Apoio à Reutilização de Processos de Software em um Ambiente de Engenharia de Software Centrado em Processo

Daniel Dias de Carvalho, Anderson J. Serra da Costa, Ernani de Oliveira Sales, Adailton Magalhães Lima, Rodrigo Quites Reis

Laboratório de Engenharia de Software – Universidade Federal do Pará (UFPA)
Belém – PA – Brasil
danieldias@ufpa.br, {anderson, ernani, adailton}@webapsee.com, quites@ufpa.br

Abstract: *Software process reuse is an important approach to reduce costs and development time while increasing the quality and the adequacy of generated processes. This paper presents the support provided by WebAPSEE environment with respect to the life cycle for software processes reuse, which comprises the following steps: 1) process modeling for reuse; 2) retrieving, tailoring and instantiation of reusable processes; 3) process enactment; and 4) evaluation and generalization of previously enacted processes.*

Resumo: *Reutilização de processos de software é uma abordagem importante para diminuição dos custos e do tempo de desenvolvimento, além de aumentar a qualidade e a adequação dos processos gerados. Este artigo apresenta o apoio fornecido pelo ambiente WebAPSEE em relação ao ciclo de vida para reutilização de processos de software, que compreende as etapas de: 1) modelagem de processos visando a reutilização; 2) recuperação, adaptação e instanciação de processos; 3) execução de processos; e 4) generalização e avaliação de processos encerrados.*

1. Introdução

A demanda por produtos de software vem aumentando e estes, por sua vez, são cada vez mais complexos [Borges e Falbo 2001] e afetam praticamente todas as atividades humanas [Pressman 2006]. À medida que a importância do software e a exigência por qualidade crescem, a comunidade de Engenharia de Software (ES) vem tentando desenvolver abordagens para melhorar o desenvolvimento e produzir software com mais qualidade [Pressman 2006].

Diversos autores afirmam que a qualidade do processo de desenvolvimento impacta diretamente na qualidade do produto a ser desenvolvido [Fuggetta 2000], [Pressman 2006] [Osterweil 1987]. O processo de software pode ser definido como um conjunto de políticas, estruturas organizacionais, tecnologias, procedimentos e artefatos necessários para conceber, desenvolver, implantar e manter um produto de software [Fuggetta 2000].

Segundo Greer e Conradi (2009), a necessidade de um processo definido tem sido reconhecida como uma estratégia de redução de riscos para a gerência de projetos. Bertollo e Falbo (2003) acrescentam que "a principal causa dos problemas no desenvolvimento de software é a falta de um processo de desenvolvimento de software claramente definido e efetivo".

Uma das abordagens propostas para melhorar a qualidade dos processos de software é prover a reutilização de processos [Kellner 1996] a partir de processos definidos previamente, disseminando conhecimentos e boas práticas entre os projetos de uma organização.

O WebAPSEE [Lima Reis e Reis 2007] é um Ambiente de Engenharia de Software Centrado em Processo (do inglês, PSEE - *Process-centered Software Engineering Environment*) desenvolvido no contexto do Laboratório de Engenharia de Software da Universidade Federal do Pará. Esse ambiente fornece um conjunto de funcionalidades que visam apoiar a definição de processos de software reutilizáveis, a adaptação de processos para projetos específicos, a execução e a generalização de processos executados.

O texto é organizado como segue. A seção 2 apresenta o ciclo de vida para reutilização de processos de software e a importância de cada uma das etapas. A seção 3 apresenta como o ambiente WebAPSEE apóia cada uma das etapas do ciclo de reutilização de processos. A seção 4 apresenta a conformidade do WebAPSEE em relação ao MPS.BR e casos reais do uso de algumas funcionalidades referentes à reutilização de processos. A seção 5 apresenta alguns trabalhos relacionados. Por fim, a seção 0 apresenta as considerações finais do trabalho.

2. Reutilização de Processos de Software

A reutilização de software é o processo de criação de sistemas de software a partir de software preexistente [Krueger 1992]. Frakes e Kang (2005) definem como "o uso de software ou conhecimento preexistente para a construção de um novo software". Essa abordagem tem sido amplamente aceita pela comunidade de ES como uma alternativa para melhorar a qualidade do desenvolvimento de software.

Diversos autores afirmam que técnicas de reutilização de produtos de software têm sido transferidas por analogia para o contexto da reutilização de processos software [Barreto *et al.* 2009] [Reis 2002] [Kellner 1996]. Segundo Kellner (1996), o conhecimento de técnicas de reutilização de produtos de software - tais como componentes reutilizáveis; repositórios para armazenar, catalogar e recuperar esses componentes reutilizáveis; e gerência de configuração dos componentes - também podem ser aplicados a processos de software.

Reutilização de processos de software é “uma ampla área de estudo e utilização prática relacionada aos diferentes aspectos envolvidos com a reutilização do conhecimento adquirido na condução de projetos anteriores” [Reis 2002]. Essa área se refere à definição de um modelo de processo através da reutilização de modelos de processo definidos previamente, ao invés de defini-lo do início [Ribó e Franch 2002].

Reutilizar processos de software é uma abordagem importante para disseminar experiências de projetos anteriores [Reis 2002]; diminuir os custos e o tempo de desenvolvimento; aumentar a qualidade e a adequação dos processos gerados; promover a reutilização de conhecimento de especialistas; e tornar a atividade acessível a profissionais menos experientes [Barreto *et al*. 2009]. Xu e Ramesh (2008) complementam que modelar um processo do início, sem adaptar processos previamente existentes, é uma atividade arriscada e muito trabalhosa.

Reis (2002) classifica, com base em [Ribó e Franch 2002], [Derniame *et al.* 1999] e [Feiler e Humphrey 1993], modelos de processo em três tipos principais: (1) Modelo Abstrato: consiste em um modelo genérico e reutilizável de processo, também denominado *template*, que estabelece um ponto de início para a construção de um novo modelo de processo; (2) Modelo Instanciado: é uma instância de um modelo abstrato pronto para execução, com objetivos e restrições específicas para o contexto em que será executado; (3) Modelo em Execução ou Executado: registra o histórico da execução de um processo, incluindo os eventos e desvios (modificações realizadas no modelo relacionado).

