

## Compartilhamento de Informações sobre Processos em Ambientes Descentralizados de Desenvolvimento de Software

Adailton Magalhães Lima, Rodrigo Quites Reis

Laboratório de Engenharia de Software (LABES)

Universidade Federal do Pará (UFPA)

Belém, Pará, Brasil

{adailton, quites}@webapsee.com

**Abstract.** *In decentralized software development the coordination and collaboration problems are hardened by the geographical and cultural distances among clients and suppliers. This paper describes a tool that provides a technological support to the problems of monitoring and coordination of decentralized software process development. This work analyses and illustrates how the Acquisition Model of the MPS.BR and some real problems extracted from the literature are supported by the proposed tool model.*

**Resumo.** *Em organizações que desenvolvem software de maneira descentralizada os problemas de coordenação e colaboração são agravados pelas distâncias geográfica e cultural dos envolvidos no processo. Este artigo descreve uma ferramenta que provê uma infra-estrutura tecnológica de apoio a monitoração e coordenação de processos descentralizados de desenvolvimento de software. Neste trabalho, as práticas de monitoração de processos do Modelo de Aquisição do MPS.BR e problemas reais coletados na literatura são analisados e relacionados às funcionalidades da ferramenta.*

### 1. Introdução

A coordenação e a colaboração são reconhecidamente aspectos essenciais para as empresas que desenvolvem software de maneira não co-localizada [Audy 2007]. No contexto de desenvolvimento de software, existem os Ambientes de Desenvolvimento de Software Centrados em Processos, que são ferramentas que permitem a modelagem e execução de processos de software e são genericamente denominados como PSEEs (*Process-centered Software Engineering Environments*) [Grhun 2002]. A finalidade principal destes ambientes é atender a requisitos organizacionais para auxiliar na coordenação das atividades relacionadas com o desenvolvimento de software. Somente a partir da metade da década de 1990, como por exemplo, os trabalhos de [Ben-Shaul 1998] e [Grundy 1998], começou a haver uma vinculação entre a oportunidade no uso e evolução da tecnologia dos PSEEs para fornecer apoio automatizado ao desenvolvimento não co-localizado de software.

Segundo Taylor (2003), descentralização é um termo diretamente relacionado com a dispersão do controle sobre as atividades no desenvolvimento de software. Pode-se afirmar assim que no desenvolvimento descentralizado não existe a obrigatoriedade de uma entidade centralizada para controle das atividades, e a coordenação entre as entidades envolvidas é resultado de decisões e acordos multilaterais. O modo contrário à descentralização é a centralização, onde existe a premissa da existência de uma entidade que centraliza as decisões e mecanismos de controle sobre o projeto. Já o termo distribuição está diretamente relacionado apenas com a dispersão física dos dados

(através de Sistemas Distribuídos [Coulouris 2005]) e as entidades envolvidas no desenvolvimento de software. Os dados compartilhados durante o projeto são considerados distribuídos em casos onde o processo de desenvolvimento envolve o envio de dados e informações entre diferentes localidades e sistemas.

Para apoio efetivo às atividades distribuídas e descentralizadas, Herbsleb (2007) considera que deve existir uma comunicação freqüente para colaboração e compartilhamento de informações sobre os processos das organizações envolvidas, de forma a permitir que o fluxo de informações entre as organizações evite repetição de erros. Segundo Wells (2001) e Herbsleb (2007), a privacidade de informações trocadas entre as diferentes organizações desenvolvedoras de software é essencial para a implantação de tais ferramentas em organizações reais.

Alinhado às políticas do Ministério de Ciência e Tecnologia, o Guia de Aquisição de Software [Softex 2008] está incluído no contexto do Modelo MPS.BR (Melhoria do Processo de Software Brasileiro) [Weber 2006] e define diretrizes que devem ser atendidas por organizações compradoras de software que desejam melhorar o processo de aquisição, em cenários diversificados – variando desde a encomenda de um produto até uma fábrica de software, sejam *in-house* ou externas. Para organizações fornecedoras de software o Guia de Aquisição serve como orientação para atender melhor seus clientes.

Inserido neste contexto e na literatura atual sobre DDS (Desenvolvimento Distribuído de Software), este trabalho apresenta uma ferramenta que representa uma extensão de um PSEE, que possui uma arquitetura centralizada para uma arquitetura descentralizada, com o objetivo de prover uma infra-estrutura tecnológica de apoio a problemas de monitoração e coordenação de processos descentralizados de desenvolvimento de software. Para apoiar os diferentes modelos organizacionais existentes em DDS (classificados como *Outsourcing* ou *Insourcing* e suas variantes [Audy 2007]), este trabalho define um contrato eletrônico como meio permitir flexibilidade na configuração de qual modelo é instanciado na organização. Relacionado às práticas e modelos na indústria, este trabalho relaciona o processo de monitoração do Modelo de Aquisição do MPS.BR e problemas reais coletados na literatura para analisar e ilustrar como problemas de DDS são tratados pela ferramenta proposta.

