

## Uma Experiência de Implementação Nível G em uma Empresa de Software Livre

Kelly Bettio<sup>1</sup>, Joselaine Valaski<sup>1</sup>, Douglas Lopes Gomes<sup>2</sup>, Edney Matias<sup>2</sup>, Sheila Reinehr<sup>1</sup>, Andreia Malucelli<sup>1</sup>

<sup>1</sup> Programa de Pós-Graduação em Informática. Pontifícia Universidade Católica do Paraná (PUCPR) - Curitiba – PR – Brasil  
{kelly.bettio,sheila.reinehr}@pucpr.br, {malu,jvalaski}@ppgia.pucpr.br

<sup>2</sup> Prognus Soluções Livres em Tecnologia da Informação Ltda  
Av. Tancredo Neves, 6731 - Parque Tecnológico Itaipu - Foz do Iguaçu - PR – Brasil  
{douglas,edney}@prognus.com.br

**Resumo.** *A qualidade de um produto de software é um fator de sucesso essencial, não somente para o software proprietário, mas também para o software livre. Preocupada com esta questão, a empresa Prognus Software Livre aderiu ao programa MPS.BR como meio de melhorar seus processos de desenvolvimento de software e oferecer maior confiança referente a seus produtos. Este artigo descreve o processo de implantação do MPS.BR nesta empresa, a adaptação da ferramenta Trac, adotada para a gerência de projetos, assim como apresenta uma comparação entre lições aprendidas em empresas desenvolvedoras de software proprietário e software livre.*

**Abstract.** *The quality of a software product is an essential success factor, not only to proprietary software, but also for free software. Concerned with this issue, Prognus Foss joined MPS.BR program as a means to improve their software development processes and provide greater confidence for their products. This paper describes the deployment process of MPS.BR in this company, the tailoring process of the tool Trac, adopted for project management, as well as presents a comparison of lessons learned in proprietary software companies and free software companies.*

### 1. Introdução

A visibilidade e o interesse pelo software livre têm aumentado nos últimos anos principalmente pelos diversos benefícios que esta abordagem proporciona à área de software, tais como: custo baixo, independência de tecnologia proprietária, robustez, segurança, facilidade de adaptação, e o suporte abundante e gratuito [Hexsel 2002].

Um produto de software livre pode ter seus códigos fonte usados, copiados, estudados, modificados e redistribuídos sem restrição [Pavlicek 2000]. O software livre, assim como um produto de software proprietário, tem sua aceitação ou rejeição influenciada pelo grau de confiança que oferece. Desta maneira, torna-se vital adotar um processo de desenvolvimento de software que garanta a qualidade do produto gerado.

As práticas de Engenharia de Software utilizadas na produção do software livre são adaptadas à realidade de desenvolvedores que trabalham de forma colaborativa em redes como a Internet, formando sistemas complexos de organização com regras e níveis de acesso diversificados, chamados “comunidades”. Essas comunidades elegem um líder que será o desenvolvedor ou empresa que decidirá a prioridade e os rumos dos projetos. Este líder é escolhido por mérito, que se dá geralmente pela quantidade e qualidade do código contribuído, além de questões relacionadas à participação ativa nas discussões desta comunidade [SOFTEX 2005].

Como forma de aperfeiçoamento de suas práticas de desenvolvimento, de forma a produzir software de melhor qualidade e confiabilidade, tanto para a comunidade de desenvolvimento quanto para seus clientes, a Prognus Software Livre, aderiu ao projeto de melhoria apoiado pelo programa MPS.BR - Melhoria de Processo do Software Brasileiro [SOFTEX 2009]. Neste contexto, este artigo tem como objetivo descrever o processo de implantação do modelo MPS na empresa Prognus Software Livre, compartilhando uma experiência prática. Além disto, apresenta uma comparação de lições aprendidas durante esta experiência com lições aprendidas em projetos de melhoria de empresas produtoras de software proprietário, buscando semelhanças como forma de contribuição para futuros projetos em empresas deste mesmo segmento de mercado.