A Figura 1 apresenta o ciclo de vida para reutilização de processos de software proposto por [Costa 2010], adaptado de [Reis 2002] e [Jørgensen e Carlsen 2001].

Figura 1 - Ciclo de Vida para Reutilização de Processos de Software [Costa 2010]

A etapa **Modelagem de Processos Visando a Reutilização** é responsável por definir, através de uma linguagem de modelagem de processo (do inglês *Process Modeling Language* - PML), os modelos de processos abstratos. Os modelos de processos de software descrevem o conhecimento de uma organização e, portanto, modelos que descrevem experiências bem sucedidas devem ser continuamente disseminados para reutilização em diferentes projetos [Reis 2002]. Dentro de uma organização de desenvolvimento de software, um Processo Padrão é um modelo de processo abstrato que engloba um conjunto fundamental de elementos (atividades, métodos, ferramentas, práticas, papéis, entre outros) que devem estar presentes em qualquer processo da organização [Borges e Falbo 2001] [ISO/IEC 2004]. Definir um processo padrão é uma boa prática que pode resultar em economia de tempo e esforço na definição de novos processos [Bertollo e Falbo 2003] [Xu e Ramesh 2008].

A etapa **Reutilização de Processos** prevê critérios, estratégias e mecanismos para auxiliar um projetista na seleção, recuperação, instanciação e adaptação de processos abstratos. Por ser genérico, o processo padrão precisa ser adaptado às necessidades específicas de cada projeto [Maia *et al*. 2005]. Borges e Falbo (2001) complementam essa ideia afirmando que "não existe um processo de software que possa ser genericamente aplicado a quaisquer projetos, visto que nenhum projeto é idêntico ao outro". Adaptar processo é uma atividade que envolve modificações, tais como a modificação sintática do modelo, o refinamento dos tipos genéricos para os tipos dependentes da organização-destino e a instanciação de agentes e recursos a partir dos tipos e restrições definidas no modelo de processo abstrato original [Reis 2002]. Essas modificações são necessárias para se reutilizar esse modelo de processo em um contexto específico [Reis 2002].

A etapa **Execução de Processos** descreve a execução de processos de software, incluindo as modificações referentes à evolução do processo. Nessa etapa as atividades modeladas para um projeto específico são realizadas, seja por agentes humanos, seja automaticamente por ferramentas. Essa etapa envolve questões importantes acerca de planejamento, controle, monitoração, garantia de conformidade com o processo modelado, treinamento, segurança e recuperação do processo [Feiler e Humphrey 1993]. A execução de processos de software pode ocorrer quando um modelo de processo está instanciado, ou seja, pronto para execução [Lima Reis 2003].

Por fim, a **Generalização e Avaliação de Processos Encerrados** extraem informações da execução de processos para alimentar o repositório de modelos de processos reutilizáveis e identificar possíveis oportunidades de melhorias.

3. Apoio à Reutilização de Processos de Software no Ambiente WebAPSEE

Esta seção apresenta uma visão geral de como o ambiente WebAPSEE apóia cada uma das etapas do ciclo de vida para reutilização de processos de software. Diversos trabalhos

relativos a Reutilização de Processos de Software, com foco especial às etapas de Modelagem [Lima *et al*. 2006], Seleção e Recuperação [Sales *et al*. 2006], Reutilização [Costa *et al*. 2007] e Generalização e Avaliação [Santos 2009] de processos já foram publicados. O presente trabalho acrescenta a descrição do mecanismo de Adaptação semi-automatizado de processos de software, que foi resultado de uma dissertação de mestrado [Costa 2010]. Nas subseções 3.1 a 3.5 serão apresentadas as funcionalidades fornecidas pelo ambiente WebAPSEE a cada uma das etapas do ciclo de vida para reutilização de processos.

3.1. Modelagem Visando a Reutilização

No ambiente WebAPSEE, existem três representações para um modelo de processo de software: processos, *templates* de processo e áreas de processo. Um processo é constituído por todos os componentes que permitem o planejamento e acompanhamento de sua execução em um projeto, tais como: atividades, conexões, artefatos, agentes, papéis, grupos, recursos, ferramentas, métricas, estimativas, entre outros. Um *template* de processo representa um processo em um nível abstrato, modelado visando a sua reutilização, não possuindo componentes específicos relativos a um projeto. Uma área de processo é um tipo particular de *template* cujo seus componentes de processo atendem a um único propósito específico, como gerência de configuração, verificação, validação, ou outro. Toda área de processo deve possuir três instâncias relativas ao nível de atendimento (baixo, médio e alto) dos seus componentes de processo ao propósito da área de processo de software que ela descreve.

A modelagem de *templates* de processo é realizada através da linguagem de modelagem de processos do ambiente, denominada WebAPSEE-PML. Essa linguagem fornece uma notação gráfica para representar componentes de processo e as conexões entre eles (Figura 2). Uma atividade decomposta representa um mecanismo de abstração do processo, através do agrupamento de outras atividades. Atividades normais são executadas pelos agentes (pessoas) e/ou grupos de agentes, com o auxílio de artefatos de entradas e recursos, e como resultado são produzidos artefatos de saída. Atividades automáticas[1] são ações realizadas por ferramentas automatizadas, sem a necessidade de alocação de agentes.

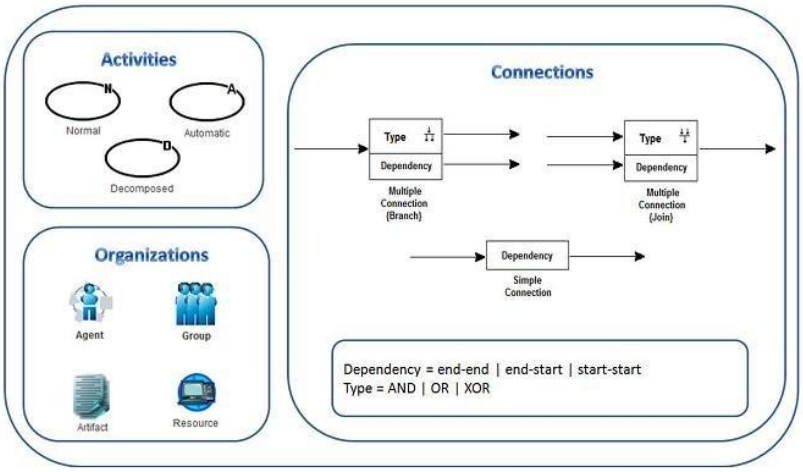

Figura 2 - Elementos da WebAPSEE-PML [Costa 2010].