O texto a seguir apresenta a seguinte organização: a seção 2 apresenta uma descrição geral dos principais componentes e aplicabilidade do ambiente WebAPSEE; a seção 3 descreve em linhas gerais o processo de aquisição de software e a relação com DDS, bem como quais características deste processo são alvo deste trabalho; seção 4 descreve a abordagem de descentralização atendida pela ferramenta deste trabalho, descrevendo uma das funcionalidades em detalhe e apresentando a aplicabilidade prática da ferramenta baseando-se em problemas retirados da literatura sobre DDS; a seção 5 discute brevemente os trabalhos relacionados; por fim, a seção 6 apresenta as conclusões e trabalhos futuros.

## 2. Ambiente WebAPSEE

O ambiente WebAPSEE é um PSEE baseado em Software Livre, desenvolvido desde 2004 pelo Laboratório de Engenharia de Software<sup>1</sup> da UFPA [Lima 2006]. Através de ferramentas clientes (*Task Agenda* e *Manager Console* – ambos descritos a seguir), os

---

<sup>1</sup>Maiores informações disponíveis em <http://www.labes.ufpa.br>

usuários obtêm acesso aos dados relacionados com o modelo e execução dos processos nele gerenciados. O ambiente se propõe a permitir a definição e implantação de modelos de processos de software através de uma notação gráfica com semântica formalmente definida [Lima 2006].

### 2.1 Arquitetura

Como a grande maioria dos PSEEs (relacionados, por exemplo, nos estudos de [Gruhn 2002]), a versão atual do WebAPSEE apresenta uma arquitetura cliente-servidor, onde os processos são executados em um único ponto e a distribuição dos dados e serviços é fornecida a partir de chamadas remotas de procedimentos pelas ferramentas clientes.

Como características técnicas desvantajosas para a configuração centralizada podem-se citar: há ponto único de falha do sistema; escalabilidade é limitada pelo poder de processamento da máquina servidora; tempo de resposta final é limitado pelo grau de concorrência no servidor.

