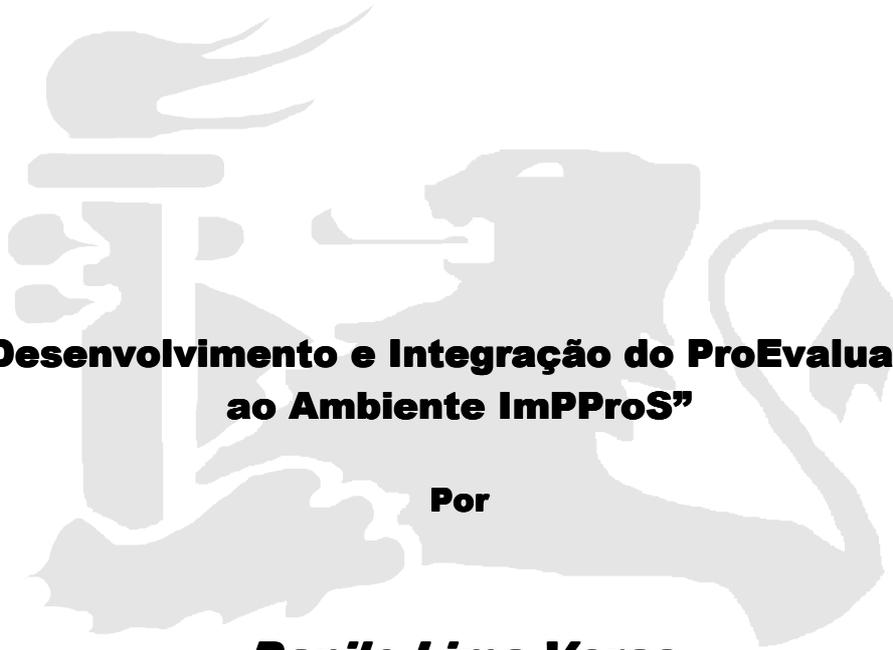




**Graduação em Ciência da Computação**



**“Desenvolvimento e Integração do ProEvaluator  
ao Ambiente ImPProS”**

**Por**

***Danilo Lima Veras***

**Trabalho de Graduação**



Universidade Federal de Pernambuco  
graduacao@cin.ufpe.br  
www.cin.ufpe.br/~graduacao

RECIFE, 22 de Junho de 2009



UNIVERSIDADE FEDERAL DE PERNAMBUCO  
CENTRO DE INFORMÁTICA  
GRADUAÇÃO EM CIÊNCIA DA COMPUTAÇÃO

DANILO LIMA VERAS

*Desenvolvimento e Integração do ProEvaluator ao  
Ambiente ImPProS*

*ESTE TRABALHO FOI APRESENTADO À GRADUAÇÃO EM CIÊNCIA  
DA COMPUTAÇÃO DO CENTRO DE INFORMÁTICA DA  
UNIVERSIDADE FEDERAL DE PERNAMBUCO COMO REQUISITO  
PARCIAL PARA OBTENÇÃO DO GRAU DE BACHAREL EM CIÊNCIA  
DA COMPUTAÇÃO.*

**ORIENTADOR:** PH. D. ALEXANDRE MARCOS LINS DE  
VASCONCELOS

RECIFE, 22 DE JUNHO DE 2009

*"Obstáculos são aquelas coisas assustadoras que você vê  
quando desvia seus olhos de sua meta."*

*(Henry Ford)*

# Agradecimentos

*Gostaria de agradecer em primeiro lugar a Deus, por ter me dado tanta perseverança durante todo esse período, por não deixar me faltar motivação para terminar esse trabalho e por me dar inspiração quando faltava.*

*Ao meu professor e orientador Alexandre Vasconcelos que durante todo esse período muito contribuiu para minha formação. Agradeço também pela ajuda, orientação, disponibilidade e paciência para me atender, mesmo quando o tempo faltava.*

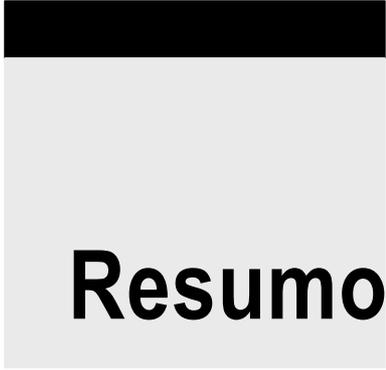
*Agradeço aos meus pais, que não só nesse período, mas por toda a minha vida acreditaram na minha capacidade e potencial.*

