Capítulo

10

Implantação e Melhoria de Processos de Software

Jair da Silva Farias

O mercado de software tem evoluído exponencialmente adjunto à popularização dos computadores e dispositivos móveis, fato este, que deriva da globalização e da necessidade de uma economia mais competitiva, onde se busca um diferencial estratégico, ocasionando uma necessidade de processos que objetivem a qualidade dos produtos de software.

A qualidade dos processos para produção de software não garante, mas aumentam a probabilidade de que os produtos sejam de qualidade. Para se atingir níveis de maturidade e qualidade dos processos de software, organizações como o SEI, Softex e Motorola, criaram modelos que acompanham ou propõem práticas e processos para produção de software e redução dos defeitos.

Portanto é importante notar que a infraestrutura criada para realizar a Melhoria do Processo de Software (MPS) deverá desempenhar um papel significativo no sucesso ou fracasso de uma iniciativa de MPS. O valor que a infraestrutura traz a uma iniciativa de MPS, a compreensão das suas funções e responsabilidades, não pode ser subestimado.

Este capítulo aborda os modelos para implantação e melhoria de processos de software, dentre os modelos que serão abordados estão: o Seis Sigma que foi criado na década de 80 para reduzir o nível de defeitos na produção da Motorola. O IDEAL que foi criado pelo SEI para melhoria de processos organizacionais e o PRO2PI criado por Salviano, que foi baseado na norma ISO/IEC 15504 e propõe uma engenharia de processo dirigida por perfis de capacidade de processo.

## Introdução a modelos para melhoria de processos de software

A primeira abordagem sistêmica de processos foi iniciada na década de 30 com Walter Shewhart [Shewhart 1980] com um trabalho em melhoria de processos com ênfase nos princípios do controle estatístico, sendo estes, refinados posteriormente na década de 80 em por de W. Edwards Deming [1986] e Joseph Juran [1997]. Entretanto, estes trabalhos eram focados na indústria de manufatura, que posteriormente seriam utilizados em fábricas de software.

Moreira [2008] descreve que na indústria de software, Watts Humphrey foi um dos primeiros estudiosos a aplicar o conceito de processo para a solução dos problemas de desenvolvimento de software. Em seu livro, [Humphrey 1989] afirma que *“o primeiro passo importante ao lidar com problemas de desenvolvimento de software é tratar esta atividade como um processo que pode ser controlado, medido e melhorado”*.

Em diversas fontes da literatura direcionadas a processos de software, [Moreira 2006] encontrou definições para o processo de software. Veja abaixo que em sua totalidade todas tem algo em comum:

* Um conjunto de atividades e resultados associados que produzem um produto de software [Sommerville 2006];
* Uma seqüência de passos executados para um determinado propósito [IEEE 2000];
* Um conjunto de atividades inter-relacionadas ou interativas que transforma entradas em saídas [ISO9000 2000];

Diante destas definições foi preciso modelar e aperfeiçoar um conjunto ou vários conjuntos de atividades, práticas e meios para a produção de software. Segundo [Moreira 2008] diversas pesquisas realizadas nos últimos anos têm apresentado a importância do uso sistemático e disciplinado de processos para que uma empresa de software possa obter sucesso. Este sucesso está relacionado a aspectos como, por exemplo: aumento de sua competitividade, capacidade para assumir maiores riscos, aumento da qualidade de seus produtos, ganhos com produtividade, menos custos e eliminação de re-trabalho. Há duas décadas, [Humphrey 1989] já afirmava que, para que empresas de software possam obter sucesso é preciso existir harmonia entre seus processos de software, focando: pessoas, produtos, processos e projetos.

Esta afirmação de Humphrey é legitimada por [Pressman 2002] que enfatiza que a falta de adoção de métodos, ferramentas e procedimentos no desenvolvimento de software têm alcançado números expressivos de projetos não concluídos, e projetos concluídos e que não atendem as necessidades do cliente.

Estudos e pesquisas tem concentrado a Engenharia de Software em uma subárea específica denominada de Melhoria de Processo de Software (*Software Process Improvement - SPI*), a MPS orienta que para desenvolver software de qualidade é preciso que os passos para seu desenvolvimento sejam acompanhados de atividades planejadas, gerenciadas, de modo a minimizar os custos e otimizar a realização das tarefas [Moreira 2008].

Pesquisadores [Habib et. Al, 2008] afirmam que “*qualquer melhoria de processo de software significante requer um investimento significativo, tempo e dinheiro*”. Então para que essas variáveis não sejam desperdiçadas é preciso um estudo de viabilidade e planejamento da mudança e da melhoria, por que *mudança não se faz da noite para o dia*.

 Nas próximas seções serão detalhados os principais modelos que orientam a implantação e a melhoria dos processos de software, são eles: IDEAL, PRO2PI, Seis Sigma e o DMAIC como ferramenta de apoio ao Seis Sigma.

## IDEAL

O IDEAL é um processo de melhoria de software criado na década de 90, que é usado para guiar o desenvolvimento de um plano estratégico integrado de melhoria a longo prazo, para o início e gestão de um programa de MPS. O objetivo desta seção e suas subseções é proporcionar ao leitor, uma descrição genérica de uma seqüência de passos recomendados para melhoria de processos de software baseada no modelo IDEAL.

O nome do modelo é formado pelo acrônimo das palavras (*Initiating, Diagnosing, Establishing, Acting, Learning*). O IDEAL é um modelo para programas de Melhoria de Processo de Software que foi desenvolvido pelo *Software Engineering Institute* (SEI) baseado no arcabouço de experiências de trabalhos de melhoria com o Governo Norte-Americano e outros clientes.

A proposta do modelo IDEAL está centrada na melhoria dos processos de software através de ciclos, onde, em cada ciclo, é executado um conjunto de atividades que são distribuídas em cinco fases: Iniciação (*Initiating*), Diagnóstico (*Diagnosing*), Estabelecimento (*Establishing*), Ação (*Acting*) e Aprendizagem (*Learning*) conforme ilustrado na Figura 1.

**Aprendizado**

**Ação**

**Estabelecimento**

**Diagnóstico**

**Iniciação**

Definir contexto
e estabelecer
patrocínios

Estímulo para
a melhoria

Estabelecer
infra-estrutura
para a melhoria

Avaliar e
caracterizar a
prática atual

Desenvolver
recomendações
e documentar
resultados
 da Fase

Definir
estratégias e
prioridades

Estabelecer
equipes de ação

Planejar ações

Planejar,
executar e
e acompanhar
implantação

Planejar e
executar pilotos

Definir
processo e
medidas

Documentar e
analisar
Lições

Revisar
abordagem
organizacional

Figura 1 - Tradução do Ciclo do modelo IDEAL, Fonte: Moreira (2008) Apud McFeeley, 1996

Para aplicar o modelo IDEAL deve ser lembrado que existem dois componentes na atividade de melhoria de processo de software, o estratégico e o tático. Se o componente estratégico for baseado nas organizações empresariais, fornecerá a orientação e priorização das atividades táticas. A figura 2 mostra uma visão bidimensional da aplicação do modelo IDEAL.

Esta seção e suas derivadas destinam-se a abordar esses dois níveis operacionais dentro de um processo de iniciativa de melhoria, visto que são os níveis abordados pelo manual oficial do SEI, escrito por [McFeleey 2006], para a implementação do IDEAL:

* O nível estratégico, em que há processos que são a responsabilidade da gerência sênior.
* O nível tático, em que os processos são modificados, criado e executado por gerentes de linha e profissionais.

Nível Estratégico

Comunicação e Envolvimento

Nível Tático

Gerenciamento do programa de melhoria de processo de software

Fase Inicial

Fase de Diagnóstico

Fase de Estabilização

Fase de Ação

Fase de Aproveitamento

Figura 2 - Duas dimensões da atividade de melhoria de processos através do Ideal, Adaptado de McFeeley, 2006.

Na próxima seção serão explicadas detalhadamente, de acordo com [McFeleey 2006], as fases do IDEAL e a atividade de gerenciamento que é essencial para que a implementação do programa de melhoria obtenha maiores chances de sucesso.

## Fases do IDEAL

A base de todo modelo de melhoria é alicerçada em ciclos, visto que, a atividade de melhoria, seja de processos ou de qualquer setor organizacional é sempre contínua, esses ciclos são baseados em fases, e estas fases são compostas por processos, práticas e atividades.

Assim também acontece com o IDEAL. O modelo é composto por cinco fases e uma atividade gerencial, as fases são formadas de atividades-processos que devem ser seguidas para um maior alcance dos objetivos organizacionais, a atividade gerencial é essencial para o controle e evolução da MPS, aquelas e esta serão explicadas nas seis seções que procedem.

