

Auxílio a Alocação de Pessoas em Projetos de Software Através de Políticas*

Marcelo A. Silva^{1,2}, Carla A. Lima Reis¹, Rodrigo Quitês Reis¹

¹Laboratório de Engenharia de Software – Universidade Federal do Pará (UFPA)
Belém – PA – Brasil

²Programa de Pós-Graduação em Ciência da Computação – Universidade Federal do Pará (UFPA) – Belém – PA – Brasil

marcelo.silva@webapsee.com, clima@ufpa.br, quitês@computer.org

Abstract. *This paper describes a tool for software process personnel allocation which is based on user defined policies. The proposed tool is integrated to an open source software process environment in order to retrieve both static and dynamic attributes from enacting process models and the software development organization. Finally, this paper also discusses the tool functionalities and feasibility within the context of a case study.*

Resumo. *Este artigo apresenta um mecanismo de apoio à alocação de pessoas em processos de software através da interpretação de políticas definidas pelo usuário. O mecanismo proposto foi integrado a um software livre para gestão de processos de software e se vale desta integração para obter informações estáticas e dinâmicas acerca dos modelos de processo e da organização de desenvolvimento de software usuária. O artigo apresenta a proposta do mecanismo e também um estudo de caso para avaliar a exequibilidade do modelo e da ferramenta proposta.*

1. Introdução

Ambientes de Engenharia de Software Centrados em Processo ou PSEEs (*Process-Centered Software Engineering Environments*) têm como objetivo fundamental auxiliar a modelagem e execução de processos de desenvolvimento de software. Para isso, é necessário apoio à especificação precisa das informações referentes tanto à organização que desenvolve o software, quanto aos elementos referentes à modelagem e execução de processos de desenvolvimento de software [Osterweil, 1999].

O desenvolvimento de PSEEs requer a integração de diversas funcionalidades voltadas para: auxílio à modelagem de processos segundo múltiplas perspectivas; auxílio à monitoração da execução de processos; simulação de processos; distribuição de informações para apoio às novas estruturas de terceirização do mercado; integração do ambiente com outras ferramentas de apoio ao desenvolvimento de software, e também o **auxílio à alocação de pessoas** e recursos de apoio nos projetos de software.

De acordo com Fuggetta (2000), PSEEs também devem apoiar a tomada de decisão no decorrer da execução do processo. Isso permite uma avaliação sistemática da qualidade do processo, dos seus componentes e dos produtos resultantes. Essa avaliação

* Trabalho apoiado pelo CNPq e ELETRONORTE.

é essencial para a implementação de estratégias que aumentem a maturidade do processo de desenvolvimento de software da organização.

A alocação de pessoas é um dos problemas que mais afeta o planejamento de um projeto e vem do fato de que processos de software lidam com atividades que demandam **pessoal especializado e limitado**. A especificação precisa das características de cada membro da equipe do projeto e sua disponibilidade representa um importante alicerce para sistematizar as práticas empregadas durante o desenvolvimento. Entretanto, a maioria dos PSEEs não permite que essa especificação seja detalhada e não provê auxílio automatizado à alocação dessas pessoas. Geralmente as ferramentas existentes permitem apenas que o gerente decida quem vai realizar qual tarefa e para isso fornecem alguns dados relevantes, tais como cargo e competências em geral. Porém, o gerente muitas vezes precisa elaborar um raciocínio adicional levando em consideração também o tipo de atividade, o estado atual do projeto (que pode estar atrasado), o grau das habilidades, carga de trabalho e experiência da pessoa a ser alocada, dentre outras variáveis importantes. Sabendo-se ainda da frequência com que ocorrem mudanças de planejamento no decorrer de um projeto, é também comum – e, irreal – exigir que um gerente saiba todas as informações sobre todas as pessoas e projetos para tomar decisões em curto prazo. Esse problema torna-se mais complexo quando constatamos que:

- Em algumas ferramentas e ambientes automatizados, as informações sobre as pessoas misturam-se com as informações sobre recursos de apoio e artefatos, tornando difícil analisar a alocação de pessoas em separado e reduzindo a precisão das informações;
- O critério de alocação para um caso pode ser generalizado para outros, mas não necessariamente serve para toda a organização e para todos os projetos e pessoas. Assim, é freqüentemente necessário que o gerente analise pessoalmente a situação e restrições correntes para cada atividade a ser instanciada;
- A organização passa a depender de estratégias de alocação de pessoas que apenas o gerente conhece e sabe aplicar. Portanto, o conhecimento obtido com a experiência gerencial não fica disponível nas ferramentas utilizadas, tampouco na organização.

Esse quadro constitui um importante obstáculo para a automatização, otimização e análise da alocação de pessoas em PSEEs. Inserido nesse contexto, este artigo apresenta **um mecanismo de auxílio à alocação de pessoas em projetos de software baseado em políticas de instanciação**, que definem detalhadamente as estratégias de alocação de pessoas de uma organização, que por sua vez podem ser aplicadas antes ou durante a execução do processo. O mecanismo foi denominado **WebAPSEE-Planner** e é um dos componentes de um PSEE chamado WebAPSEE, o que garante que a alocação ocorra integrada com as outras fases do processo. O artigo apresenta o funcionamento do mecanismo, destacando o mapeamento de estratégias reais de alocação de pessoas (obtidas através de estudos de caso) para processos de software modelados e executados no ambiente, buscando testar sua eficácia e capacidade de reutilização.

O artigo está organizado como segue: a seção 2 trata dos aspectos relacionados à gerência da alocação de pessoas em projetos de software; a seção 3 apresenta a proposta de extensão da arquitetura do ambiente WebAPSEE; a seção 4 descreve a linguagem de políticas de instanciação utilizada; a seção 5 apresenta um exemplo de instanciação no WebAPSEE-Planner; a seção 6 trata dos trabalhos relacionados; e a seção 7 apresenta as considerações finais.

2. Gerência da Alocação de Pessoas em Projetos de Software

O aumento da complexidade de projetos de desenvolvimento de software, aliado à necessidade de prazos de entrega cada vez menores, resultou em uma taxa de 74% de falha nos projetos de desenvolvimento de software das empresas americanas no ano de 2001, conforme pesquisa realizada pelo Standish Group (2001). Segundo essa pesquisa, as principais causas de falhas nos projetos estão associadas a dificuldades com os seguintes temas: apoio da alta gerência, envolvimento do usuário, experiência do gerente do projeto e definição clara das regras do negócio e escopo do projeto. O entendimento do processo de desenvolvimento de software sob um enfoque social no qual a interação entre os membros da equipe é a chave para o sucesso tem como foco os esforços do gerente de projeto na definição de quais tarefas devem ser executadas, por quem, quando e apoiado por qual ferramenta [Wastell et al. 1999].

