



Proposta de Modelo de Processo de Desenvolvimento de Software Orientado ao Gerenciamento de Processo de Negócio

Eduardo F. S. Freire¹, Diogo A. Gouveia¹, Simone V. Silva¹

¹Instituto Federal de Educação, Ciência e Tecnologia Fluminense (IFFluminense)
Rua Dr. Siqueira, 273 – 28.030-130 – Campos dos Goytacazes – RJ – Brazil
eduardofreire@gmail.com, diogo.azvg@gmail.com, simonevsinfo@gmail.com

Abstract. *This paper brings a proposition of a model of software development process based on business process management. For that, we discourse over the traditional developing methods in comparison against the proposed loose coupling development as the support of Business Process Management.*

Resumo. *Este artigo traz uma proposta de um modelo de processo de desenvolvimento de software orientado ao gerenciamento de processo de negócio. Para tanto, discorremos sobre os métodos de desenvolvimento tradicionais em comparação com o desenvolvimento em baixo acoplamento como suporte ao gerenciamento de processo de negócio proposto.*

1. Introdução

O uso de Tecnologias da Informação e Comunicação – TIC para suportar o gerenciamento de processos de negócios exerce um papel de grande importância, dado que não mais se concebe a gestão das mais distintas organizações no séc. XXI sem o uso de aparatos tecnológicos seja no campo dos equipamentos, seja no campo dos softwares. Por outro lado, equipamentos e softwares uma vez postos em funcionamento tornam a gestão dos negócios estática segundo os requisitos em que foram desenvolvidos, mesmo que a organização seja dinâmica e exija o mesmo dinamismo dos sistemas que a suportam.

Neste contexto, é possível entender que para cada mudança requerida pelo dinamismo da gestão do negócio, esforços no sentido de adaptar os sistemas computacionais de suporte devem ser empreendidos. Tais esforços, segundo os conceitos de processo de desenvolvimento de software tradicionais, se traduzem em um planejamento e execução de um projeto de mudança que ensejará uma nova análise parcial de sistemas, seguida de alteração na codificação dos programas, além de toda sorte de testes, checagens, e outros aparatos garantidores da qualidade. Assim sendo, não se pode negar que todo esse processo de manutenção ou adaptação de software demanda recursos humanos, materiais, e principalmente, tempo.

Por isso, no intuito de debater os conceitos de processo de desenvolvimento de software em um contexto onde a atualização de requisitos de software deveria ser mais dinâmica e mais alinhada aos processos de negócio, propomos a formalização de uma metodologia de desenvolvimento baseada no conceito de baixa acoplagem e no uso de ferramentas *Business Process Management Suite – BPMS*.

2. O Processo de Desenvolvimento de Software

O modelo de desenvolvimento de sistemas de informações mais difundido se baseia em um artefato monolítico, que traduza todos os requisitos necessários para que um software seja capaz de suportar um determinado processo ou negócio (PRESSMAN, 2011). Neste contexto, algumas etapas do desenvolvimento de software são executadas a fim de construir a solução tecnológica de acordo com os requisitos levantados na análise. A análise sistema leva em conta os aspectos computacionais que serão alvos dos esforços de desenvolvimento e implantação do software. Para Pressman (2011), o desenvolvimento de softwares pode seguir um ciclo de vida em cascata (Figura 1) iniciando na engenharia de sistemas, que motiva a análise e o projeto de desenvolvimento do novo sistema ou da melhoria em um sistema existente, passando pela codificação, testes, e manutenção. Na visão de Pressman (2011), ainda há outros modelos de processo de desenvolvimento de software, o incremental, os evolutivos e a prototipação.