A próxima seção apresenta uma caracterização da empresa Prognus Software Livre. O projeto de melhoria implantado é descrito na Seção 3. A Seção 4 descreve a ferramenta Trac, adaptada para auxiliar na execução das atividades do processo definido. A Seção 5 apresenta as lições aprendidas, as quais foram comparadas com experiências de empresas voltadas à produção de software proprietário e a Seção 6 conclui este artigo.

## 2. Prognus Software Livre

A Prognus Software Livre é uma empresa de desenvolvimento de soluções de comunicação empresarial, que atua no mercado há mais de cinco anos. Dentre as tecnologias oferecidas, a Prognus destaca o desenvolvimento e contribuição do Expresso Livre. O Expresso Livre é uma ferramenta de comunicação colaborativa (*groupware*), que centraliza diversos módulos de serviços em uma única aplicação (e-mail, agenda de compromissos, catálogo de endereços, *workflow*, entre outros). Esta ferramenta é mantida por uma comunidade de usuários que, reunidos, garantem a sua manutenção e evolução. Por motivos estratégicos a empresa focou 100% de suas atividades no desenvolvimento de soluções abertas em ferramentas de comunicação e colaboração corporativa, tendo o Expresso Livre como seu produto foco.

A atuação da empresa no desenvolvimento do Expresso Livre resultou em uma posição de destaque no Comitê Gestor do Expresso Livre, que é uma associação voluntária, formada por empresas como Serpro, Celepar, Dataprev e Prodeb, sendo todas estas empresas públicas, exceto a Prognus Software Livre.

Para garantir a qualidade do produto gerado, tanto para a própria comunidade mantenedora, quanto para os clientes conquistados, a Prognus passou a se preocupar em definir um processo de desenvolvimento de software baseado em um modelo de melhoria de processo. Este seria um grande desafio, pois, a própria característica do modelo de código aberto possibilita a livre manutenção e contribuição por qualquer

pessoa que respeite as licenças de software desta natureza, não definindo um modelo de desenvolvimento de software [OpenSource 2011] e, conseqüentemente, não garantindo muitas vezes que o produto seja gerado sob um processo de qualidade de acordo com a expectativa da comunidade e do cliente.

Devido ao sucesso demonstrado por meio de micro e pequenas empresas que adotaram o modelo MPS, a Prognus aderiu ao projeto de implantação deste modelo entre um grupo de empresas de desenvolvimento de software do norte e oeste do Paraná, tendo como Instituição Implementadora (II) o Centro Internacional de Tecnologia de Software (CITS), objetivando a definição e institucionalização de um modelo de desenvolvimento de software baseado no nível de maturidade Parcialmente Gerenciado (G). Será descrito na próxima seção como o projeto de melhoria da Prognus foi planejado e executado.

### 3. Projeto de Melhoria

O início do projeto se deu por meio da execução de um Diagnóstico (*Gap Analysis*) com a finalidade de fornecer a situação dos processos da empresa, anteriores à adoção do modelo MPS. As primeiras atividades foram realizadas em dezembro de 2009 e foram divididas em quatro fases: planejamento, definição, institucionalização e a avaliação do processo definido.

#### 3.1 Planejamento

O início do planejamento se deu após a contratação dos serviços de consultoria fornecidos pelo CITS que em conjunto com a SOFTEX organizou o planejamento do projeto como um todo. Este planejamento consistiu na programação das atividades a serem realizadas, na definição de um cronograma macro, um plano de projeto de melhoria e na alocação de uma equipe na empresa dedicada à definição dos processos de software, composta por um recurso tempo integral e um recurso tempo parcial.

#### 3.2 Definição

A definição dos processos iniciou com a determinação de um ciclo de vida padrão para os projetos de desenvolvimento de software baseado no *Rational Unified Process* (RUP) e adaptado à realidade dos projetos de evolução do Expresso Livre.