As subseções seguintes tratam da alocação de pessoas no contexto do processo de desenvolvimento e dos elementos para tomada de decisão para alocação.

2.1. Alocação de Pessoas no Ciclo de Vida de Software

A gerência da alocação de pessoas pode ser vista como um processo que acompanha todo o ciclo de vida de um projeto de software, estando presente desde a fase de planejamento até a conclusão do projeto como um todo [Schneider 2003].

Considerando o aspecto central das pessoas no desenvolvimento de software, muitos estudos têm buscado entender melhor o comportamento dos profissionais, o que esperam, do que gostam e o que os faz trabalhar mais e melhor [Acuña 2003], [Agarwal 2000], [Lee 2002], [PMI 2004]. Uma vez que as pessoas representam um fator tão determinante nos processos de desenvolvimento de software, atribuir tarefas adequadas à capacidade de cada profissional é de grande importância para que os níveis de produtividade e qualidade desejados em cada atividade possam ser atingidos.

Schneider (2003) sugere, a partir da discussão sobre gerência de competências em uma Organização, um processo de gerência de alocação de recursos humanos que consiste em três etapas principais: (1) busca das competências necessárias – definição das competências em si, a busca dos profissionais da organização que as possuem e a seleção do(s) profissional(is) desejado(s); (2) alocação de pessoas – processos de alocação e liberação de profissionais; (3) desenvolvimento de competências – avaliação dos recursos humanos para posterior capacitação dos mesmos.

A norma NBR ISO 10006 [ABNT 2000] inclui os aspectos especificamente quantitativos da gerência de pessoas, e conseqüentemente do subconjunto de gerência da alocação de pessoas, no conjunto de processos relacionados aos recursos de forma geral. Esses aspectos visam criar um ambiente no qual a equipe possa contribuir efetiva e eficientemente para o projeto. Neste conjunto a norma identifica dois grandes processos: (1) planejamento da alocação de pessoas – correspondendo as atividades de identificação, estimação, programação e alocação de pessoas; e (2) controle da alocação de pessoas – comparando a utilização real e planejada das pessoas alocadas para tomar as devidas providências quando necessário.

De acordo com Barreto et al. (2005b), um aspecto importante que deve ser considerado na alocação de pessoas em um projeto de software refere-se à agregação de valor para o projeto e para a organização. Tendo ainda que o desenvolvimento de software envolve tempo, talento e dinheiro, o bom uso de cada recurso disponível torna-se um diferencial muito importante [Boehm 2000]. Como um dos recursos mais

importantes na agregação de valor para um projeto é justamente o recurso humano a ele alocado, o processo de alocação deve não apenas considerar os aspectos técnicos – características necessárias para executar as atividades e características possuídas pelos profissionais; mas também deve ser feito de modo a priorizar algum fator que agregue valor ao projeto. Dentre os fatores específicos de cada equipe, podem ser citados: a experiência, o custo e o tamanho da mesma [Barreto et al. 2005a; 2005b].

2.2. Elementos críticos para Alocação de Pessoas

O problema da alocação de pessoas no processo de software envolve, basicamente, um conjunto de profissionais e um conjunto de atividades. Cada profissional possui uma série de características, que o gerente de projeto precisa analisar ao decidir quem deve ser alocado para qual atividade. Algumas dessas características são [Barreto et al. 2005b], [Kroll 2003], [Lima Reis 2003], [Schnaider 2003]: (a) **conjunto de competências** como habilidades, conhecimentos, experiências, formação acadêmica, certificações, entre outras. Cada competência com uma determinada intensidade (por exemplo, grande conhecimento em modelagem UML, noções de linguagens de programação, entre outras); (b) **conjunto de cargos ou papéis** que cada profissional está apto a assumir em atividades de projetos de software (exemplos de cargos podem ser: Analista, Projetista, Testador, etc.). Cada cargo estando relacionado à experiência e às competências de cada profissional; (c) **grupos de trabalho** referentes às equipes que o profissional vier a formar para desenvolver atividades cooperativas; (d) **graus de afinidade com outros profissionais**, representando de forma indireta a produtividade do profissional em atividades que exijam trabalho cooperativo.

Além disso, diferentes pessoas possuem **experiência** em diferentes tipos de atividades (como análise, codificação ou teste) e podem ser caracterizados por diferentes métricas na Organização (como por exemplo, produtividade diária em pontos por função) [Barreto et al. 2005b], [Schnaider 2003]. Finalmente, um profissional possui um **custo** por hora de trabalho (ou outra unidade), além de uma **carga de trabalho** correspondente ao número de atividades nas que este profissional está alocado em um determinado período [Lima Reis 2003].

Cada atividade, por sua vez, possui: (a) **um conjunto de competências exigidas**: quais características um profissional precisa possuir para executar a atividade e, se relevante, em que intensidade mínima; (b) **conjunto de cargos requeridos**: onde cada cargo representa um profissional a ser alocado. Uma atividade indica ainda o **período** em que precisa ser realizada (informação esta, normalmente, contida no cronograma do projeto) e qual o seu **tipo** (se atividade de requisitos, codificação, etc.), conforme visto em [Barreto et al. 2005b], [Lima Reis 2003], [Schnaider 2003].

É de relevância ainda considerar o critério de **reutilização** das estratégias de alocação, para que dessa forma, boas práticas de alocação possam nortear futuras decisões dos gerentes de projeto [Basili et. al. 2001], [Schnaider 2003], [Staab 2001].

Um fator importante que tem sido negligenciado corresponde à **situação do processo de software no momento da alocação**, pois a mesma não pode ser definida *a priori*. E nas mudanças que surgem ocorrem as principais oportunidades de tomada de decisão do gerente. Uma razão para esse aspecto não estar sendo considerado em outras propostas pode ser causada pela dificuldade em integrar a alocação com as fases de modelagem e execução de um processo nos ambientes existentes.

A seção seguinte apresenta a proposta do mecanismo WebAPSEE-Planner e como a solução atende o problema da alocação de pessoas com os elementos citados.

3. WebAPSEE-Planner

WebAPSEE é um PSEE que tem como principal objetivo o aumento da flexibilidade e automação para a modelagem e execução de processos de software. O ambiente tem diversas funcionalidades em desenvolvimento para auxiliar a simulação, reutilização, execução descentralizada de processos, coleta de métricas, dentre outras, que visam aumentar a maturidade da organização usuária facilitando a implantação de processos de software [Lima 2006a; 2006b]. O ambiente permite modelagem visual de processos através da WebAPSEE-PML [Lima Reis 2003] e a execução flexível de processos, isto é, o ambiente apóia (entre outras funcionalidades) mudanças dinâmicas no processo, mantendo a consistência do mesmo a partir do uso de regras implementadas no sistema.

