

Odyssey-CCS: Uma ferramenta flexível para o controle de modificações em software

Luiz Gustavo B. Lopes^{1,2}, Leonardo Murta¹, Cláudia Werner¹

¹ COPPE/UFRJ - Programa de Engenharia de Sistemas e Computação
Caixa Postal 68511 CEP. 21945 – 970 – Rio de Janeiro - RJ

² Banco Central do Brasil - Setor Bancário Sul (SBS) Quadra 3 Bloco B
Caixa Postal 08670 CEP. 70074-900 - Brasília - DF

luiz.lobes@bcb.gov.br, {murta,werner}@cos.ufrj.br

Abstract. *Software is subject to change. However, different problems may arise when changes are performed without control. This paper describes Odyssey-CCS, a tool for change control that provides resources to support the communication and coordination of work teams.*

Resumo. *Todo software está sujeito a modificações. Contudo, diversos problemas podem surgir quando as modificações são feitas sem controle. Este artigo descreve a Odyssey-CCS, uma ferramenta para o controle de modificações em software que possui recursos que auxiliam a comunicação e coordenação de equipes de trabalho.*

1. Introdução

Todo software está continuamente sujeito a modificações [5, 6, 14], que podem ser motivadas por diversos fatores. Exemplos são a correção de erros encontrados por desenvolvedores e usuários, o surgimento de novas tecnologias, as alterações no ambiente de operação e o desenvolvimento incremental.

Quando as modificações no software são realizadas sem controle, diversos problemas podem surgir [6, 14]. Por exemplo, com a falta de coordenação das pessoas envolvidas, uma mesma tarefa de modificação pode ser realizada mais de uma vez por pessoas diferentes, causando desperdício de recursos. Mais ainda, as modificações podem ser realizadas em uma ordem de prioridade errada e pode ocorrer demora no atendimento às requisições de modificação. Além disso, o gerente do projeto perde informações, como o número de modificações em andamento, finalizadas e em espera, quais foram os motivos de cada modificação, quem foi que as realizou e quais artefatos foram afetados. Se não forem mantidas as informações sobre quais versões de artefatos estão presentes em cada cliente, quando um determinado cliente requisitar uma modificação, será muito difícil atender à sua solicitação.

Desta forma, o objetivo deste trabalho é fornecer uma ferramenta para o controle de modificações, denominada Odyssey-CCS, que seja configurável com relação aos processos de controle de modificações e à coleta de informações, e que auxilie na detecção das responsabilidades entre fornecedores e clientes. Esse tipo de ferramenta atua como um maestro das equipes envolvidas com as modificações, auxiliando na

comunicação e coordenação do trabalho e mantendo informações sobre as modificações realizadas (e.g.: quem fez o quê, quando, onde, como e por que). Maiores detalhes sobre a abordagem, de mesmo nome, que baseia a ferramenta Odyssey-CCS, podem ser encontrados em [7].

Este trabalho está organizado em quatro seções, além desta introdução. A Seção 2 fornece informações sobre a utilização da Odyssey-CCS. A Seção 3 detalha os seus aspectos técnicos. A Seção 4 discute alguns trabalhos relacionados. Finalmente, a Seção 5 apresenta as contribuições, limitações e possíveis trabalhos futuros.

2. Utilização da Ferramenta

A ferramenta Odyssey-CCS auxilia na realização de tarefas relacionadas ao controle de modificações de software. A seguir, são descritas estas atividades e o apoio fornecido pela Odyssey-CCS.

1) Modelar processo: O Gerente de Configuração deve modelar um processo de controle de modificações a ser seguido para cada software produzido na empresa [3, 4]. Este processo deve ser adaptado do processo padrão definido para a organização, adequando-se às características específicas de cada projeto.

A Odyssey-CCS auxilia esta atividade através do Modelador SPEM (Figura 1). Este módulo permite modelar graficamente um processo de controle de modificações através da notação SPEM [11], que é própria para a modelagem de processos de software. A SPEM é baseada no MOF [10], e utiliza alguns diagramas da UML [12] adaptados para a modelagem de processos. Assim, esta notação deve ser facilmente compreendida pelos profissionais que conhecem a UML, linguagem muito difundida na área de Engenharia de Software.

