

Análise dos métodos de gerenciamento ágil de projetos segundo os níveis de maturidade em melhoria contínua

Luís Fernando Magnanini de Almeida

Department of Production Engineering – Universidade Federal de São Carlos (UFSCar).
Rod. Washington Luiz, km 235, ZipCode: 13565-905 – São Carlos - SP/ Brazil.
Telephone: 55 16 33518237
e-mail: luisfernando@dep.ufscar.br

Daniel Jugend

Department of Production Engineering – Universidade Estadual Paulista (UNESP).
Av. Eng. Luiz Edmundo Carrijo Coube, 1401. ZipCode: 17.033-360. Bauru – SP/ Brazil.
Telephone: 55 14 31036122
e-mail: daniel@feb.unesp.br

Sérgio Luís da Silva

Department of Information Sciences – Universidade Federal de São Carlos (UFSCar).
Rod. Washington Luiz, km 235, ZipCode: 13565-905 – São Carlos - SP/ Brazil.
Telephone: 55 16 33518374
e-mail: sergiol@ufscar.br

José Carlos de Toledo

Department of Production Engineering – Universidade Federal de São Carlos (UFSCar).
Rod. Washington Luiz, km 235, 13565-905 – São Carlos - SP/ Brazil.
Telephone: 55 16 33518237
e-mail: toledo@dep.ufscar.br

Daniel Capaldo Amaral

Department of Production Engineering – São Carlos Engineering School – University of São Paulo (USP).
Av. Trabalhador São Carlense, 400, ZipCode: 13.566-590, São Carlos - SP/ Brasil.
Telephone: 55 16 3373 8289
e-mail: amaral@sc.usp.br

Análise dos métodos de gerenciamento ágil de projetos segundo os níveis de maturidade em melhoria contínua

Resumo

Um dos princípios do gerenciamento ágil de projetos é a reflexão sobre as atividades realizadas, de modo que equipe que o aplica se torne mais eficaz, se adaptando conforme as necessidades percebidas. Esse princípio está diretamente ligado à melhoria contínua e ao aprendizado organizacional, sendo esse aspecto evidenciado pelos autores que tratam desse tema. No entanto, em pesquisa bibliográfica, não foram encontrados estudos que demonstrassem a maior eficiência dos métodos de gerenciamento ágil com relação à melhoria contínua e nem que os classificasse segundo algum modelo de maturidade. Este trabalho apresenta uma comparação teórica entre três métodos de gerenciamento ágil (XP, SCRUM e IVPM2) tendo como referência um modelo de maturidade em melhoria contínua proposto por Bessant et al. (2001). Os resultados apontam para um elevado grau de maturidade dos três métodos e também mostram algumas dificuldades, como, por exemplo, a difusão da melhoria para a organização como um todo.

Abstract

One of the principles of Agile Project Management is the critical observation about the work done and the design of new strategies to make the staff more effective, adapting the process as the perceived needs. This principle is directly linked to continuous improvement and organizational learning, and this is emphasized by the authors dealing with this issue. However, in literature review, no studies were found that show the efficiency of agile methods regarding to continuous improvement nor that rank those according to some maturity model. This paper presents a theoretical comparison between three agile methods (XP, SCRUM and IVPM2) using the maturity model for continuous improvement proposed by Bessant et al. (2001). The results indicate a high degree of maturity in all methods analyzed and also shows some difficulties, for example, the dissemination of the improvement to in the organization.

1. Introdução

As mudanças no cenário produtivo contemporâneo, em particular a maior busca pela inovação, as mudanças constantes de requisitos dos produtos devido a mudança nos desejos dos clientes e ciclo de vida de produtos menores têm levado as empresas a competirem por meio de projetos complexos e inovadores, com processos de desenvolvimento flexíveis (ALMEIDA, 2012).

Autores como Dawson e Dawson (1998) e Suikki et al. (2006) afirmam que os métodos tradicionais de gerenciamento de projetos (GP) não são tão adequadas para esse novo contexto, principalmente devido as limitações no planejamento e controle e sua demasiada generalidade, isto é, a aplicação indiscriminada das mesmas ferramentas e técnicas para qualquer tipo de projeto.

Como alternativa aos métodos tradicionais de GP foram desenvolvidos, em meados da década de 1990 e início dos anos 2000, os novas alternativas, agrupadas sobre a denominação de “métodos ágeis para o gerenciamento de projetos”. Esses, além de se proporem a oferecer alternativas para as dificuldades de planejamento e controle em ambientes de rápidas mudanças e contemplar as particularidades de cada projeto, também buscam ajudar no aprendizado organizacional (AO) e melhoria contínua (MC), tanto do processo como do produto.

Nesse sentido, Highsmith (2004) aponta o aprendizado contínuo como necessário para a execução de atividades complexas no decorrer do projeto, e enfatiza que no Gerenciamento Ágil de Projetos (GAP), ele acontece durante as interações, particularmente nos ciclos de planejamento, execução e controle.

Para Cohn (2005) e Boehm (2003), ao revisar o plano constantemente, durante o planejamento evolutivo, a equipe de projetos aprende com as mudanças até o ponto ótimo de congelar o plano de projeto e fazer entregas de valor ao cliente.

Contudo, apesar de os autores que tratam do GAP justificarem como alguns de seus diferenciais o AO e a MC, não foram encontrados, na revisão bibliográfica realizada, estudos robustos que comprovassem a sua eficiência nesses aspectos.