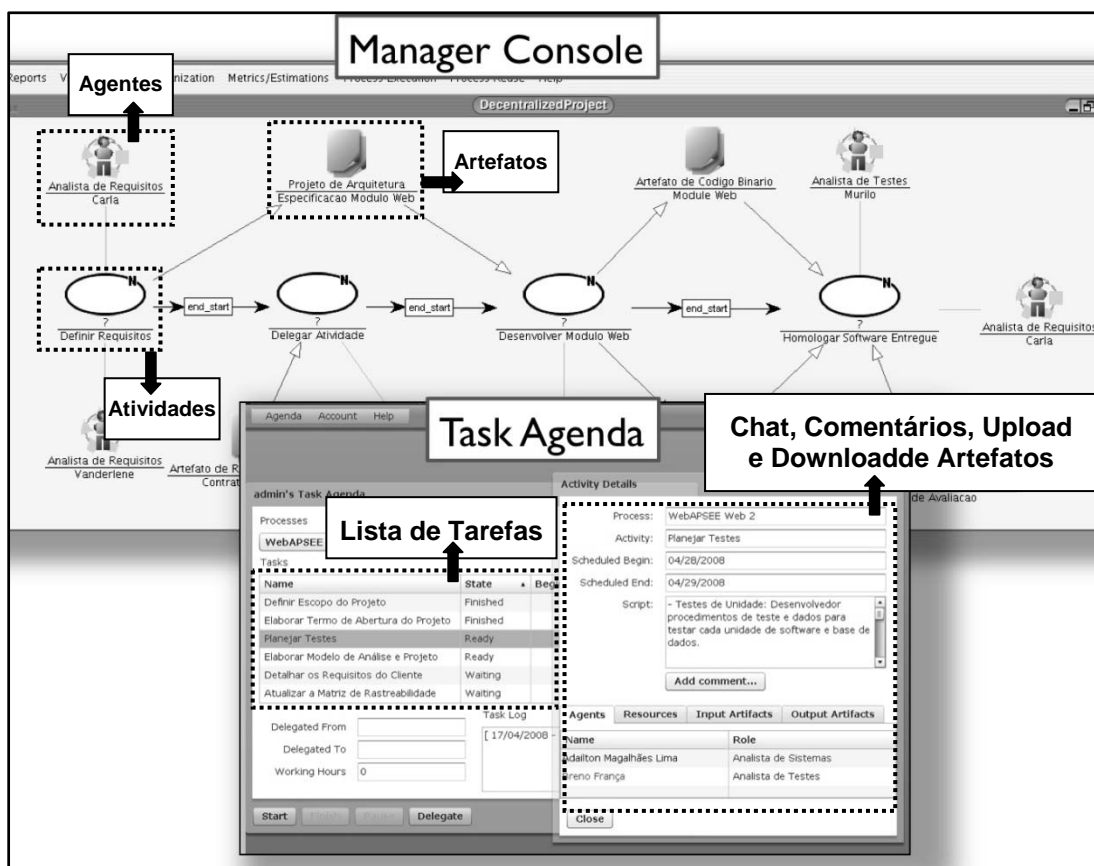


Figura 1 – Visão Geral das Ferramentas do Ambiente WebAPSEE

Além das desvantagens técnicas citadas, a arquitetura cliente-servidor possui desvantagens do ponto de vista organizacional para empresas. Se houver a necessidade de compartilhamento de informações a respeito dos processos de uma organização, o acesso aos dados aos interessados poderá abrir brechas para que informações sigilosas sobre os processos da organização sejam acessadas indevidamente, o que é totalmente desaprovado segundo [Wells 2001].

### 2.2 Manager Console

É utilizado um formalismo de modelagem que permite representação gráfica de processos, como ilustrado na parte superior da Figura 1. O ambiente WebAPSEE

permite a integração de vários serviços de gerência de processos, incluindo modelagem, execução, visualização, instanciação e resposta a eventos da execução.

Através do *Manager Console*, o gerente do processo de desenvolvimento de software pode: modelar processos, gerenciar execução de processos, visualizar relatórios do processo e gerenciar informações da organização (como artefatos, pessoas e recursos) através de um Editor Gráfico.

### 2.3 Task Agenda

Através da *Task Agenda*, o desenvolvedor visualiza os processos de software em execução nos quais está inserido como uma lista de tarefas a serem realizadas. Assim, o desenvolvedor interage com a agenda fornecendo *feedback* sobre o andamento dessas tarefas.

A Figura 1 contém na parte inferior a ilustração de uma tela de uso da ferramenta *Task Agenda*. A Agenda representa a visão de um desenvolvedor a partir das tarefas que lhe são atribuídas para um determinado processo.

### 2.4 Estratégia de Implantação de Melhoria de Processos

O ambiente WebAPSEE é parte de uma estratégia de implementação de melhoria de processos de software, descrita em detalhes em [Covre 2008]. O papel principal deste ambiente no contexto de melhoria de processos de software é a definição de um repositório comum para definição e acompanhamento dos processos no contexto de uma organização.

Segundo [Hillegersberg 2007], para que ferramentas atendam aos requisitos exigidos pelo contexto de DDS é necessário tratar da integração de processos entre diferentes organizações. Alinhado a esta necessidade relatada na literatura e aos modelos de maturidade de processos na aquisição de software (como MPS.BR e CMMI), este trabalho descreve uma extensão ao modelo cliente-servidor do ambiente WebAPSEE como apoio específico à etapa de monitoração em processos de aquisição de software, que inclui a integração dos processos organizacionais para alcançar os objetivos de monitoração e integração necessários para as ferramentas em DDS.

## 3. Monitoração de Aquisição de Software e DDS

A contratação de serviços de desenvolvimento de software, seja um produto fechado ou módulos específicos, está inserida no contexto de DDS quando alguma parte do processo de desenvolvimento está remotamente localizada. Seja para monitorar o desenvolvimento de um produto, seja para monitorar o desenvolvimento de um módulo que faz parte de um produto e processo desenvolvimento maior, a monitoração das atividades permite o acompanhamento do andamento das atividades através de medidas de tempo e revisão de artefatos envolvidos no processo [Hayes 2005].

Notadamente a monitoração é um processo crítico em projetos de DDS que envolvem *Outsourcing*, pois neste caso as diferenças organizacionais podem gerar conflitos de interesses para a integração de sistemas e acesso às informações sobre projetos [Wells 2001]. Em processo de *Insourcing* os aspectos citados anteriormente podem não ser tão críticos, mas mesmo assim ainda existe a necessidade de monitoração na medida em que as distâncias geográfica e cultural permitem diferentes ambientes de trabalho na mesma organização (por exemplo, matriz nos EUA e subsidiária na Índia).