Através do Modelador SPEM, é possível modelar as atividades que devem ser realizadas, sua sequência de execução, os artefatos consumidos e produzidos por cada atividade, os papéis responsáveis por sua execução, entre outros. O processo modelado também pode ser exportado segundo o padrão XMI [13], potencializando a sua utilização por outras ferramentas de modelagem e execução de processos que venham a utilizar o metamodelo SPEM.

2) Modelar formulário: O Gerente de Configuração também deve definir quais informações precisam ser coletadas ao longo do trabalho de controle de modificações, para que os envolvidos na modificação possam realizar as atividades e tomar as decisões necessárias, e para que as modificações sejam compreendidas futuramente.

A Odyssey-CCS fornece um módulo para modelagem gráfica de formulários, o Modelador de Formulários (Figura 2), que permite definir as informações que deverão ser coletadas. As informações são coletadas através de campos contidos nos formulários. Os tipos de campos implementados permitem coletar texto, selecionar opções, marcar alternativas, anexar arquivos e referenciar outros formulários. À medida que um formulário é construído, uma pré-visualização deste é exibida.

3) Configurar processo: Após a modelagem do processo e dos formulários a serem utilizados para o controle de modificações, o Gerente de Configuração deve fazer a atribuição de responsabilidades, designando os papéis modelados no processo às pessoas que participarão do controle de modificações do software. Deve também

identificar o momento de coleta de cada informação, através da associação de formulários a produtos modelados no processo. Além disso, deve configurar o envio de notificações por e-mail, com relação ao conteúdo, destinatários e momento. Isso é realizado através do preenchimento e associação de um formulário especial de notificação aos produtos modelados no processo.

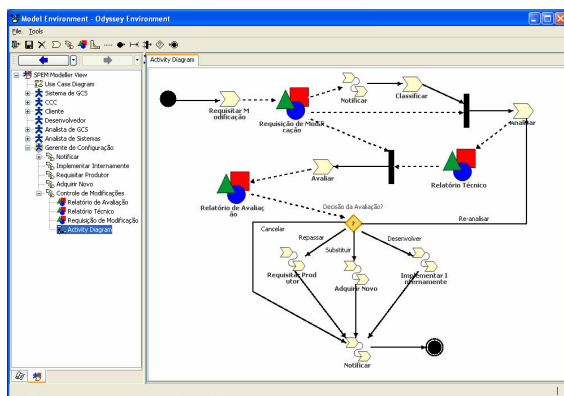


Figura 1. Modelador SPEM

Figura 2. Modelador de Formulários

A Odyssey-CCS possui um módulo de software que auxilia a realização deste trabalho passo a passo. Ele traduz o processo selecionado para utilização no contexto do software em foco, exibindo os papéis e produtos modelados, e também acessa as informações cadastradas no sistema, como os usuários e os formulários, permitindo que o mapeamento descrito anteriormente seja realizado. Após a configuração, o processo é colocado em execução e os clientes podem enviar requisições de modificação.

4) Verificar as atividades e decisões pendentes de execução: As pessoas envolvidas com o controle de modificações devem verificar quais atividades e decisões estão pendentes para execução. A Odyssey-CCS exibe uma tela para cada usuário, apresentada na Figura 3, com as suas pendências em todos os projetos nos quais ele participa. Dependendo da configuração do processo, os usuários também podem receber notificações por e-mail sobre o progresso das instâncias de execução dos processos.

5) Realizar as atividades e decisões pendentes de execução: Após verificar suas pendências, cada usuário realizará as suas atividades. Através de uma interface da Odyssey-CCS, exibida na Figura 3, o usuário pode marcar que está realizando uma atividade ou tomando uma decisão, de forma que outras pessoas que desempenhem o mesmo papel tenham ciência deste fato. O usuário também tem acesso a todas as informações coletadas anteriormente em cada instância de execução do processo, possivelmente necessárias para a execução da atividade ou decisão corrente.

Por exemplo, dependendo da configuração realizada, o usuário terá acesso a informações sobre os contratos firmados entre fornecedores e clientes, como exibido na Figura 4. Estas informações auxiliam a identificar se o fornecedor é realmente responsável por uma requisição de manutenção e sob que condições essa manutenção deve ser executada (e.g.: gratuita ou paga). Além disso, é possível detectar quais outros clientes devem ser avisados quando uma determinada modificação for executada.