Os métodos de GAP abordados neste trabalho são: o SCRUM (SCHWABER, 2004), o Extreme Programming – XP (BECK, 2000) e o Iterative and Visual Project Management Method – IVPM2 (AMARAL et al., 2011). O objetivo principal do estudo é realizar uma

análise desses três métodos tendo como base os conteúdos prescritos, nas publicações básicas desses métodos, e como referência o conceito de níveis de maturidade em melhoria contínua, propostos por Bessant et al. (2001), de modo a analisar se eles contribuem realmente para a MC e para o AO. Como objetivos secundários situa as atividades e práticas desses métodos com relação as fases do projeto segundo o PMI (2008) para verificar em quais etapas o aprendizado e a melhoria contínua ocorrem. Como método de pesquisa é utilizado a revisão bibliográfica e análise de conteúdos.

2. Métodos de Gerenciamento Ágil de Projetos

De forma a se contrapor as críticas sobre a aplicabilidade dos métodos tradicionais de gerenciamento de projetos em ambientes inovadores, nos quais a mudança constante é comum, alguns novos métodos de GP foram desenvolvidos, tendo como principal diferencial a autogestão, simplicidade, uso de iteração, envolvimento do cliente e uso da visão no lugar do escopo (AMARAL et al., 2011).

Esses novos métodos se autodenominaram métodos ágeis para o gerenciamento de projetos (EDER et al, 2010), sendo os principais, segundo Almeida (2012): Scrum (SCHWABER; BEEDLE, 2001; SCHWABER, 2004), Lean Software Development (POPPENDIECK e POPPENDIECK, 2003), Crystal (COCKBURN, 2004), Feature Driven Development – FDD (PALMER e FELSING, 2002), Adaptive Software Development (HIGHSMITH, 2000), Dynamic System Development Method – DSDM (STAPLETON, 1997), Extreme Programming (BECK, 2000), Iterative and Visual Project Management Method - IVPM2 (CONFORTO, 2009).

De todos esses métodos, três foram selecionados para mais discussões e análises no atual artigo, utilizando como critérios seus empregos disseminados – como é o caso do SCRUM e o XP, ou por sua completude e desenvolvimento recente, como o IVPM2.

2.1. SCRUM

O método SCRUM, criado por Jeff Sutherland, Ken Schwaber e Mike Beedle (SCHWABER, 2004), está ilustrado na figura 1.

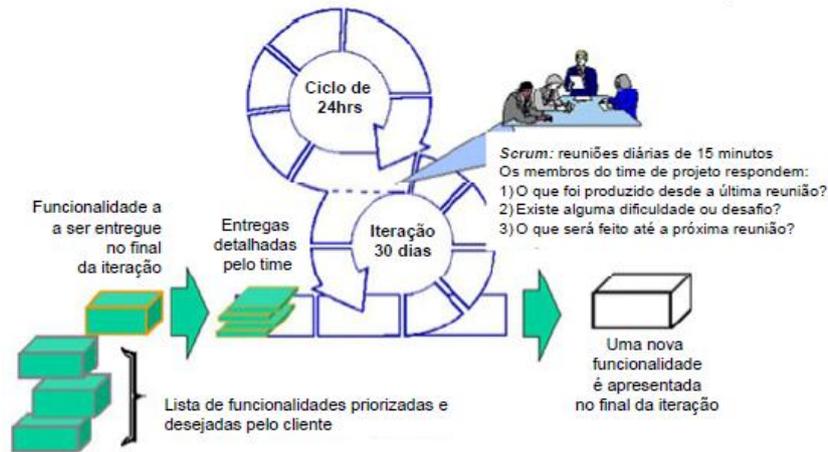


Figura1: Modelo de referência para gerenciamento com SCRUM. **Fonte:** Conforto (2009)

Os principais elementos do SCRUM, segundo SCHWABER (2004), são:

- **Product Backlog:** conjunto de requisitos definidos com investidores, parceiros e outras pessoas envolvidas no projeto, formando uma lista de atividades a serem desenvolvidas durante o empreendimento;
- **Sprint Planning Meeting:** reunião que define o número e quais atividades serão desenvolvidas na Sprint (ciclo de desenvolvimento no gerenciamento iterativo o qual geralmente tem duração de 15 dias);
- **Sprint Backlog:** subconjunto do Product Backlog é uma lista de atividades que serão desenvolvidas na Sprint.
- **Sprint:** são executados os itens do Product Backlog, sua duração são cerca de 15 a 30 dias;
- **Daily Scrum:** rápida reunião que ocorre todos os dias entre os membros da equipe, definindo quais tarefas serão realizadas no dia na qual também se discute os resultados das atividades do dia anterior;
- **Sprint Review Meeting:** acontece após termino da Sprint, na qual a equipe discute erros e acertos.

Segundo Schwaber (2004), o método também possui algumas outras particularidades. A primeira são os papéis do SCRUM, que são basicamente três: Product Owner; Time de Projetos (Project Team); Scrum Master.

O Product Owner é responsável por representar os interesses do cliente e servir como suporte no projeto e no sistema resultante. Ele também deve ajudar na confecção de uma lista de requisitos denominado Product Backlog. Nessa lista, estão as funcionalidades priorizadas segundo o valor para o produto final, devendo ser as com maior valor as primeiras a serem realizadas (SCHWABER, 2004).

O Time de Projetos (Project Team) é responsável por desenvolver uma funcionalidade. Os times são auto geridos, auto-organizados, e multifuncionais, sendo responsáveis por transformar os requisitos em funcionalidades, assumindo uma responsabilidade coletiva pelo sucesso do projeto (SCHWABER, 2004).

O Scrum Master é responsável pelo sucesso do processo do SCRUM, ensinando o processo para todos os envolvidos no projeto, também deve conseguir a conciliação entre a cultura da empresa e o método e entregar os resultados esperados, assegurando que todos sigam as regras e práticas do Scrum (SCHWABER, 2004).