Segundo [Sabherwal 2003], o tipo de monitoração que oferece menor custo variável durante o projeto é a **monitoração automática**, que é auxiliada pelo uso de

ferramentas que capturam automaticamente os dados sobre o andamento do projeto. Caso a monitoração ocorra de forma manual, mas seguindo algum processo ou metodologia é chamada de **monitoração formal**, e caso ocorra sem seguir nenhum processo ou metodologia é chamada de **monitoração ad hoc**. Por último, caso não haja nenhuma atividade de monitoração e a aquisição de dados sobre o andamento do projeto seja informal é chamada de **monitoração informal**. A Figura 2 destaca o processo de monitoração do fornecedor no contexto geral do Modelo de Aquisição do MPS.BR [Softex 2008]. Os quatro objetivos marcados com setas indicam onde são gerados artefatos que descrevam o *status* do andamento do processo. O *status* do andamento do processo é um documento tem a finalidade de registrar em uma determinada data a situação da aquisição no que diz respeito a prazo e requisitos atendidos.

Em trabalho recente, [Kuipers 2007] defende como solução aos problemas monitoração de projetos em *Outsourcing* uma abordagem baseada na monitoração da qualidade do software. Seu trabalho considera que através da monitoração direta das métricas relacionadas com o código fonte desenvolvido sejam gerados gráficos para análise e tomada de decisões. No contexto da Engenharia de Processos de Software, este trabalho considera a monitoração do progresso de atividades tanto pela monitoração de metas e planos quanto pela monitoração dos produtos desenvolvidos como um meio fundamental para que haja coordenação efetiva de atividades de desenvolvimento de software em DDS. Segundo [Ebert 2007], a coordenação baseada no andamento de atividades do processo de engenharia dos produtos permite um melhor controle e planejamento dos próximos passos do projeto.

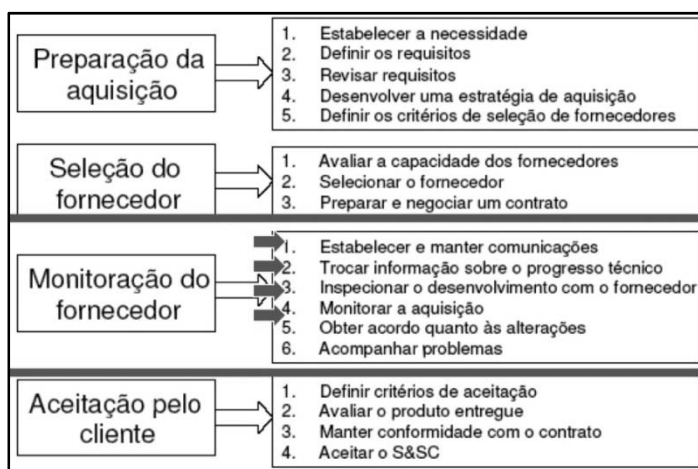


Figura 2 – Destaque à Necessidade de Monitoração do *Status* do Processo no Modelo de Aquisição do MPS.BR [adaptado de Softex (2008)]

#### 4. Descentralização de Processos de Software em um Ambiente Automatizado

Este trabalho estende as funcionalidades definidas em [Lima 2007], onde é definido um modelo geral baseado em contratos eletrônicos para interação entre cliente e fornecedor. O mecanismo de compartilhamento de contexto de processos de software em ambientes descentralizados é uma extensão ao trabalho anterior, que era constituído basicamente na troca de dados distribuídos sobre atividades e artefatos envolvidos para realização das atividades acordadas entre cliente e fornecedor. Como extensões providas por este trabalho têm-se: componentes para persistir mensagens trocadas; implementação de protocolo para controlar o recebimento *online* de notificações; definição da visualização

de contexto de processos, que inclui informações adicionais sobre as atividades acordadas entre cliente e fornecedor.

#### 4.1. Visão Geral do Trabalho

A Figura 3 representa em um diagrama de implantação a visão geral da ferramenta descrita neste trabalho. O componente **P2Process Server** atua como um *proxy* entre instâncias diferentes do ambiente WebAPSEE (componente **WebAPSEE Server**) que se comunicam através da Internet. As diferentes instâncias das ferramentas ilustradas na Figura 3 representam a ligação entre o ambiente de cliente e fornecedor proposta neste trabalho, que atua diretamente sobre o problema de integração de processos em ambientes de DDS citado em Prikladnicki (2007).