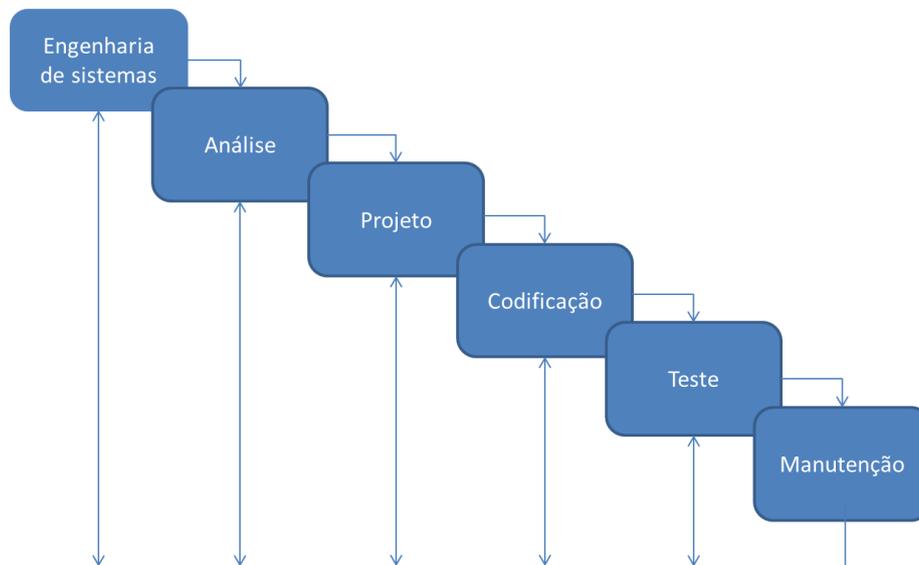


Figura 1. Ciclo de vida de software. Fonte: adaptado de Pressman (2011).

Sommerville (2007) por sua vez, também define o processo de desenvolvimento de software como um ciclo compreendendo as etapas: especificação, projeto e implementação, validação e evolução. Ainda há o modelo unificado ou *Rational Unified Process* – RUP, que inicia na modelagem do negócio, passando pelo planejamento, elaboração dos requisitos, análise e projeto que vão ditar como deve ser a implementação. Após testado e avaliado quanto aos requisitos, o software poderá ser entregue conforme podemos visualizar na Figura 2 (KROLL; KRUCHTEN, 2003).

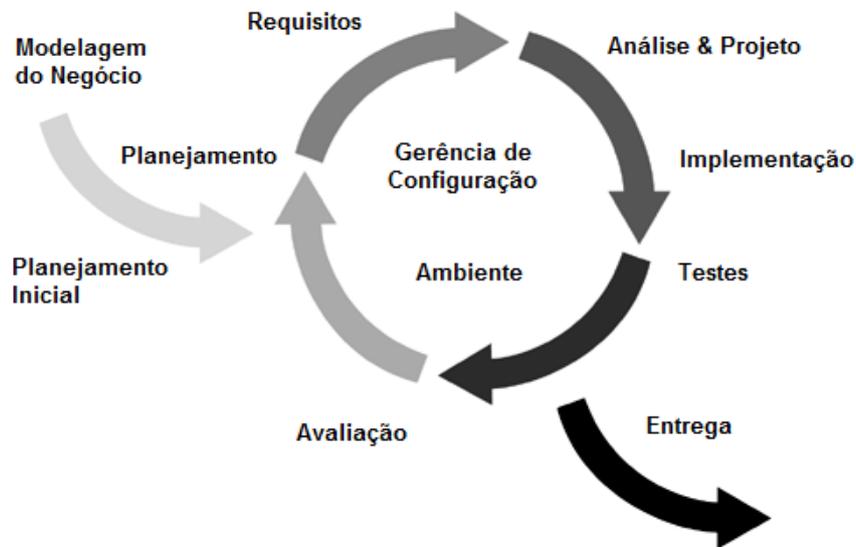


Figura 2. Desenvolvimento Iterativo no RUP. Fonte: adaptado de Kroll e Kruchten (2003)

Para Jalote (2005) um dos maiores problemas no desenvolvimento de software é a satisfação do cliente, isso entendendo o software como um produto. Essa afirmação se confirma, na opinião do autor, pelo fato dos requisitos do cliente atravessarem diversas etapas até a codificação propriamente dita, e com os problemas tradicionais de comunicação, muita ambiguidade é ocasionada. Além disso, quanto maior a complexidade do software, mais complexa deve ser a gestão do processo, aumentando o tempo de execução do projeto e também a necessidade de garantias de qualidade e seu próprio custo.