Desse modo, percebe-se que existe uma forte interação entre o cliente (na figura do product owner), scrum master e o time de projetos. Os produtos são melhorados a cada iteração, seja pela introdução de novas funcionalidades, seja pela validação dos clientes ao fim de um ciclo. Já o processo, também é melhorado e adaptado, principalmente nas reuniões de Sprint planning e Sprint Review.

A troca de conhecimentos entre o cliente e a equipe acontece explicitamente na Sprint planning e na Sprint review. De forma indireta, ele também ocorre durante a confecção do product backlog. Já o aprendizado da equipe ocorre no dia a dia, principalmente durante a daily scrum e as Sprint review e nas conversas informais para resolução de problemas.

2.2. Extreme Programming (XP)

O Extreme Programming, ou XP, é um método de gerenciamento ágil que apresenta cinco fases: exploração, planejamento, iterações para entrega, produção e manutenção, conforme apresentado na figura 2, sendo este método, assim como o SCRUM, exclusivo da área de software.

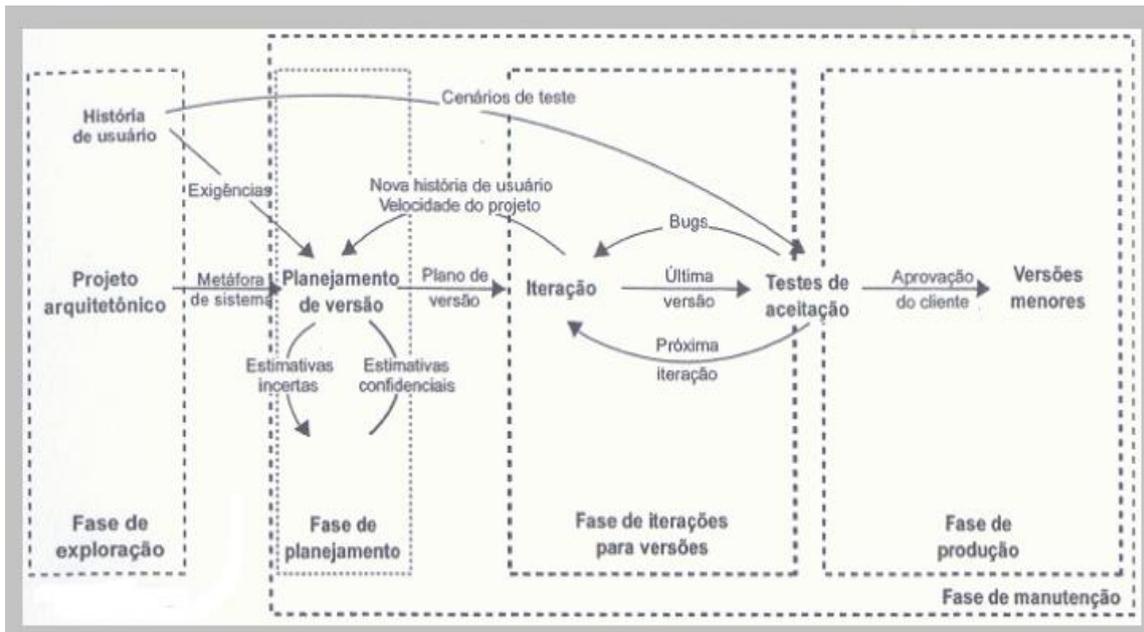


Figura2: Modelo de referência para gerenciamento com XP. Fonte:Beck (2000)

Segundo Beck (2000), as principais atividades de cada fase são:

- **Fase de Exploração:** é a primeira fase pela qual passa o projeto, que inclui o desenvolvimento da base da arquitetura e das primeiras histórias dos usuários, sendo aconselhado que os usuários forneçam histórias suficientes para a execução de uma boa primeira versão. Essas histórias de usuários são um guia básico do método XP, fornecendo requisitos em alto nível para o sistema e sendo cruciais para o processo de planejamento. Ele também funciona como uma forma de “escopo” (AMBLER, 2002);
- **Fase de Planejamento:** logo após a fase de exploração é a de planejamento, que tem como principal propósito o alinhamento entre o cliente e a equipe de projetos sobre a data na qual o menor e melhor conjunto de histórias de usuários será implementada (BECK, 2000).
- **Fase de iterações para versões:** nesta fase é que ocorre o principal trabalho do projeto XP que são as iterações, sendo que, neste momento, ocorre o maior trabalho de desenvolvimento, incluindo modelagem, programação, teste e integração. Também nessa fase que acontece grande parte do aprendizado, tanto na troca de informações entre a equipe e o cliente (em caso de dúvidas), quanto entre os membros da própria equipe, pelo aprendizado e comunicação das lições aprendidas durante o desenvolvimento das histórias (BECK, 2000).

- **Fase de Produção:** nessa fase o foco é a garantia de que o software esteja pronto para entrar em produção, isto é, com atividades como testes de sistema, de carga e de instalação (BECK, 2000).
- **Fase de manutenção:** segundo Ambler (2002) essa fase é realmente o “estado normal” dos projetos XP, pois o desenvolvimento continua no decorrer do tempo, compreendendo as fases de planejamento, iteração para versões e produção.

Logo, verifica-se que o software é melhorado constantemente, assim como no SCRUM, e que boa parte do aprendizado ocorre por trocas de conhecimento entre os membros da equipe durante a fase de iterações.

O aprendizado com o cliente ocorre, em sua maior parte, entre as fases de exploração e planejamento e durante a fase de produção, na aprovação ou não pelo cliente da parte do produto entregue. Outro aspecto interessante é que a velocidade para a execução das histórias (formato como o escopo é transmitido a equipe, com situações de uso) é constantemente revisada devido às melhorias no processo de produção, conferindo um caráter de melhoria contínua tanto no desenvolvimento de produto quanto no processo em si.