## Fase inicial (Initiating)

Este é o passo inicial para implementação do modelo IDEAL. Nesta fase a gerência sênior da organização compreende a necessidade de melhoria de processo de software (MPS), compromete-se com um programa de MPS, e define o contexto da MPS.

Esta etapa é similar à definição de um novo sistema. Um plano inicial de alto nível de MPS e um cronograma para as tarefas iniciais de MPS são desenvolvidos, e ainda, os principais elementos funcionais são definidos com uma chave de interfaces e os requisitos também são definidos e acordados. Este plano de alto nível vai orientar a organização até a conclusão da fase de estabelecimento, na qual um plano de ação para MPS será concluído. Normalmente, uma equipe é formada para explorar as questões e desenvolver uma proposta para MPS à gerência sênior. Após a aprovação da proposta da MPS, a infra-estrutura para o lançamento do programa de MPS será formada.

A organização precisa decidir como vai organizar os esforços de melhoria que serão envolvidos, tanto na prática e nos níveis de gestão, quanto no tempo das pessoas que serão afetadas. Além disso, é nesta fase que são formados os grupos para apoiar e facilitar o programa. Geralmente são formados dois grupos para o apoio a MPS, são eles:

* Grupo de Gerenciamento *(Management Steering Group - MSG)*: responsável por vincular o programa de melhoria de processo à visão e missão da organização, representando dessa forma, o patrocinador. Este grupo é responsável por alocar e monitorar o progresso dos recursos, prover direcionamento e aplicar ações corretivas para o programa, quando necessário;
* Grupo de Processos de Engenharia de Software *(Software Engineering Process Group - SEPG)*: indicado pela alta gerência para ser responsável por coordenar o programa de melhoria. Este grupo é responsável por promover, instruir e acompanhar as atividades, garantindo o bom andamento do programa dentro da organização.

A Figura 3 ilustra os dez processos sugeridos pelo guia de implantação do IDEAL para a primeira fase, após esta, seguiremos com a fase de diagnóstico da organização para dar continuidade a MPS.

Primeiros Passos

Construir a proposta de melhoria do processo de software

Identificando as necessidades do negócio e o caminho para melhoria

Treinamento, conscientização e construção do suporte a MPS

Obter aprovação da proposta de MPS e dos recursos iniciais

Estabelecer uma infra-estrutura para melhoria do processo de software

Avaliar o clima de MPS

Definir metas gerais do MPS

Definir os princípios orientadores
do Programa de SPI

Lançamento do Programa

Figura 3 -Fluxo do processo para fase de iniciação, Adaptado de [McFeeley 2006]

## Fase de diagnóstico (Diagnosing)

O grupo de gerenciamento (*MSG*) deve compreender a base da organização e do processo atual de software para desenvolver um plano, que permita atingir o negócio na mudança específica do processo de software e nas metas da MPS da organização. As atividades realizadas na fase de diagnóstico vão fornecer essas informações para o planejamento e priorização da MPS.

É necessário para fornecer orientações claras para a melhoria de processos um plano estratégico de ação para a MPS. Através deste, diversas ações serão tomadas nos próximos anos, além disso, deverá fornecer, de forma clara e mensurável, as necessidades de negócio para a condução do programa de MPS, ligada ao plano de negócios da organização e da visão empresarial.

As linhas de base irão fornecer informação sobre como a organização atualmente realiza suas atividades de software. O conhecimento dos pontos fortes e oportunidades para melhoria é um pré-requisito essencial para a identificação e priorização de um eficaz e eficiente programa de MPS.

O resultado principal desta fase são as conclusões finais e relatório de recomendações, que é produzido como resultado das atividades de *baseline*[[1]](#footnote-1). Saídas secundárias podem ser revisões à visão da organização e do plano de negócios, um conjunto mínimo recomendado de *baselines* inclui:

* Organização de maturidade do processo de *baseline*.
* Descrição inicial do processo (mapa inicial do processo de software)
* Métricas de *baseline* (nível inicial do negócio e métricas de processo para medir o progresso).

Para cada *baseline*, muitos métodos eficazes de coleta de informação estão disponíveis. Para a *baseline* de maturidade do processo, um avaliador autorizado pode levar em conta a conduta da organização baseado no *Capability Maturity Model Integration* (CMMI) ou os colaboradores da própria organização pode ser treinado para avaliar o seu processo de maturidade. O MSG deve escolher o número e o tipo de *baseline* que melhor atingir os objetivos que fixou para que um relatório e recomendações possam ser obtidas a partir de cada um. Manter uma dinâmica da melhoria de processos entre *baselining* e a implantação de MPS é muito importante.

A Figura 4 ilustra os seis processos sugeridos pelo guia de implantação do IDEAL para a fase de diagnóstico, após esta, seguiremos com a fase de estabilização da organização para dar continuidade a MPS.

Determinar necessidade de *baseline* (s)

Plano para os *baseline* (s)

Orientar

*baseline* (s)

Resultados atuais

Desenvolver as conclusões finais e Relatório de Recomendações

Comunicar os resultados e
Recomendações para
Organização

Figura 4 - Fluxo do Processo da fase de Diagnóstico, Adaptado de [McFeeley 2006]

## Fase de estabilização (Diagnosing)

Criar um plano de ação estratégico para a melhoria de processo de software (MPS) é um dos passos mais críticos e negligenciados da iniciativa de MPS. Por isso, é necessário que a equipe de gestão desenvolva ou atualize um plano de ação estratégico baseado na visão da organização, o plano de negócios, e as lições aprendidas dos esforços de melhoria passados, adjuntos aos resultados do *baselining* deesforços.

A fase de estabilização é repetida conforme necessário. Geralmente é desencadeado pela falta de um plano de ação para uma organização em seu primeiro ciclo, através do modelo IDEAL. Para aquelas organizações em um ciclo posterior, este passo pode ser desencadeado por uma necessidade de atualizar o plano anterior, objetivos, ou metas.

Nesta fase criar um plano de ação sólido é muito importante, as experiências mostram que, sem um planejamento cuidadoso, os esforços acabarão por falhar e haverão distorções, ou não correspondendo às expectativas escritas pelo alto gerenciamento. A razão que leva à necessidade de planos estratégicos bem elaborados, não é apenas identificar as melhorias, mas atender as necessidades críticas do negócio com a instalação dessas melhorias em toda a organização [McFeeley 2006].

A identificação das melhorias é muitas vezes a parte mais fácil. Fazer com que todos em toda a organização mudem a maneira como eles fazem as coisas é sempre a
parte mais difícil de todo o esforço de melhoria.

O objetivo desta fase está centrado em desenvolver ou aperfeiçoar um plano estratégico de ação, que irá fornecer orientações e diretrizes para o programa de MPS, que terá duração de três a cinco anos, este é o tempo indicado pelo guia de implantação do IDEAL, para uma atualização do plano estratégico [McFeeley 2006]. A saída principal desta etapa é o plano de ação estratégica de MPS, saídas secundárias podem ser as revisões da visão da organização e do plano de negócios.

Além da produção de um plano estratégico de MPS é preciso integrá-lo com iniciativas já previstas ou em andamento de Gestão da Qualidade Total (*TQM*), com as conclusões e recomendações da *baseline* no plano de ação estratégico e alinhá-lo com o plano de negócio da organização, missão e visão. A Figura 5 ilustra os quatorze processos sugeridos pelo guia de implantação do IDEAL para a fase de estabilização, após esta, seguiremos com a fase de ação organizacional que é onde serão empregados os esforços planejados nas fases anteriores para dar continuidade a MPS.

Rever a visão da Organização

Treinamento, conscientização e construção do suporte a MPS

Construir a proposta de melhoria do processo de software

Seleção e treinamento no Planejamento Estratégico de

Processo

Revisão do plano de negócio da organização

Determinar os pontos chaves do negócio

Revisão dos esforços passados de melhoria

Descrever as motivações para melhoria

Identificar os esforços atuais e

Futuros (Previsão)

de melhoria

Transformar as metas genéricas de MPS para metas específicas e mensuráveis

Criar ou atualizar o plano estratégico de MPS

Conciliar os planos/ esforços

Existentes de melhoria com com o *baseline*

e recomendações

Formar o Grupo Técnico de Trabalho(GTT)

Construir um consenso, revisar, aprovar o plano estratégico e disponibilizar recursos para ação

Figura 5 - Fluxo do processo da fase de estabilização, Adaptado de [McFeeley2006]

## Fase de ação (Acting)

A fase de ação é a fase onde as melhorias são desenvolvidas e implantadas em toda a organização. Aqui as melhorias sugeridas pelos grupos de trabalho terão seu valor colocado a “prova”. O grupo de gestão (*MSG*) e o de processo de engenharia de software (*SEPG*) farão não só a gestão, mas também, o apoio ao desenvolvimento, controle, e implantação das melhorias.