*Agradeço a minha namorada por sua eterna dedicação e preocupação comigo. Apesar das minhas dificuldades e impaciência, sempre esteve ao meu lado me apoiando.*

*Agradeço a todos os meus amigos de graduação que, junto comigo, também venceram mais esse obstáculo.*

*Aos meus amigos, pelo incentivo e pelo apoio, sendo sempre responsáveis por grande parte dos meus momentos de alegria.*

*Não sei o que seria de mim sem cada um de vocês! Obrigado por tudo!*

A graphic element consisting of a black horizontal bar at the top, followed by a light gray rectangular area containing the word "Resumo" in a large, bold, black sans-serif font.

# Resumo

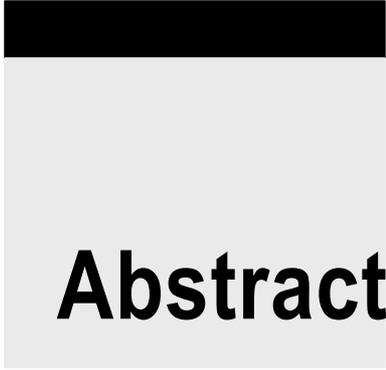
No final da década de 60, foi identificado um fenômeno no desenvolvimento de software batizado de “Crise do Software”. Grandes empresas e órgãos governamentais amargavam uma série de prejuízos na produção de sistemas informatizados, que na maioria das vezes, não satisfaziam suas necessidades. Tudo porque o desenvolvimento de software era realizado de maneira ad-hoc. Os escassos casos de sucesso eram resultado do esforço individual de profissionais comprometidos.

Para padronizar as boas práticas no desenvolvimento de sistemas, surgiram os modelos de processos de desenvolvimento de software. A partir daí, aumentou-se a expectativa de que todos os problemas de desenvolvimento seriam resolvidos com a adoção de um processo de desenvolvimento. Isso não ocorreu porque não existe um processo ideal para todas as empresas de software. O que existem são modelos de processos que devem ser customizados para cada organização, o que se torna um grande desafio saber se um processo foi adequadamente definido e como mantê-lo em constante melhoria. Para se certificar quanto a essas questões é necessário realizar uma avaliação de seus processos de software.

A mudança das necessidades de mercado, vem fazendo com que as empresas tenham que definir processos em diferentes modelos e conseqüentemente possuir diferentes processos de avaliação.

Este trabalho propõe uma ferramenta Web genérica baseada no processo de avaliação da ISO 15504. Ela qual permite o cadastro de qualquer modelo ou método de avaliação. Também permite atribuir notas ou conceitos aos componentes dos modelos, entre outras funcionalidades. Essa ferramenta também deve ser integrada ao ambiente de implementação progressiva de software, o ImPProS.

**Palavras-chave:** ferramentas de avaliação de processos de software, genérica, modelos e normas de qualidade de software, métodos de avaliação, ISO 15504, CMMI, MPS.BR.



# Abstract

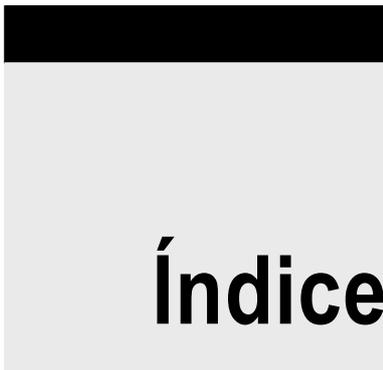
In the late 60s, a phenomenon in software development was identified and known as “Software crisis”. Big companies and governmental bodies endured a series of losses in the IT systems production, which, most of times, did not meet their needs. This was due to software development being done ad-hoc. The scarce success cases were the result of individual effort from committed professionals.

Models of software development processes then appeared to standardize the best practices in systems development. From that day on, there was an increase in the expectations that all issues in development would be addressed by adopting a development process. That did not occur because there is no ideal process for all software companies. What truly exist are models of processes that must be customized for each organization. Define a process correctly and define a continuous process improvement are the biggest challenges for the companies. To analyze whether a model is appropriate to a given organization, software processes evaluation must be carried out.

The new business requirements have made the companies to define different processes with a different models. So, the companies also needs to have different kinds of Assessments Methods.

This work proposes a generic Web tool to support the process assessment activities described in the ISO 15504. The tool allows the record of any kind of model or assessment methods. It also allows to rate the model’s component. This tool is integrated with the ImPProS.