Como principais críticas, têm-se o fato de que o desenvolvimento depende muito do cliente, como na maioria dos métodos de gerenciamento ágil. Além disso, o XP é específico para software e o aprendizado organizacional no nível fora da equipe é comprometido, isto é, a equipe aprende e troca informações entre si, mas inexiste, pelo menos na teoria, troca de conhecimentos entre equipes de projetos distintos.

2.2. Iterative and Visual Project Management Method - IVPM2

O IVPM2, de criação mais recente que os anteriores, pretende ser uma alternativa para os métodos de gerenciamento ágil de projetos mais difundidos, contudo, muito específicos para a área de software. Desse modo, o método incorpora os principais princípios ágeis, como o planejamento iterativo, simplicidade, uso de ferramentas visuais, adaptando-os para serem usados tanto em projetos de produtos físicos, quanto de software e, por esta particularidade, foi escolhido para análise.

Esse método foi publicado inicialmente na dissertação de mestrado de Conforto (2009), mas, assim como no caso do Scrum e do XP, foi considerada a versão mais completa e atualizada

publicada em livro (AMARAL et al., 2011). A figura 3 ilustra as principais etapas e atividades para a implantação do método, explicadas nos tópicos seguintes.

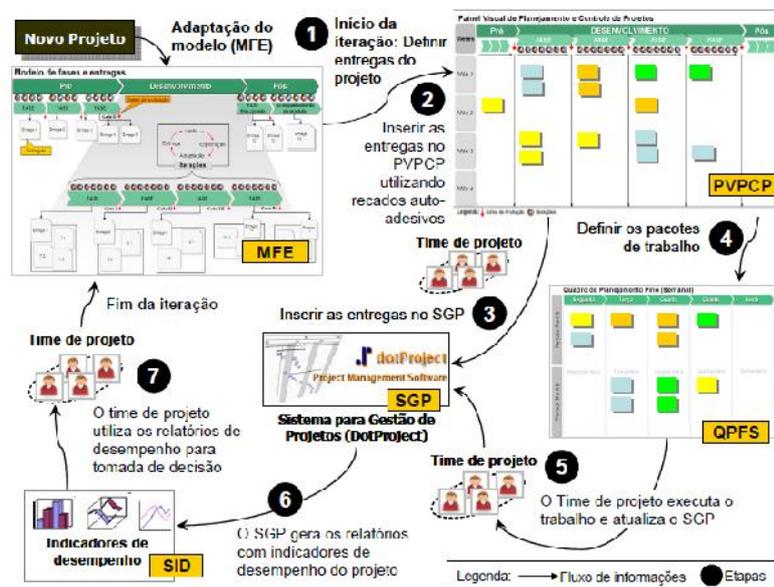


Figura 3: Atividades, etapas e fluxos no IVPM2. **Legenda:** MFE – modelo de Fases e Entregas; PVPCP – Painel Visual de Planejamento e Controle de Projetos; SGP – Sistema para Gerenciamento de Projetos; QPFS – Quadro de Planejamento Fino Semanal; SID – Sistema de Indicadores de Desempenho. **Fonte:** Conforto (2009)

- Modelo de Fases e Entregas (MFE):** é definido como um modelo de processo de negócio simplificado, permitindo uma descrição uniforme do processo de desenvolvimento na empresa, devendo conter um conjunto mínimo de padrões, suficientes para as entregas mais simples. O conjunto mínimo para a aplicação do IVPM2 contém: plano de projeto ágil; documento para avaliação de transição de fases; modelo para atas de reunião de projeto; modelo para capa de pasta de projeto; modelo de documento para avaliação e encerramento do projeto;
- Painel visual de planejamento e controle de projetos (PVPCP):** foi desenvolvido a partir de múltiplas fontes, mesclando teoria tradicional e ágil, e busca integrar conceitos de processo de desenvolvimento de produtos, visão de fases, uso de iterações e planejamento por entregas. Trata-se de um painel visual físico que contempla o planejamento e controle das entregas de projetos. No final de cada iteração, existe uma entrega, e essa é acompanhada pelo quadro visual.
- Quadro de planejamento fino semanal (QPFS):** esse quadro de apoio foi desenvolvido para planejar as atividades e pacotes de trabalho durante um espaço curto de tempo, neste caso, uma semana. Um dos benefícios desse quadro é que ele

ajuda na obtenção de resultados a cada semana, ajudando a equipe a se orientar por resultados.

- **Sistema para gerenciamento de projetos (SGP):** é um sistema para gerenciamento de projetos que utiliza uma combinação de ferramentas simples e visuais com sistemas de informação.
- **Sistema de indicadores de desempenho (SID):** o IVPM2 adota um software de gerenciamento para o armazenamento de dados dos projetos. Desse modo, pode utilizar tanto os indicadores clássicos, de tempo, custo e qualidade, além de indicadores mais voltados para o GAP, sendo eles em grande parte qualitativa, como a satisfação do cliente e a qualidade das entregas.

Os autores também dividem a implantação do método em sete etapas descritas em detalhe em Amaral et al (2011). Aqui será dado o foco apenas na etapa 7 que contém aspectos de melhoria contínua.

Essa etapa é denominada “avaliação dos resultados, tomada de decisão, aprendizado”. Nela, os resultados de uma ou mais iterações são avaliadas com apoio dos relatórios de desempenho do projeto. Nessa avaliação, deve ser abordado o que “deu certo” ou não, discutir os problemas e desafios enfrentados, os resultados alcançados e as soluções criativas que podem ser utilizadas em outras iterações e situações (AMARAL et al., 2011).