A fase de ação une a missão do programa de MPS com a missão da organização no desenvolvimento de produtos. Este é o ponto culminante dos esforços de MPS. Para o planejamento e a introdução de melhorias, devem ser estudadas e avaliadas a estrutura atual da organização e as práticas utilizadas na criação dos produtos de software, para que elas sejam totalmente compreendidas e documentadas.

Também é importante um mecanismo ou métricas que identifiquem os efeitos da mudança em uma determinada área. Estes efeitos devem ser identificados o mais cedo possível para que eles possam ser tratados em tempo hábil. Para ajudar a compreender as práticas, é preciso se utilizar das técnicas disponíveis para modelar e avaliar a práticas atuais em “como estão”, e assim determinar a áreas de melhoria, e como os processos candidatos a melhoria devem ser examinados e avaliados.

Após essa avaliação e criação do estado atual dos processos, a organização precisa definir um "onde chegar" e escolher a solução adequada para atingir o estado desejado dos processos candidatos. Após esta avaliação e seleção, informar as decisões a serem tomadas para a seleção candidatos e a tecnologia a ser utilizada para a melhoria. A identificação de onde se quer chegar no estado do processo é muito importante para o sucesso global da fase de ação.

Esta fase do IDEAL é onde os Grupos de Trabalho Técnicos (*GTT*) desenvolvem melhorias específicas para processos específicos. Há duas abordagens básicas para concepção de soluções: foco na resolução de problemas específicos; incremento de um determinado processo.

Para esta fase, é essencial a utilização de projetos piloto para validar, refinar e testar os refinamentos das soluções para a MPS, esta fase pode necessitar de mais tempo que as demais por ser também uma fase experimentação. A Figura 6 ilustra os doze processos sugeridos pelo guia de implantação do IDEAL para a fase de ação, após esta, seguiremos com a fase de alavancagem da organização para dar continuidade a MPS.

Concluir plano tático para o Grupo de Trabalho de Técnico(GTT)

Soluções piloto potenciais

Refinar processo (Processo centrado em abordagem)

Analisar e corrigir os problemas (Problema centrado em abordagem)

Desenvolver Soluções

Selecionar as soluções fornecidas

Determinar a necessidade de apoio em longo prazo

Desenvolver uma estratégia de *Rollcut* e *template* do plano

O pacote de melhoria e *Turn over* para o Grupo de Engenharia de Processos de Software (SEPG)

Dissolver o GTT

*Rollcut* soluções

Construir um consenso, revisar, aprovar o plano estratégico e disponibilizar recursos para ação

Figura 6 - Fluxo do processo da fase de ação, Adaptado de [McFeeley2006]

## Fase de aproveitamento (Leveraging)

Agora que a organização completou um ciclo através do IDEAL, é necessário rever o que aconteceu durante esse ciclo e se organizar para o próximo ciclo, através do modelo. Ao invés de re-introduzir o IDEAL na fase de iniciação, esta fase vai retorna a fase de diagnóstico. Esta fase, além de preparação para o próximo ciclo através IDEAL, dá a oportunidade de ajuste a melhoria do processo de software (MPS) antes de iniciar o processo novamente.

Geralmente existem alguns “falsos começos” de melhoria em determinadas áreas da organização ou omissões e algumas atividades que foram planejadas para serem feitas mais que não ocorreram durante o primeiro ciclo do IDEAL. Uma vez que se tem documentado uma lista completa das lições aprendidas em cada uma das atividades de MPS, agora é preciso aplicá-las durante a fase de aproveitando a tornar o processo de MPS um trabalho mais eficiente e eficaz durante o próximo ciclo através do modelo IDEAL. Segundo [McFeeley 2006] algumas tarefas necessárias nesta fase são:

* Rever e analisar as lições aprendidas com as fases anteriores;
* Incorporar melhorias nos processos de MPS;
* Motivar a revisão das atividades de MPS;
* Analisar e avaliar as metas;
* Avaliar o patrocínio e empenho do envolvidos;
* Desenvolver um plano para proporcionar uma orientação contínua para o SPI programa.

Como visto nos tópicos acima, esta fase é dirigida a análise e revisão das tarefas que foram envolvidas no trabalho de MPS, através dessas análises e revisões é dado início a caracterização da melhoria contínua, baseada nas mudanças de melhoria dos processos e nas lições aprendidas com a MPS.

A Figura 7 ilustra os sete processos sugeridos pelo guia de implantação do IDEAL para a fase de aproveitamento, após esta, segue-se com a fase de gerenciamento do programa de melhoria do processo de software da organização para dar continuidade a MPS.

Determinar necessidade de *baseline* (s)

Analisar as lições

Revisar a abordagem organizacional

Revisão dos patrocinadores e compromisso

Estabilizar metas de alto nível

Desenvolver nova revisão da proposta de melhoria do processo de software (MPS)

Continuar com a MPS

Figura 7 - Fluxo do processo da fase de aproveitamento, Adaptado de [McFeeley2006]

## Fase de gerenciamento do programa de melhoria do processo de software (Manage)

A melhoria do processo de software é uma iniciativa muito importante para uma organização. Para coordenar as diversas atividades que irão ocorrer no decurso de um programa de melhoria do processo de software (MPS) é necessária a previsão de uma efetiva infra-estrutura de apoio. Além disso, a infra-estrutura deve ser capaz de reagir de forma oportuna para as demandas do programa de MPS.

No início do programa de MPS, uma infra-estrutura de MPS inicial deve ter sido posta em prática para gerir as atividades da organização durante o programa. Esta fase de gerenciamento do programa de melhoria é o melhor momento para rever a forma como esta infra-estrutura tem realizado suas tarefas é após algum tempo criar o grupo comprometido com o apoio a MPS e a obtenção de patrocínios, avaliação do empenho de todos, completando as atividades *baselining* e o planejamento de ação para o próximo ciclo de IDEAL.

Com o programa de MPS em curso, uma infra-estrutura possivelmente como a ilustrada na Figura 8 deve ser desenvolvida e posta em prática. Esta infra-estrutura terá a responsabilidade de fornecer orientação para o programa de MPS. Na maioria dos casos, haverá três componentes para a organização: Grupo de Processos de Engenharia de Software (SEPG); Grupo de Gestão (MSG); Grupo de Trabalho Técnico (GTT).

Presidente

Recursos Humanos

Financeiro

Manufatura

Desenvolvimento

SEPG

Grupo de Gerenciamento

Figura 8 – Componentes típicos de uma infra-estrutura de MPS , Adaptado de [McFeeley2006

Estes são os nomes genéricos e podem variar de organização a organização. Os componentes da infra-estrutura e sua relação com os outros são em grande parte determinados por fatores, como tamanho da organização e a diversidade geográfica.

Em [McFeeley 2006] são citadas algumas perguntas a responder sobre o desempenho da infra-estrutura que inicialmente foi posta em prática:

* A infra-estrutura está efetivamente ligada ao programa de MPS para a missão da organização e da visão organizacional?
* A infra-estrutura foi capaz de obter e alocar recursos suficientes para garantir conquistas oportunas?
* A infra-estrutura acompanhou corretamente o programa de MPS e forneceu orientação e correção necessárias?

As atividades de melhoria não irão ocorrer em um vácuo nem ocorrerão em série. Depois que o programa de MPS está em curso, haverão várias atividades de melhoria ocorrendo em diferentes unidades organizacionais. Como exemplo, podem haver grupos técnicos de trabalho (GTT) abordando gerenciamento de configuração, gerenciamento de requisitos, planejamento do projeto, e as análises comparativas, todos podem ocorrer simultaneamente. A infra-estrutura de apoio deve manter o controle de tudo isso e estar preparada para fornecer a necessária supervisão e orientação a todas as atividades do programa de MPS.

A infra-estrutura de apoio deve estar ciente de que os GTT’s podem e provavelmente vão funcionar em paralelo. De acordo com [McFeeley 2006], a qualquer momento, o grupo de apoio deve estar preparado para:

* Oferecer suporte para uma tecnologia que está sendo introduzida;
* Formação e coordenação de recursos;
* Continuação, construção e fornecimento de patrocínio;
* Proporcionar conhecimento de planejamento;
* Avaliar o impacto organizacional;
* Mostrar as lições aprendidas.

O IDEAL é um modelo, bem descrito e de fácil compreensão, que acompanha todas as fases de um programa de MPS, para maiores informações sobre como implementar e gerir o modelo em uma organização, veja as sugestões de leitura no final desse capítulo, no tópico IDEAL, lá você vai encontrar como adquirir o manual gratuito e completo do modelo.