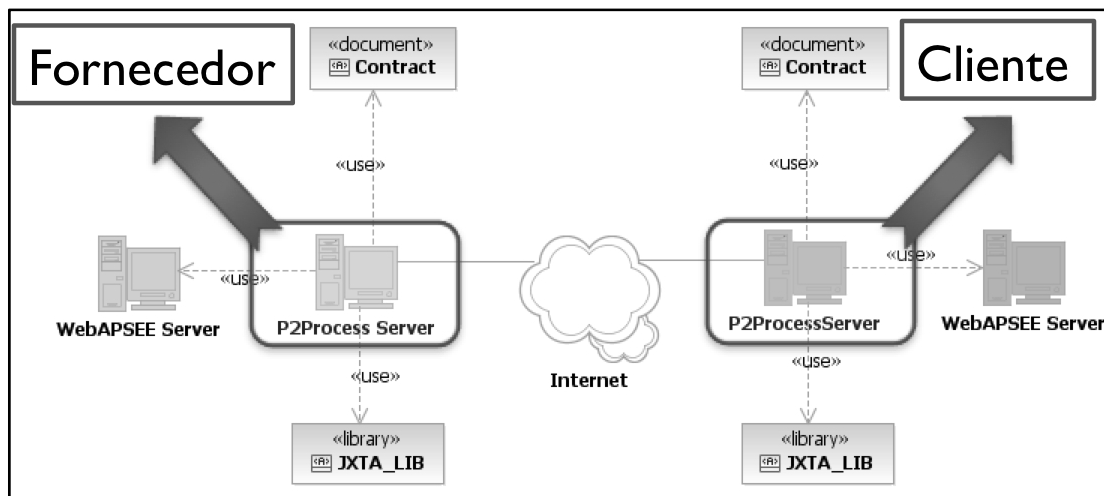


Figura 3 – Diagrama de Implantação ilustrando um caso ferramenta em uso

A comunicação entre os componentes remotos é realizada através da troca de mensagens que são enviadas quando os eventos relacionados à execução dos processos são gerados no ambiente WebAPSEE. Como uma vantagem arquitetural da abordagem adotada pode-se afirmar que se reduz os custos de desenvolvimento, propicia-se o reuso de conhecimento e projeto, além de prover alta extensibilidade para a aplicação ser utilizada em diferentes contextos [De Souza 2002].

#### 4.2. Contrato Eletrônico

Como parte da ferramenta proposta, o contrato eletrônico (artefato UML **Contract** na Figura 3) define quais as organizações fazem parte de um contexto de descentralização de processos e estabelece regras para controlar o acesso a informações sobre processos remotos (neste caso, o contrato atua como um controle de acesso para proteger informações organizacionais confidenciais sobre os processos das diferentes instâncias de ferramentas envolvidas).

Apesar da existência de um contrato ser mais usual em projetos que envolvem *Outsourcing*, implicitamente projetos que envolvem *Inourcing* possuem alguma definição contratual de como é a relação entre matriz e subsidiária [Audy 2007]. Neste trabalho existe uma versão eletrônica do contrato que permite a flexibilidade de modificações dinâmicas do conteúdo para adaptar-se a diferentes contextos organizacionais (de *Outsourcing* a *Inourcing*). A ferramenta deste trabalho possui regras de negócio que definem obrigatoriedade de modificações conjuntas no contrato (com a atualização em todos os envolvidos), o que pode estabelecer uma relação direta

com a confiança no uso da ferramenta na medida em que o comportamento anômalo de um dos envolvidos (violação do contrato) é detectado automaticamente.