- Planejamento: nesta fase o escopo do projeto é definido e o projeto é planejado;
- Execução: esta fase trata da execução do que foi planejado, sendo dividida em três subfases denominadas especificação, codificação e teste;
- Lançamento: esta é a ultima fase do ciclo, trata da liberação para testes das demandas desenvolvidas para a comunidade e para os clientes.

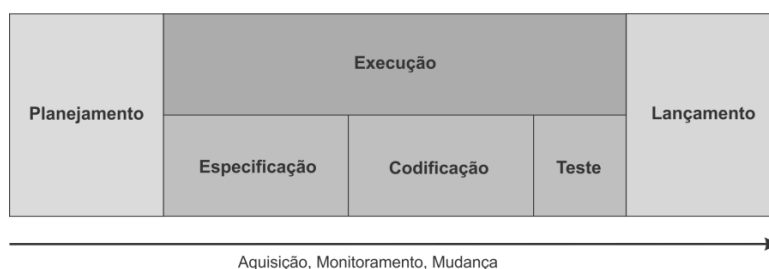


Figura 1. Ciclo de Vida Padrão da Prognus Software Livre

A partir da existência de um modelo de ciclo de vida padrão para os projetos, foram definidos cinco processos contendo atividades do fluxo de trabalho da Prognus: *Processo Principal*, correspondente ao ciclo de vida apresentado na Figura 1; *Aquisição*, que responde pela aquisição (coleta) de requisitos com os clientes; *Monitoramento*, que visa garantir a execução conforme o planejamento do projeto; *Mudança e Qualidade*, que apoiam os demais processos no controle de mudanças e na garantia de aderência aos padrões definidos, respectivamente.

O conjunto destes processos foi denominado Processo de Desenvolvimento da Prognus Software Livre, conforme Figura 2. Cada atividade dos processos foi descrita de maneira clara, definindo entradas e saídas, responsáveis e participantes envolvidos, modelos de documentos e ferramentas que poderiam auxiliar na sua execução.

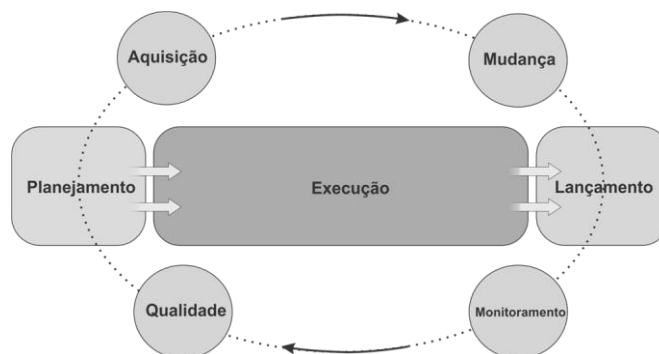


Figura 2. Processo de Desenvolvimento da Prognus Software Livre

### 3.3 Institucionalização

A institucionalização dos processos definidos na Prognus ocorreu de forma gradativa. Todos os colaboradores foram motivados e apresentados ao processo, realizando leitura e estudo integral das definições. Após esta etapa, um workshop de institucionalização, onde todos os colaboradores foram responsáveis pela apresentação de um processo específico aos demais colegas foi promovido, com o objetivo de consolidar a institucionalização do processo na Prognus Software Livre.

Também fez parte da institucionalização, a execução de um projeto piloto com tamanho pequeno, contendo uma funcionalidade a ser implementada. A equipe do projeto foi composta por um gerente de projeto, dois analistas e dois programadores, cada qual exercendo um ou mais papéis do processo proposto.

Uma simulação de avaliação denominada *Mini Gap-Analysis* foi realizada em Julho de 2010, para averiguar a institucionalização do processo, além de prover a obtenção de *feedback* dos colaboradores em relação às expectativas quanto ao processo definido e ao nível de aceitação dentro da empresa. Não foram encontradas dificuldades na absorção dos processos pela equipe de desenvolvimento.