## PRO2PI

O aumento nos estudos voltados ao auxílio das atividades de produção software tem gerado uma diversidade de métodos, ferramentas, práticas, processos e metodologias para o desenvolvimento de software, entretanto estes são acompanhados de estudos de melhoria no decorrer do tempo.

Salviano (2006) cita que a melhoria de processo de software tem apontado na prática ser uma abordagem eficaz e eficiente para a necessária melhoria das organizações que produzem software. A comunidade tem relatado vários casos de sucesso, como, por exemplo, [Herbsleb et al. 1994, DACS 1999 e Card 2002].

O PRO2PI[[2]](#footnote-2), criado por Salviano, surgiu baseado em “*Uma Proposta Orientada a Perfis de Capacidade de Processo para Evolução da Melhoria de Processo de Software*”, foi fruto resultante da junção dos estudos em Melhoria de Processos de Software, Modelos de Capacidade de Processo, Gerações de Arquiteturas de Modelos de Capacidade de Processo e Engenharia de Processos dirigida por perfis de capacidade.

Os estudos de Salviano apontam para uma grande quantidade de normas na literatura relativas às abordagens para melhoria de processo, como, por exemplo, IDEAL (McFeeley 1996), ISO/IEC 15504 [ISO/IEC TR 15504-7 1998, ISO/IEC 15504-4 2004], problemas e metas [Porter e Sakry 2002] e as orientações para a melhoria de [O’Toole 2000], que utilizam como referência um modelo de processo que sistematiza e representa as melhores práticas, definem uma medição para avaliação da capacidade dos processos e provê um roteiro racional para a melhoria dos processos.

 Exemplos de modelos mais utilizados são a Norma ISO/IEC 12207 (1998[[3]](#footnote-3)), a ISO/IEC 15504 que é também conhecida como *SPICE (Software Process Improvement and Capability Determination)* (ISO/IEC 15504 1998), o CMMI (*Capability Maturity Model Integration*) [Chrissis et al. 2003], e a aplicação para software da ISO 9000, principalmente a versão 2000 com o par coerente 9001 e 9004 (ISO 9001 2000, ISO 9004 2000).

Essas normas, metodologias, modelos e abordagens estão bem difundidas na comunidade e descritas em várias publicações [Rocha et al. 2001], inclusive com considerações sobre o relacionamento entre os modelos citados [Sheard 2001].

Na próxima seção o processo base de formação do PRO2PI será iniciado, baseado na engenharia de processo dirigida por perfis de capacidade e seus fundamentos. É importante observar a Figura 1 que descreve os princípios do processo de definição de um PRO2PI baseado nas necessidades organizacionais.

## Engenharia de processo dirigida por perfis de capacidade e seus fundamentos

Para que haja um processo de descoberta de oportunidades para evolução na área de melhoria de processo, é preciso estudar a engenharia de processo dirigida por níveis de capacidade de processo, nesse estudo é possível refletir sobre propostas de mudança de melhoria dos processos de software.

 A definição para perfil de capacidade de processo aponta para um modelo que representa um processo segundo o aspecto de capacidade de processo. O processo de uma empresa deve ser representado por um perfil de capacidade de processo, que é uma abstração do processo, segundo o aspecto capacidade de processo. O par consistente formado por perfil de capacidade de processo e processo é ilustrado na Figura 1, onde um processo é uma parte do mundo (M0) e é representado, segundo o aspecto capacidade de processo, pelo modelo perfil de capacidade de processo no espaço de modelagem (M1) [Salviano 2006].



Figura 9 - Perfil de capacidade de processo e processo, Fonte: (Salviano 2006)

De acordo com [Salviano 2006] uma forma simples de entender a relação entre o perfil e o processo é com o seguinte questionamento: Se o perfil de capacidade de processo representado por um nível de maturidade do modelo CMMI/DEV tivesse sido definido para o processo de uma determinada unidade organizacional, em um determinado momento, qual seria este nível de maturidade de forma a representar o processo atual ou o processo alvo para uma melhoria alinhada ao contexto e objetivos estratégicos da organização.

A melhoria de processo deve ser evoluída para uma engenharia que trate esse par consistente (perfil e processo) no centro dessa engenharia. É proposto então, como uma evolução da atual melhoria de processo de software baseada em modelos de maturidade, uma Engenharia de Processo Dirigida por Perfil de Capacidade de Processo (*Process Capability Profile Driven Process Engineering* - PCDE) aplicada a software.

Na próxima seção a definição da abordagem PRO2PI apresentada detalhadamente, é importante a observação das ilustrações, pois estas modelam a forma de funcionamento da aplicação do PRO2PI possibilitando um melhor entendimento.

## O PRO2PI

As propriedades do PRO2PI foram definidas a partir de seis fases, onde foram organizados os modelos de maturidade e referência em melhoria de processos. Um resumo das fases de formação de um PRO2PI está ilustrado na Figura 4 e podemos chamá-lo de “processo de montagem do PRO2PI”.

A Figura 10 exemplifica o modelo base para o ciclo de melhoria PRO2PI que é composto por melhores de práticas dos modelos de maturidade, práticas de desenvolvimento de software e modelos de gestão.

Figura 10 - Diagrama da utilização corrente da melhoria de processo, Fonte (Salviano, 2006).

A fase 2 do processo de formação do PRO2PI que é representada através da função de definição ou atualização de PRO2PI “defineP”, acontece a partir do momento em que uma organização seleciona elementos de um ou mais modelos de referências e define um perfil de capacidade de processo que representa os elementos selecionados desses modelos e de qualquer outra fonte. Um ciclo de melhoria de processo é realizado objetivando evoluir os processos para atender a esse perfil alinhado com o contexto e objetivos estratégicos da organização.

O perfil pode conter boas práticas de referência selecionadas dos modelos mais genéricos existentes e de outras fontes, de forma a representar orientações relevantes à organização. O perfil pode ser alterado a qualquer momento em função de novas percepções, alterações do contexto ou dos objetivos estratégicos e dos resultados da utilização da versão corrente do perfil. Esse uso é representado pela função “defineP” (define, ou atualiza, perfil de capacidade de processo) da Figura 3, [Salviano 2006].

Já a fase 3 é perpassada pela função de utilização do PRO2PI, o uso desse perfil é utilizado para orientar as ações de melhoria e, é representado pela função “usaP” (usa perfil de capacidade de processo) na Figura 4. Neste caso as ações de melhoria devem ser satisfatórias para que o processo resultante atenda a todos os requisitos representados no perfil de capacidade de processo.

A fase 4 que conta com a função de avaliação de capacidade de processo em relação a um PRO2PI. O processo da organização pode ser examinado com uma avaliação de capacidade de processo em relação ao perfil de capacidade de processo. Esse exame é representado pela função “avaliaPr” (avalia capacidade de processo em relação a um PRO2PI) na Figura 4. Os resultados de capacidade de processo gerados por essa avaliação podem ser utilizados como referências adicionais para a definição, ou alteração do perfil de capacidade de processo [Salviano 2006].

O ciclo com a função de definição de modelos de capacidade de processo mais específicos com a abordagem PRO2PI é apresentado na Figura 3. A partir do contexto de negócio de um segmento, como, por exemplo, o segmento Metalúrgico, ou de um domínio, como, por exemplo, engenharia de testes, pode ser definido um modelo mais específico. Esse modelo pode ser composto por áreas de processo ou perfis de capacidade de processo relativos um segmento ou domínio.

Figura 11 - Diagrama da definição de modelos mais específicos, Fonte: (Salviano, 2006)

A abordagem PRO2PI é caracterizada pela busca contínua ao alinhamento do perfil de capacidade de processo como uma representação, segundo o aspecto de capacidade de processo, do processo da unidade organizacional. Essa relação está descrita na Figura 1.

Em concordância com a Figura 3, a melhoria de processo com o PRO2PI é realizada com a definição e utilização de um PRO2PI que pode ou não conter boas práticas selecionadas dos modelos de capacidade de processo, modelos com outras finalidades, modelos mais específicos de um segmento ou domínio ou de qualquer outra fonte, que se adaptem as necessidades estratégicas da organização.

É importante salientar que a abordagem PRO2PI é baseada no ciclo de utilização das funções (“defineM”, “usaP”, “avaliaPr”, “defineP”) que giram em torno do PRO2PI e do modelo mais específico descrito anteriormente.

Figura 12 - Diagrama da abordagem PRO2PI para modelos e melhoria de processo, Fonte: (Salviano, 2006)

A Figura 5 demonstra uma analogia à utilização da abordagem PRO2PI ilustrada na Figura 4 com a utilização da engenharia de requisitos no desenvolvimento de software, que podemos denominar de desenvolvimento de software orientado por requisitos.