O **WebAPSEE-Planner** está sendo proposto como extensão ao WebAPSEE, para prover um mecanismo de auxílio à alocação de pessoas em projetos de software, tendo como base a interpretação de **políticas** definidas pelo usuário. A proposta original desse mecanismo está em [Lima Reis 2003], sendo que o atual consiste em implementação completa do mecanismo de forma integrada ao processo de software e avanços na experiência com estratégias reais de alocação [Silva 2007b].

Uma característica importante que influencia a solução proposta para instanciação é que o WebAPSEE permite a modelagem de processos abstratos que podem ser instanciados pelo gerente enquanto o processo executa (pois é possível executar processos incompletos). Assim, a proposta constitui um mecanismo de alocação de pessoas **integrado** a um ambiente que executa processos, permitindo que cada atividade seja instanciada assim que estiver pronta para executar (se essa for a opção do gerente), sendo que esta alocação pode ser decidida de forma autônoma pelo sistema, baseado nas estratégias de alocação habilitadas (ou políticas habilitadas).

O **WebAPSEE-Planner** utiliza os seguintes elementos: políticas de instanciação, características das pessoas e necessidades do processo. Com isso, produz: sugestões de instanciação (lista ordenada de sugestões) e modificação do processo (altera o processo automaticamente alocando as pessoas mais indicadas).

Para o funcionamento do mecanismo, o usuário deve definir e habilitar políticas de instanciação. Para definir políticas existe uma linguagem apropriada (detalhada na seção 4) enquanto que para habilitar políticas deve-se escolher onde a política se aplica: em uma atividade específica (através do editor visual de processos), em um processo (incluindo todas as suas atividades) ou na organização como um todo. A Figura 1 resume o funcionamento do WebAPSEE-Planner através de um diagrama usando a WebAPSEE-PML (Linguagem de Modelagem de Processos adotada pelo ambiente WebAPSEE onde elipses representam atividades e artefatos têm formato de papel, enquanto que a dependência entre atividades e a entrada e saída de artefatos é representada por setas).

Conforme pode ser visto na figura, na primeira etapa (“Geração de Sugestões”) são geradas as sugestões de instanciação, que levam em consideração o processo a ser instanciado, as políticas definidas, o perfil de agente requerido e as características dos agentes existentes na organização. Neste nível inicialmente são identificados os agentes compatíveis com a atividade, ou seja, quais agentes estão disponíveis e podem exercer o cargo e/ou habilidades requeridos. Em seguida, identifica-se qual política deve ser interpretada a fim de gerar a lista de sugestões (detalhes sobre o processo de

identificação da política a ser aplicada podem ser encontrados em [Silva 2007b]). Tendo identificado a política a ser aplicada, bem como o conjunto de agentes compatíveis, o WebAPSEE-Planner seleciona os agentes mais adequados (de acordo com os critérios de restrição e ordenação expressos na política aplicada) à atividade sendo instanciada. A segunda etapa (“Instanciação do Processo”) obtém como resultado um modelo de processo de software instanciado através do uso das sugestões obtidas da etapa anterior.

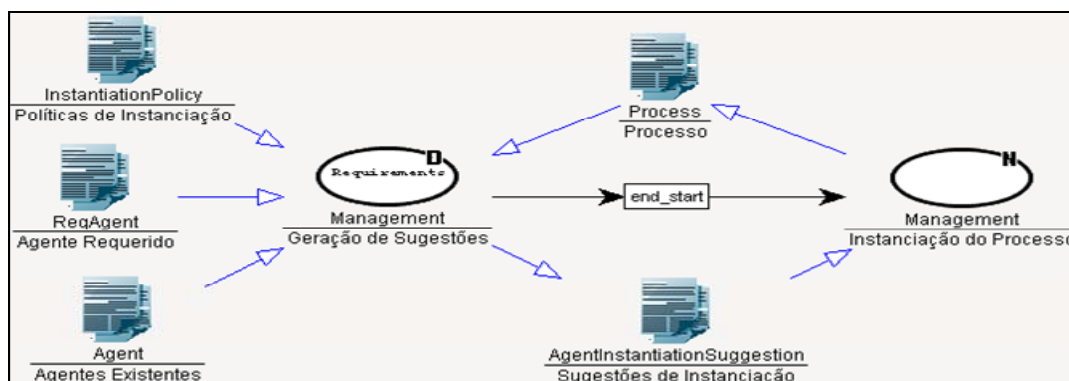


Figura 1 - Instanciação de um Processo através de Políticas [Lima Reis 2003]

O WebAPSEE-Planner pode ser ativado em dois momentos: (1) Durante a modelagem de uma atividade (por solicitação do usuário); (2) Durante a execução do processo (imediatamente antes da atividade começar): prevendo a necessidade de definição dos agentes, o mecanismo de execução ativa a instanciação automática.

4. Linguagem para definir Políticas de Instanciação

Segundo Feiler (1993), políticas de processos de software consistem em “princípios que conduzem o desenvolvimento e/ou a execução de processos de software”. No WebAPSEE, uma política pode ser informalmente descrita como um conjunto de propriedades que atuam na formação e execução de modelos de processos de software, representando um conhecimento gerencial genérico e reutilizável para diferentes contextos. Um objeto de política de instanciação é composto por uma especificação textual contendo critérios organizacionais definidos pelo usuário. Uma especificação detalhada e formal sobre a linguagem é apresentada em [Lima Reis 2003].

Uma política para a instanciação de pessoas é descrita principalmente por: **interface da política** (*PolInterface*), a qual especifica o tipo de objeto WebAPSEE cujas propriedades são tratadas na condição da política (podendo ser *Process* ou *Activity*); **O cargo ao qual a política se aplica** (*ApplyTo*); **A condição** (*Condition*) lógica que deve ser satisfeita para que os critérios da política sejam seguidos no momento da instanciação (as condições permitem que as situações correntes do processo e da organização sejam avaliadas para que a política seja aplicada); **critérios de restrição** (*RestrictBy*). Se a condição for satisfeita, os critérios de restrição permitem gerar o conjunto de pessoas compatíveis com a necessidade; **critério de ordenação** (*OrderBy*), que serve para ordenar a lista de pessoas de acordo com o critério escolhido. Caso não haja critério de ordenação, a sugestão será ordenada pela carga de trabalho (*workload*) das pessoas.

Os critérios de restrição que o usuário pode escolher incluem: a comparação com o valor de uma métrica cadastrada (por exemplo, escolher pessoas com métrica de produtividade maior que 100 pontos por função por mês); a comparação pelo custo de contratação (por exemplo, somente os que custam menos que R\$ 100 por hora), a carga

de trabalho (e.g., somente os que estão com menos de três atividades já alocadas no período da tarefa); pela experiência (pode-se consultar processos passados para retornar somente pessoas com experiência em determinada linguagem de programação).