### 3.4 Avaliação

Para uma avaliação nível G do modelo MPS são necessários dois projetos, sendo um projeto concluído e um projeto em andamento [SOFTEX 2009]. Como preparação para a avaliação inicial, as reuniões de consultoria focaram na verificação dos artefatos gerados nos projetos e no esclarecimento de dúvidas sobre a institucionalização dos processos.

A avaliação inicial essencialmente verificou os artefatos gerados no projeto concluído e no projeto em andamento. Cada artefato foi confrontado diretamente com os resultados esperados dos processos de Gerência de Projetos e Gerência de Requisitos obrigatórios para o nível G [SOFTEX 2009], conforme previsto no processo de avaliação MPS.

Para a avaliação final a consultoria, juntamente com a equipe de processos, focaram os trabalhos na realização de simulações de entrevistas com os membros da equipe que participaram dos projetos avaliados. A avaliação final foi realizada e o êxito dos trabalhos da equipe de processos pôde ser confirmado com a obtenção do nível de maturidade G do MPS.

#### 4. Ferramenta Trac

O maior desafio na definição do processo de desenvolvimento da Prognus foi a seleção das ferramentas adequadas para a realização de cada atividade do processo definido. A ferramenta selecionada para atender as necessidades foi o Trac, por oferecer flexibilidade de personalização por meio de *plug-ins* e componentes que podem ser adicionados conforme necessidade [Trac 2011].

Esta ferramenta já era utilizada anteriormente na empresa, o que facilitou a adequação de acordo com as necessidades estabelecidas pelo processo tornando-se a principal ferramenta para a gerência dos projetos.

Inicialmente voltado para o controle de mudanças em projetos de desenvolvimento de software, o ambiente da ferramenta foi adaptado para cumprir o papel de organizar os artefatos gerados pelos projetos, centralizar as informações gerenciais e realizar o monitoramento da execução dos projetos. Todas as necessidades dos membros de um projeto foram integradas em um único ambiente de forma simples e flexível. A tela principal do Trac já adaptado é apresentada na Figura 3.



Figura 3. Tela principal do ambiente Trac da Prognus Software Livre

No Trac, todos os artefatos são páginas Wiki geradas a partir de um *template* definido. O responsável por executar uma atividade do processo e por gerar o *template* deve preencher as informações necessárias do documento e salvá-lo diretamente no Trac e assim torná-lo disponível aos demais membros da equipe.

O Gerente de Projeto tem à disposição relatórios de execução de atividades, gráficos que refletem o andamento do projeto e acesso aos documentos produzidos pelos integrantes de sua equipe. O monitoramento do projeto é realizado por meio do acompanhamento dos *tickets* que o Trac disponibiliza. Desta forma todas as atividades do cronograma de execução são *tickets*, cada qual com o responsável pela sua execução. O apontamento de horas também é realizado no próprio *ticket*, uma vez que é possível registrar o tempo de execução das tarefas.

Também foram adicionados ao Trac, plug-ins de personalização dos *tickets*, geração de relatórios de acompanhamento e modelagem de diagramas no padrão *Unified Modeling Language* (UML).

## 5. Lições Aprendidas

A implementação do MPS na Prognus, de uma forma geral, foi considerada um desafio especial para todos os colaboradores. Algumas lições aprendidas durante o projeto de melhoria foram comparadas com lições aprendidas fornecidas por empresas voltadas à produção de software proprietário publicadas em eventos de qualidade de software dos últimos cinco anos, com o objetivo de identificar se existem e quais são as semelhanças entre as abordagens.