Figura 13 - Diagrama, análogo ao de PRO2PI, do desenvolvimento de software, Fonte: (Salviano, 2006)

Na Figura 6, a função “defineRS” (define requisitos) é utilizada para definir, ou atualizar, o conjunto de requisitos, a função “usaRS” (usa requisitos de software) será utilizada para guiar o desenvolvimento do software, de forma que, seja dirigida pelos requisitos. A função “verificaSw” (testa software) será utilizada para buscar problemas no software e com isto sugerir o quanto o software desenvolvido está em concordância com os requisitos. A função “defineRLPS” (define requisitos de linha de produto de software) representa o desenvolvimento de requisitos para uma linha de produto de software, de tal forma que esses requisitos sejam um sistema, sendo que os requisitos foram baseados em diferentes conjuntos de requisitos, de diferentes stakeholders. Esta forma de desenvolvimento de requisitos é análoga à forma de desenvolvimento de modelos mais específicos representada na Figura 4 [Salviano 2006].

As próximas seções deste capítulo referentes ao PRO2PI identificam um conjunto de quatro elementos que descrevem a abordagem e define cada elemento desse conjunto. Esses elementos são propriedades de um PRO2PI, um modelo de PRO2PI, medições de PRO2PI e um ciclo de melhoria com PRO2PI.

## PRO2PI-PROP: Propriedades de PRO2PI

De acordo com Salviano (2006) para ser útil e efetivo como orientação para a melhoria de uma organização, um PCP deve ter, em um grau suficiente, pelo menos as propriedades de ser relevante, oportuno, viável, representativo e específico em um determinado momento e com uma determinada previsão de investimento em ações de melhoria, ser sistêmico e dinâmico, e ser rastreável a modelos relevantes, conforme ilustrado na Figura 9.

Figura 8 - Propriedades de PRO2PI, Fonte: (Salviano, 2006)

A Tabela 1, que é baseada na descrição de Salviano das propriedades de um PRO2PI, explica detalhadamente cada propriedade e as descreve com exemplos, após esta seguiremos para a seção do elemento modelo de PRO2PI (*PRO2PI-MODEL*).

Tabela 1 – Propriedades de um PRO2PI, Adaptado de Salviano 2006

|  |  |
| --- | --- |
| **Propriedade** | **Descrição** |
| **Relevante** | Para ser relevante um PCP deve representar um estado do processo da organização que esteja alinhado com a estratégia de negócio da organização, e com isto represente melhorias importantes para a organização.  |
| **Viável** | Um PCP deve representar um estado de processo que requeira uma quantidade de esforço e de recursos para seu atendimento, a partir do estado atual, que seja viável de ser disponibilizado. Para isto uma análise deve ser feita. É necessário a utilização de uma medição para esta quantidade que pode ser baseada na unidade de melhoria definida em PRO2PI-MEAS. |
| **Oportuno** | Um PCP deve considerar, e apontar oportunidades disponíveis. Por exemplo, mesmo que a área de configuração não esteja com prioridade alta, mas existam recursos disponíveis a baixo custo, pode ser viável incluir devido à oportunidade. |
| **Sistêmico** | Um PCP deve representar um estado do processo que seja um sistema. Desta forma o processo representado por esse estado funcionará na organização, gerando os resultados necessários. Como um sistema, esse estado será ao mesmo tempo um situação auto sustentável e uma etapa para a melhoria. |
| **Representativo** | Um PCP deve representar uma abstração do processo, segundo o aspecto de capacidade de processo. Esta abstração deve ser completa, ou seja, representar todo o processo, nos termos e elementos de PCP. Todas as características representadas no PCP têm que estar sendo realizadas no processo e o processo realiza apenas o que está representado no PCP. |
| **Rastreável** | Como os modelos genéricos mais utilizados são consolidações de melhores práticas utilizadas por várias organizações, pode ser importante manter uma rastreabilidade do PCP com elementos relevantes, como, por exemplo, áreas de processo e níveis de maturidade, dos modelos relevantes para a organização.  |
| **Específico** | Como é importante manter um relacionamento um para um entre um PCP e o estado do processo, e cada organização tem sua especificidade, é fundamental que o PCP tenha elementos específicos para a organização. |
| **Dinâmico** | Como os vários fatores que influenciam o que deveria ser o estado do processo podem mudar, é importante que os PCPs possam ser ajustados. Também como o conhecimento sobre esses mesmos fatores tende a aumentar, os PCPs podem ser ajustados para refletir esse melhor conhecimento. Portanto eles devem ser dinâmicos. |

## PRO2PI-MODEL: Modelo de PRO2PI

O modelo de PRO2PI, denominado de *PRO2PI-MODEL* foi definido por Salviano (2006), para atender aos seguintes requisitos:

* Unificar os elementos e estruturas dos modelos de capacidade de processo considerados mais relevantes, especificamente dos modelos iCMM v2.0, CMMI-SE/SW v1.1, ISO/IEC 15504-5:2006 e MR-MPS v1.0, e com isto permitir a representação de praticamente qualquer elemento desses modelos em um PRO2PI;
* Buscar representar em um PRO2PI os elementos de outros modelos de capacidade de processo, incluindo PMI OPM3, ITsqc eSCM-SP, ITGI COBIT, SMMM, KMMM, PMMM, UMM, TMM, OOSPICE, SPICE4SPACE, Automotive SPICE, 15504MPE e S9K;
* Permitir também a representação em um PRO2PI de elementos selecionados de outros modelos de referência, que não sejam modelos de capacidade de processo, especificamente dos modelos PMBOK e ISO 9001, e incluindo ISO/IEC 12207 Amd2, IEEE SWEBOK 2004, EFQM, FPNQ PNQ e RUP.
* É especificado em uma referência conceitual adequada na linha de MDE;
* Minimizar a quantidade de tipos de elementos de outros modelos.

Salviano (2006) definiu o PRO2PI-MODEL direcionado à unificação e nas extensões de dois frameworks para modelos de capacidade de processo: o framework da ISO/IEC 15504 [ISO/IEC 15504-2 2003] e o do CMMI. A Figura X relaciona os principais elementos da estrutura dos modelos CMMI-SE/SW v1.1 e ISO/IEC 15504-5:2006 e indica como esses elementos são considerados na estrutura de PRO2PI-MODEL.



Figura 14 - Elementos da estrutura do CMMI e 15504-5 em PRO2PI-MODEL, Fonte: [Salviano, 2006]

Além disso, elementos de outros modelos, como ISO 9001, SWEBOK, PMBOK, OPM3 e SW-CMM, podem também ser modelados e associados ao PRO2PI-MODEL. Com isto é possível definir o perfil de capacidade de processo baseando-se em elementos de múltiplos modelos.

## PRO2PI-MEAS: Medições para PRO2PI

As medições de PRO2PI, denominadas de PRO2PI-MEAS (PRO2PI Measurements) são um conjunto de medições relacionadas a PRO2PI definidas por [Salviano 2006], estas medições podem servir para medir a viabilidade de um PRO2PI, complexidade e efetividade do investimento a ser realizado na MPS.

Segundo [Salviano 2006] o PRO2PI-MEAS utiliza o modelo de informação de medição definido na norma [ISO/IEC 15939 2002] e utilizado no PSM [McGarry et al. 2002] e nas áreas de processo de medição e análise dos modelos CMMI-SE/SW e ISO/IEC 15504-5. Para tanto são definidos:

* Uma medida derivada da complexidade de uma melhoria de processo representada por um PRO2PI (*Derived Measure Process Improvement Complexity* DM-PIC) em relação à situação atual de uma unidade organizacional, em termos de Unidades de Melhoria de Processo (*Unit for Process Improvement Complexity* U-PIC);
* Uma medida derivada da efetividade de investimento para melhoria de processo em uma unidade organizacional (Derived Measure for Investment Effectiveness for Process Improvement DM-IEPI), em termos de Unidades de Eficiência de Investimento para Melhoria de Processo (Unit for Investment Effectiveness for Process Improvement U-IEPI);
* O produto de informação denominado Viabilidade de PRO2PI, em termos de uma interpretação de DM-PIC e DM-IEPI.

A Figura 9 ilustra o modelo PRO2PI de medição, segundo a estrutura definida na Norma ISO/IEC 15939.

Em tese [Salviano 2006] explica que o modelo de análise da viabilidade é uma função da complexidade de um PRO2PI e da efetividade do investimento de uma unidade organizacional, e tem valor normalizado da viabilidade igual a 1, quando a complexidade da melhoria modelada por PRO2PI é compatível com a capacidade da unidade organizacional. O valor de viabilidade é maior que 1, quando a melhoria é maior que a capacidade, e menor que 1, quando a melhoria é menor que a capacidade. Um PRO2PI é considerável viável para uma unidade organizacional quando o valor normalizado da viabilidade estiver no intervalo [0.8 – 1.2], ou seja, dentro da faixa de 20% para cima ou para baixo. Para informações completas não só do modelo de medição, mas do PRO2PI por completo veja as sugestões de leitura no final capítulo.



