grupo de Garantia da Qualidade do Processo e Produto - GQPP - em um processo nível G) e outros variantes, ou seja, será possível escolher um entre alguns componentes que implementam uma mesma característica (por exemplo, estimativa por ponto de função ou por ponto de caso de uso). Assim, definir um processo a partir de uma linha de processos consistirá em incluir ou não os componentes opcionais e escolher um dentre os possíveis componentes variantes em cada ponto de variação. Todo o conhecimento restante já estaria encapsulado na própria linha de processos.

Uma característica (*feature*), neste contexto, poderia ser, por exemplo, o apoio a uma área de processo, ou o apoio a uma determinada técnica ou método. Seria possível, assim, determinar uma matriz, composta pelos produtos candidatos e pelas características candidatas, como ilustrado na tabela 1. Os itens apresentados na tabela 1 podem parecer simples em um primeiro momento, mas vale lembrar que tanto produtos quanto características podem ser compostos de outros produtos, características e componentes individuais. Cada um desses tem suas restrições e relacionamentos com os demais que precisam ser respeitados, o que torna o problema consideravelmente mais complexo.

Tabela 1 – Relação de Produtos e das Características que os compõem

Produto/Característica	Apoio a GPR	Apoio a GRE	Apoio a GQA	Apoio a GC
Processo Nível G	X	X			
Processo Nível F	X	X	X	X	...
Processo Nível E	X	X	X	X	...

A figura 1 exibe um exemplo bastante simples da proposta, apenas para ilustrar seu funcionamento pretendido. Podemos ver na parte mais à esquerda, uma linha de processos, que define alguns componentes obrigatórios, define atividades do GQPP como sendo opcionais e as atividades de estimativas como sendo variantes, ou seja, deve ser escolhida uma das três formas de realizar a atividade. Na parte central da figura, vemos uma linha de processos já especializada, em que as atividades do GQPP agora são obrigatórias. Essa especialização pode representar, por exemplo, o processo definido para uma organização nível G, que decidiu utilizar o GQPP. Vale lembrar que poderiam ser definidas várias linhas de processos especializadas, em mais de um nível de especialização, se necessário. Finalmente, na parte mais à direita, vemos um processo já instanciado, em que todas as decisões a serem tomadas em relação a linha de processos já foram tomadas. No caso, escolheu-se usar estimativas por pontos de casos de uso. Este processo poderia ser, por exemplo, um processo para um projeto específico.

Neste trabalho, padrões de processo também serão considerados. Sua função será complementar o conhecimento sobre a execução dos componentes de processo. Tanto os componentes, como frameworks ou linha de processos poderão ter associados padrões de processo para auxiliar na execução dos processos, ou seja, ao instanciar um processo a partir de, por exemplo, uma linha de processos, os padrões de processo relevantes já serão incluídos

no processo instanciado. Além disso, poderão ser definidos padrões para auxiliar na definição, guiando a escolha de elementos variantes e opcionais em uma linha de processos.



Figura 1 – Exemplo de uso de linhas de processos.

Outras questões bastante relevantes serão também consideradas neste trabalho, entre elas:

- Gerência de configuração de processos: definição de uma abordagem que considere, entre outros: versionamento de processos e de componentes de processos e sua evolução; rastreabilidade entre componentes e normas, requisitos; relacionamentos entre os processos instanciados e os itens que os geraram e outros.
- Comunidade de compartilhamento de conhecimento de processos: definição de uma espécie de comunidade de prática, onde os membros possam consultar, obter e disponibilizar conhecimento de processo. Ainda há muito a ser definido, mas essa comunidade poderia ser aberta a diversas organizações, e os itens de conhecimento poderiam ter diferentes tipos de licença de uso. Por exemplo, alguns poderiam ser gratuitos, em outros o apoio ferramental seria pago, a outros poderiam estar associadas horas de um consultor especializado no uso do item de conhecimento, etc.
- Definição de processos de definição de processos: definir e descrever as atividades que devem ser realizadas para definir um processo, segundo a abordagem proposta neste trabalho, considerando a definição de processos com e para reutilização.

4. Metodologia e Estado Atual do Trabalho

O trabalho aqui descrito foi motivado pela necessidade observada nas experiências da própria COPPE/UFRJ em definição de processos, em que muitos processos semelhantes precisam ser definidos e não há uma maneira sistemática de reutilizar em novas definições o conhecimento empregado em definições anteriores.