3. Desenvolvimento em Baixo Acoplamento

Desde os anos 80 os softwares eram reconhecidos como os protagonistas da nova engenharia de negócios (HAMER e CHAMPY, 1997). Mas, a reengenharia centrada em sistemas monolíticos de planejamento de recursos empresariais, os *Enterprise Resource Planning* – ERP, se mostrou pouco dinâmica e fortemente alinhada não aos processos, mas aos requisitos de cada ERP.

O paradigma de desenvolvimento de sistemas baseados no armazenamento de dados é a marca dos ERP, de modo que a análise do sistema parte da visão da necessidade de construir o software para atender às demandas de armazenamento de dados. Neste sentido, o processo ao qual um determinado sistema deveria suportar não se torna relevante aos analistas a não ser para levantar os requisitos de entrada, processamento e saída de dados (DEVENPORT, 1998). Por definição, um ERP pode conter módulos ou subsistemas (Figura 3) que se integram entre si, evitando que dados sejam duplicados e funcionalidades sobrepostas (DEVENPORT, 1998).



Figura 3. Composição de um ERP. Fonte: adaptado de Devenport (1998).

Contudo, a partir do conceito de baixo acoplamento cunhado primeiramente por Wieck (1976), que preconizava a necessidade das organizações de se tornarem mais flexíveis dados os fenômenos externos que influenciam as atividades internas, foi demasiadamente importante na visão de que a engenharia empresarial, a partir do advento dos sistemas monolíticos de ERP, encontrava demasiada resistência à mudança uma vez que estariam fortemente acoplados aos ERP (VIECK e SUTCLIFFE, 2001).

Por isso, como forma de reduzir o engessamento dos sistemas monolíticos algumas iniciativas foram introduzidas no mundo do desenvolvimento de software, tais como modularização, bibliotecas dinâmicas, componentes e arquitetura orientada a serviços, em inglês *Service-Oriented Architecture – SOA* (ERL, 2009). Estes serviços, também conhecidos por *web services*, ficam armazenados em servidores de internet ou intranet e podem ser acessados por um cliente remotamente, obedecendo aos parâmetros definidos em um contrato baseado em um protocolo definido em *Simple Object Access Protocol – SOAP* (ERL, 2009). Mais recentemente, o uso do intensivo do protocolo *Hypertext Transfer Protocol – HTTP* promoveu o surgimento de outro padrão de *web services* denominado *Representational State Transfer – REST*, que popularizou e difundiu ainda mais a filosofia de desenvolvimento de soluções de Tecnologia da Informação – TI em baixo acoplamento (ERL *et al.*, 2012).

Os *web services* podem ser utilizados em duas topologias diferentes, uma é chamada “colaboração” ou “choreografia”, onde cada serviço fica disponível para ser acessado por outro serviço (Figura 4). Outra topologia existente se baseia em um elemento central que controla o fluxo do processo e a interação com o usuário como um maestro regendo uma orquestra, daí vem a denominação “orquestração”. Neste

último, com o advento das linguagens de execução de processos (*Business Process Execution Language – BPEL*), foi possível controlar e concentrar o foco no processo (KARANDE *et al.*, 2011).