Figura 15 - Produto de informação Viabilidade de PRO2PI, Fonte: (Salviano, 2006)

## PRO2PI-CYCLE: Processo para ciclo de melhoria

Um programa de MPS baseado em modelos é realizado com a utilização de abordagens. Entre elas podem ser destacadas a abordagem IDEAL, citada na seção 1.1 deste capítulo, o ciclo de melhoria da ISO/IEC 15504 e a abordagem AMP1. O IDEAL foi definida para arquitetura estagiada fixa, e ciclo 15504-4 e AMP1 são definidas para arquitetura estagiada fechada.

O processo para ciclo de melhoria com PRO2PI utiliza como referência as três abordagens citadas. Um ciclo de melhoria com PRO2PI pode entendido como composto por fases correspondentes às fases das abordagens citadas com o acréscimo de uma atividade para definição e utilização de PRO2PI que pode ser utilizada em qualquer fase [Salviano 2006].

O programa é iniciado com a decisão e comprometimento da organização em realizar a melhoria. As atividades desse ciclo são alinhadas com o contexto e os objetivos estratégicos da organização e podem utilizar experiências e resultados de outras organizações. Segundo [Salviano 2006] esse alinhamento é essencial para a melhoria porque o objetivo é a melhoria do negócio da organização, por meio da melhoria do processo de produção software.

A referência principal para a melhoria é o perfil de capacidade de processo, cuja definição e utilização é encapsulada em “define e utiliza PRO2PI” na Figura 10 que ilustra as seis fases e outros elementos do processo PRO2PI-CYCLE. O objetivo principal das atividades de definição e utilização de PRO2PI é definir e utilizar PRO2PI. O produto de entrada e de saída é o perfil de capacidade de processo. As atividades, que podem ser consideradas como uma implementação das práticas base definidas para o processo de estabelecimento de processo. Os objetivos de PRO2PI-CYCLE incluem [Salviano 2006]:

* Identificar e analisar os objetivos, estratégia, contexto e/ou qualquer outro aspecto relevante de negócio da unidade organizacional e da organização, para subsidiar e orientar a definição dos objetivos de melhoria;
* Identificar os objetivos da melhoria, incluindo objetivos mais específicos para o próximo ciclo de melhoria e objetivos mais gerais para o programa de melhoria, sempre alinhados aos objetivos, estratégia, contexto e/ou qualquer outro aspecto relevante de negócio identificados;
* Estabelecer critérios de qualidade para avaliar e melhorar um perfil de capacidade de processo

Assim é finalizada a seção de PRO2PI, para informações com maior nível de detalhe, veja a seção de sugestões de leitura no final deste capítulo. A próxima seção apresenta o Seis Sigma, que neste capítulo será abordado com um método estatístico para a melhoria do processo de software.

## Seis Sigma

Esta seção irá abordar fatores relevantes sobre o Seis Sigma, PDCA e DMAIC que é uma ferramenta utilizada no Seis Sigma, baseada no PDCA, ela possibilita a melhoria de processos de software através de ciclos de atividades de melhoria.

O Seis Sigma é um método estatístico que é representado pela 18ª letra do alfabeto grego, o sigma (σ), que é também o símbolo de desvio padrão na estatística. Em [Wang 2008] o Seis Sigma é definido como uma abordagem que melhora a qualidade através da análise de dados estatísticos. Nos últimos anos tem havido um aumento significativo em sua utilização.

Em pesquisa feita por [Moreira 2008] ela afirma que Seis Sigma é uma metodologia de melhoria de processo criada por Bill Smith para a Motorola em 1986 [Eckes 2001] com o objetivo de ajudar a empresa a reduzir custos e a melhorar a qualidade dos produtos através de ferramentas analíticas de medição e controle. Inicialmente o foco do Seis Sigma era melhorar os processos de manufatura, com o passar do tempo esta abordagem ganhou maturidade tornando-se amplamente utilizada por organizações de diversas áreas para melhoria de processos organizacionais.

“*Seis Sigma é a inflexível e rigorosa busca da redução da variação em todos os processos críticos para alcançar melhorias contínuas e quânticas que impactam os índices de uma organização e aumentam a satisfação e lealdade dos clientes. É uma iniciativa organizacional projetada para criar processos de manufatura, serviço ou administrativo. A ferramenta de melhoria empregada na implantação dos projetos Seis Sigma é o DMAIC: acrônimo que representa: Definir-Medir-Analisar-Implementar-Controlar*” [Rasis 2002]. Esta ferramenta será definida com mais detalhes na seção 1.5.2.

Existem várias definições para Seis Sigma, cada uma varia conforme a atividade onde está sendo aplicada a metodologia.*“Seis Sigma se refere a um processo em que o intervalo entre a média de um processo de medição da qualidade e o mais próximo da especificação limite é pelo menos, seis vezes o desvio padrão do processo”* [Wang 2008].

Six Sigma possibilita uma capacidade de processo que deve gerar apenas 3,4 defeitos por milhão de oportunidades (DPMO), apresentando 99,99966% de perfeição [Donegan 2005]. Com isso, pode-se inferir que, atingir o nível de Seis Sigma é um processo lento que exige muito planejamento e comprometimento com a qualidade do produto, fato este, que é medido através do histórico das variações dos defeitos ocorridos na produção e pode ser acompanhado pelo DMAIC.

A Tabela X demonstra uma relação numérica de níveis Sigma de acordo com o percentual de perfeição e DPMO do projeto, fazendo uma relação com o tempo desperdiçado causado pelos defeitos existentes.

Tabela x - Níveis Sigma (Fonte: Donegan, 2005)

|  |  |  |  |
| --- | --- | --- | --- |
| **Nível Sigma** | **Percentual Correto (%)** | **DPMO** | **Tempo Desperdiçado por Século** |
| 3 | 93,3193 | 66.807 | 3 ½ meses |
| 4 | 99,3790 | 6.210 | 2 ½ dias |
| 5 | 99,9767 | 233 | 30 minutos |
| 6 | 99,99966 | 3,4 | 6 segundos |

Segundo [Scatolin 2005] não se pode aceitar a ilusão de que Seis Sigma é a solução dos problemas para toda empresa. Deve-se fazer uma análise crítica e verificar se a metodologia é a mais adequada a depender do momento em que a empresa está.

O objetivo do Seis Sigma é suprir as necessidades de uma empresa em melhorar seus processos de forma contínua e sustentável. Através de um forte foco na capacitação e treinamento de seus colaboradores, as empresas que implementam esta metodologia têm a finalidade de diminuir o desperdício com defeitos e possíveis re-trabalhos e um aumento agressivo nos lucros, e com isso, proporcionar uma evolução contínua dos seus processos internos, incentivando o crescimento e melhorando o aproveitamento dos seus funcionários.

Portanto, a estratégia de melhorar o desempenho de processos, o aproveitamento de recursos materiais e o atendimento ao cliente, acompanhados por elevados investimentos no treinamento dos colaboradores e incentivando a criatividade, faz do Seis Sigma uma metodologia que consegue promover o atendimento dos objetivos da empresa que o adota, desde que, uma análise crítica entre custo e benefício seja realizada previamente.

Entretanto, Seis Sigma não é apenas uma simples medida de defeitos. Ela é uma abordagem holística para melhoria dos negócios que abrange: filosofia, medida de desempenho, metodologia para melhoria, e um conjunto de ferramentas. De acordo com Siviy et al. 2008], por causa de suas diversas dimensões adaptáveis as necessidades das organizações, o Seis Sigma tanto pode servir como um modelo de governança empresarial, através da melhoria de processos da alta administração, como um mecanismo técnico de melhoria, através da coleta de métricas de defeitos na produção de um determinado produto.

Por ser a base para formação do DMAIC, a próxima seção abordará o ciclo PDCA que é a ferramenta mais importante e mais utilizada para MPS, é nela que foram baseadas a definição da abordagem IDEAL [McFeeley 1996] e o ciclo de melhoria da ISO/IEC 15504.

## PDCA

Inicialmente o ciclo PDCA foi idealizado por Walter Shewart na década de 30, mas o mesmo só viria a ser disseminado largamente na indústria por [Deming 1986]. O nome do modelo PDCA é derivado do acrônimo das palavras (*Plan-Do-Check-Action,* Planejamento-Execução-Verificação-Ação) o modelo é visto na literatura como o precursor no conceito de melhoria de processos [Moreira 2008, Salviano 2006].