Para ordenar o grupo resultante são escolhidos os critérios de carga de trabalho, custo, habilidade, afinidade com outras pessoas alocadas e ainda pelo valor de qualquer métrica existente na base de métricas do ambiente.

A linguagem para definição de condições (campo *Condition*) foi proposta por Reis (2002a; 2002b) e estabelece um conjunto de funções aplicáveis aos tipos *Activity*, *Artifact*, *Process*, *Resources*, *Agent*, *Groups*, *Roles* e *Connections*. Estas funções são utilizadas durante a modelagem e execução de processos e foram usadas aqui para consultar as informações sintáticas acerca do modelo de processo a ser instanciado. Algumas funções disponíveis² são mostradas na Figura 2.

Get_roles → set of roles	Is_active → bool
Is_performed_by_agents → bool	Get_successors → set of activities
Get_duration → real	Get_successors_oftype(type) → set of activities
Get_agents → set of agents	Get_artifact_connections_to(activity) → set of activities
Is_enacting → bool	Get_types_reqResources → set of string
Is_late_to_begin → bool	Get_input_artifacts → set of artifacts
Is_waiting → bool	Get_output_artifacts → set of artifacts

Figura 2 - Funções sobre Atividades para Políticas de Instanciação

5. Exemplo de Política de alocação no WebAPSEE-Planner

A tecnologia de processos de software, quando aplicada em contextos reais, permite testar sua viabilidade e adequação. Essa seção mostra uma estratégia capturada em uma empresa real sendo modelada e testada no WebAPSEE-Planner.

5.1. Identificação de Política de Instanciação de uma empresa CMM 2

Sabe-se que a qualidade do planejamento de um novo projeto está diretamente relacionada com a capacidade da organização em manter e utilizar o histórico de sucessos e fracassos de seus projetos já executados. Essa seção apresenta a definição de uma política de instanciação de pessoas a partir de um estudo etnográfico realizado por Silva (2007a) em uma empresa de desenvolvimento de software certificada em CMM nível 2 [SEI 2002] localizada no norte do Brasil. Inicialmente, Silva utilizou técnicas como observação não participativa direta (quando o pesquisador acompanha os acontecimentos e situações no local em que ocorrem, porém sem participar do contexto) e observação participativa (quando o pesquisador se torna um participante na cultura ou contexto que está sendo observado) para acompanhar e compreender o contexto de desenvolvimento de software na organização estudada. Em seguida, Silva pôde realizar um conjunto de entrevistas semi-estruturadas (com uso de um roteiro para as perguntas) com 10 profissionais da organização (distribuídos em 7 cargos diferentes nos processos da empresa), com tempo de serviço variando de 1 a 37 anos.

Em síntese, durante o seu estudo da empresa em questão, Silva (2007a) identificou, dentre outros resultados, as seguintes estratégias adotadas para a alocação de pessoas: (1) o gerente de uma atividade de Controle de Qualidade de Software (CQS)

² Em função do caráter internacional do projeto é utilizada denominação em inglês de seus componentes.

não pode estar alocado a nenhuma outra atividade no mesmo projeto; (2) com exceção do gerente do projeto, um agente (pessoa) pode ser alocado em mais de um projeto, porém exercendo cargos diferentes; (3) a equipe de um projeto é alocada de acordo com a carga de trabalho da mesma.

5.2 Mapeamento da Política identificada para a linguagem do WebAPSEE-Planner

Os aspectos identificados por Silva (2007a) apresentados na seção anterior foram utilizados na definição da política “Gerente_CQS”, apresentada na Tabela 1. Vale ressaltar que a política representa apenas as estratégias referentes à alocação de um Gerente de Controle de Qualidade de Software, os demais cargos existentes (por exemplo, gerente de projeto) na empresa demandam outras políticas de instanciação, que não são mostradas nesse texto.

Tabela 1 – Política para alocar Gerente de Controle de Qualidade de Software

Policy ID:	“Gerente_CQS”
Name:	“Gerente de Qualidade de Software”
Description:	“Se a atividade envolve o Controle da Qualidade do Software, selecionar os Gerentes de Qualidade de Software que não estejam alocados a nenhuma outra atividade desse processo e que não estejam exercendo nem o cargo de Gerente de Qualidade de Software nem o cargo de Gerente de Projeto em nenhum processo corrente. Dar preferência aos agentes com menor carga de trabalho.”
Interface:	a: Activity
ApplyTo:	Role “SoftwareQualityManager”
Condition:	a.get_type() sub_type_of “SoftwareQualityControl”
RestrictBy:	get_metric(“workload_same_process”, = , 0) get_metric(“workload_as_SoftwareQualityManager”, = , 0) get_metric(“workload_as_ProjectManager”, = , 0)
OrderBy:	low_workload()

De acordo com o campo *ApplyTo* da política, esta deve ser aplicada quando for necessário instanciar um Gerente de Qualidade de Software (ou *SoftwareQualityManager*). O campo *Condition* indica que esta política deve ser aplicada somente se a atividade envolver Controle de Qualidade de Software (ou *SoftwareQualityControl*). O campo *RestrictedBy* é composto por três restrições: a **primeira restrição** indica que o agente não pode estar alocado a nenhuma atividade no processo corrente, conforme o solicitado na estratégia (1); a **segunda restrição** indica que o agente não pode estar exercendo o cargo de Gerente CQS em nenhum projeto em execução da organização, ainda de acordo com a estratégia (1); a **terceira restrição** também referencia a estratégia (2), indicando que o agente não pode estar exercendo o cargo de Gerente de Projeto em nenhum outro projeto sendo executado na organização. Finalmente, o campo *OrderBy* referencia a estratégia (3), indicando que deve ser dada preferência aos agentes com maior disponibilidade (ou menor carga de trabalho).

5.3. Modelagem de Políticas no WebAPSEE-Planner

A Figura 3 representa a política de instanciação “Gerente_CQS” sendo modelada no WebAPSEE-Planner. A referida figura encontra-se dividida em quatro partes, cada parte destacando um dos campos editáveis da referida política: (a) *Condition*; (b) *ApplyTo*; (c) *RestrictedBy* e (d) *OrderBy*. Em (a) é destacado o botão de acesso à edição do campo em destaque (4) no caso, o campo *Condition*. Em (a) são destacados ainda os campos não editáveis nesta tela: (1) identificador; (2) nome e (3) descrição da política.