Foram selecionados 10 trabalhos entre relatos de experiência e trabalhos técnicos relacionados a implementações MPS para software proprietário e comparadas com as lições aprendidas identificadas na Prognus durante o projeto. As lições aprendidas selecionadas, listadas abaixo, foram citadas por pelo menos 40% dos trabalhos e se aplicam de forma similar a uma implementação MPS de empresa atuante no segmento de software livre como a Prognus, sendo:

- A existência de uma forma de comunicação eficiente entre os participantes dos projetos possibilitando a troca de ideias e sugestões [Borssatto and Moro 2007; Brietzke et al. 2007; Marçal et al. 2007; Santos et al. 2009; Mendes et al. 2010].
- A alocação de pelo menos um membro da equipe de processos dedicado em tempo integral ao projeto de melhoria, utilizado sabiamente devido à limitação de recursos de uma organização pequena [Borssatto and Moro 2007; Brietzke et al. 2007; Marçal et al. 2007; Santos et al. 2009; Mendes et al. 2010; Rodrigues and Kirner 2010].
- A inclusão de profissionais experientes e o investimento na capacitação dos membros da equipe de processos como fator determinante para o sucesso da iniciativa de melhoria, além da realização de treinamentos/workshops internos com a participação de todos os colaboradores facilitando a institucionalização dos processos [Guerra et al. 2006; Borssatto and Moro 2007; Santos et al. 2009; Mendes et al. 2010; Rodrigues and Kirner 2010; Salgado et al. 2010].
- A possibilidade e a viabilidade da definição de um processo simplificado que atendesse as necessidades da empresa sem se tornar “engessado” e a criação de uma biblioteca de ativos permitindo que as informações transitassem de forma

rastreável e transparente entre os colaboradores [França et al. 2009; Mendes et al. 2010; Rodrigues and Kirner 2010; Salgado et al. 2010].

- Trabalho em equipe de forma colaborativa envolvendo todos os membros que serão afetados com a iniciativa, principalmente aqueles que exercem maior influência nas equipes aumentando o comprometimento no alcance das metas do projeto [Brietzke et al. 2007; Marçal et al. 2007; Santos et al. 2009; Mendes et al. 2010].
- O apoio efetivo da alta direção e a participação de níveis gerenciais no grupo de processos é essencial para o bom andamento das iniciativas de melhoria [Brietzke et al. 2007; Borssatto and Moro 2007; Mendes et al. 2010; Rodrigues and Kirner 2010].
- A existência de uma área de qualidade no apoio a execução dos processos e na avaliação dos documentos produzidos [Guerra et al. 2006; Brietzke et al. 2007; Borssatto and Moro 2007; Mendes et al. 2010].

## 6. Considerações Finais

Entre os vários benefícios que um programa de melhoria de processo pode trazer para uma empresa desenvolvedora de software, destaca-se a mudança cultural na organização e o despertar de uma nova visão sobre a qualidade. A Prognus Software Livre pode observar estas mudanças principalmente na questão de senso de comprometimento e responsabilidade de todos os colaboradores com o produto final a ser disponibilizado ao cliente e à comunidade.

Foi verificada a importância em manter conciliados e alinhados os interesses entre a comunidade mantenedora do Expresso Livre e os clientes da Prognus, mesmo cada uma tendo necessidades diferentes. Além da satisfação de ambas às partes quando se mantém o princípio da livre colaboração estabelecido pelo software livre.

A adaptação da ferramenta Trac de acordo com as necessidades geradas pelo processo definido foi um grande facilitador para alcançar o nível de maturidade esperado. A ferramenta contribuiu com a organização das tarefas, o controle dos projetos, a integração das atividades e artefatos e principalmente possibilitou maior visibilidade do processo por todos os colaboradores.

Este relato teve como objetivo descrever a experiência de um programa de melhoria de processo em uma empresa desenvolvedora de software livre e mostrar a sua viabilidade. Também foi destacada neste trabalho, a importância do apoio de uma ferramenta aderente ao processo da organização como agente facilitador na integração dos artefatos gerados e atividades desenvolvidas pelos diferentes papéis que os colaboradores desempenham. Além da identificação de lições aprendidas oriundas de implementações de software proprietário que se aplicam com similaridade a uma empresa de software livre.