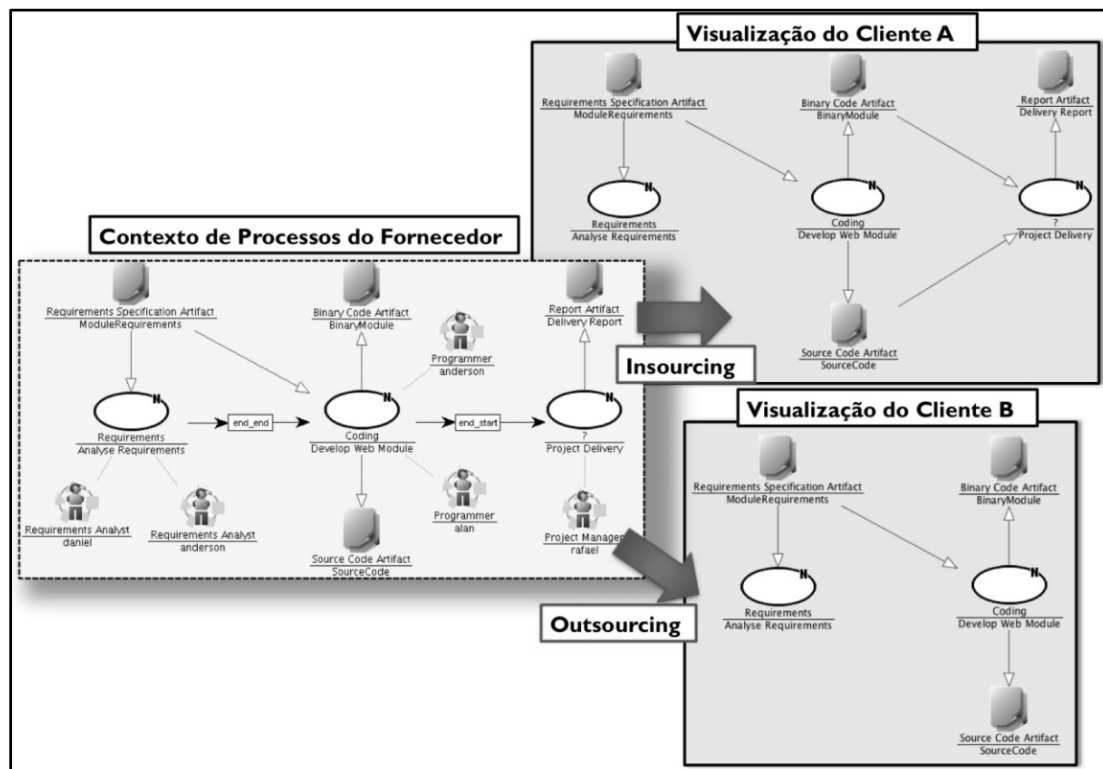


Figura 4 – Visualização do Contexto de Processos Remoto em diferentes Contextos Organizacionais

O “Contexto de Processo do Fornecedor” ilustrado na Figura 4 representa o modelo de processos do fornecedor que contém informações sobre as atividades relacionadas com a atividade descentrada pelo cliente (cujo nome é “Develop Web Module” e produz os artefatos “SourceCode” e “BinaryModule”) e as pessoas alocadas às atividades. Do ponto de vista do cliente, a atividade “Develop Web Module” irá produzir artefatos de software por ele requeridos. Do ponto de vista do fornecedor, o modelo “Contexto de Processo do Fornecedor” representa todos os passos necessários para entrega dos produtos, bem como a alocação das pessoas que efetivamente irão trabalhar para executar as tarefas de desenvolvimento.

A partir da configuração do contrato eletrônico, podem ser estabelecidas regras para, por exemplo, impedir ou habilitar a visualização e monitoração dos detalhes dos processos internos adotados pelos fornecedores. O controle é de granulosidade fina, o que permite que o contrato estabeleça de fato quais elementos do processo são propagados entre as organizações – por exemplo, detalhes sobre as atividades, artefatos (intermediários e finais) produzidos, além de detalhes sobre o estado do andamento do processo. A idéia geral da funcionalidade de compartilhamento de contexto de processos é que clientes possam requisitar uma visão sobre processos do fornecedor e visualizar apenas os componentes do processo permitidos no contrato. Com este mecanismo configurável, podem existir diferentes casos atendidos, dependendo da configuração mais restritiva ou aberta do contrato eletrônico. Por exemplo, podem-se ter casos de parceiros internos da organização (o caso de *Insourcing* ilustrado pela “Visualização do Cliente A” na Figura 4), onde as regras permitem que todas as atividades incluindo as remotas possam ser visualizadas pelo cliente. Outro caso é o de

parceiros externos da organização (o caso de *Outsourcing* pela “**Visualização do Cliente B**” ilustrada na Figura 4), onde o caráter mais restritivo do contrato é evidenciado pelo bloqueio da visualização de informações sobre a atividade “**Project Delivery**”.

### 4.3. Aplicabilidade Prática

Dentre os modelos organizacionais *Outsourcing* ou *Insourcing* (*Offshore* ou *Onshore*) [Audy 2007], diferentes problemas que podem ser tratados por ferramentas especializadas são relatados pela literatura [Prikladnicki 2007] [Ebert 2007] [Hillegersberg 2007]. A seguir quatro destes problemas citados na literatura são discutidos sob a ótica das funcionalidades da ferramenta descrita neste trabalho:

