

Os princípios ágeis na dinâmica do desenvolvimento distribuído de software

Alexandre L'Erario¹

¹Coordenação de Informática – Universidade Tecnológica Federal do Paraná (UTFPR)
Cornélio Procópio, Paraná CEP 86300-000, Brasil

alerario@utfpr.edu.br

***Abstract.** The objective of this paper is to show how agile processes can contribute to the dynamics of the distributed software development. In this research, the model M3DS, presented by L'Erario (2007) was configured in an organization of global software development. Thus it was possible to identify in which states of the model M3DS agile methods could be used. The result indicates the impact of using agile processes related to steps in the process of distributed development.*

***Resumo.** O objetivo deste artigo é apresentar como os processos ágeis podem contribuir na dinâmica do desenvolvimento distribuído de software. Nesta pesquisa, o modelo M3DS, apresentado por L'Erario (2007) foi configurado em uma organização de desenvolvimento global de software. Desta maneira foi possível identificar em quais estados do modelo M3DS os métodos ágeis poderiam ser utilizados. O resultado indica o impacto do uso de processos ágeis relacionado a etapas do processo de desenvolvimento distribuído.*

1. Introdução

O gerenciamento ágil (*Agile Project Management* – APM) pode ser considerado uma alternativa para a gestão projetos. Griffiths (2005) indica em sua publicação as diferenças básicas entre gestão ágil de projetos e o modelo tradicional de gestão. Estas diferenças centram-se, em sua maioria, em mecanismos sinérgicos de gerenciamento, propondo uma integração e interação mais eficaz entre os *stakeholders* de um determinado projeto. Alguns princípios, da metodologia ágil parecem que são incompatíveis com o desenvolvimento distribuído de software. Entretanto, pesquisas como apresentadas por Paasivaara (2006) e Smits (2007) sugerem que há compatibilidade com algumas restrições.

Este artigo tem o objetivo de explorar o gerenciamento ágil de projetos na dinâmica do desenvolvimento distribuído de software. Um estudo de caso foi configurado no M3DS (L'ERARIO, 2007) para que pudesse ser identificada em quais estados do desenvolvimento distribuído a metodologia ágil pode ser utilizada e quais os possíveis impactos.

2. O modelo de dinâmica M3DS

A dinâmica de um ambiente de DDS revela o funcionamento da rede de produção, da sua concepção inicial até seu encerramento. Porém, abordar individualmente os sites na rede pode não revelar a influência que o projeto tem sobre a existência na rede.

Portanto, a dinâmica da rede, representada na figura 1, aborda conjuntamente dois componentes fundamentais em um ambiente de DDS. O primeiro componente indica a dinâmica de uma unidade de produção de software (site). O segundo elemento é atrelado ao projeto desenvolvido em uma rede de produção de software. Sobre esta dinâmica, há uma intersecção que ocorre quando o site precisa desenvolver um subproduto de rede, e há uma instância do projeto para vários sites. Os estados são descritos com mais detalhes em L'Erario (2009).

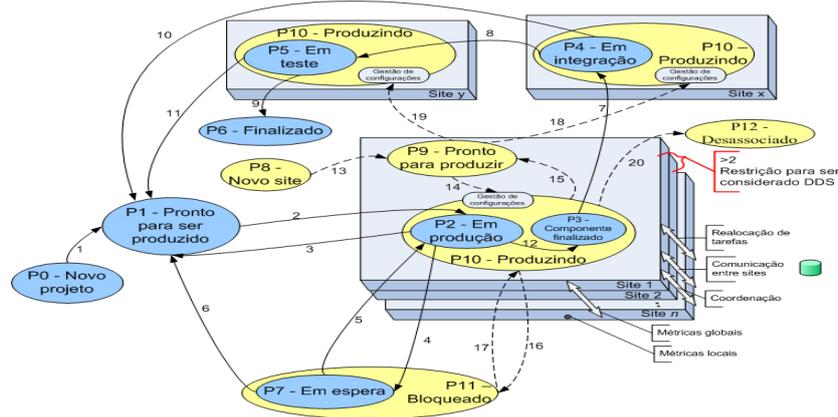


Figura 1 - M3DS

3. Métodos ágeis

Em 2001, houve um marco neste estilo de gerenciamento de projetos, quando Kent Back e mais 16 signatários lançaram o movimento ágil denominado de “*Manifesto for Agile Software Development*”. Segundo Turk (2002), a organização *agile alliance*, mantém uma lista de princípios do desenvolvimento ágil de projetos. Estes princípios, citados na tabela 1, basicamente reduzem a documentação e estimulam a interação informal entre os membros do projeto.