As informações modeladas visualmente com a WebAPSEE-PML podem ser complementadas por descrições textuais fornecidas através de formulários. O *templates* de processo e as áreas de processo modeladas são armazenados no repositório de *templates* do WebAPSEE, para que seja possível sua reutilização posteriormente. A Figura 3 apresenta um *template* de processo modelado no WebAPSEE.

[1] Apesar de ser definida pela linguagem WebAPSEE-PML, a modelagem de atividades automáticas ainda não está disponível na atual versão (1.7) do ambiente WebAPSEE.

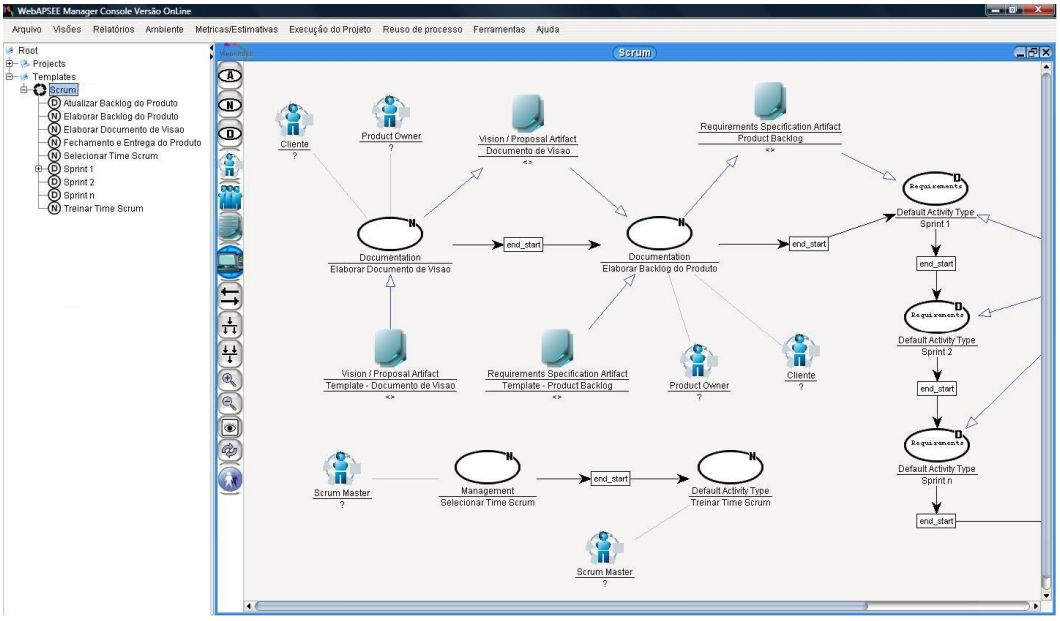

Figura 3 - Modelagem de *Template* de Processos.

Para que o caráter abstrato dos *templates* seja garantido, algumas regras são verificadas durante a modelagem. Por exemplo, não é permitida a alocação de agentes, apenas a definição de papéis necessários para a execução das atividades. Além disso, um *template* não pode ser executado, devendo ser primeiramente transformado em um processo.

O WebAPSEE também possui o recurso de controle de versões dos *templates* de processo, permitindo o registro do histórico de suas modificações. Tal recurso permite ainda que um *template* seja continuamente aperfeiçoado e, quando necessária, haja a substituição da versão atual por uma mais nova, mantendo as versões anteriores no repositório [Costa *et al.* 2007].

3.2. Reutilização

3.2.1. Recuperação e Instanciação de *Templates* de Processo

A recuperação de *templates* de processo pode ser realizada manualmente ou através de um mecanismo que utiliza a técnica de Raciocínio Baseado em Casos (CBR – *Case Based Reasoning*, em inglês) [Bergmann *et al*. 1999].

Na recuperação manual o usuário escolhe o *template* de processo, dentre os disponíveis no repositório, mais adequado ao contexto do projeto e, então, aciona o mecanismo de instanciação. O mecanismo de instanciação gera um modelo de processo executável correspondente ao *template* de processo selecionado.

Na recuperação com o mecanismo CBR, o usuário fornece informações sobre o contexto do projeto (Figura 4) e cabe a esse mecanismo escolher o *template* de processo a ser recuperado e instanciado. As informações de contexto são agrupadas em 5 dimensões: Projeto (porte, objetivo, tempo estimado, esforço e orçamento), Processo (tipo de processo, objetivo, tipos de atividades e tipos de artefatos), Software (nível de funcionalidade, nível de confiabilidade, nível de usabilidade, nível de eficiência, nível de manutenibilidade e nível de portabilidade), Organização (cargos, habilidade, nível de competência dos agentes e nível de conhecimento dos requisitos) e Ambiente (tipos de recursos e tipos de ferramentas).

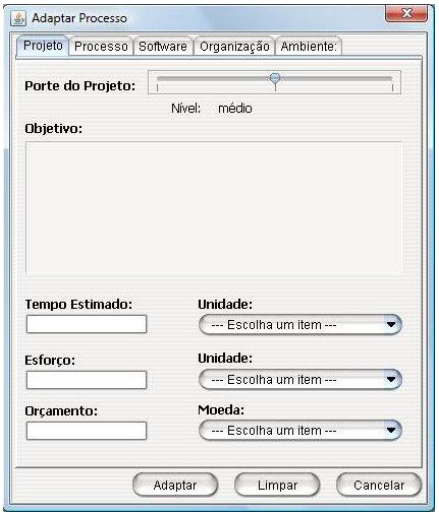

Figura 4 - Formulário de Informações de Contexto.