- a. **Definição Formal do Processo:** a definição formal do processo está relacionada com utilização de alguma linguagem para definir o modelo de processo de software, como a linguagem de modelagem utilizada no ambiente WebAPSEE [Lima 2006]. Na coordenação descentralizada de atividades este problema refere-se à definição de quais atividades serão compartilhadas entre as organizações. Neste trabalho ao definir no contrato quais os elementos de processo que são compartilháveis entre cliente e fornecedor há definição formal do processo na medida em que o contrato é tratado como um formalismo entre os envolvidos.
- b. **Visualização do Contexto do Projeto:** a visualização do contexto do projeto está diretamente relacionada com a requisição de informações sobre o processo remoto, para visualização do contexto das atividades envolvidas no desenvolvimento dos produtos acordados entre as organizações descentralizadas. A Figura 4 exemplifica o contexto visualizado pelo fornecedor quando no contexto da atividade remota. Segundo [Ebert 2007] a visualização dos processos remotos é necessária para permitir a adequação dos processos a normas de maturidade de processos, como CMMI [Hayes 2005] ou MPS.BR [Weber 2006]. Este problema é relatado por empresas estudadas em [Prikladnicki 2007], e o objetivo deste trabalho é tratar diretamente este problema. Como um trabalho futuro considera-se a experimentação e coleta de evidências empíricas dos resultados do uso da ferramenta.
- c. **Monitoração Automática do Projeto:** A notificação de eventos de execução atua como um mecanismo de monitoração provido continuamente pela ferramenta, sem a necessidade de requisições do usuário para obter os dados sobre o andamento das atividades. Este problema é relatado por [Hillegersberg 2007] e [Ebert 2007], e é tratado neste trabalho pelas funcionalidades oferecidas para monitoração do andamento do processo de software. Esse conjunto de funcionalidades é classificado por [Sabherwal 2003] como **monitoração automática**.
- d. **Integração dos Processos:** a integração dos processos organizacionais está diretamente relacionada com a comunicação sobre os as atividades e artefatos dos processos definidos no contexto descentralizado de coordenação. A integração entre dois ambientes de processos (exemplificado pela integração de duas instâncias do ambiente WebAPSEE na Figura 3) representa umas formas de realizar a integração entre os processos de diferentes organizações.

## 5. Trabalhos Relacionados

Entre os PSEEs que utilizam a definição de mecanismos de segurança na integração de dados sobre processos (como **DPE/PAC** [Chou 2005] e **Oz** [Ben-Shaul 1998]), não



existe a definição clara da necessidade de definir um acordo formal que seja multilateral e promova o consenso entre os envolvidos na forma de contrato eletrônico. Assim, tais propostas quando implantadas em ambientes organizacionais mais restritos (tais como *Outsourcing Offshore*) podem gerar desconfiança entre os envolvidos quanto ao consenso dos acessos remotos. Como exemplo, os trabalhos de [Ben-Shaul 1998] e [Chou 2005] definem regras de controle de acesso diretamente no modelo de cada *site*.

No caso deste trabalho, a existência de um contrato relaciona um representante de cada organização envolvida. Assim, a definição do contrato tem o objetivo de promover o consenso entre as partes na medida em que exige a definição a priori das regras de segurança e com a ciência de todos os envolvidos. O consenso entre as partes envolvidas é citado por [Hillegersberg 2007] como um dos requisitos essenciais que devem ser atendidos por ferramentas que atuem no escopo de DDS.

A necessidade de um contrato entre cliente e fornecedor é também um dos requisitos citados pelo modelo de Aquisição do MPS.BR [Softex 2008] para apoio ao processo de aquisição de software. Não foram encontradas ferramentas que relacionem diretamente quais requisitos do modelo de Aquisição MPS.BR são atendidos. No caso deste trabalho, o contrato eletrônico definido atende diretamente os requisitos de proteção de informações organizacionais citados em [Wells 2001] e [Ebert 2007], além da integração de processos atacar diretamente os requisitos destacados na Figura 2.

### 6. Conclusões e Trabalhos Futuros

O apoio automatizado à monitoração de processos de software descentralizados em DDS é necessário para agilizar a troca de informações entre clientes e fornecedores [Hillegersberg 2007]. As funcionalidades da ferramenta descrita neste artigo são potenciais facilitadores da relação entre cliente e fornecedor, principalmente no contexto de *Outsourcing* onde a distância gerencial entre os envolvidos remete a uma relação mais formal para troca de dados [Sabherwal 2003]. Com a flexibilidade permitida pelo uso de contrato eletrônico, mesmo casos de *Insourcing* são aplicados dependendo da configuração do contrato. A coleta de evidências empíricas da efetividade das funcionalidades descritas neste trabalho é definida como um trabalho futuro.

A versão atual da ferramenta implementada permite apenas a integração entre diferentes instâncias do ambiente **WebAPSEE**. Considera-se a implementação da integração com outras ferramentas um trabalho futuro que pode apoiar a uma implantação menos invasiva em organizações que já possuam outras ferramentas consolidadas no ambiente operacional da organização.