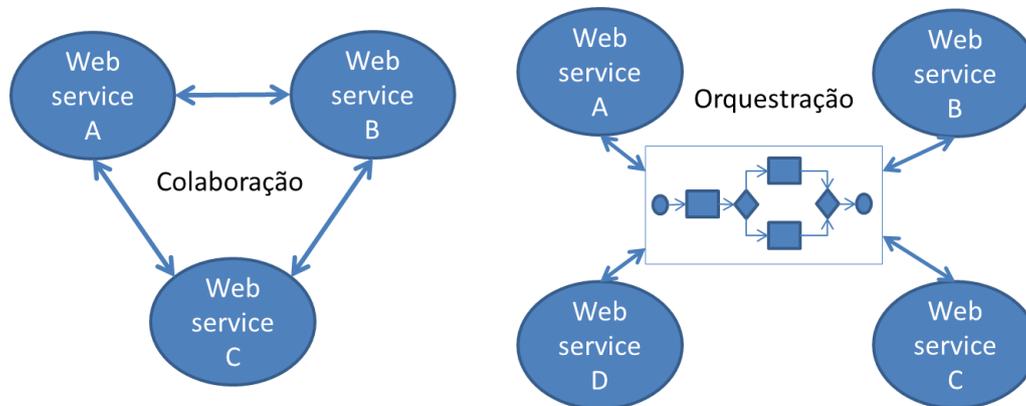


Figura 4. Topologias de integração de *web services*. Fonte: adaptado de Karande *et al.* (2011).

4. Gerenciamento de Processos de Negócios

No final do séc. XX a reengenharia se tornou célebre, pois consistia em um novo método de gestão baseado em reescrever todos os processos, substituindo-os por processos com suporte em TI, principalmente através do uso de ERP (HAMMER e CHAMPY, 1994).

De certo, o ERP aperfeiçoara e melhorara vários processos que estavam centrados em informação, tais como controle, monitoramento, análise de performances, etc. Porém, softwares se mantêm estáticos até que alguém os modifiquem, daí o problema da manutenção e atualização de software que estaria perfeitamente alinhado com os requisitos iniciais e que demandaria certo tempo para ajustá-lo a um novo contexto, uma vez que os requisitos de software mudam dinamicamente em função das mudanças do mercado ou até mesmo das estratégias organizacionais (SMITH e FINGAR, 2007). Simultaneamente, surgiram tecnologias de informação e comunicação (internet, servidores de serviços, portais, linguagens de modelagens de processos, simuladores de processos, linguagens e motores de execução de tarefas, etc.) que impulsionam ainda mais a necessidade de mudança.

Dentre os principais avanços estariam as ferramentas de modelagem de processos, que poderiam ser exportadas, lidas e convertidas por ferramentas de execução de processos. Essa integração abriu um novo mundo de possibilidades na gestão de negócios frente a um mundo dinâmico e que deram condições de formulação de estratégias agressivas e transformadoras de processos, com forte lastro na agregação de valor, tais como o envio de email e SMS aos clientes a cada etapa concluída, ou até a possibilidade do cliente poder acompanhar o andamento do processo dentro da organização (SMITH e FINGAR, 2007).

Não obstante, gestores, consultores e pesquisadores, tais como Smith e Fingar (2007), Spanyi (2003 e 2006) e até mesmo o próprio Hammer (1997) entenderam que havia sido criadas condições para gerir negócios dando ênfase às mudanças rápidas do mundo ao qual as organizações estariam inseridas, lançando mão de uma visão holística, integrando equipes e entregando melhores produtos e serviços aos clientes.

Portanto, no início do séc. XXI nasce o gerenciamento de processos de negócios, em inglês *Business Process Management* – BPM, uma disciplina que reúne técnicas de gestão, produção, organização empresarial e tecnologia da informação e comunicação (ABPMP, 2013). O principal conceito em BPM é o uso do ciclo *Plan-Do-Check-Act* – PDCA de Deming (2003) como base para o ciclo BPM, conforme ilustrado na Figura 5. No ciclo BPM, a análise representa a gestão do conhecimento para diagnosticar a necessidade de mudança no processo, em seguida, um desenho do novo processo deve ser implementado para que o controle e monitoramento da execução de cada instância do processo garanta a entrega de valor aos clientes. Ao identificar pontos de melhoria, um refinamento deve ser empreendido fazendo retornar ao planejamento inicial do ciclo ora iniciado.