Tabela 1 - Princípios dos processos ágeis

| ID | Princípio |
|----|---|
| 1 | Nossa maior prioridade é satisfazer o cliente, mediante a entrega rápida e contínua de software; |
| 2 | Empresários e desenvolvedores devem trabalhar juntos diariamente ao longo de todo o projeto. |
| 3 | As evoluções dos requisitos são bem vindas, mesmo durante o desenvolvimento. |
| 4 | Entregar software funcionando com mais frequência. |
| 5 | O trabalho de software é a principal medida do progresso |
| 6 | Construa os projetos em torno de pessoas motivadas. Dêem-lhes o ambiente e o apoio de que necessitam, e confie neles para fazer o trabalho. |
| 7 | As melhores arquiteturas, requisitos e designs emergem de equipes auto-organizadas. |
| 8 | O método mais eficiente e eficaz de transmitir informações entre os indivíduos em um projeto de desenvolvimento é a conversa face-a-face. |
| 9 | Processos ágeis promovem o desenvolvimento sustentável. |
| 10 | Atenção contínua a excelência da técnica e bons designs aumentam a agilidade |
| 11 | Simplicidade é essencial |
| 12 | Equipes de projeto avaliam sua eficácia em intervalos regulares e ajustam as suas conformidades. |

4. Metodologia

A metodologia empregada neste trabalho é o estudo de caso, baseado nos delineamentos apresentados por Yin (2005). O caso investigado foi uma organização multinacional, que desenvolve software para tecnologia embarcada. A revisão literária sobre APM e

DDS possibilitou a construção do protocolo de pesquisa - conceito norteado por Yin (2005) – que foi aplicado em campo com o objetivo de gerar o resultado final.

5. Estudo de caso

Antes da aplicação do estudo de caso, o modelo M3DS foi mapeado na organização em questão. Para tanto, mapeou-se as atividades do desenvolvimento distribuído de software utilizado pela organização, nos estados do modelo M3DS. Nesta etapa, cada estado agregou um conjunto de atividades do processo de desenvolvimento. Após este mapeamento, uma análise sobre a utilização dos princípios, na organização em questão foi efetuada.

Uma análise dos princípios do desenvolvimento ágil sobre a dinâmica do desenvolvimento distribuído de software revelou que para cada princípio, um conjunto de estados e conseqüentemente transações devem ser disparados sobre certas circunstâncias. Estas circunstâncias revelaram as restrições do gerenciamento ágil em projetos distribuídos de software.

Para que o princípio 1 e 4, indicados na tabela 1, sejam almeçados e o produto seja entregue a sequencia de estados P0, P1, P2, P3, P4, P5 e P6 precisa ser concluída. O ciclo de P0 a P6 é longo. Há uma grande trajetória constituída de um conjunto variado de atividades que aumentam a latência entre o pedido do software e sua entrega. Foi identificado que o estado P1 tem grande influencia na velocidade da entrega. Se as atividades desenvolvidas por P1 utilizarem as mesmas bases tecnológicas de projetos anteriores, o processo de desenvolvimento da arquitetura do software tende a fluir com maior naturalidade. Conseqüentemente, os estados P2, P3, P4 e P5 compartilham da mesma restrição.

A distribuição pode ter duas conseqüências do ponto de vista do desenvolvimento ágil. A primeira é acelerar o processo de desenvolvimento, caso as atividades alocadas em P1 a P5 sejam executadas por uma equipe com experiência. Entretanto pode acontecer o oposto, principalmente quando P1 precisa consolidar alguma plataforma de desenvolvimento ou consolidar uma tecnologia.