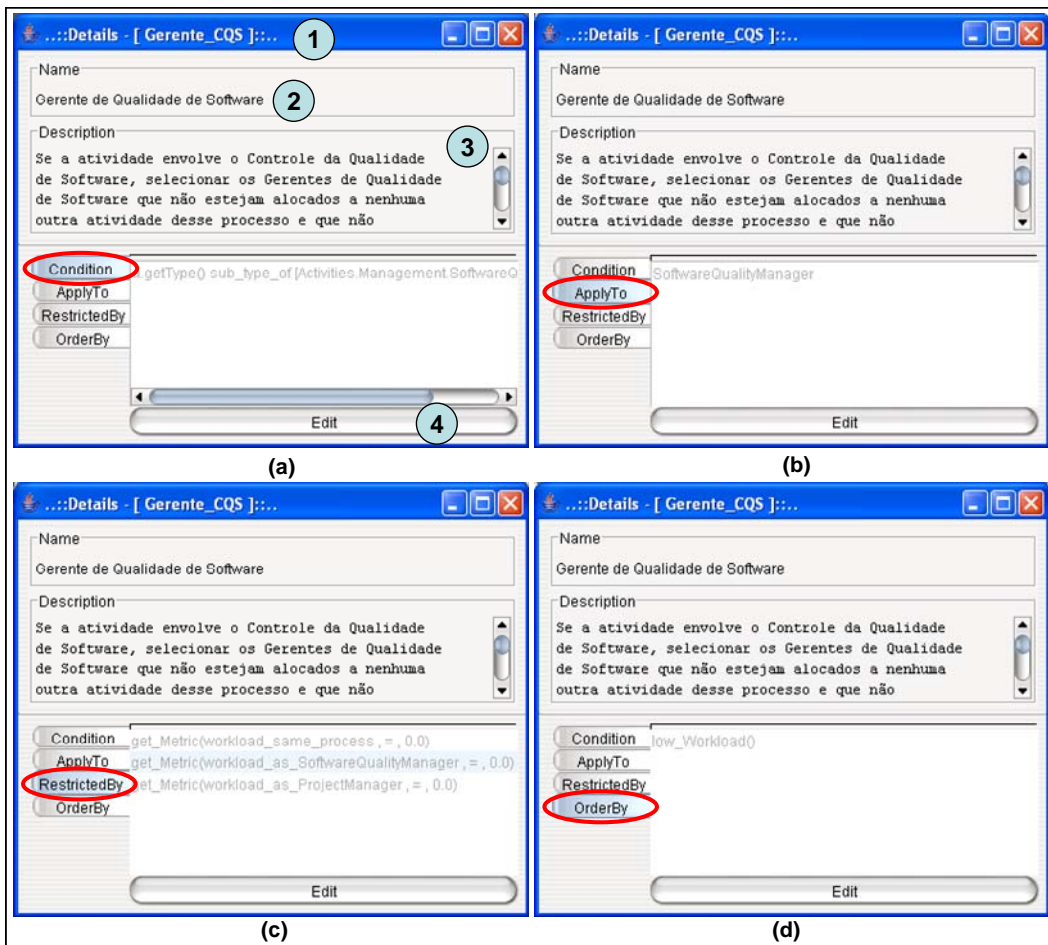


Figura 3 - Edição dos Detalhes da Política

A Figura 4 mostra a edição de uma condição através de um formulário, que guia a edição e valida cada passo, evitando erros de sintaxe e tirando do usuário a obrigação de conhecer a estrutura de uma condição. A referida figura apresenta uma condição sintaticamente incorreta (2). Neste caso, o operando selecionado (1) está correto (o operando “a.getType()” é válido de acordo com a linguagem de condições), todavia a associação “a.getType() intersection True” está incorreta sintaticamente.

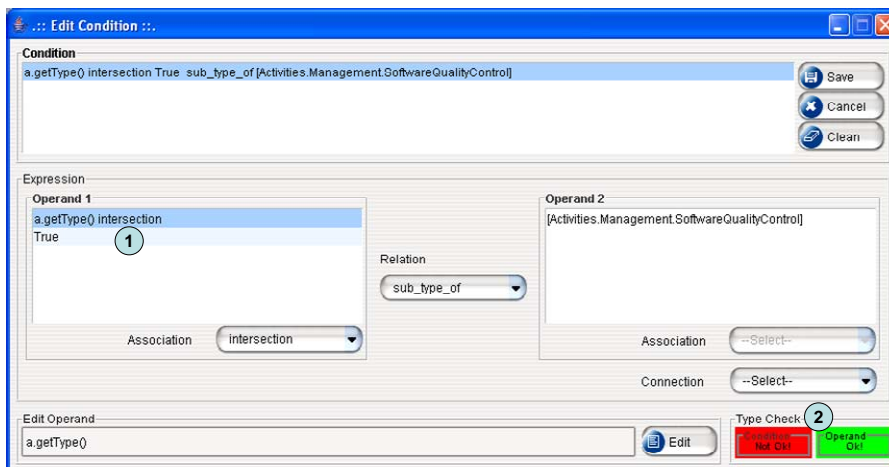


Figura 4 - Especificação de uma condição no WebAPSEE

5.4 Exemplo de Alocação usando a Política modelada

Esta subseção apresenta um exemplo de instanciação de pessoas apoiada pelo WebAPSEE-Planner, tomando como base todo o contexto apresentado nas subseções anteriores. Para ilustrar a instanciação, foram inseridas no ambiente WebAPSEE informações exemplo (não reais) sobre agentes, atividades e processos.

A situação atual dos agentes da organização é representada na Tabela 2, onde o termo “GQS” representa o cargo “Gerente de Qualidade de Software” (no WebAPSEE, este cargo está representado por “*SoftwareQualityManager*”).

Tabela 2 - Características dos Agentes da Organização

	Lúcia	Maria	Paulo	Pedro
Cargos	Programador/ Analista / GQS	ProjetistaDeSistema/ GQS	GerenteDeProjeto / Programador / GQS	Gerente / Analista
Carga de Trabalho (Total para o Período)	3	1	3	2
Carga de Trabalho por Cargo	Programador (2) / Analista (1)	ProjetistaDeSistema (1)	GerenteDeProjeto (1) / Programador (2)	Analista (2)
Carga de Trabalho no Processo Atual	0	0	1	0
Produtividade (PF ao Dia)	5	5	7	4
Atraso nas tarefas	25%	28%	17%	9%

A seguir é explicada cada linha da tabela em questão: **cargos** – representa os possíveis cargos que um agente possui; **carga de trabalho total** – indica em quantas atividades cada agente encontra-se alocado no momento da instanciação; **carga de trabalho no processo atual** – indica a carga de trabalho do agente, relativa ao processo atual no momento da instanciação; **carga de trabalho por cargos** – indica a carga de trabalho de cada agente, distribuída pelos cargos que o mesmo está exercendo, por exemplo, “CargoX (Y)” significa que o agente está exercendo o “CargoX” em “Y” atividades; **produtividade** – indica a produtividade diária do agente, medida em pontos por função; **atraso nas atividades** – indica o percentual de atraso das tarefas do agente. As duas últimas linhas são obtidas a partir da base de métricas do WebAPSEE.