### Referências

- Audy, J.; Prikladnicki, R. (2007) “Desenvolvimento Distribuído de Software: Desenvolvimento de software com equipes distribuídas”. Rio de Janeiro, Editora Elsevier, 2007.
- Ben-Shaul, Israel Z.; Kaiser, G.E. (1998) “Federating Process-Centered Environments: The OZ Experience”. In: Automated Software Engineering, Vol. 5.
- Covre, V; Lima Reis, C. A.; Favero, E. (2008) “Metodologia para Implementação do MPS.BR Utilizando o Ambiente WebAPSEE”. In: VII Simpósio Brasileiro de Qualidade de Software (SBQS 2008), Florianópolis, Santa Catarina.
- Coulouris, George F.; Dollimore, J.; Kindberg, T. (2005) “Distributed Systems: concepts and design”. Pearson Education Limited, Edinburg, England.

- Chou, Shih-Chien; Hsu, Wei-Chuan; Lod, Wei-Kuang (2005) "DPE/PAC: decentralized process engine with product access control". In: Journal of Systems and Software, Volume 76, Issue 3, Junho.
- De Souza, C. R. B., Basaveswara, S. D., Redmiles, D. F. (2002) "Supporting Global Software Development with Event Notification Servers". In: 24th International Conference on Software Engineering, International Workshop on Global Software Development, pp. 9-13, Orlando, Florida, EUA.
- Ebert, Christopher (2007) "Optimizing Supplier Management in Global Software Engineering". In: Second International Conference on Global Software Engineering, ICGSE 2007, Munique, Alemanha.
- Gruhn, Volker (2002) "Process-Centered Software Engineering Environments: A Brief History and Future Challenges". In: Annals of Software Engineering 14, 363–382. Kluwer Academic Publishers.
- Hayes, Will; et al. (2005) "CMMI® SCAMPI Distilled Appraisals for Process Improvement". Editora Addison Wesley Professional, 2005.
- Herbsleb, James D. (2007) "Global Software Engineering: The Future of Socio-technical Coordination". In: International Conference on Software Engineering, Future of Software Engineering, IEEE Computer Society, Washington, DC, USA (2007).
- Hillegersberg, Jos van; Herrera, Miles (2007) "Tool Support for Distributed Software Development: The past – present – and future of gaps between user requirements and tool functionalities". In: Tools for Managing Globally Distributed Software Development (TOMAG 2007), Munique, Alemanha, 2007.
- Kuipers, Tobias; Visser, Joost; Vries, Gerjon de (2007) "Monitoring the Quality of Outsourced Software". In: Tools for Managing Globally Distributed Software Development (TOMAG 2007), Munique, Alemanha, 2007.
- Lima, A. M.; Reis, R. Q.; Lima Reis, Carla A. (2006) "Gerência Flexível de Processos de Software com o Ambiente WebAPSEE". In: XIX Simpósio Brasileiro de Engenharia de Software – Sessão de Ferramentas.
- Lima, A. M.; Reis, Rodrigo Q. (2007) "Uma Proposta de Ferramenta para Execução Descentralizada de Processos de Software". In: VI Simpósio Brasileiro de Qualidade de Software (SBQS 2007), Porto de Galinhas, Pernambuco.
- Prikladnicki, R.; Audy, J. L. N.; Damian, D.; Oliveira, T. C. (2007) "Distributed Software Development: Practices and challenges in different business strategies of offshoring and onshoring". In: Proceedings. International Conference on Global Software Engineering (ICGSE 2007), Munique, Alemanha.
- Sabherwal, R. (2003) "The evolution of coordination in outsourced software development projects: a comparison of client and vendor perspectives". In: Information and Organization, Volume 12, número 3, pp. 153-202.
- Softex, Guia MPS.BR de Aquisição de Software, versão 1.2 . Disponível em [http://www.softex.br/mpsbr/\\_guias/MPS.BR\\_Guia\\_de\\_Aquisicao\\_v1.2.pdf](http://www.softex.br/mpsbr/_guias/MPS.BR_Guia_de_Aquisicao_v1.2.pdf). Acesso em junho de 2008.
- Taylor, Richard N.; Erenkrantz, J.R. (2003) "Supporting Distributed and Decentralized Projects: Drawing Lessons from the Open Source Community". In: Proceedings of 1st Workshop on Open Source in an Industrial Context, California, Outubro, 2003.
- Weber, Kival; et al. (2006) "Melhoria de Processo do Software Brasileiro (MPS.BR): um programa Mobilizador". In: 32ª Conferência Latino-Americana de Informática (CLEI 2006), Santiago, Chile, Agosto.
- Wells, Thomas O.; Braunfeld, Roger. (2001) "Protecting Your Most Valuable Asset: Intellectual Property". In: IEEE Special Issue in IT Professional, Volume: 3, Issue: 2, pag. 11–17, Março-Abril.