Figura 5. Ciclo BPM. Fonte: adaptado de ABPMP (2013).

A implementação das soluções tecnológicas na gestão por processos apregoada nos conceitos BPM se dá por intermédio de um sistema de controle e monitoramento de processos chamado *Business Process Management Suite* – BPMS. Essas ferramentas são compostas por outras ferramentas que objetivam darem suporte a gestão dos processos, tais como: modeladores e motores de processos, motores de regras de negócios, conectores de bancos de dados, etc (ABPMP, 2013).

Por baixo da execução de processos está o conceito de orquestração de serviços humanos e tecnológicos. A orquestração destes serviços deve ser feita pelo motor de execução de processos que possui tanto o caminho para os mesmos, quanto

o mapeamento do fluxo de trabalho envolvido no processo a partir dos modelos de processos criados nas ferramentas BPMS utilizando a linguagem de modelagem e execução de processos *Business Process Modeling Notation* – BPMN (ABPMP, 2013).

Durante a modelagem dos desenhos dos processos é possível definir quais atividades deverão ser realizadas por seres humanos e quais os dados esses seres humanos deverão da entrada no sistema. Para tanto, a própria ferramenta provém um construtor de formulário que será acionado quando o processo estiver em execução. Da mesma forma, as atividades tecnológicas serão acionadas automaticamente a fim de executar os serviços instalados em servidores de apoio passando-lhes os dados obtidos durante o processo e recebendo de volta dados oriundos de bases de dados ou de outros sistemas, incluindo os ERP (ABPMP, 2013).

5. Modelo de Processo de Desenvolvimento Proposto

Objetivando adequar o processo de desenvolvimento de software aos conceitos de baixo acoplamento e orquestração via modelagem de processos, propomos um modelo de processo de desenvolvimento de software observando as características do novo ordenamento dos artefatos computacionais, a partir da definição do fluxo de trabalho e da integração das equipes, passando pela necessidade de consulta e gravação de dados nas bases de dados e nos sistemas de gerenciamento empresariais, passando pelo controle do fluxo e pelos conceitos de melhoria contínua e agregação de valor.

5.1. Etapas do processo de desenvolvimento de software

Propomos as etapas para a construção dos artefatos tecnológicos de informação e comunicação conforme a Figura 6. A primeira etapa, o desenho de processos, é a mais importante, pois é onde a equipe de análise de processos define quais atividades serão realizadas por membros das equipes de execução de processos e quais deverão ser automatizadas. O controle interno do processo deve ser feito a partir de variáveis que serão informadas pelos usuários ou por dados retornados de consultas às bases de dados. De posse dessas variáveis, e das demais variáveis que serão interagidas com os usuários, os projetistas e analistas de processos poderão construir as telas de entrada e saída de dados. A etapa de definição de regras de negócios será responsável pela análise dos critérios de desvios nos *gateways* dos fluxos de trabalho, de modo a possibilitar o armazenamento das regras em motores de processos e liberando a necessidade de inclusão dessas regras nos códigos dos artefatos, ou *web services* de acesso a dados.

A princípio, poderá haver formas de consultas diretamente às bases de dados, seja por meio de conectores presentes nas próprias ferramentas BPMS, quanto por *web services* existentes ou *Application Programming Interface* – API de sistemas previamente preparados para servirem dados pela *web*.