O mecanismo CBR analisa a similaridade de diferentes *templates*, armazenados no repositório de *templates*, em relação às informações fornecidas pelo usuário, recuperando e instanciando aquele *template* com o maior grau de similaridade. O cálculo de similaridade é realizado utilizando dois conceitos de similaridades: similaridade local e similaridade global. A similaridade global é calculada pela soma das similaridades locais, que, por sua vez, representam as similaridades dos atributos fornecidos pelo usuário em relação às respectivas características presentes nos *templates* de processo. Mais detalhes sobre o funcionamento desse mecanismo são apresentados em [Sales *et al.* 2006] e em [Costa 2010].

3.2.2. Composição

A composição de processos consiste na inserção de um *template* de processo em uma atividade decomposta (Figura 5). Ao compor uma atividade decomposta com um *template*, os componentes de processo inerentes ao *template* são copiados para o modelo de processo referenciado pela decomposta. Tal operação pode ser realizada tanto na modelagem de processos quanto de *templates* de processo, além de ser utilizada pelo mecanismo de adaptação de processos.

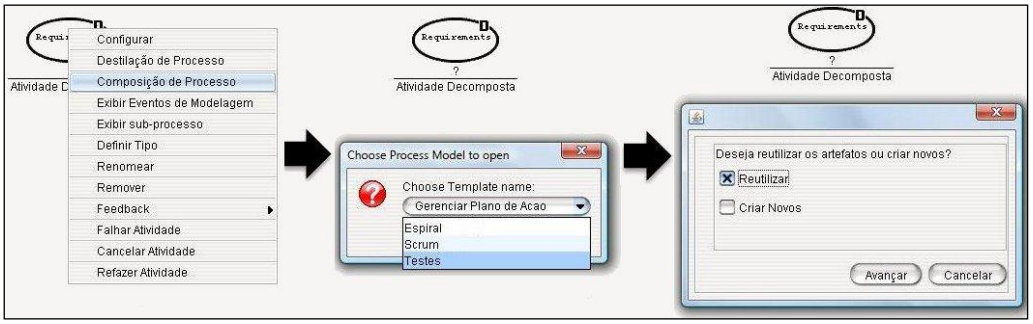

Figura 5 - Composição de Processos no WebAPSEE.

3.2.3. Adaptação

A adaptação pode ser realizada manualmente ou de forma semi-automatizada. Na adaptação manual o usuário tem a responsabilidade de realizar todas as modificações necessárias no *template* selecionado e instanciado previamente, como a adição e remoção de papéis, artefatos e recursos, além da composição de processos.

O mecanismo semi-automatizado de adaptação auxilia o usuário, selecionando, recuperando e instanciando automaticamente um *template* de processo e sugerindo

modificações nesse processo instanciado. Para o funcionamento do mecanismo semi-automatizado de adaptação de processos do WebAPSEE é necessário fornecer uma base de regras *Fuzzy* [Zadeh 1989]. Essas regras são baseadas na sintaxe "Se <*condições*> Então <*conclusão*>", onde as condições são valores de atributos referentes ao contexto do projeto (Figura 4). A conclusão se refere ao nível de rigor de uma área de processo a ser inserida no *template* de processo recuperado. É importante ressaltar que o conjunto de regras deve ser especificado por especialistas em adaptação, não sendo papel do usuário informá-las durante a adaptação. A seguir, é apresentado um exemplo de regra *Fuzzy* gerada para o mecanismo:

- **SE** o Porte do Projeto for baixo **E** a Usabilidade do Software for baixa **E** a Funcionalidade do Software for baixa **E** a Manutenibilidade do Software for baixa **E** a Competência dos Agentes for alta, **ENTÃO** o Processo de Verificação deve possuir nível de rigor baixo.

O mecanismo de adaptação semi-automatizado já fornece algumas regras, mas é recomendável a modificação, criação ou remoção de algumas delas, de modo a se adequarem às necessidades e às diretrizes de adaptação de processos da organização alvo. Após a configuração das regras *Fuzzy*, o mecanismo recebe como entrada a caracterização do contexto (Figura 4).

Com base nessas informações de contexto e nas regras *Fuzzy*, o *template* de processo mais similar às informações fornecidas é selecionado, recuperado e instanciado automaticamente do repositório de *templates* de processo. Paralelamente, também baseado nas informações de contexto e nas regras *Fuzzy* previamente configuradas, o mecanismo sugere uma lista de áreas de processo que devem ser adicionadas ao processo instanciado, além de determinar os seus respectivos níveis de rigor. O usuário, então, seleciona as áreas de processo relevantes e indica as atividades decompostas em que cada uma delas será inserida no processo (Figura 6).

Figura 6 – Seleção e Composição de Áreas de Processo [Costa 2010].

Após a inserção das áreas de processo sugeridas pelo mecanismo semi-automatizado, o usuário continua com a possibilidade de realizar adaptações manuais conforme as necessidades específicas do projeto sendo executado.

3.3. Execução de Processos

A máquina de execução do WebAPSEE coordena o fluxo de execução das atividades, com base nas conexões entre elas. Atividades podem ser executadas apenas quando todas as precondições são satisfeitas, garantindo que o processo adaptado para o projeto seja seguido. Além disso, a execução de processos no WebAPSEE segue uma abordagem flexível, permitindo mudanças dinâmicas no processo e execução de processos incompletos [Lima *et*

al. 2006]. Para evitar inconsistências durante as modificações, um conjunto de regras, descritas em [Lima Reis 2003], são verificadas dinamicamente.

O acompanhamento do processo pode ser feito visualmente, através de rótulos que identificam o estado atual de cada atividade (*Ready*, *Waiting*, *Active*, *Paused*, *Finished* ou *Canceled*). Quando conexões entre atividades são definidas; artefatos são modificados, agentes são alocados, entre outros eventos, a máquina de execução é notificada e estabelece novos estados para as atividades envolvidas. Dessa forma a gerência tem uma visão geral do andamento do processo. A Figura 7 apresenta uma visão de um processo em execução no WebAPSEE.