Vale ressaltar que a ferramenta Odyssey-CCS está integrada com a ferramenta Odyssey-VCS [9], permitindo que todas as versões dos artefatos afetados pela modificação sejam automaticamente associadas à modificação para consulta futura.

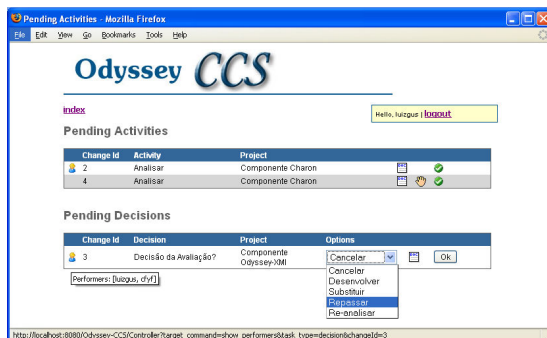


Figura 3. Atividades e decisões pendentes

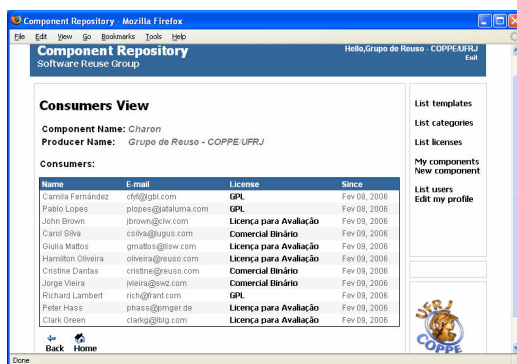


Figura 4. Contratos entre fornecedores e clientes

6) Finalizar as atividades e decisões: Ao realizar as suas atividades e decisões, as pessoas envolvidas com o controle de modificações devem fornecer as informações necessárias para que a modificação possa ser compreendida no futuro. Através da Odyssey-CCS, o usuário finaliza uma atividade ou decisão e fornece as informações requisitadas. Isso é feito por meio do preenchimento dos formulários, que foram associados aos produtos de atividades na etapa de configuração do processo. Desta forma, quando uma atividade é finalizada, o formulário associado ao seu produto é exibido e preenchido pelo usuário. Após esta ação, a Odyssey-CCS armazena as informações fornecidas, infere as próximas atividades ou decisões do processo e disponibiliza estas informações para os usuários responsáveis pelas novas pendências.

3. Aspectos técnicos

A Odyssey-CCS (Figura 5) foi desenvolvida, em sua maior parte, com tecnologia J2EE. Assim, a maioria dos módulos possui uma interface gráfica para Internet, desenvolvida em JSP. Além disso, eles possuem controladores desenvolvidos como Servlets e a camada de negócios desenvolvida com o uso de EJB. Foi utilizado também o repositório MDR [8], que facilita o armazenamento e manipulação de modelos baseados no MOF, como os processos modelados em SPEM. Uma instalação exemplo da Odyssey-CCS está disponível em <http://reuse.cos.ufrj.br/Odyssey-CCS> sob licença gratuita para

utilização. O módulo Modelador SPEM é o único desenvolvido com a tecnologia J2SE. Ele foi desenvolvido como um *plugin* do Ambiente Odyssey [15], que está disponível em <http://reuse.cos.ufrj.br/site/pt/>, também sob licença gratuita para utilização.

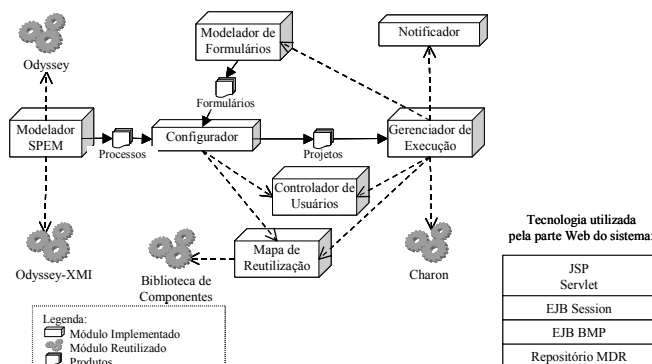


Figura 5. Visão Geral da Odyssey-CCS

4. Trabalhos Relacionados

Há diversas ferramentas para controle de modificações, como ClearQuest [16] e Bugzilla [1], entre outras [2]. Porém, não foi encontrada uma que utilizasse uma notação padronizada de modelagem de processos, como a SPEM. Muitas delas não permitem a modelagem dos processos a serem utilizados. As que permitem, geralmente, não fornecem notação gráfica ou utilizam diagramas de transição de estados como notação. Porém, a notação de diagrama de transição de estados não possui alguns elementos importantes para modelagem de processos, como os produtos consumidos e produzidos, os papéis responsáveis pelas atividades e a definição de subatividades. Além disso, são poucas as que permitem a configuração da coleta de informações. Outra constatação é que nenhuma das ferramentas de controle de modificações pesquisadas possui recursos que auxiliem a identificação da responsabilidade de manutenção entre os fornecedores e os clientes de cada software.