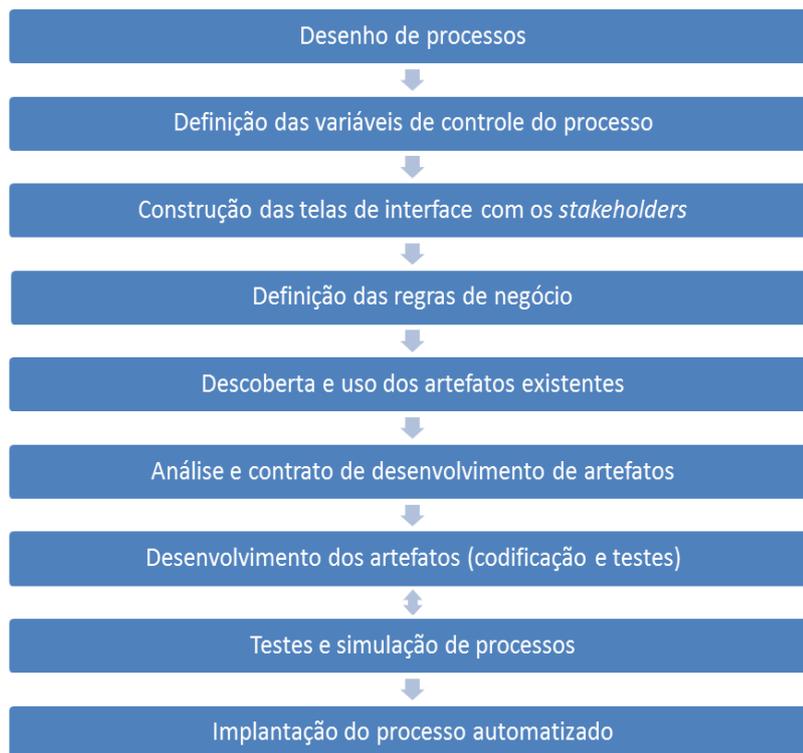


Figura 6. Etapas do processo. Fonte: elaboração própria.

Da necessidade remanescente de acesso a dados serão encomendados às equipes de desenvolvimento de artefatos de software os *web services* de acordo com contratos contendo: dados de entrada e saída e requisitos de funcionalidade. O desenvolvimento desses artefatos deverá ser condicionado a um determinado padrão de qualidade, além de haver o comprometimento com testes e simulação dos processos em ambiente não operacional a fim de detectar falhas e validar os requisitos funcionais e não funcionais previstos.

Por fim, a ferramenta BPMS será colocada em produção e a qualquer momento, sob demanda, manutenções podem ocorrer para garantir o perfeito funcionamento dos processos em relação aos artefatos de serviços automatizados.

5.2. Análise e desenvolvimento dos artefatos

Os artefatos de automatização de processos e de acesso a dados a serem utilizados pelas ferramentas BPMS deverão ser objeto de um esforço de análise, projeto e garantia de qualidade de desenvolvimento de software. Para tanto, a equipe de analistas de processos, em consórcio com os analistas de sistemas, deverá elaborar um contrato contendo: (i) requisitos funcionais e não funcionais; (ii) tipos de dados de entrada e de saída; (iii) análise de dados envolvidos nos serviços e (iv) casos de testes para cada artefato. Na Figura 7 dispomos os papéis dos principais atores nesse contexto de desenvolvimento de artefatos para suportar os processos de negócios automatizados e orquestrados pelas ferramentas BPMS.