Os autores também atentam para o fato de ser mandatória a revisão dos riscos e oportunidades do projeto antes do encerramento de cada iteração, assim como a atualização do PVPCP e SGP, e a avaliação dos membros da equipe, criando um ambiente de aprendizado contínuo, motivação e reconhecimento.

Apesar da grande flexibilidade de utilização e de se embasar em vários outros métodos de GAP, o IVPM2 também apresenta limitações como, por exemplo, ser específico para as atividades de planejamento e controle de escopo e tempo e, portanto, assim como vários outros métodos de gerenciamento ágil de projetos ainda carece e terá como desafio futuro abordar as outras áreas de conhecimento propostas pelo PMI (2008), como, por exemplo, no gerenciamento de riscos, recursos humanos e aquisições.

3. Níveis de maturidade em melhoria contínua

Existem alguns modelos que definem maturidade em melhoria contínua, conforme apresentados na posterior seção 5. Nesta seção é abordado o modelo de maturidade proposto por Bessant et al. (2001), selecionado como referencial teórico para a pesquisa.

Esses autores partem do princípio de que existem rotinas-chaves e habilidades associadas com a melhoria contínua (MC), e os organizam em cinco níveis, conforme apresentado na figura 4. Essas rotinas-chaves e habilidades seriam:

- **Entender a melhoria contínua** – a habilidade de articular os valores básicos da MC.
- **Adquirir o hábito da melhoria contínua** - habilidade de gerar o envolvimento contínuo em MC.
- **Foco na melhoria contínua** – habilidade de alinhar as atividades de MC aos objetivos estratégicos da companhia.
- **Liderando o caminho** – habilidade de liderar, direcionar e dar suporte a criação de comportamentos sustentáveis de MC.
- **Alinhando a melhoria contínua** – habilidade de criar consistência entre os valores da MC e o comportamento no contexto organizacional.
- **Compartilhar a resolução dos problemas** – habilidade de levar a MC além das fronteiras da organização.
- **Melhoria contínua da melhoria contínua** – habilidade de gerenciar estrategicamente a melhoria contínua.
- **Organização que aprende** – gerar a habilidade de permitir que o aprendizado ocorra e seja absorvido em todos os níveis da organização

Esse modelo foi escolhido, pois é genérico (isto é, pode ser aplicado em várias áreas como, por exemplo, na manufatura de produtos físicos e software), foi o mais citado na revisão bibliográfica realizada além de apresentar cinco níveis de melhoria em uma forma lógica e evolutiva, conforme apresentado na figura 4.



Figura4: Níveis de maturidade em melhoria contínua segundo modelo de Bessant et al. (2001).

Bessant et al. (2001) descrevem cada nível da seguinte forma:

- **Nível 1** – Pré MC: o interesse em melhoria contínua é desencadeado por uma crise, seminário ou visita a outra organização, mas a sua implantação ocorre de modo ad hoc.
- **Nível 2** – MC Estruturada: existe um comprometimento formal na construção de um sistema que irá desenvolver a melhoria contínua através da organização.
- **Nível 3** – MC orientada a objetivos: existe o comprometimento em ligar os comportamentos “locais” em melhoria contínua com uma estratégia mais ampla de toda organização.
- **Nível 4** – MC Proativa: exista uma tentativa de desenvolver autonomia e dar mais autonomia aos indivíduos e grupos para gerenciar e dirigir seus próprios processos.
- **Nível 5** – Capacidade total em MC: A organização se aproxima do conceito de “Organização que aprende”

4. Método de Pesquisa

O objetivo da pesquisa é analisar três métodos de gerenciamento ágil de projetos sobre o prisma do nível de maturidade em melhoria contínua. Para isso, foi realizado um levantamento bibliográfico sobre modelos de maturidade em MC. A escolha dos métodos de GAP analisados se deu pelo número de citações, sendo observados apenas os principais métodos, segundo ALMEIDA (2012). A figura 5 ilustra as etapas da pesquisa.

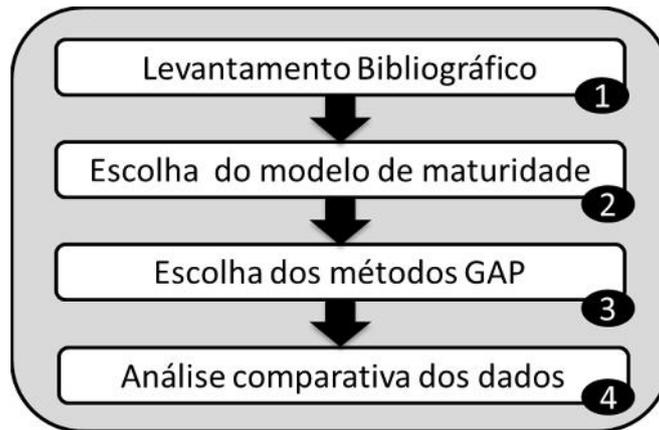


Figura5: Etapas da pesquisa

1. **Levantamento bibliográfico:** Foi feito um levantamento bibliográfico sobre modelos de maturidade em melhoria contínua, sendo utilizadas as palavras chaves: maturity; continuous; improvement.
2. **Escolha do modelo de maturidade:** Como critérios para a escolha de um modelo para ser o referencial de pesquisa, foram adotados o número de citações e ser genérico, isto é, poder ser aplicado a diversas áreas.
3. **Escolha dos métodos GAP:** Os principais métodos de GAP, segundo Almeida (2012), foram utilizados como base, passando pelo seguinte processo de seleção: o livro que continha a descrição do método serviu como base para a busca referente ao número de citações, utilizando o Google Scholar. Os dois métodos com maior número de citações foram escolhidos para participar da análise comparativa. Um terceiro método, o IVPM2, também foi selecionado por ser o único método encontrado não específico para a área de software. Os resultados da busca encontram-se na seção 5.
4. **Análise comparativa dos dados:** primeiramente, as etapas dos métodos foram classificadas segundo o modelo de maturidade em melhoria contínua escolhido, e posteriormente os três métodos GAP analisados foram comparados entre si. Para a comparação, as atividades foram classificadas segundo o modelo de Bessant et al. (2001) e distribuídas conforme as fases do projeto proposta pelo PMI (2008).