**Keywords:** software process assessment tools, generic, software quality models, assessment methods, ISO 15504, CMMI, MPS.BR.

A graphic element for the index, consisting of a black horizontal bar at the top, a light gray rectangular area below it, and the word "Índice" in a large, bold, black sans-serif font centered within the gray area.

# Índice

<b>1 INTRODUÇÃO .....</b>	<b>1</b>
1.1 CONTEXTUALIZAÇÃO .....	1
1.2 MOTIVAÇÃO.....	3
1.3 OBJETIVOS.....	4
1.4 METODOLOGIA DE TRABALHO .....	7
1.5 ESTRUTURA DO DOCUMENTO .....	7
<b>2 MODELOS PARA MELHORIA E AVALIAÇÃO DE PROCESSOS DE SOFTWARE ...</b>	<b>9</b>
2.1 PROCESSO DE SOFTWARE .....	10
2.2 AVALIAÇÃO DE PROCESSO DE SOFTWARE.....	11
2.3 MODELOS E NORMAS DE QUALIDADE DE SOFTWARE .....	13
2.3.1 ISO/IEC 12207 .....	15
2.3.2 ISO/IEC 15504 .....	17
2.3.3 CMMI – Capability Maturity Model Integration.....	23
2.3.4 MPS.BR - Programa de Melhoria do Processo de Software Brasileiro.....	32
2.4 MÉTODOS DE AVALIAÇÃO DE PROCESSOS DE SOFTWARE.....	38
2.4.1 ISO/IEC 15504 .....	39
2.4.2 SCAMPI – Standard CMMI Appraisal Method for Process Improvement.....	40
2.4.2.1 Detalhamento das etapas: .....	41
2.4.2.2 Classes de Avaliação.....	42
2.4.3 MA-MPS.....	44
2.5 CONSIDERAÇÕES FINAIS:.....	45
<b>3 IMPROS - UM AMBIENTE DE IMPLEMENTAÇÃO PROGRESSIVA DE PROCESSO DE SOFTWARE.....</b>	<b>47</b>
3.1 CONTEXTUALIZAÇÃO .....	47
3.2 AMBIENTES DE DESENVOLVIMENTO DE SOFTWARE ORIENTADOS A PROCESSO .....	48
3.2.1 O Ambiente ImPProS.....	50
3.2.1.1 Arquitetura .....	51
3.2.1.2 Ciclo de Vida de Processo no Ambiente ImPProS .....	55
3.3 CONSIDERAÇÕES FINAIS .....	58
<b>4 FERRAMENTAS PARA AVALIAÇÃO DE PROCESSOS DE SOFTWARE .....</b>	<b>59</b>
4.1 APPRAISAL ASSISTANT .....	59
4.2 AMBIENTE DE APOIO ÀS AVALIAÇÕES MPS.BR .....	60
4.3 SEAL .....	62
4.4 PISA .....	63
4.5 PROEVALUATOR – MPS.BR .....	64
4.6 COMPARATIVO ENTRE AS FERRAMENTAS.....	65
4.7 CONSIDERAÇÕES FINAIS .....	67
<b>5 PROEVALUATOR: UMA FERRAMENTA PARA AVALIAÇÃO DE PROCESSOS DE SOFTWARE .....</b>	<b>69</b>

5.1 VISÃO GERAL .....	70
5.2 REQUISITOS .....	71
5.3 ARQUITETURA.....	74
5.3.1 <i>Design Patterns</i> .....	75
5.3.1.1 Padrões Estruturais .....	76
5.3.1.2 Padrões Comportamentais.....	76
5.3.1.3 Padrões Criacionais.....	77
5.4 INTEGRAÇÃO COM O IMPPROS.....	77
5.5 CONSIDERAÇÕES FINAIS .....	79
<b>6 CONCLUSÃO.....</b>	<b>81</b>
6.1 CONTRIBUIÇÕES DO TRABALHO .....	82
6.2 TRABALHOS FUTUROS.....	83