A Figura 5 mostra parte de um processo modelado no WebAPSEE, onde se quer alocar um agente para a atividade “RevisarArtefatos”. Essa atividade requer um Gerente de Controle de Qualidade de Software para ser executada. Na figura, é exibida a lista de operações disponíveis para o componente referente ao perfil requerido pela atividade. Ao selecionar a opção “DefineInstance”, o usuário solicita ao mecanismo WebAPSEE-Planner as sugestões de instanciação para o referido perfil.

A primeira etapa para a geração de sugestões consiste em identificar os agentes que podem exercer o cargo requerido. De acordo com as características exibidas na Tabela 2, os agentes selecionados seriam: [Lúcia, Maria e Paulo].

A política aplicada (“Gerente_CQS”) é verificada e sua condição (campo *Condition*) é satisfeita por causa do tipo da atividade (representado logo abaixo da elipse com o nome de *SoftwareQualityControl*). Os critérios de restrição estabelecem que o agente não pode estar alocado a nenhuma atividade do processo corrente, nem exercendo o cargo de Gerente de Projeto (*ProjectManager*) ou Gerente de CQS (*SoftwareQualityManager*). Portanto a lista de agentes candidatos é reduzida para: [Lúcia, Maria]. Finalmente, o critério de ordenação define que a lista de sugestões deva ser ordenada pela menor carga de trabalho [*low_workload()*], o que resulta na lista [Maria, Lúcia], conforme destacado no lado direito da Figura 5.

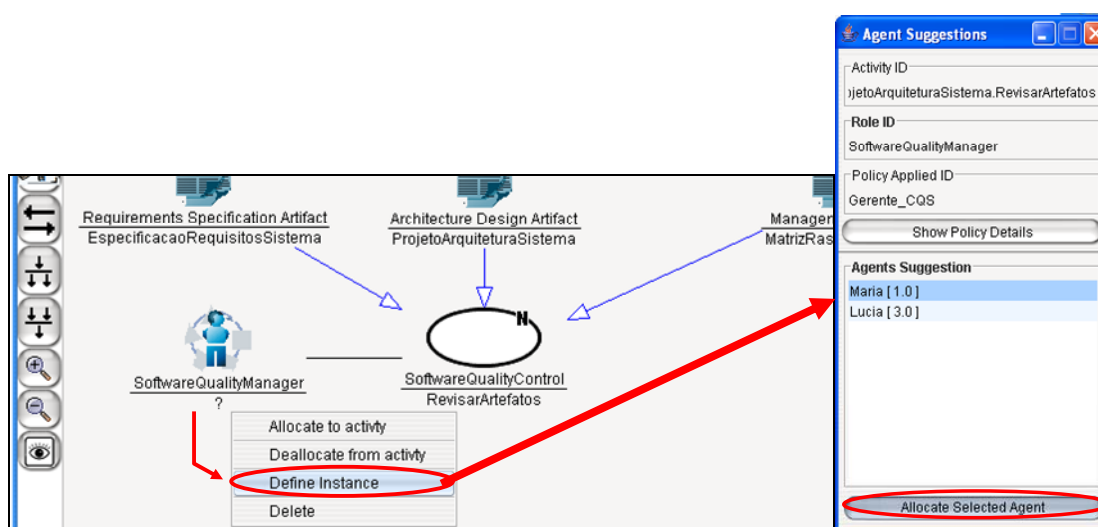


Figura 5 - Definindo Instância de Agente para Cargo Requerido

Para que o Gerente possa avaliar o impacto da sua escolha de alocação, o mecanismo disponibiliza ainda o valor utilizado para ordenar a lista de sugestões (na Figura 5, pode-se verificar que a carga de trabalho da agente Maria é 1 e da agente Lúcia é 3). Dessa forma, o Gerente pode verificar o critério de ordenação adotado e depois alocar o agente para a função requerida (o usuário não é obrigado a escolher o primeiro indicado). A Figura 6 mostra a agente “Maria” alocada na atividade e a alteração (feita de forma automática pelo WebAPSEE) do estado da atividade “RevisarArtefatos”, que passa a ter o estado *Ready* (essa mudança ocorreu porque a única pendência para que a atividade executasse era a alocação, mas outras dependências são verificadas). Da mesma forma, a referida atividade passa a constar na lista de tarefas da agente alocada, que pode ser visualizada através da agenda do desenvolvedor (aplicativo separado).

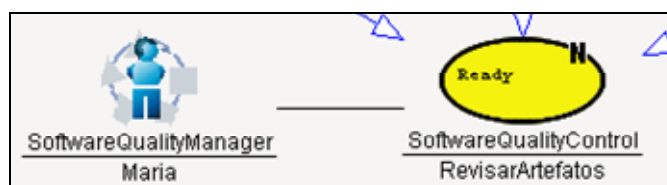


Figura 6 - Resultado da aplicação da política

A partir dessa alocação, qualquer outra atividade nessa organização que precisar de um *SoftwareQualityManager*, com certeza não terá como sugestão a Maria, pois a política garante que ela não pode atuar nesse cargo em outro projeto simultâneo. O exemplo mostrou a alocação de uma pessoa somente, mas como já mencionado, o mecanismo pode instanciar todos os agentes de um processo.

6. Trabalhos Relacionados

Uma deficiência da maioria das abordagens existentes é a não distinção entre pessoas e recursos [Podorozhny 1999]. Pessoas são normalmente tratadas da mesma forma que recursos de apoio, e são alocadas para atividades, enquanto que os recursos de apoio nem sempre possuem informação detalhada sobre seu estado, impedindo verificação automática e adoção de estratégias para alocação. Plekhanova (1999) propõe auxílio à alocação de pessoas através de programação linear usando informações (como competências) sobre os profissionais requeridos e disponíveis. Porém, além de não diferenciar pessoas de recursos, somente auxilia alocação em modelos de processos

previamente definidos, restringindo assim a sua adoção em ambientes de gerência de processos dinâmicos. Também podem ser citados os trabalhos de Lerner et. al. (2000), Podorozhny (1999) e Huang (1999) como trabalhos relacionados. Porém tais propostas não possibilitam a definição de políticas genéricas (reutilizáveis em diferentes contextos), sendo necessário definir explicitamente em qual atividade a política irá ser aplicada, além de não oferecer critérios para ordenação das sugestões.