5. Resultados

Para a escolha do modelo de maturidade em melhoria contínua a ser utilizado como referencial para essa pesquisa foram utilizados dois critérios. O primeiro foi o número de citações do artigo em que ele é apresentado segundo o Google Scholar. O segundo critério foi o modelo ser genérico, isto é, não ser específico para uma área específica como, por exemplo, software. A tabela 1 ilustra os resultados dessa busca.

Autores	O modelo é genérico?	Número de citações (Google Scholar)
PAULK et al., 1993	Não	3480
BESSANT et al., 2001	Sim	201
CAFFYN, 1999	Sim	152
NIAZI; WILSON; ZOWGHI, 2005	Não	109
KWAK; WIBBS, 2002	Não	96
BURNSTEIN; SUWANASSART, 1996	Não	84

Tabela 1: Citações dos modelos de maturidade em melhoria contínua segundo o Google Scholar. **Fonte:** Autoria Própria.

Com relação aos métodos de gerenciamento ágil de projetos, também foi efetuado um processo semelhante, verificando o número de citações, desta vez dos livros sobre o método, sendo escolhidos os dois com maior número. Um terceiro método também foi incluído na análise, o IVPM2, pois ele se aplica em outras áreas que não o desenvolvimento de software. Os resultados estão presentes na tabela 2.

Desse modo, o modelo de maturidade em melhoria contínua escolhido como referência foi o proposto por Bessant et al. (2001) e os métodos de gerenciamento ágil foram o Scrum, XP e IVPM2, sendo as análises comparativas realizadas na seção 5.1.

Método	Autores	Número de citações (Google Scholar)
<i>Dynamic System Development Method – DSDM</i>	STAPLETON, 1997	261
<i>Crystal</i>	COCKBURN, 2004	430
<i>Feature Driven Development – FDD</i>	PALMER e FELSING, 2002	437
<i>Lean Software Development</i>	POPPENDIECK e POPPENDIEK, 2003	555
<i>Adaptative Software Development</i>	HIGHSMITH, 2000	619
Scrum	SCHWABER; BEEDLE, 2001	1699
Extreme Program – XP	BECK, 1999	7342

Tabela 2: Citações dos principais métodos de gerenciamento ágil de projetos (ALMEIDA, 2012) segundo o Google Scholar.

5.1. Análise comparativa de acordo com os níveis de maturidade de Bessant et al. (2001)

A primeira análise realizada é referente a presença das habilidades para a melhoria contínua em alguma prática ou fase dos métodos ágeis escolhidos para o estudo. Vale ressaltar que essa análise baseou-se exclusivamente nas práticas e atividades apresentadas pelos autores dos métodos em seus livros, não sendo verificadas a sua aplicação em campo, o que poderia evidenciar que o método não é aplicado em sua totalidade ou que adaptações foram realizadas de forma a adquirir essas habilidades.

O quadro 1 ilustra essa análise, sendo destacadas as habilidades presentes nos métodos. As células foram hachuradas quando o método possuía práticas ou atividades que pudessem levar ao desenvolvimento da habilidade em questão.

Habilidade segundo Bessant et al. (2001)	Método Ágil		
	SCRUM	XP	IVPM2
Entender a melhoria contínua			
Adquirir o hábito da melhoria			
Foco na melhoria			
Liderando o caminho			
Alinhando a melhoria contínua			
Compartilhar a resolução dos problemas			
Melhoria contínua da melhoria			
Organização que Aprende			

Quadro 1: Análise dos métodos SCRUM, XP e IVPM2 segundo as habilidades em melhoria contínua propostas por Bessant et al. (2001). **Fonte:** Autoria Própria.

A habilidade “entender a melhoria contínua” está inserida nos princípios que norteiam os métodos GAP, se manifestando tanto no desenvolvimento iterativo que exige reflexão sobre o que foi realizado, quanto pela própria valorização do aprendizado e desenvolvimento da equipe.

Já o “hábito da melhoria contínua” está fortemente ligado no GAP com algumas atividades chave como, por exemplo, os rituais do SCRUM que são mandatórios em cada iteração e o ajuste de velocidade no XP, na qual sempre se busca aumentar ou manter a velocidade por meio de melhorias dos processos Também está presente na fase de “avaliação dos resultados, tomada de decisão, aprendizado” no IVPM2. Da mesma forma, a habilidade “liderando o caminho” está presente em todos os métodos, às vezes de forma implícita pelas atribuições do gerente de projetos e da equipe, ou de forma explícita como no caso do Scrum Master.

A “melhoria contínua da melhoria contínua” também está presente em todos os métodos, pois a reflexão, aprendizado sobre os processos internos e a realização da MC é essencial para o

sucesso de atividades como o Sprint review (SCRUM), ajuste de velocidade (XP) ou nas análises e melhoria dos indicadores (IVPM2).