# Lista de Figuras

FIGURA 1.1 - CICLO DE MELHORIA CONTÍNUA [ISO/IEC 15504-1, 2004].....	3
FIGURA 2.1 – PRINCIPAIS ELEMENTOS DE UM PROCESSO DE AVALIAÇÃO [ISO/IEC 15504-1, 2004].....	12
FIGURA 2.2 - PROCESSOS DA NORMA ISO/IEC 12207.....	17
FIGURA 2.3 - COMPONENTES DA ISO/IEC 1550.....	19
FIGURA 2.4 – ESTRUTURA DO MODELO DE REFERÊNCIA UTILIZADO NA PARTE 5 DA ISO/IEC 15504 [ISO/IEC 15504-5, 2006].....	20
FIGURA 2.5- DIAGRAMA DE CLASSES DA ISO/IEC 15504 [WANGENHEIM, 2005].....	23
FIGURA 2.6 – ABORDAGEM ESTAGIADA E CONTÍNUA DO MODELO CMMI [CMU/SEI, 2006]. ...	25
FIGURA 2.7 - EQUIVALÊNCIA ENTRE A REPRESENTAÇÃO CONTÍNUA E ESTAGIADA DO CMMI....	29
FIGURA 2.8 - COMPONENTES DO CMMI [CMU/SEI, 2006].....	30
FIGURA 2.9 - DIAGRAMA DE CLASSES DO CMMI [WANGENHEIM, 2005].....	31
FIGURA 2.10 - COMPONENTES DO PROGRAMA MPS.BR [MPS.BR, 2007A].....	33
FIGURA 2.11 – NÍVEIS DE MATURIDADE DO MR-MPS [MPS.BR, 2007A].....	34
FIGURA 2.12 - ESTRUTURA DO MPS.BR .....	35
FIGURA 2.13 - PROCESSOS DO MPS.BR [MPS.BR, 2007A].....	36
FIGURA 2.14 - DIAGRAMA DE CLASSES DO MPS.BR [WANGENHEIM, 2005].....	38
FIGURA 2.15 - CICLO PARA AVALIAÇÃO DE PROCESSO NA ISO/IEC 15504.....	39
FIGURA 2.16 - PROCESSOS DE AVALIAÇÃO DO MDD [CMU/SEI, 2001].....	41
FIGURA 2.17 - PROCESSOS MA-MPS [MPS.BR, 2007].....	44
FIGURA 3.1 - ARQUITETURA DO IMPPROS [OLIVEIRA, 2005].....	52
FIGURA 3.2 - RELACIONAMENTO DOS MÓDULOS DE GERENCIAMENTO DO IMPPROS .....	53
FIGURA 3.3 - FERRAMENTAS DE APOIO AO AMBIENTE IMPPROS.....	55
FIGURA 3.4 - CICLO DE VIDA DE PROCESSO NO IMPPROS [OLIVEIRA, 2005].....	56
FIGURA 4.1 - REGISTRO DE EVIDÊNCIAS CMMI .....	60
FIGURA 4.2 - CADASTRO DA ORGANIZAÇÃO AVALIADA .....	61
FIGURA 4.3 - DEFINIÇÃO DE PROCESSOS NA FERRAMENTA SEAL .....	63
FIGURA 4.4 - ATRIBUIÇÃO DE NOTAS DA FERRAMENTA PISA.....	64
FIGURA 4.5 - AVALIAÇÃO DE PROCESSO NA FERRAMENTA [MOURA, 2007].....	65
FIGURA 4.6 – COMPARATIVO DAS FERRAMENTAS DE AVALIAÇÃO.....	66
FIGURA 5.1 – FASES CONTEMPLADAS PELO PROEVALUATOR .....	70
FIGURA 5.2 - DIAGRAMA DE CASOS DE USO .....	74
FIGURA 5.3 - PACOTES DO PROEVALUATOR .....	75
FIGURA 5.4 - ESTRUTURA DE INTEGRAÇÃO DO IMPPROS [OLIVEIRA, 2007].....	77
FIGURA 5.5 – EXECUÇÃO DO PROEVALUATOR ATRAVÉS DO IMPPROS.....	79

# Introdução

*Este capítulo introduz o trabalho descrito neste trabalho de graduação, descrevendo em linhas gerais a pesquisa desenvolvida. Na seção 1.1 é apresentado o contexto no qual o trabalho está inserido. Na seção 1.2 é descrita a sua motivação, apresentando as principais necessidades existentes, que levaram ao seu desenvolvimento. A seção 1.3 descreve os principais objetivos da pesquisa realizada, definindo a abordagem utilizada para alcançar tais objetivos. A seção 1.4 descreve a metodologia aqui utilizada. A seção 1.5 apresenta a organização desta dissertação, definindo o assunto abordado em cada um dos capítulos.*

## 1.1 Contextualização

No final da década de 60, foi identificado um fenômeno no desenvolvimento de software batizado de “Crise do Software”. Grandes empresas e órgãos governamentais amargavam uma série de prejuízos na produção de sistemas informatizados, que na maioria das vezes, não satisfaziam suas necessidades. Tudo porque o desenvolvimento de software era realizado de maneira ad-hoc. Os escassos casos de sucesso eram resultado do esforço individual de profissionais comprometidos. Para padronizar as boas práticas no desenvolvimento de sistemas, surgiram os modelos de processos de desenvolvimento de software.