Existem também abordagens nacionais interessantes sobre alocação de pessoas. Schnaider (2003) define um processo de planejamento da alocação de recursos humanos integrado à Estação Taba a partir do mecanismo *RHManager*. Entretanto, a ferramenta apóia a modelagem do processo de planejamento da alocação de recursos humanos, deixando a execução desse processo a cargo do gerente do projeto. Além disso, não parece ser possível especificar e manter as estratégias tomadas na alocação dos profissionais, o que faz com que o sucesso na gerência de alocação de pessoas dependa essencialmente da experiência e conhecimento do gerente do projeto.

Barreto (2005b) define a questão de alocação de pessoas como um problema de satisfação de restrições. O mecanismo proposto por Barreto utiliza diversos algoritmos de inteligência artificial para gerar sugestões de equipes que satisfaçam as restrições de uma atividade. Porém não foram observados: a possibilidade de avaliação da situação da organização no momento da instanciação, e um apoio para definição e reutilização de estratégias de alocação. Nesse sentido, é sempre necessário efetuar uma busca em todas as possíveis combinações de equipes que podem ser geradas na organização. Essa mesma limitação também é identificada em abordagens baseadas no problema de *job-shop* [Blazewicz, 1996]. Nessas abordagens é estabelecida uma busca por um escalonamento de recursos que minimize o tempo total de produção (*makespan*), ou nesse caso, o tempo total do processo de desenvolvimento.

Assim, a abordagem de políticas de instanciação de pessoas proposta neste artigo diferencia-se das outras pelos seguintes fatores: (1) políticas de instanciação de pessoas descrevem especificamente estratégias de alocação de profissionais em projetos de software (políticas para recursos de apoio são tratadas em outro trabalho [Lima Reis 2001]); (2) a linguagem de definição de políticas representa um formalismo compacto que possibilita a modelagem de estratégias genéricas de alocação de pessoas reutilizáveis em diferentes contextos através de construções definidas de forma independente; (3) a partir do momento em que as estratégias de alocação de pessoas estão representadas através de políticas, a gerência de recursos humanos em projetos de software torna-se menos dependente dos conhecimentos e experiências individuais do gerente de projeto; (4) a condição de uma política provê à presente abordagem a capacidade de avaliar a situação da organização no momento da instanciação do processo; (5) políticas de instanciação oferecem diferentes critérios de ordenação das sugestões de instanciação de pessoas, o que permite flexibilizar a escolha dependendo da estratégia da empresa.

7. Considerações Finais

Este trabalho apresentou um mecanismo denominado WebAPSEE-Planner para o auxílio à alocação de pessoas em projetos de software através de Políticas definidas pelo usuário. Para gerar sugestões de instanciação, o mecanismo WebAPSEE-Planner utiliza a informação expressa nas políticas habilitadas, além da informação referente às pessoas existentes e das pessoas requeridas por cada atividade. Detalhes sobre o meta-modelo que fornece a base para esse mecanismo podem ser encontrados em Silva (2007b) e Lima

Reis (2003). Este último trabalho citado definiu uma versão preliminar do mecanismo de políticas de instanciação que foi implantada no ambiente PROSOFT-APSEE (desenvolvido pelo Instituto de Informática da UFRGS). O presente artigo apresenta a nova proposta integrada ao WebAPSEE, e apresenta sua avaliação com uma política real capturada a partir de um estudo etnográfico em uma empresa de software.

A definição das políticas, aliada aos conhecimentos histórico, técnico e social da organização, provê ao WebAPSEE-Planner a capacidade de gerar sugestões de alocação (de pessoas) adequadas às necessidades de cada situação, de cada atividade. A linguagem de definição de Políticas permite ainda o estabelecimento de estratégias de alocação genéricas, facilitando assim a sua reutilização em diferentes modelos de processos. Além disso, as políticas representam uma extensão da WebAPSEE-PML, fornecendo uma estrutura capaz de armazenar informações ortogonais ao meta-modelo de um processo. Em adição a estas características, a estrutura compacta da linguagem utilizada pelo mecanismo proposto no presente trabalho introduz uma nova forma de inserir conhecimento gerencial em um PSEE. A abordagem declarativa adotada pela linguagem de Políticas é uma tentativa de obter níveis cada vez mais altos na automatização da gerência de processos de software. Essa linguagem possui ainda fundamentos matemáticos (por causa da semântica formal), conforme apresentado em Lima Reis (2003), que podem ser úteis para desenvolver futuras extensões e adaptações a outros ambientes de desenvolvimento de software.

A solução fornecida para o problema da gerência da alocação de pessoas em projetos de software busca contribuir com o planejamento de um projeto, oferecendo apoio no decorrer de todo o ciclo de vida de desenvolvimento e não apenas no início. Maiores níveis de maturidade podem ser obtidos com a aplicação dessa proposta, pois ela trabalha tanto com informações qualitativas (como os diferentes cargos que uma pessoa pode exercer), como informações quantitativas (como por exemplo a carga de trabalho de um determinado profissional) sobre os processos de uma empresa e consegue inferir situações a partir das políticas definidas pelo usuário para facilitar a tomada de decisão na otimização do processo. Trabalhos futuros incluem a experimentação de mais políticas de alocação em diversas empresas e também o aumento o poder de expressão dos critérios de restrição das políticas.

Referências

- ABNT. NBR ISO 10006. Gestão da Qualidade – Diretrizes para a Qualidade no Gerenciamento de Projetos. 2000. Associação Brasileira de Normas Técnicas, Rio de Janeiro, RJ, Brasil.
- ACUÑA, S. T.; JURISTO, N. Modelling Human Competencies in the Software Process. 2003. ProSim'03, Portland, Estados Unidos.
- AGARWAL, R.; FERRAT, T. W. Retention and the career motives of IT professionals. 2000. Proceedings of the 2000 ACM SIGCPR conf. on Computer personnel research.
- BARRETO, A. S.; BARROS, M. O.; WERNER, C. M. L. Staffing a Software Project: a Constraint Satisfaction Approach. 2005a. 7th International Workshop on Economics-Driven Software Engineering Research, St. Louis, Estados Unidos.
- BARRETO, A. S.; BARROS, M. O.; WERNER, C. M. L. Apoio à Alocação de Recursos Humanos em Projetos de Software: Uma Abordagem Baseada em Satisfação de Restrições. 2005b. IV Simpósio Brasileiro de Qualidade de Software - SBQS 2005. Porto Alegre, Brasil.