Percebe-se que nenhum dos três métodos possui a habilidade de “alinhar a melhoria contínua” e nem de “compartilhar a resolução dos problemas”. Esse fato ocorre devido ao alto grau de independência e certo grau de “isolamento” entre os times de projeto e à quase exclusividade de transmissão do conhecimento por meio da socialização (NONAKA e TAKEUCHI, 1997) o que dificulta a transferência de conhecimento entre os diferentes projetos. Desse modo, o compartilhamento do aprendizado entre equipes se torna o principal ponto fraco dos métodos GAP quando se trata da melhoria contínua e aprendizado organizacional.

Já o IVPM2 foi o único método que apresentou o “foco na melhoria”, isso por que, com a ajuda do software de gestão, gera análises quantitativas e qualitativas do desempenho do projeto e de processo e as utiliza para melhorar seus processos, tendo medidas que possam se alinhar com os objetivos da empresa e não apenas qualitativas do projeto. Apesar de todos os métodos realizarem melhoria contínua, o único que possui ferramentas para acompanhar o desenvolvimento dessa no tempo é o IVPM2.

Por fim, nenhum método possui a habilidade “organização que aprende”. Quando um projeto acaba e as equipes são realocadas, cada membro leva seu aprendizado para o novo time o conhecimento pode ser transmitido de forma tácita durante o novo projeto. Contudo, não existem meios formais de transferência de conhecimento entre times ou para o banco de dados da organização, não podendo a mesma ser considerada uma organização que aprende no sentido mais amplo da definição, pois está baseado na socialização e na “boa vontade” dos colaboradores em partilhar os conhecimentos adquiridos. Mais uma vez, a transferência de conhecimentos entre equipes se destaca como fator negativo do GAP.

Outra análise realizada, apresentada no quadro 2, é a classificação dos métodos de gerenciamento ágil de projetos segundo os níveis de melhoria propostos por Bessant et al. (2001). Da mesma forma, as células hachuradas são as que o método possui completamente os requisitos para aquele nível de maturidade.

Níveis de maturidade segundo Bessant et al. (2001)	Método Ágil		
	SCRUM	XP	IVPM2
Nível 1 – Pré MC			
Nível 2 – MC estruturada			
Nível 3 – Melhoria contínua orientada a objetivos			
Nível 4 – Melhoria contínua pró ativa			
Nível 5 – Organização que aprende			

Quadro 2: Análise dos métodos SCRUM, XP e IVPM2 segundo os níveis de maturidade em melhoria contínua propostas por Bessant et al. (2001).

Todos os métodos analisados apresentaram-se prioritariamente no nível 4. Isto é, se aplicadas corretamente as práticas e atividades a empresa poderia evoluir do nível 1 ao 4 de maturidade em melhoria contínua seguindo apenas a proposta original dos métodos.

Percebe-se que todos os métodos conferem autoridade ao time de projetos para gerenciar seus próprios processos e aprendizados e, assim, buscar a melhoria contínua, o que é feito de forma pró ativa conforme os requisitos para o nível 4 de maturidade. Para atingir o nível 5 o conhecimento deveria ser transmitido a toda organização, desse modo, essa dificuldade de transferência de conhecimento entre equipes, já apresentada anteriormente, impede que o GAP atinja o nível mais alto de maturidade em melhoria contínua, segundo o modelo de Bessant et al. 2001.

5.2. Análise comparativa entre os métodos ágeis segundo as fases do ciclo de vida do projeto

A última análise realizada foi a comparação das práticas de melhoria contínua com as fases do ciclo de vida de um projeto tradicional (PMBok, 2008). Essa análise foi feita para verificar em quais fases existem atividades de melhoria contínua e aprendizado organizacional e como elas estão distribuídas no ciclo de vida do projeto. Na maioria dos projetos tradicionais esse aprendizado se concentra no fim, principalmente quando no preenchimento dos documentos de lições aprendidas.

Fases do Projeto (PMBok, 2008)	Método de Gerenciamento Ágil de Projetos		
	Scrum	XP	IVPM2
Iniciação			
Planejamento			
Execução			
Controle			
Encerramento			

Quadro 3: Análise das práticas de melhoria contínua dos métodos SCRUM, XP e IVPM2 segundo as fases do ciclo de vida de projetos proposta pelo PMBoK (2008).

Na fase de iniciação, todos os métodos possuem atividades de melhoria contínua, seja para definir a melhor configuração de atuação da equipe, seja para definir quais práticas e ferramentas se adaptam melhor aos envolvidos e ao tipo de projeto. O aprendizado entre organizações se dá na elaboração da visão ou das histórias.

O aprendizado organizacional também se faz presente na fase de planejamento, em particular, na sprint planning no SCRUM e na elaboração do plano de versão (XP), nos quais a equipe deve adquirir conhecimentos sobre o negócio do cliente a fim de contribuir com o melhor

conjunto de funcionalidades ou requisitos a serem implementados para entregar produtos de maior valor. Considerou-se que, apenas no IVPM2 esse aprendizado não ocorre devido a um maior distanciamento da equipe e do cliente nessa fase, o que é inerente ao método.

Durante a fase de execução o aprendizado organizacional e melhorias pontuais nas práticas dos métodos ocorrem principalmente durante as conversas informais e discussões sobre o projeto. Já a fase de controle concentra boa parte da MC e do aprendizado inter-organizacional, principalmente na sprint review e demonstração (SCRUM), nos testes de aceitação e ajuste de velocidade (XP) e nas avaliações das entregas e análises do processo (IVPM2).

Por fim, na fase de encerramento a MC e o AO também ocorre, mas em menor número, sendo esse fator bem diferente dos projetos tradicionais, nos quais essa fase concentra quase todo o aprendizado. Desse modo, percebe-se que no GAP o aprendizado e melhoria contínua se dão de forma mais diluída e constante, apresentando dificuldades em sua formalização e difusão para toda empresa enquanto que nos métodos tradicionais ele é mais formalizado e difundido e concentrado na fase de encerramento.