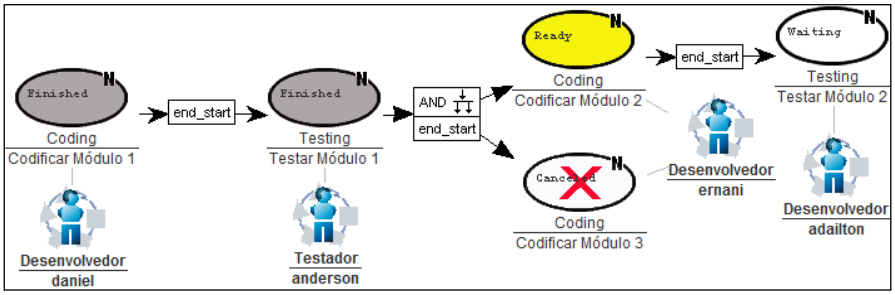

Figura 7 - Acompanhamento da Execução do Processo.

3.4. Avaliação de Processos Encerrados

O modelo de gerência de eventos do ambiente WebAPSEE é responsável por registrar eventos gerados por ações do usuário ou internamente pelo sistema [Paxiúba *et al*. 2007]. São registrados no *log* do WebAPSEE eventos como ativação, execução e encerramento de um processo; início e término de execução de atividades; alocação de recursos; e outros. Para cada evento registrado, é mantida uma rastreabilidade entre o tipo do evento, a data de ocorrência e a sua origem. A partir desses dados armazenados, é possível gerar relatórios gerenciais, que podem ser utilizados para o acompanhamento de processos em execução ou avaliação de processos encerrados.

3.5. Destilação/Generalização de Processos Encerrados

Processos executados e encerrados podem ser generalizados, através do acionamento, pelo usuário, do mecanismo de generalização [Costa e Sales 2007] (Figura 8). Esse mecanismo reinicializa todas as variáveis modificadas para o projeto – tais como data de início, data de finalização e estimativa de tempo - e remove todas as informações modeladas especificamente para o contexto em que o processo foi executado. Por exemplo, um agente alocado para a realização da atividade é removido, permanecendo apenas o papel definido. Ao final dessas operações, o resultado é um novo *template* de processo que poderá ser armazenado no repositório de *templates* e reutilizado posteriormente, completando o ciclo de vida para reutilização de processos.

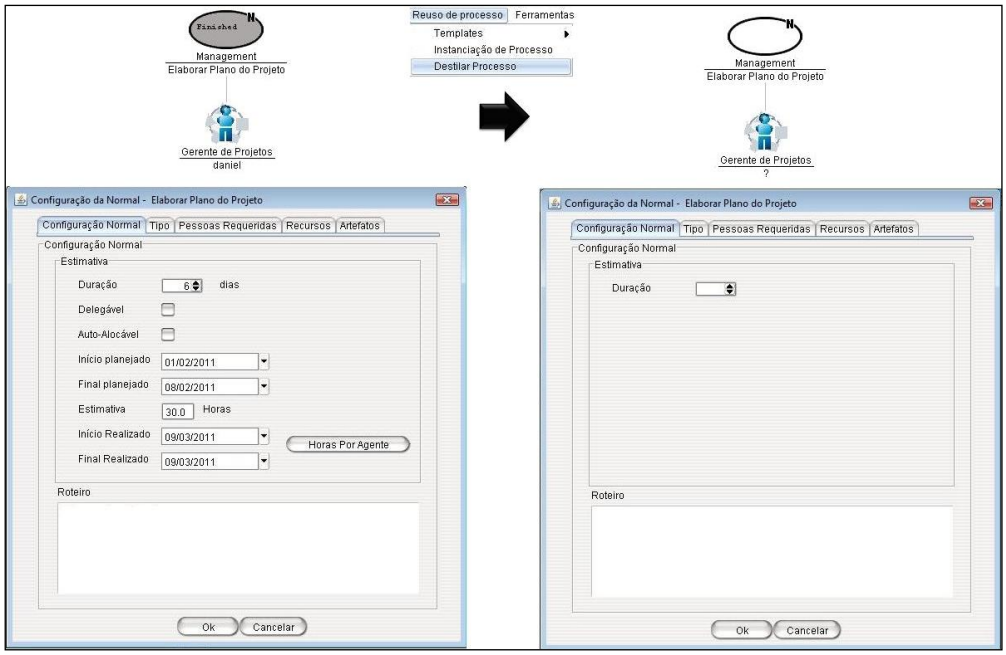

Figura 8 - Acionamento do Mecanismo de Generalização

4. Conformidade com o MPS.BR e Experiências Práticas com o WebAPSEE

Esta seção apresenta, primeiramente, como o WebAPSEE atende aos resultados esperados do MPS.BR. Em seguida, são descritas experiências reais no uso desse ambiente para reutilização de processos.

4.1. Conformidade com o MPS.BR

O Modelo de Referência do MPS.BR [SOFTEX 2009] trata questões relativas a definição e adaptação de processos de software através do processo Definição do Processo Organizacional (DFP). De acordo com esse modelo, o propósito do processo DFP é "estabelecer e manter um conjunto de ativos de processo organizacional e padrões do ambiente de trabalho usáveis e aplicáveis às necessidades de negócio da organização". A Tabela 1 apresenta os resultados esperados do processo DFP e como o WebAPSEE atende a cada um deles.

Tabela 1 – Mapeamento entre o DFP e funcionalidades do WebAPSEE.