5. Conclusão

Este trabalho apresentou uma ferramenta para controle de modificações que possibilita a modelagem gráfica via notação padronizada do processo de controle de modificações, assim como das informações a serem coletadas durante a execução desse processo. Além disso, a ferramenta mantém informações referentes aos contratos entre fornecedores e clientes, possibilitando a identificação de responsabilidades de manutenção. Mais ainda, os clientes podem ser notificados sobre o andamento do processo sempre que necessário.

Porém, algumas limitações foram detectadas, como a falta de recursos para realização de buscas complexas no sistema, a falta de alguns elementos SPEM, que podem ser necessários para a modelagem de processos mais complexos, e a falta de um repositório de dados mais robusto, como um SGBD. Como trabalhos futuros, podemos citar o fornecimento de recursos para a análise das informações coletadas, através de gráficos e tabelas, o apoio à tarefa de análise de impacto, auxiliando a identificação de componentes afetados por uma requisição de modificação, e a avaliação da ferramenta em ambientes reais de utilização.

Agradecimentos

Agradecemos ao CNPq, CAPES e Banco Central do Brasil pelo apoio financeiro.

Referências Bibliográficas

1. Barnson, M.P., Steenhagen, J. and Weissman, T. The Bugzilla Guide - 2.17.5 Development Release, The Bugzilla Team, 2003.
2. CMCrossroads. Defect Tracking Software, 2006.
3. EIA. EIA 649 - National Consensus Standard for Configuration Management, Electronic Industries Alliance, 1998.
4. IEEE. Std 828 - IEEE Standard for Software Configuration Management Plans, Institute of Electrical and Electronics Engineers, 2005.
5. Lehman, M.M., Perry, D.E., Ramil, J.F., Turski, W.M. and Wernick, P.D., Metrics and Laws of Software Evolution: The Nineties View. in, (Albuquerque, NM, 1997), pp. 20-32.
6. Leon, A. *A Guide to Software Configuration Management*. Artech House Publishers, Norwood, MA, 2000.
7. Lopes, L.G.B., Murta, L.G.P. and Werner, C.M.L., Odyssey-CCS: A Change Control System Tailored to Software Reuse. in *International Conference on Software Reuse*, (Torino, Italy, 2006), pp. 170-183.
8. Matula, M. NetBeans Metadata Repository, NetBeans Community, 2005.
9. Oliveira, H.L.R., Murta, L.G.P. and Werner, C.M.L., Odyssey-VCS: Um Sistema de Controle de Versões Para Modelos Baseados no MOF. in *Simpósio Brasileiro de Engenharia de Software, Seção de Ferramentas*, (Brasília, Brasil, 2004), pp. 85-90.
10. OMG. Meta Object Facility (MOF) Specification, version 1.4, Object Management Group, 2006.
11. OMG. Software Process Engineering Metamodel (SPEM), Version 1.1, Object Management Group, 2006.
12. OMG. Unified Modeling Language (UML) Specification, Version 1.5, Object Management Group, 2003.
13. OMG. XML Metadata Interchange (XMI) Specification, Version 2.1, Object Management Group, 2006.
14. Pressman, R.S. *Software Engineering: A Practitioner's Approach*. McGraw-Hill, 2005.
15. Werner, C.M.L., Mangan, M.A.S., Murta, L.G.P., Souza, R.P., Mattoso, M., Braga, R.M.M. and Borges, M.R.S., OdysseyShare: an Environment for Collaborative Component-Based Development. in *IEEE Conference on Information Reuse and Integration (IRI)*, (Las Vegas, USA, 2003), pp. 61-68.
16. White, B.A. *Software Configuration Management Strategies and Rational ClearCase: A Practical Introduction*. Addison-Wesley, 2000.