6. Conclusões

A análise comparativa de conteúdo dos métodos GAP (XP, SCRUM e IVPM2), com relação ao modelo de maturidade proposto por Bessant et al. (2001), indicou que os métodos possuem um elevado nível de maturidade (4) em melhoria contínua, corroborando para autores como Highsmith (2004) e Cohn (2005) que sugerem o impacto positivo da utilização do GAP no aprendizado organizacional e melhoria contínua.

Nenhum método analisado se mostrou completo com relação à MC, mas cada um tem pontos fortes e fracos, como, por exemplo, o Scrum com as reuniões de revisão e o IVPM2 com a coleta e análises de dados. Essas atividades poderiam ser combinadas ou outras equivalentes introduzidas de modo a produzir melhores resultados.

Também evidenciou que a melhoria contínua no GAP acontece durante todo o projeto, diferentemente dos projetos tradicionais em que se concentra no seu término. Contudo, fica clara a dificuldade de compartilhar esse aprendizado com toda a organização, de modo que cada equipe de projetos possui seu aprendizado e processo próprio.

O estudo apresenta limitações, dentre elas ser baseado em revisão bibliográfica e análise de conteúdos. Da mesma forma, apenas três métodos de GAP foram analisados. Como estudos futuros propõe-se a realização de estudos de caso para verificar como a MC acontece nas empresas que adotam o GAP e quais adaptações foram feitas aos métodos (se é que foram feitas) para suprir as dificuldades apontadas. Outro aspecto evidenciado foi a falta de procedimentos sistemáticos que garantam o alinhamento da MC com as estratégias da organização.

Também, se propõe o desenvolvimento de procedimentos sistemáticos para a transferência de conhecimentos entre equipes de projeto e que garantam o alinhamento das iniciativas de melhoria contínua com a estratégia da organização.

Referências

- ALMEIDA, L.F.M.** *Características, fatores críticos e indicadores de agilidade no gerenciamento de projetos de produtos inovadores*. 157 p. Dissertação (mestrado) – Universidade Federal de São Carlos, São Carlos, São Paulo, 2012
- AMARAL, D.C; CONFORTO, E.C; BENASSI, J.L.C; ARAÚJO, C.** *Gerenciamento Ágil de Projetos – aplicação em produtos inovadores*. Saraiva: São Paulo, 2011.
- AMBLER, S. W.** *Modelagem Ágil: Práticas eficazes para a programação eXtrema e o processo unificado*. Bookman: Porto Alegre, 2002.
- BESSANT, J; CAFFYN, S.; GALLAGHER, M.** *An evolutionary model of continuous improvement behaviour* Technovation, v. 21, n. 2, pp 67-77, 2001.
- BECK, K.** *Extreme Programming Explained*. Pearson Education, Inc, 2000.
- BOEHM,B.; TURNER, R.** *Balancing Agility and Discipline: a guide for the perplexed*. Addison-Wesley: Boston, 2004.
- BURNSTEIN, I.; SUWANASART, T.; CARLSON, R.** *Developing a testing maturity model for software test process evaluation and improvement* Test Conference, 1996. Proceedings., International , vol., no., pp.581-589, 20-25 Oct 1996.
- CAFFYN, S.** *Development of a continuous improvement self assessment tool* International Journal of Operations & Production Management, v. 19, n. 11, pp 1138-1153, 1999.
- COCKBURN, A.** *Crystal Clear – a human-powered methodology for small teams*. Agile Software Development Series, 2004.
- CONFORTO, E. C.** *Gerenciamento ágil de projetos: proposta e avaliação de método para gestão de escopo e tempo*. 2009. 306 f. Dissertação (Mestrado) – Escola de Engenharia de São Carlos, Universidade de São Paulo, São Carlos, 2009.
- DAWSON, R. & DAWSON, C.** *Practical proposals for managing uncertainty and risk in project planning*. International Journal of Project Management, v.16, n.5, p.299-310, 1998.
- HIGHSMITH, J.** *Agile Project Management: creating innovative products*. Addison-Wesley: Boston, 2004.
- KWAK, Y. H.; IBBS, W.** *Project Management Process Maturity (PM)² Model* Journal of Management in Engineering , v. 18, n.3, pp.150-155, 2002.
- NAZI, M.; WILSON, D.; ZOWGHI, D.** *A maturity model for the implementation of software process improvement: an empirical study* Journal of Systems and Software, v. 74, n. 2, pp 155-172, 2005.
- NONAKA, I.; TAKEUCHI, H.** *Criação de Conhecimento na Empresa*. Campus, 1997.
- PALMER, S.R; FELSING, J.M.** *A practical guide to feature-driven development (the coad series)*. Primeira edição, Prentice Hall PTR, USA, 2002.
- PAULK, M. C.; CURTIS, B.; CHRISSIS; M. B.; WEBER. C. V.** *Capability maturity model, version 1.1* Software, IEEE, v. 10, n. 04, pp. 18-27, 1993.
- PROJECT MANAGEMENT INSTITUTE – PMI.** *Guia PMBOK: Um guia do conjunto de conhecimentos do gerenciamento de projetos*. Pennsylvania: Project Management Institute, 4th. ed, 2008.

SCHWABER, K. *Agile Project Management with SCRUM*. Microsoft Press: Washington, 2004.

SUIKKI, R.; TROMSTEDT, R.; HAAPASALO, H. *Project management competence development framework in turbulent business environment*. *Technovation*, v.26, n.5, p.723-738, 2006.