Resultado Esperado	Como o WebAPSEE Atende
DFP1. Um conjunto definido de processos padrão é estabelecido e mantido, juntamente com a indicação da aplicabilidade de cada processo	Um conjunto de *templates* de processo pode ser modelado e armazenado no repositório do WebAPSEE. A aplicabilidade de cada *template* e área de processo são definidas através da técnica de CBR e de regras *Fuzzy*, respectivamente.
DFP2. Uma biblioteca de ativos de processo organizacional é estabelecida e mantida	Ativos de processo podem ser armazenados no repositório do WebAPSEE

DFP3. Tarefas, atividades, papéis e produtos de trabalho associados aos processos padrão são identificados e detalhados, juntamente com o desempenho esperado do processo	As informações necessárias em relação aos *templates* de processo podem ser definidas durante a modelagem, utilizando-se a linguagem WebAPSEE-PML e descrições textuais.
DFP4. As descrições dos modelos de ciclo de vida a serem utilizados nos projetos da organização são estabelecidas e mantidas	Podem ser modelados diferentes *templates* de processo que definem diferentes modelos de ciclo de vida. O modelo de ciclo de vida mais adequado a cada projeto poderá ser recuperado manualmente ou automaticamente durante a adaptação.
DFP5. Uma estratégia para adaptação do processo padrão é desenvolvida considerando as necessidades dos projetos	Na adaptação semi-automatizada, o modelo de ciclo de vida e as áreas de processo são recuperadas do repositório com base em informações de contexto do projeto fornecidas pelo usuário. Os critérios de qual modelo de ciclo de vida e de que áreas de processos devem ser recuperadas são definidos através da técnica de CBR e de regras *Fuzzy*, respectivamente.
DFP6. O repositório de medidas da organização é estabelecido e mantido	O WebASEE possui integrado um plano de medição organizacional, que permite a definição das metas organizacionais, métricas, questões e indicadores. Podem ser definidas métricas referentes a diversos componentes de processo, como atividades, agentes, grupos de agentes, artefatos e recursos. As métricas são coletadas e armazenadas no repositório do ambiente, permitindo a geração de relatórios gerenciais.
DFP7. Os ambientes padrão de trabalho da organização são estabelecidos e mantidos	O WebAPSEE permite o cadastro de recursos compartilhados e ferramentas organizacionais. Exemplos de recursos são impressoras, computadores, roteadores ou qualquer recurso que pode ser compartilhado entre os projetos. Esses recursos cadastrados podem ser inseridos posteriormente nos *templates* de processo padrão da organização, através da linguagem WebAPSEE-PML.

4.2. Relatos de Experiências sobre Reutilização de Processos com o WebAPSEE

Funcionalidades do ambiente WebAPSEE referentes ao ciclo de vida para reutilização de processos de software têm sido utilizadas na prática por empresas de Belém para auxiliar na implantação de processos.

França *et al.* (2009) descreve a experiência na utilização do WebAPSEE para a implantação do Nível G do MR-MPS no Centro de Tecnologia da Informação e Comunicação (CTIC) da Universidade Federal do Pará (UFPA). Outro relato de experiência sobre a utilização dessa ferramenta para a implantação do Nível G do MR-MPS, na empresa Equilibrium Web, é descrito em [Sales *et al.* 2010].

Em ambos os casos, o ambiente WebAPSEE permitiu: a modelagem e instanciação de *templates* de processo através de uma linguagem visual (apesar da modelagem de *templates* de processo não ser um requisito para o Nível G); a adaptação manual do processo instanciado para o contexto dos projetos, antes do início da execução; adaptações manuais posteriores, como a modificação do fluxo de execução dos processos mediante eventos significativos (atrasos, mudanças nos requisitos, entre outros); monitoração dos projetos por meio da geração de relatórios; visualização do modelo de processo em execução e do estado das atividades possibilitando maior percepção sobre o processo sendo executado.

Os principais pontos positivos identificados pelos usuários do WebAPSEE nesses dois relatos são:

- Definição dos processos (tarefas, prazos, agentes, recursos, etc.) através de uma linguagem visual;
- Funcionalidades que permitem criar *templates* de processos de desenvolvimento de software e instanciá-los nos projetos da organização;
- Possibilidade de modificação do fluxo de execução dos processos mediante eventos significativos (atrasos, mudanças nos requisitos, entre outros);
- Apoio e orientação para execução das tarefas através do ambiente;
- Acesso as atividades diárias e visualização do modelo de processo em execução possibilitando maior percepção (*awareness*) sobre o processo.

5. Trabalhos Relacionados

Esta seção apresenta a análise comparativa de quatro ferramentas (sendo uma delas a descrita nesse artigo) utilizadas para reutilização de processos, tendo como critério analisar duas ferramentas amplamente utilizadas pela indústria de software e duas ferramentas bastante difundidas no meio acadêmico.

5.1. *Rational Team Concert*

A ferramenta *Rational Team Concert* (RTC) [IBM 2011] foi desenvolvida pela IBM para se integrar à plataforma Jazz, uma arquitetura para gerência e automação do ciclo de vida de software baseada na colaboração de ferramentas. Dentro da plataforma Jazz, o RTC fornece funcionalidades referentes à definição, adaptação e reutilização de processos. No contexto da plataforma Jazz, um *template* de processo define as funções dos papéis, suas permissões para execução de operações dentro da ferramenta - como alteração do estado de um item de trabalho - fluxos de trabalho, iterações e condições prévias de operações.

Durante a criação de um novo projeto, um *template* de processo pode ser recuperado do repositório e todas as informações podem ser adaptadas para o contexto específico. A modelagem dos processos pode ser realizada diretamente em arquivos *Extensible Markup Language* (XML) ou via manipulação de itens dispostos hierarquicamente no formato de árvores. Ao contrário do WebAPSEE, pode-se ainda definir níveis mais detalhados de adaptação para atender às necessidades das diferentes equipes e áreas dentro do projeto. Em cada nível dentro da hierarquia de adaptação, o processo-pai pode definir quanta flexibilidade seus filhos têm para sobreporem ou modificarem o processo para suas necessidades [IBM 2011]. Entretanto, não existe um mecanismo semi-automatizado para a adaptação de processos.

A execução do processo pode ser controlada através do mecanismo de condições prévias, que reforçam as regras do processo. Por exemplo, é possível definir que a realização de uma atividade implica na geração de determinados artefatos. Também é possível acompanhar os processos através de barras de progresso, que mostram como está o andamento do processo.