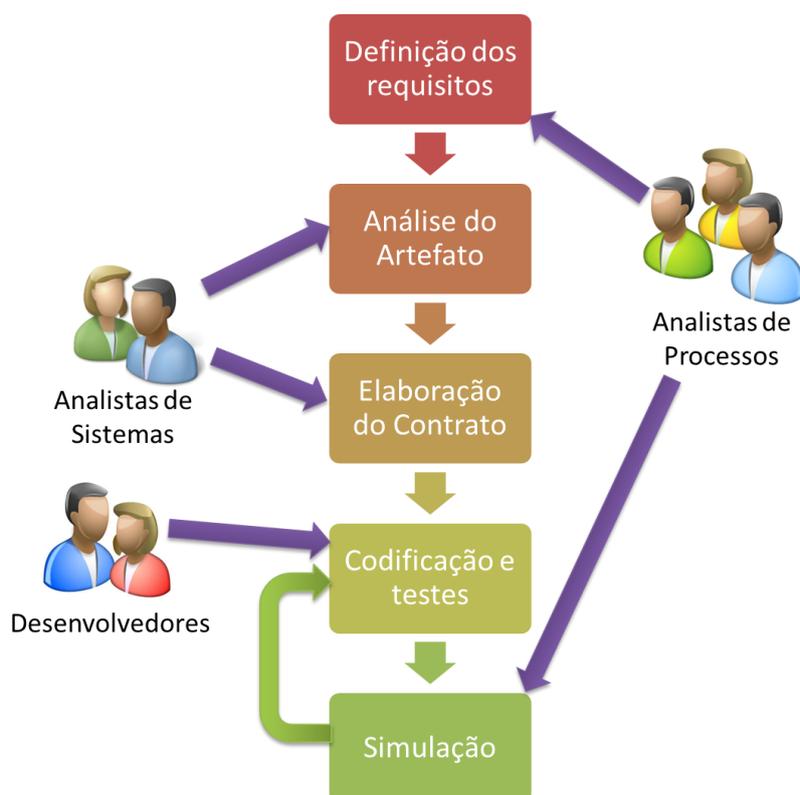


Figura 7. Atores e seus papéis no processo de artefatos para BPMS. Fonte: elaboração própria.

Os requisitos funcionais deverão indicar que dados serão retornar através dos artefatos, assim como que tipo de transformação de dados deverão ocorrer a partir dos dados de entrada. Os requisitos não funcionais terão principal aspecto na segurança da informação e na performance.

Quanto às bases de dados de suporte ao processo, essas poderão existir como parte de outro sistema de informação ou também será objeto de criação de bases específicas. No caso de bases de dados existentes, haverá a necessidade do conhecimento de tais bases por parte da equipe de análise a fim de viabilizar o trabalho de desenvolvimento específico para cada interação com tais bases existentes. Neste caso, talvez seja necessário que os analistas façam uma engenharia reversa para descobrir a função de cada elemento da base de dados ao ponto de impedir que armazenamentos sejam feitos contra os projetos iniciais, principalmente em caso de ERP que ainda se encontrarem em operação.

Outro ponto relevante nesse modelo é a criação de casos de testes pela equipe de análise de sistemas juntamente com a equipe de análise de processos a fim de garantir que o desenvolvimento dos artefatos tenha como ser testado e validado.

Uma vez entregue a primeira versão do artefato, uma simulação do mesmo em funcionamento no processo deve ser empreendida para que se garanta a qualidade e a aderência do artefato de acordo com os requisitos originais.



Os métodos e técnicas de desenvolvimento de software, pelas características de dinamismo dos conceitos de gerenciamento de processos de negócios, devem ser os mais ágeis possível, chegando ao ponto de avaliar se os papéis de análise de processos e análise de sistemas devam ser desempenhados pelas mesmas equipes de profissionais, ao ponto de evitar ambiguidades na interpretação dos requisitos e reduzir o tempo de execução do próprio processo de desenvolvimento dos artefatos.

6. Considerações finais

O desenvolvimento de software para suporte ao gerenciamento de processos de negócios, dada a característica do próprio modelo de gestão por processos de ter como característica básica a necessidade de observar o baixo acoplamento de recursos humanos e tecnológicos, deve ser adaptado em relação aos métodos tradicionais de desenvolvimento de softwares já consagrados, sem, com isso, negar que exista espaço para os métodos mais tradicionais, pois ainda há de haver necessidade de desenvolvimento de sistemas mais adensados e até monolíticos para áreas da computação e até para outros modelos de gestão.