O ciclo PDCA é divido em quatro fases, na primeira fase é feito um planejamento das ações a serem tomadas, estabelecimento de metas e fixação das atividades necessárias para atingir as metas. Na segunda fase são realizadas as atividades fixadas na primeira fase. Já na terceira fase é feita uma análise e verificação dos resultados atingidos. Finalmente, a quarta fase envolve correções de metas e se for necessário, são tomadas ações corretivas ou de melhoria se necessário [Salviano 2006].

## DMAIC

O DMAIC é a ferramenta do Six Sigma mais utilizada na melhoria de processos de software (MPS), ela aborda as fases e as atividades necessárias dentro do ciclo de melhoria utilizado para atingir as metas organizacionais.

Em pesquisa realizada por [Moreira 2008] é citado que o modelo DMAIC foi desenvolvido inicialmente pela Motorola como o modelo MAIC (*Measure, Analyse, Improve e Contro -* Medir, Analisar, Melhorar e Controlar) como evolução do ciclo PDCA e depois adotado pela GE como DMAIC, em que D (*Define*) significa a fase de definição. Esse método passou a ser a base operacional da ruptura do Seis Sigma para essas empresas, sendo fundamental para o sucesso que alcançaram [Harry e Schroeder, 1998, Pande et al.1998, Eckes 2001].

Para uma definição enxuta e clara do ciclo, observe a Figura 11, para uma definição mais detalhada, leia as próximas seções baseadas em pesquisa feita por Morreira [2008] do ciclo DMAIC.

Figura 16 - Fases do ciclo DMAIC, Adaptado de [Siviy et al. , 2008]

## Definir

Essa fase de definição serve como plataforma para a equipe organizar-se, determinar seus papéis e responsabilidades, estabelecer metas e marcos (*milestones*) e rever passos do processo. Os pontos-chave a serem definidos são os requisitos do cliente, escopo do projeto, priorização de causa e escopo, e o planejamento do projeto. Cada um desses passos pode ser alinhado com o cliente e é essencial apreciar e entender essa ligação com o cliente antes e durante esse estágio do modelo.

As ferramentas utilizadas nesta fase incluem dados dos clientes, análise custo benefício, desenho dos macro-processos prioritários.

## Medição

A fase de medição apresenta os objetivos de confirmar e quantificar o problema; identificar variáveis importantes de entrada no processo; medir os passos do processo atual; se necessário, revisar o problema; definir os resultados esperados e exibir as variações usando Diagrama de Pareto, histogramas, *run charts*.

A fase Medição é um importante passo da abordagem, pois os seus resultados são utilizados para a tomada de decisões baseadas em fatos. É também durante esta fase que é calculado o desempenho atual do processo (capacidade Sigma atual do processo).

## Análise

A fase Análise tem como objetivo principal analisar os dados coletados na fase Medição através ferramentas de análise para identificar as causas primárias dos problemas e propor soluções para os mesmos. Pode ser realizado um *brainstorming* durante esta fase, a fim de determinar as melhorias de maior impacto nos requisitos do cliente, levando em consideração os riscos associados.

Nesta fase é utilizada a análise de dados exploratória e descritiva para ajudar a entender os dados. Ferramentas estatísticas como teste de hipóteses, análise de variância e regressão são utilizadas para apoiar esta fase.

## Melhoria

Na fase melhoria o planejamento e as análises realizadas são executados. As decisões baseadas em fatos foram realizadas na fase de análise utilizando métricas da fase de medição, e agora as melhorias do processo resultante poderão ser implementadas, após serem aprovadas.

A solução proposta na fase anterior é avaliada e validada nesta fase e, se conveniente, é aplicada em larga escala na organização. Métodos como Delineamento de Experimentos (DOE) e gráficos estatísticos são empregados para validar a melhoria e o desempenho do processo (sigma) é recalculado.

## Controle

Esta fase é projetada para garantir que os ganhos conseguidos nas fases anteriores não sejam perdidos, medindo as melhorias e garantindo que sejam sustentadas. Para tanto, são elaborados procedimentos para medição e controle do processo (Controle estatístico do processo – CEP). Estes procedimentos são validados e documentados pela equipe do programa de melhoria. Neste momento é validado o desempenho e o retorno financeiro do projeto junto com os patrocinadores e a equipe. Ferramentas como gráficos de controle são utilizados nesta fase do ciclo.

## Considerações Finais

Este capítulo apresentou de forma detalhada os principais modelos para implantação e melhoria do processo de software, o IDEAL, PRO2PI, fundamentado na norma 15504, e o Seis Sigma.

Diante do arcabouço de melhoria de processo de software, fica claro que o mesmo é de uma complexidade, que muitas vezes se torna até subjetiva devido à quantidade de variáveis que envolve a melhoria de processos de software. É preciso que esta atividade passe por uma fase de planejamento e estudo detalhada, como todas as abordagens aqui citadas propõem, nenhuma delas garante que o retorno do investimento será imediato, por isso, é preciso muita cautela, um planejamento estratégico e uma equipe de apoio bem formada e treinada, para que seja possível dar início ao processo de melhoria.

##  Exercícios

Descreva o que é melhoria de processo de software e qual a sua importância para organização.

Descreva o que é IDEAL e qual o objetivo de implementá-lo.

Quantas e quais são as fases do modelo IDEAL? Descreva cada uma delas.

Segundo o modelo Ideal, quais os componentes típicos de uma infra-estrutura de MPS?

Quais as propriedades de PRO2PI? Descreva cada uma delas.

Quais são as funções utilizadas em um ciclo de PRO2PI?

O que é Seis Sigma e qual o seu objetivo principal?

Explique o que é PDCA e quais as suas fases.

Explique o que é DMAIC e quais as suas fases. Descreva cada uma delas.

Descreva um conjunto de práticas que possa unir as práticas do IDEAL, PRO2PI e Seis Sigma, para MPS.

## Sugestões de leitura

Para entender melhor o que é Melhoria de processo de software lei a norma ISO/IEC 15504-4/2004.

Para um estudo detalhado sobre o PRO2PI leia tese de doutorado de Clênio Salviano, “Uma Proposta Orientada a Perfis de Capacidade de Processo para Evolução da Melhoria de Processo de Software”. Disponível em: http://libdigi.unicamp.br/document/?code=vtls000380495

Para um estudo detalhado sobre IDEAL leia o guia oficial de implantação produzido pelo SEI, “IDEAL - A User's Guide for Software process Improvement”. Disponível em: http://www.sei.cmu.edu/library/abstracts/reports/96hb001.cfm

## Tópicos de pesquisa

Visto que a tarefa de MPS é realizada por pessoas os **aspectos humanos** envolvidos na MPS devem ser estudados com maior detalhe**.**

**Aspectos psicológicos e sociais** da MPS que podem influenciar da eficiência e eficácia de um programa de MPS.

Como a tarefa de MPS é complexa e muitas vezes subjetiva, estudar um **conjunto de Métricas eficazes para MPS** é uma tarefa delicada e que exige muitos anos de pesquisa**.**

**O comprometimento dos Recursos Humanos** com a MPS, ainda deve ser muito pesquisado.

**Novas normas e padrões de MPS surgirão com o advento das Metodologias Ágéis,** visto que a necessidade de Time-to-Marketing diminui.