A ferramenta RTC fornece a possibilidade de generalização de um processo de um projeto em um *template* de processo, através do acionamento da funcionalidade *Generate Process Template*. O *template* gerado pode então ser compartilhado para posterior reutilização em outros projetos. Além de permitir a modelagem de *templates* personalizados, o RTC fornece diversos *templates* de processo predefinidos, como o Scrum, o OpenUp e o Cloudburst *Sample*. *Templates* de processo adicionais também podem ser recuperados a partir do endereço jazz.net.

5.2. Team Foundation Server

A ferramenta *Team Foundation Server* (TFS) [Microsoft 2011] é uma solução desenvolvida pela Microsoft para integrar a plataforma Visual Studio ALM (*Application Lifecycle Management*), que, assim como o Jazz, consiste na colaboração de ferramentas para gerenciar e automatizar o ciclo de vida de desenvolvimento de software.

A ferramenta armazena *templates* de processo em um repositório. Os dois *templates* instalados por padrão em conjunto com a ferramenta são o MSF (Microsoft *Solutions Framework*) *for Agile* e o MSF *for* CMMI (*Capability Maturity Model Integration*). Também é possível a modelagem e o armazenamento de *templates* de processo próprios. Ao criar um novo projeto, um *template* armazenado no repositório do TFS deve ser selecionado, recuperado e adaptado para atender às necessidades específicas do projeto. O usuário pode estabelecer o *template* que será selecionado por padrão durante a criação de novos projetos.

5.3. ODE

O ambiente ODE (*Ontology based software Development Environment*) [Bertollo e Falbo 2003] é um PSEE baseado em ontologias. Esse ambiente segue um modelo de definição de processos baseado em níveis (processo padrão, processo especializado e processo de projeto).

Um processo padrão define os elementos básicos que compõem todos os processos da organização. Pode-se modificar um processo padrão para levar em consideração aspectos tecnológicos ou domínios de aplicação, gerando processos especializados. A partir dos processos especializados ou dos processos padrão, processos de projetos são então definidos.

Durante a definição de processos padrão/especializado existe um certo grau de automação, pois alguns componentes de processo são sugeridos para compor o processo sendo definido, com base no repositório de conhecimento da ferramenta. Entretanto, não é possível a seleção e recuperação automática do processo padrão/especializado durante a adaptação de um processo para o contexto de um projeto, como ocorre com o mecanismo CBR do WebAPSEE.

5.4. Análise Comparativa

Os critérios utilizados para realizar a comparação entre as ferramentas para reutilização de processos foram levantados em [Reis 2002], baseados em trabalhos de outros autores especialistas na área. A Tabela 2 apresenta as ferramentas e os critérios que cada uma atende.

Tabela 2 – Comparativo de Ferramentas para Reutilização de Processos

Critério / Ferramenta	Modelagem de Processos Abstratos	Representação Gráfica de Processos	Recuperação de Processos	Adaptação de Processos	Generalização de Processos Executados	Execução de Processos
WebAPSEE	X	X	X	X	X	X
RTC	X		X	X	X	X
TFS	X		X	X		X
ODE	X		X	X		X

É importante destacar que a proposta do WebAPSEE fornece apoio a todos os critérios em um único ambiente. Além disso, as características específicas de um projeto são levadas em consideração tanto na recuperação quanto na adaptação de processos no momento em que tais operações são realizadas (*on-the-fly*), o que não é implementado nas demais

ferramentas. O mecanismo de CBR e o sistema de regras *Fuzzy* também são relevantes por utilizarem técnicas de inteligência artificial em tomada de decisões durante o ciclo de reutilização.

6. Considerações Finais

Este artigo teve como foco apresentar como o ambiente WebAPSEE atende a cada uma das etapas do ciclo de vida para reutilização de processos de software.

Um ponto fraco do mecanismo semi-automatizado de adaptação de processos é a dificuldade em se modelar o conjunto adequado de regras *Fuzzy* necessárias para o seu funcionamento. Essa dificuldade se deve ao grande número de variáveis de entradas consideradas para cada regra e à necessidade de modelar uma grande quantidade de regras para se ter uma abrangência suficiente dos possíveis contextos de desenvolvimento de software. Esses fatores podem tornar a especificação das regras uma atividade imprecisa e bastante trabalhosa.

Um trabalho futuro em relação à essa funcionalidade é a avaliação da quantidade mínima de regras necessárias para garantir um nível adequado de abrangência e a implantação de melhorias de usabilidade para que a funcionalidade possa ser utilizada e avaliada em um caso prático de adaptação de processo.

Agradecimentos

Os autores agradecem a Coordenação de Aperfeiçoamento de Pessoal de Nível Superior – CAPES – pelas bolsas de mestrado para o desenvolvimento deste trabalho, e a Fundação de Amparo à Pesquisa do Estado do Pará – FAPESPA – pelo apoio ao projeto ao qual esta ferramenta está inserida.

Referências

Barreto, A., Murta, L. e Rocha, A. R. C. (2009) “Componentizando Processos Legados de Software Visando a Reutilização de Processos”. In: VIII Simpósio Brasileiro de Qualidade de Software (SBQS), Ouro Preto, Brasil.

Bergmann, R., Althoff, K.-D., Breen, S., Göker, M., Manago, M., Traphöner, R. e Wess, S. (1999) “Developing Industrial Case-Based Reasoning Applications”, Berlin: Springer-Verlag. (Lecture Notes in Artificial Intelligence, v. 1612).

Bertollo, G. e Falbo, R. A. (2003) “Apoio Automatizado à Definição de Processos de Software em Níveis”. In: II Simpósio Brasileiro de Qualidade de Software (SBQS), Fortaleza, Brasil.

Borges, L. M. S. e Falbo, R. A. (2001) “Gerência de Conhecimento sobre Processos de Software”. In: XV Simpósio Brasileiro de Engenharia de Software (SBES), Rio de Janeiro, Brasil.

Costa, A. J. S. (2010) “Um Mecanismo de Adaptação de Processos de Software”. Dissertação de Mestrado. PPGCC/UFPA, Belém, Brasil.

Costa, A. J. S. e Sales, E. O. (2007) “Uma Proposta para Reutilização de Processos de Software para o Ambiente WebAPSEE”. Trabalho de Conclusão, Bacharelado em Ciência da Computação, UFPA, Belém, Brasil.