Partindo desta necessidade, foi feita uma busca na literatura procurando identificar as abordagens existentes que tratam de reutilização de processos. Foi escopo da pesquisa na literatura, também, a investigação de técnicas de reutilização de produtos de software, procurando identificar as semelhanças existentes e o que poderia ser reutilizado da própria reutilização de produtos de software, no contexto de processos de software. A partir da busca realizada até o momento, foram definidas as principais características do trabalho e quais são os resultados esperados mais importantes. Para complementar a pesquisa na literatura e

garantir um nível adequado de cobertura na pesquisa, será também realizada uma pesquisa baseada em revisões sistemáticas da literatura, sem o rigor original proposto por essa técnica, mas utilizando alguns conceitos básicos, não com o intuito de confirmar hipóteses, mas apenas de garantir uma boa cobertura para a pesquisa.

Para consolidar as idéias do trabalho, são realizados periodicamente seminários em que o trabalho é apresentado para o grupo de alunos de mestrado e doutorado, além dos orientadores da COPPE/UFRJ, de forma que todos possam criticar e contribuir com sugestões para o trabalho. A própria participação no WTDQS terá grande importância para a consolidação das idéias do trabalho e para avaliar a aceitação deste pela comunidade acadêmica, constituindo uma base mais sólida para a continuidade do desenvolvimento do trabalho.

No estágio atual do trabalho está sendo iniciado o detalhamento de como será de fato o uso de componentes, frameworks, linhas e padrões de processos para apoiar a definição de processos, considerando também aspectos de gerência de configuração e de gerência de conhecimento. Especialistas serão consultados para opinar sobre as informações necessárias nos componentes e sobre a granularidade mais adequada. Após o término desse detalhamento, será realizado um exemplo de uso da abordagem para avaliar se o que foi definido parece estar adequado e revisões serão feitas até que se chegue a um modelo mais adequado. Serão então definidas e desenvolvidas ferramentas de apoio para a abordagem. Seu uso será avaliado nas definições de processo da própria COPPE/UFRJ e também em organizações na indústria. Espera-se que o exame de qualificação ocorra, aproximadamente, em agosto de 2007, após um ano e meio do início do doutorado.

Referências

- [1] Osterweil, L., 1987, "Software Processes Are Software Too". In: *International Conference on Software Engineering*, pp. 2-13, Monterey, Estados Unidos, April.
- [2] Kellner, M.I., 1996, "Connecting Reusable Software Process Elements and Components". In: *International Software Process Workshop*, pp. 8-11, Dijon, France, June.
- [3] Murta, L.G.P., 2006, *Gerência de Configuração no Desenvolvimento Baseado em Componentes*, Tese de D.Sc., Programa de Engenharia de Sistemas e Computação, COPPE/UFRJ, Rio de Janeiro, Brasil.
- [4] Gary, K.A., Lindquist, T.E., 1999, "Cooperating Process Components". In: *International Computer Software and Applications Conference (COMPSAC)*, pp. 218-223, Phoenix, United States, October.
- [5] Fusaro, P., Visaggio, G., Tortorella, M., 1998, "REP - Characterizing and Exploiting Process Components: Results of Experimentation". In: *Working Conference on Reverse Engineering*, pp. 20-29, United States.
- [6] Ru-Zhi, X., Tao, H., Dong-Sheng, C., *et al.*, 2005, "Reuse-Oriented Process Component Representation and Retrieval". In: *ICCIT*, pp. 911-915, Shanghai, China, September.
- [7] Bosch, J., 2000, *Design and use of software architectures: adopting and evolving a product-line approach*, 1st ed. New York, United States, Addison Wesley Professional.
- [8] Johnson, R.E., 1997, "Frameworks = (components + patterns)", *Communications of the ACM*, v. 40, n. 10 (October), pp. 39-42.
- [9] Foundation, T.E., 2007, "Eclipse Process Framework Project". In: <http://www.eclipse.org/epf/>, 27/03/2007.
- [10] Garg, A., Critchlow, M., Chen, P., *et al.*, 2003, "An Environment for Managing Evolving Product Line Architectures". In: *International Conference on Software Maintenance*, pp. 358-367, Amsterdam, Netherlands.
- [11] Rombach, H.D., 2005, "Integrated Software Process and Product Lines". In: *International Software Process Workshop*, pp. 83-90, Beijing, China, May.
- [12] Jaufman, O., Münch, J., 2005, "Acquisition of a Project-Specific Process". In: *Product Focused Software Process Improvement*, pp. 328-342, Oulu, Finland, June.
- [13] Washizaki, H., 2006, "Building Software Process Line Architectures from Bottom Up". In: *Product-Focused Software Process Improvement*, pp. 415-421, Amsterdam, Netherlands, June.
- [14] Ambler, S.W., 1998, *Process Patterns: Building Large-Scale Systems Using Object Technology* New York, United States, Cambridge University Press.
- [15] Huang, H., Zhang, S., 2003, "Hierarchical Process Patterns: Construct Software Processes in a Stepwise Way". In: *IEEE International Conference on Systems, Man and Cybernetics*, v. 2, pp. 1353-1358, Washington, United States, October.