**MPS X Metodologias ágeis,** ainda existem poucos estudos relacionados a esse tema.

|  |
| --- |
| Referências Chrissis, M. Beth, Konrad, Mike and Shrum, Sandy (2003) “CMMI: Guidelines for Process Integration and Product Improvement”, Addison-Wesley Pub Co, 2003. |
| Data & Analysis Center for Software, A Business Case for Software Process Improvement Revised - Measuring Return on Investment from Software Engineering and Management, A DACS State-of-the-Art Report, Contract Number SP0700-98-4000, Prepared By Thomas McGibbon, 30 September 1999. (disponível em http://www.dacs.dtic.mil/techs/roispi2/, acessado mais em 23/08/2009) |
| David N. Card, Published Sources of Benchmarking Data, memorandum, 5 pages, Software Productivity Consortium, March 2002. |
| Deming, W. E. “Out of the Crisis”. MIT Center for Advanced Engineering Study, Cambridge, MA. 1986. |
| Donegan, P. M. “Medição de Qualidade de Software sob a Perspectiva do Six Sigma e CMMI”. Monografia de conclusão do curso de Ciência da Computação da Universidade Estadual do Ceará, Fortaleza. 2005. |
| Eckes, G. “Six Sigma for Everyone”. John Wiley & Sons, Inc. 2001. |
| Habib, M., Ahmed, S., Rehmat, A., Khan, M. J., & Shamail, S. (n.d.). Blending Six Sigma and CMMI - An Approach to Accelerate Process Improvement in SMEs. Framework, 386-391 p., 2008. |
| Harry, M. e Schroeder, R. “Six Sigma: The Breakthrough Management Strategy Revolutionizing the World’s Top Corporations”. Currency. 1998. |
| Humphrey, W. S. “Managing the software process”. Boston: Addison-Wesley. 1989. |
| IEEE. “IEEE Standard 610.12-1990 Glossary of Software Engineering Terminology”. IEEE CS Press. 1990. |
| ISO 9001 (2000) “The International Organization for Standardization - Associação Brasileira de Normas Técnicas, NBR ISO 9001:2000 – Sistemas de gestão da qualidade – Requisitos”, Rio de Janeiro, 2000. |
| ISO 9004 (2000), “The International Organization for Standardization - Associação Brasileira de Normas Técnicas, NBR ISO 9004:2000 – Sistemas de gestão da qualidade – Diretrizes para melhorias de desempenho”, Rio de Janeiro, 2000. |
| ISO/IEC 12207 (1998) “Associação Brasileira de Normas Técnicas, NBR ISO/IEC 12207 – Tecnologia de Informação - Processos de ciclo de vida de software”, Rio de Janeiro, 1998. |
| ISO/IEC 15504-4 (2004) “The International Organization for Standardization and the International Electrotechnical Commission, ISO/IEC 15504 - Information Technology - Process Assessment – Part 4: Guidance on using assessment results”, 2004. |
| ISO/IEC 15939 (2002) The International Organization for Standardization and the International Electrotechnical Commission, ISO/IEC 15939 – Software engineering – Software measurement process, 37 pages. |
| ISO/IEC TR 15504 (1998) “The International Organization for Standardization and the International Electrotechnical Commission, ISO/IEC TR 15504 - Information Technology - Software Process Assessment”, document set with nine parts: ISO/IEC TR 15504-1 to ISO/IEC TR 15504-9, 1998. |
| ISO/IEC TR 15504-7 (1998) “The International Organization for Standardization and the International Electrotechnical Commission, ISO/IEC TR 15504-7 - Information Technology - Software Process Assessment - Part 7 : Guide for use in process improvement”, Technical Report, 1998. |
| James Herbsleb, A. Carleton, J. Rozum, J. Siegel, David Zubrow, Benefits of CMM-Based Software Process Improvement: Initial Results CMU/SEI-94-TR-013, Software Engineering Institute, Carnegie Mellon University, Pittsburgh, PA: 1994. |
| McFeeley, Bob (1996) “IDEAL - A User's Guide for Software process Improvement”, Handbook CMU/SEI-96-HB-001, 236 pages, 1996.  |
| McGarry, J., Card, D., Jones, C., Layman, B., Clark, E., Dean, J., Hall, F. (2002) Practical Software Measurement: Objective Information for Decision Makers, PSM, Addison Wesley Professional, ISBN: 0201715163, 304 p., 2002. |
| Moreira, R. T. (2008) “Uma Abordagem para Melhoria do Processo de Software baseada em Medição”, dissertação de mestrado. |
| Neil S. Potter and Mary E. Sakry (2002) “Making Process Improvement Work: A Concise Action Guide for Software Managers and Practitioners”, Addison-Wesley Professional, ISBN 0201775778, 2002. |
| O’Toole, P. (2000) “Do’s and Don’ts of Software Process Improvements”, slides from tutorial presented at SIMPROS 2000 (Salviano e Santana 2000). |
| Pande, P. S., Neuman, R. P. e Cavanagh, R. R. “The Six Sigma Way: How GE, Motorola, and Other Top Companies Are Honing Their Performance”. McGraw-Hill. 1998 |
| Paulk, Mark C., Weber, Charles V., Curtis, Bill and Chrissis, M. Beth (1994) “The Capability Maturity Model - Guidelines for Improving the Software Process”, CMU-SEI, Addison-Wesley, 441 pages, 1994. |
| Pressman, R. “Software Engineering: A Practitioner’s Guide”. McGraw-Hill. 2002. |
| Rasis, D., Gitlow, H.S., Popovich, E. Paper Organizers International: A Fictitious Six Sigma Green Belt Case Study I. Quality Engineering, 15 (1), pp.127-145, 2002. |
| Rocha, Ana R. C. da, Maldonado, José C. e Weber, Kival C. (2001), “Qualidade de Software: Teoria e Prática”, Prentice Hall, 303 páginas, 2001. |
| Salviano, C. F. “Uma Proposta Orientada a Perfis de Capacidade de Processo para Evolução da Melhoria de Processo de Software”. Tese (Doutorado) - Universidade Estadual de Campinas, Campinas, 2006. |
| Salviano, Clênio F. e Filho, Ozeas V. S. (Editores) (2000), Anais e Slides das Apresentações do SIMPROS 2000: Segundo Simpósio Internacional de Melhoria de Processo de Software, São Paulo, 442 páginas, 2000. |
| Scatolin, André Celso, Application of Six Sigma Methodology in order to Reduce Waste of a Manufacturing Process, Campinas: Faculdade de Engenharia Mecânica, Universidade Estadual de Campinas, 2005. 137 p. Trabalho Final de Mestrado Profissional. |
| Sheard, Sarah A., “Evolution of the Frameworks Quagmire”, IEEE Computer, July 2001, pp. 96-98. |
| Shewhart, W. A. “Economic Control of Quality of Manufactured Product”. American Society for Quality Control. 1980. |
| Sommerville, I. “Software Engineering”. Addison-Wesley. 2006 |
| Wang, H. (2008). A Review of Six Sigma Approach: Methodology, Implementation and Future Research. 2008 4th International Conference on Wireless Communications, Networking and Mobile Computing, 1-4. IEEE. |
| Yang, K. e El-Haik B. (2003) “Design for SIX SIGMA: A roadmap for product development”, McGraw-Hill. P. 12-13. |

Índice

[10.1 Introdução a modelos para melhoria de processos de software 2](#_Toc243989095)

[10.2 IDEAL 3](#_Toc243989096)

[10.2.1 Fases do IDEAL 5](#_Toc243989097)

[10.2.1.1 Fase inicial (Initiating) 6](#_Toc243989098)

[10.2.1.2 Fase de diagnóstico (Diagnosing) 8](#_Toc243989099)

[10.2.1.3 Fase de estabilização (Diagnosing) 10](#_Toc243989100)

[10.2.1.4 Fase de ação (Acting) 12](#_Toc243989101)

[10.2.1.5 Fase de aproveitamento (Leveraging) 14](#_Toc243989102)

[10.2.1.6 Fase de gerenciamento do programa de melhoria do processo de software (Manage) 16](#_Toc243989103)

[10.3 PRO2PI 18](#_Toc243989104)

[10.3.1 Engenharia de processo dirigida por perfis de capacidade e seus fundamentos 19](#_Toc243989105)

[10.3.2 O PRO2PI 20](#_Toc243989106)

[10.3.2.1 PRO2PI-PROP: Propriedades de PRO2PI 25](#_Toc243989107)

[10.3.2.2 PRO2PI-MODEL: Modelo de PRO2PI 26](#_Toc243989108)

[10.3.2.3 PRO2PI-MEAS: Medições para PRO2PI 28](#_Toc243989109)

[10.3.2.4 PRO2PI-CYCLE: Processo para ciclo de melhoria 29](#_Toc243989110)

[10.4 Seis Sigma 30](#_Toc243989111)

[10.4.1 PDCA 33](#_Toc243989112)

[10.4.2 DMAIC 33](#_Toc243989113)

[10.4.2.1 Definir 34](#_Toc243989114)

[10.4.2.2 Medição 34](#_Toc243989115)

[10.4.2.3 Análise 35](#_Toc243989116)

[10.4.2.4 Melhoria 35](#_Toc243989117)

[10.4.2.5 Controle 35](#_Toc243989118)

[10.5 Considerações Finais 36](#_Toc243989119)

[10.6 Exercícios 37](#_Toc243989120)

[10.7 Sugestões de leitura 38](#_Toc243989121)

[10.8 Tópicos de pesquisa 39](#_Toc243989125)

[Referências 40](#_Toc243989126)

1. Uma *baseline* é um conjunto de especificações sobre os quais foram feito um acordo, que serve como base para as ações posteriores. [↑](#footnote-ref-1)
2. PRO2PI: *PROcess capability PROfile to Process Improvement*, onde o 2 representa ao mesmo tempo a repetição de PRO e também o termo *to*. PRO2PI deve ser pronunciado em inglês, de forma semelhante a “pró tru pai”. (Salviano, 2006). [↑](#footnote-ref-2)
3. [↑](#footnote-ref-3)