- BASILI, V., LINDVALL, M., COSTA, P. Implementing the Experience Factory concepts as a set of Experience Bases. 2001. In: Software Engineering and Knowledge Engineering - SEKE 2001, Buenos Aires, Argentina, June.
- BLAZEWICZ, J., DOMSCHKE, W., and PESCH, E. The job shop scheduling problem: Conventional and new solution techniques. *European Journal of Operational Research*, 93:1–33. 1996.
- BOEHM, B.; SULLIVAN, K. Software economics: a roadmap. 2000. In: The Future of Software Engineering, 22nd International Conference on Software Engineering..
- DINGSOYR, T.; ROYRVIK, E. Skills Management as Knowledge Technology in a Software Consultancy Company. 2001. In Proceedings of the Third international Workshop on Advances in Learning Software Organizations (September 12 - 13, 2001). K. Althoff, R. L. Feldmann, and W. Müller, Eds. Lecture Notes In Computer Science, vol. 2176. Springer-Verlag, London, 96-105.
- FEILER, P. H.; HUMPHREY, W. S. Software Process Development an Enactment: Concepts and Definitions. In: INTERNATIONAL CONFERENCE ON THE SOFTWARE PROCESS, 2., 1993, Berlin.
- FUGGETTA, A.; AMBRIOLA, V.; CONRADI, R. Assessing Process-centered Software Engineering Environments. 6., 1997. ACM Transactions on Software Engineering and Methodology. New York, USA.
- FUGGETTA, Alfonso. Software process: a roadmap. Proceedings of the Conference on The Future of Software Engineering, p.25-34, June 2000, Limerick, Ireland.
- HUANG, Y.; SHAN, M. Policies in a Resource Manager of Workflow Systems: Modeling, Enforcement and Management. 1999. International Conference on Data Engineering, 15. Proceedings... Sydney, Australia. March, 1999.
- KROLL, P.; KRUCHTEN, P. The Rational Unified Process Made Easy, Addison-Wesley, 2003.
- LEE, P. C. B. The social context of turnover among information technology professionals. 2002. Proceedings of the 2002 ACM SIGCPR conference on Computer personnel research.
- LERNER, B. S.; et al. Modeling and Managing Resource Utilization in Process, Workflow and Activity Coordination. 2000. Tech. Report. Dept. of C.S., Univ. of Massachusetts (UM-CS-2000-058) Aug., 2000.
- LIMA, A. et al. Gerência Flexível de Processos de Software com o Ambiente WebAPSEE. 2006a. 19º Simpósio Brasileiro de Engenharia de Software – Sessão de Ferramentas. Outubro, 2006.
- LIMA, A. et al. Análise do Ambiente WebAPSEE no atendimento aos requisitos de Gerência de Processos de Software. 2006b. XX SEMANA PARAENSE DE INFORMÁTICA (SEPAI/CTIC 2006). Belém, PA. Outubro, 2006.
- LIMA REIS, C. A. Uma Abordagem Flexível para Execução de Processos de Software Evolutivos. 2003. Tese de Doutorado – Programa de Pós-Graduação em Computação, Universidade Federal do Rio Grande do Sul, Porto Alegre.
- LIMA REIS, C.A. et al. A Abordagem APSEE para Modelagem e Gerência de Recursos em Ambientes de Processos de Software. 2001. Simpósio Brasileiro de Engenharia de Software. Florianópolis, Brasil, 2001.
- MONTONI, M. et al. Uma Abordagem de Garantia de Qualidade de Processos e Produtos de Software com Apoio de Gerência de Conhecimento na Estação TABA. V Simpósio Brasileiro de Qualidade de Software - SBQS06. Vila Velha, 2006.

- OSTERWEIL, L. J.; PODORZHNY, R. M.; LERNER, B. S. Modeling Resources for Activity Coordination and Scheduling. Proceedings of the 3rd International Conference on Coordination Models and Languages. Springer-Verlag, 1999.
- PLEKHANOVA, V. Capability and Compatibility Measurement in Software Process Improvement. 1999. Proceedings of the 2nd European Software Measurement Conference – FESMA'99. Amsterdam, The Netherlands, October, 1999
- PMI - Project Management Institute. A Guide to the Project Management Body of Knowledge - PMBOK Guide. 3rd ed. 2004.
- PODOROZHNY, R. et al. Modeling Resources for Activity Coordination and Scheduling. Intl. Conf. on Coordination Models and Languages. 3. Proceedings... April 1999. LNCS, v.1594.
- REIS, R. Q. APSEE-REUSE: Um Meta-modelo para apoiar a reutilização de processos de software, 2002a. Tese de Doutorado – Programa de Pós-Graduação em Computação, Universidade Federal do Rio Grande do Sul, Porto Alegre.
- REIS, R.Q. et al. Automatic Verification of Static Policies on Software Process Models. 2002b. In: WANG, Y.; BRYANT, A. (Eds.) Annals of Software Engineering. Vol. 14, Special volume on Process-based Software Engineering, p.197-234. 2002b. Boston, MA, USA: Kluwer Academic Publishers.
- ROZENBLIT, J.W.; KOCOUREK, C. Concepts for computer assisted engineering process management. Workshop on Engineering of Computer-Based Systems (ECBS '97), 1997, Monterey, CA, USA.
- SCHMIDT, K.; BANNON, L. Taking CSCW Seriously: Supporting Articulation Work. 1992. Computer Supported Cooperative Work (CSCW): An International Journal, vol. 1, 1992, no. 1-2, pp. 7-40. [DOI 10.1007/BF00752449].
- SCHNAIDER, L.R.C. Planejamento da alocação de recursos humanos em ambientes de desenvolvimento de software orientados à organização. 2003. Dissertação de Mestrado – Programa de Pós-Graduação em Engenharia de Sistemas e Computação, Universidade Federal do Rio de Janeiro, COPPE/UFRJ, Rio de Janeiro, RJ, Brasil.
- SEI-SOFTWARE ENGINEERING INSTITUTE. CMMi for Software Engineering. Staged Representation (CMU/SEI-02TR029). Pittsburg: Carnegie Mellon University, 2002.
- SILVA, A C. M. Um Estudo Etnográfico de Uma Organização Pública Certificada no Nível 2 do CMM. 2007a. Trabalho Individual. Programa de Pós-Graduação em Engenharia Elétrica. Universidade Federal do Pará. Belém.
- SILVA, M. A. WebAPSEE-Planner: Auxílio À Alocação De Pessoas Em Projetos De Software Através De Políticas. Trabalho de Conclusão de Curso – Bacharelado em Ciência da Computação – UFPA, 2007b.
- STAAB, S. Human Language Technologies for Knowledge Management. 2001. IEEE Intelligent Systems, vol. 16, n. 6 (November/December), pp. 84-94.
- WASTELL, D. et al. The Human Dimension of the Software Process. 1999. Software Process: Principles, Methodology, Technology, Springer.