## 7. Referências

- Borssatto, I. B. and Moro, A. R. (2007) “Medições de uma implementação de MPS.BR nível F”, In: VI Simpósio Brasileiro de Qualidade de Software, Porto de Galinhas, Brasil, p. 397-404.

- Brietzke, J., López, P. A. P., Albertuni, I. and Richter, L. A. (2007) “A Conquista do MPS.BR Nível F na Qualidade Informática : Um Caso de Sucesso”, In: VI Simpósio Brasileiro de Qualidade de Software, Porto de Galinhas, Brasil, p. 357-364.
- França, B. B. N., Sales, E. O., Reis, C. A. L. and 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, Ouro Preto, Brasil, p. 310-317.
- Guerra, E., Travassos, G. H., Santos, G., Mafra, S., Barreto, A. and Rocha, A. R. (2006) “Melhoria de Processos no Desenvolvimento de Software e Hardware – O Caso Maxtrack”, In: V Simpósio Brasileiro de Qualidade de Software, Vila Velha, Brasil, p. 326-333.
- Hexsel, R. A. (2002) “Propostas de Ações de Governo para Incentivar o Uso de Software Livre.”, Relatório Técnico RT-DINF 004/2002.
- Marçal, A. S. C., Buchmann, C. A. F., Morais, E. M. and Maciel, T. M. M. (2007) “Implantando uma Gestão Colaborativa de Processos aderente ao CMMI em um Instituto de Inovação”, In: VI Simpósio Brasileiro de Qualidade de Software, Porto de Galinhas, Brasil, p. 381-388.
- Mendes, F. F., Nascimento, H. A. D., Fernandes, P. G., Nunes, R. S. and Mota, C.C. (2010) "Implantação de Melhoria de Processos em um Setor de Produção de Software de uma Universidade Federal", In: IX Simpósio Brasileiro de Qualidade de Software, Belém, Brasil, p. 359-365.
- OpenSource (2011) “Open source initiative”, <http://www.opensource.org>, February.
- Pavlicek, R.C. (2000) “Embracing insanity: Open source software development”, In: Sams, Indianapolis.
- Rodrigues, J. F. and Kirner, T. G. (2010) "Benefícios, Fatores de Sucesso e Dificuldades da Implantação do Modelo MPS.BR", In: IX Simpósio Brasileiro de Qualidade de Software, Belém, Brasil, p. 41-55.
- Salgado, A., Melcop, T., Acchar, J., Rego, P.A., Ferreira, A.I.F, Katsurayama, A.E., Montoni, M and Zanetti, D. (2010) "Aplicação de um Processo Ágil para Implantação de Processos de Software baseado em Scrum na Chemtech", In: IX Simpósio Brasileiro de Qualidade de Software, Belém, Brasil, p. 351-358.
- Santos, G., Katsurayama, A. E., Zanetti, D., Montoni, M., Filho, R. C. S., Barreto, A. O. S. and Rocha, A. R. (2009) “Lições Aprendidas em uma Iniciativa de Melhoria de Processos de Software sob Diferentes Perspectivas: Membros da Organização, Implementadores e Avaliadores”, In: VIII Simpósio Brasileiro de Qualidade de Software, Ouro Preto, Brasil, p. 334-341
- SOFTEX (2005) “O Impacto do Software Livre e de Código Aberto na Indústria de Software do Brasil”, Campinas, Brasil.
- SOFTEX (2009) "MPS.BR – Guia Geral v. MR-MPS: 2009”, [http://www.softex.br/mpsbr/\\_guias/guias/MPS.BR\\_Guia\\_Geral\\_2009.pdf](http://www.softex.br/mpsbr/_guias/guias/MPS.BR_Guia_Geral_2009.pdf)
- Trac (2011) “Ferramenta Trac”, <http://trac.edgewall.org>, February.