Portanto, o desenvolvimento em baixo acoplamento é uma necessidade do modelo de gestão por processos, sendo ainda possível imaginar outros sistemas e ferramentas com suas partes altamente acopladas, como os utilitários para escritórios ou os sistemas especializados, por exemplo.

Como proposta, foi sugerido neste trabalho um modelo de processo de desenvolvimento de software para dar conotação finalística ao baixo acoplamento e ao desenvolvimento de artefatos para dar suporte ao gerenciamento de processos de negócios, de modo que foi introduzida a figura de analistas de processos, encarregados de desenhar os processos, ligando os usuários aos sistemas de armazenamento de dados de uma forma ordenada e padronizada.

Com o advento das ferramentas BPMS a orquestração de serviços tecnológicos, também conhecidos como *web services* foi levada a, em última análise, possibilitar o acoplamento centralizado mas controlado. Ou seja, o fluxo do trabalho agora já pode ser objeto de uma modelagem onde o objetivo principal é a melhoria do processo baseado num ciclo BPM. Essa orquestração de *web services*, aliada ao controle do processo entre os *stakeholders* proporciona uma dinamização das funcionalidades finais da parte tecnológica, e resolvendo um problema descoberto no final do séc. XX, o engessamento dos processos pela adoção do conceito de ERP.

Referências

- ABPMP. (2013) “BPM CBOK: guia para o gerenciamento de processos de negócio; corpo comum de conhecimento. Versão 3.0”: ABPMP.
- Davenport, T. H. (1998) “Putting de enterprise into the enterprise system”. Harvard Business Review. pp. 1221-1231, jul./ago.
- Deming, W. E. (2003) “Saia da crise: as 14 lições definitivas para controle da qualidade”. São Paulo: Futura.



- Erl, T.; Carlyle, B.; Pautasso, C; Balasubramanian, R. (2012) "SOA with REST: principles, patterns & constraints for building enterprise solutions with REST". Prentice Hall.
- Hamel, G.; Prahalad C. K. (1995) "Competindo pelo futuro: estratégias inovadoras para obter o controle do seu setor e criar os mercados de amanhã". Rio de Janeiro: Campus.
- Hammer, M.; Champy, J. (1994) "Reengenharia: Revolucionando a empresa em função dos clientes, da concorrência e das grandes mudanças de gerência". 23 ed. Rio de Janeiro: Campus.
- Hammer, M. (1997) "Além da reengenharia : como organizações orientadas para processos estão mudando nosso trabalho e nossas vidas". Tradução de Ana Beatriz Rodrigues e Priscilla Martins Celeste. Rio de Janeiro: Campus.
- Jalote, P. (2005) "An Integrated Approach to Software Engineering". 3 ed. New York: Springer.
- Karande, A.; Karande, M. e Meshram, B.B. (2011) "Choreography and orchestration using business process execution language for soa with web services". IJCSI International Journal of Computer Science Issues, Vol. 8, Issue 2, Mar.
- Kroll, P., Kruchten, P., (2003) "The Rational Unified Process Made Easy: A Practitioner's Guide to the RUP". Addison Wesley.
- Weick, K. E. (1976). "Educational organizations as loosely coupled systems". Administrative Science Quarterly, 21, 1–19.
- Weick K. E. e Sutcliffe K. M. (2001) "Managing the unexpected: assuring high performance in an age of complexity". San Francisco, CA: Jossey-Bass.
- Pressman, R.S. (2011) "Engenharia de Software: Uma Abordagem Profissional". 7 ed. McGraw-Hill.
- Smith, H.; Fingar, P. (2007) "Business process management: the third wave". Tampa, FL, USA: Meghan-Kiffer Press.
- Sommerville, I. (2007) "Engenharia de Software", 8 ed. Pearson Addison-Wesley.
- Spanyi, A. (2003) "Business process management is a team sport: play it to win!". Anclote Press.
- Spanyi, A. (2006) "More for less: the power of process management". Meghan Kiffer Press.