Costa, A. J. S., Sales, E. O., Reis, R. Q. e Lima Reis, C. A. (2007) “Apoio a Reutilização de Processos de Software através de Templates e Versões”. In: VI Simpósio Brasileiro de Qualidade de Software (SBQS), Porto de Galinhas, Brasil.

Derniame, J., Kaba, B. e Wastell, D. (1999) "Software Process: Principles, Methodology and Technology". Berlin: Springer-Verlag. (Lecture Notes in Computer Science, v. 1500).

Feiler, P. e Humphrey, W. (1993) "Software Process Development and Enactment: Concepts and Definitions". In: 2nd International Conference on the Software Process (ICSP), Berlin, Germany.

Frakes, W.B. e Kang, K. (2005) "Software Reuse Research: Status and Future", IEEE Transactions on Software Engineering, 31(7), July, p. 529-536.

França, B. B. N. de, Sales, E. O., Lima Reis, Carla A. e Reis, R. Q. (2009) "Utilização do Ambiente WebAPSEE na implantação do nível G do MPS.BR no CTIC-UFPA". In: VIII Simpósio Brasileiro de Qualidade de Software (SBQS), Ouro Preto, Brasil.

Fuggetta, A. (2000) "Software Process: A Roadmap". In: 22nd International Conference on on Software Engineering (ICSE), Proceedings of the Conference on The Future of Software Engineering, pp. 25-34., New York: ACM Press.

Greer, D. e Conradi, R. (2009) "Software project initiation and planning – an empirical study", IET Software, 3(5), p. 356–368.

IBM. (2011) "Rational Team Concert", http://publib.boulder.ibm.com/infocenter/rtc/v2r0m0/index.jsp, Março.

ISO/IEC 15504 (2004) "Information Technology – Software Process Assessment".

Jørgensen, H. e Carlsen, S. (2001). "Writings in Process Knowledge Management of Knowledge Captured by Process Models". Technical report, SINTEF Telecom and Informatics, Oslo STF40 A00011, ISBN 82-14-01928-1.

Kellner, M. I. (1996) "Connecting Reusable Software Process Elements and Components". In: 10th International Software Process Workshop, pp. 8-11, Dijon, France.

Krueger, C. W. (1992) "Software Reuse", ACM Computing Surveys, 24(2), p. 131-184.

Lima Reis, C. A. (2003) "Uma Abordagem Flexível para Execução de Processos de Software Evolutivos". Tese de Doutorado. PPGC/UFRGS, Porto Alegre, Brasil.

Lima Reis, A. M., Costa, A. J. S., França, B. B. N., Reis, C. A. L. e Reis, R. Q. (2006) "Gerência Flexível de Processos de Software com o Ambiente WebAPSEE", In: XX Simpósio Brasileiro de Engenharia de Software (SBES), Florianópolis, Brasil.

Lima Reis, C. A. e Reis, R. Q. (2007) "Laboratório de Engenharia de Software e Inteligência Artificial: Construção do ambiente WebAPSEE", ProQuality (UFLA), 3, p. 43-48.

Maia, A., Freitas, A. e Nunes, D. (2005) "Um Modelo para Auxiliar a Adaptação de Processos de Software". In: IV Congresso Brasileiro de Computação, Itajaí. Brasil.

Microsoft (2011). "Microsoft Visual Studio Team Foundation Server", http://www.microsoft.com/business/smb/pt-br/servidores-e-ferramentas/visual-studio.mspx

Osterweil, L. (1987) "Software Process Are Software Too". In: 9th International Conference on Software Engineering (ICSE), Monterey, Estados Unidos, p. 2-13.

Paxiúba, C. M. C., Pereira, M., Reis, R. Q. e Lima Reis, C. A. (2007) "Acompanhamento e Avaliação de Projetos através da Monitoração de Eventos em um Ambiente de Gestão de Processos de Software". In: VI Simpósio Brasileiro de Qualidade de Software (SBQS), Porto de Galinhas, Brasil.

Pressman, R. S. (2006) "Software Engineering: A Practitioner's Approach", 6th Edition, New York: McGraw-Hill, 2006.

Reis, R. Q. (2002) "APSEE-Reuse: Um Meta-modelo para Apoiar a Reutilização de Processos de Software". Tese de Doutorado. PPGC/UFRGS, Porto Alegre, Brasil.

Ribó, J. e Franch, X. (2002) “Supporting Process Reuse in PROMENADE”. Research Report, No. LSI-02-14-R. Barcelona: Departament de Llenguatges i Sistemes Informàtics, Universitat Politécnica de Catalunya.

Sales, E., Freitas, S. e Reis, R. (2006) “Uma Ferramenta para Recuperação de Modelos de Processo de Software Reutilizáveis”. In: XX Simpósio Brasileiro de Engenharia de Software – Sessão de Ferramentas (SBES), Florianópolis, Brasil.

Sales, E., Reis, C. A. L., Reis, R. Q., Nascimento, L. e Farias Júnior, S. (2010) “Uso do Ambiente WebAPSEE na Implementação do Nível G do MPS.BR na Equilibrium Web”. In: VI Workshop Anual do MPS (WAMPS), Campinas, Brasil.

Santos, V. A. (2009) “Aprendizado Organizacional e Melhora Contínua de Processos de Software Através do Reuso de Processos de Software”. Dissertação de Mestrado. Universidade Estadual do Ceará, Fortaleza, Brasil.

SOFTEX (2009) “MPS.BR – Melhoria de Processo do Software Brasileiro – Guia de Implementação: Parte 3”, versão 2009.

Xu, P. e Ramesh, B. (2008) “Using Process Tailoring to Manage Software Development Challenges”, IT Professional, 10(4), p 39-45. IEEE Educational Activities Department Piscataway, Estados Unidos.

Zadeh, L.A. (1989) "Knowledge representation in fuzzy logic", IEEE Transactions on Knowledge and Data Engineering, 1(1), pp. 89-100, IEEE Educational Activities Department Piscataway, NJ, USA.