6. Conclusões

Neste caso, foi notado que quando o projeto atinge certo ponto de maturidade, após consolidação da arquitetura, os conceitos de gerenciamento ágil de projetos podem ser empregados. Após a definição da arquitetura, que já deve ter certo grau de estabilidade, os desenvolvedores praticamente executam o restante do projeto com um grau maior de interação, sem muito formalismo.

As tecnologias, plataformas e arquitetura da solução precisam ser amplamente difundidas e conhecidas para que os desenvolvedores passem a utilizar alguns princípios de APM. A figura 2 ilustra como o APM pode ser utilizada. Somente após consolidação da arquitetura e da plataforma de desenvolvimento, em um novo projeto é possível que parte do desenvolvimento utilize métodos ágeis.

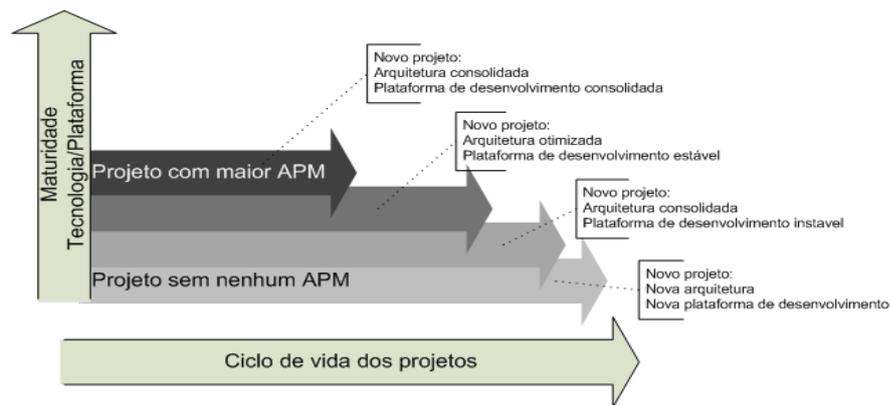


Figura 2 - Ciclo de vida dos projetos com o uso da APM

Este caso em particular, comprovou que a APM pode ser utilizada juntamente com o DDS sobre certas circunstâncias. Entretanto, esta pesquisa pode restringir a generalização do modelo – fato constatado pela metodologia adotada - o que pode tornar inválido o esquema apresentado na figura 2 para outras organizações. Como trabalho futuro, este mesmo protocolo de pesquisa deverá ser aplicado em outras organizações.

Referências

- GRIFFITHS, M., "Teaching agile project management to the PMI" Agile Conference, 2005. Proceedings , vol., no., pp. 318-322, 24-29 July 2005
- L'ERARIO, Alexandre, PESSÔA, Marcelo Schneck de Paula. An Analysis of the Dynamics and Properties of the Distributed Development of Software Environments: A Case Study. Software Engineering Research and Practice 2007: 471-477
- L'ERARIO, Alexandre, PESSÔA, M. S. P., LAURINDO, F. J. B. A gestão do conhecimento na ruptura e integração de componentes no processo de desenvolvimento distribuído de software In: ALTEC - Seminário Latino Ibero-americano de Gestión Tecnológica, 2009, Cartagena de Indias.
- PAASIVAARA, M. and Lassenius, C. Could Global Software Development Benefit from Agile Methods?, in Global Software Engineering, 2006. ICGSE'06. International Conference, p{ 109--113 }, 2006.
- SMITS, H. and Pshigoda, G. 2007. Implementing Scrum in a Distributed Software Development Organization. In Proceedings of the AGILE 2007 (August 13 - 17, 2007). AGILE. IEEE Computer Society, Washington, DC, 371-375. DOI= <http://dx.doi.org/10.1109/AGILE.2007.34>
- TURK, D., R. France, and B. Rumpe (2002, May). Limitations of agile software processes. In Third International Conference on eXtreme Programming and Agile Processes in Software Engineering (XP 2002), Alghero, Sardinia, Italy.
- YIN, Robert K.. Estudo de caso: planejamento e métodos. 3. ed. Porto Alegre: Bookman, 2005. 212 p.