As empresas então dando cada vez mais importância à qualidade do software que estão produzindo. Devido à alta concorrência, uma empresa que não produz software com o mínimo de qualidade aceitável (considerando os requisitos de diversos pontos de vista) está fadada ao fracasso. Uma das evoluções mais importantes no estudo da qualidade está em notar que a qualidade do produto é algo bom, mas que qualidade do processo de produção é ainda mais importante [Junior 1997].

Para uma boa especificação de um processo de software, é necessário levar em consideração características da organização, metas estratégicas, assim como características da equipe e do projeto. Daí, a impossibilidade de se obter um processo único para todas as situações. Um processo mal especificado pode levar a um longo tempo de desenvolvimento, não aderência aos requisitos do cliente ou até um alto custo do projeto, entre outros problemas, ou seja, o software terá uma baixa qualidade. Por isso, é imprescindível uma correta especificação do processo de forma que ele se adéque ao perfil da organização.

Hoje em dia, pode-se consultar normas e padrões tanto para produtos quanto para processos. Entretanto, é comum encontrar empresas que perseguem os dois tipos de padrão de qualidade. Os estudos sobre qualidade mais recentes são, na sua maioria, voltados para o melhoramento do processo de desenvolvimento de software. Não é que a qualidade do produto não seja importante, o fato é que, ao garantir a qualidade do processo, já se está dando um grande passo para garantir também a qualidade do produto [Junior 1997].

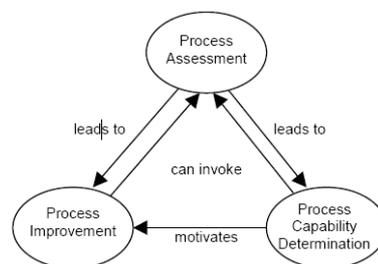
Por isso, as empresas estão controlando os elementos envolvidos no desenvolvimento do produto produzido. Esse controle só é possível fazendo-se uma avaliação dos seus processos. Para se implementar a melhoria de processo numa organização, é necessário fazer uma auto-avaliação, comparar o resultado com os processos de outras organizações, e por fim, traçar as metas que se deseja alcançar.

Quando uma organização realiza uma avaliação seu objetivo é medir a qualidade dos seus processos e, conseqüentemente, conseguir melhorá-los. Neste contexto, a empresa vai utilizar o resultado da avaliação para identificar oportunidades de melhoria, assim como manter o que já está funcionando bem.

Outra abordagem possível é a organização realizar uma avaliação dos processos de seus fornecedores. Assim, ela pode usar os resultados dessa avaliação para analisar a capacidade dos processos de seus fornecedores e, assim, identificar os melhores fornecedores. Em qualquer das duas situações, é fundamental que seja seguido um método de avaliação de processos, como por exemplo um método baseado na norma ISO/IEC 15504 [ISO/IEC 15504-5, 2006], o SCAMPI [CMU/SEIa, 2002] ou até mesmo o MA-MPS [MPS.BR, 2006b].

Por outro lado, para apoiar a modelagem e a execução de processo, têm sido propostos ambientes de desenvolvimento de software centrados no processo, os quais englobam, além das ferramentas de apoio ao desenvolvedor, ferramentas que permitem modelagem do processo de software e execução do mesmo. Desta forma, o ambiente “conhece” o processo a ser seguido e pode, assim, orientar os desenvolvedores na execução de suas tarefas, além de executar automaticamente tarefas repetitivas [Reis 2000].

A utilização de um processo de avaliação inserido num contexto de um ambiente de implementa de processos proporcionam um ciclo de melhoria contínua [ISO/IEC 15504-1, 2004], conforme pode ser visto na figura 1.1.



**Figura 1.1 - Ciclo de melhoria contínua [ISO/IEC 15504-1, 2004]**

## 1.2 Motivação

Há diversos anos vimos aumentar consideravelmente a qualidade dos sistemas de software produzidos [STANDISH, 1995]. Porém, esse avanço ainda não é suficiente. Precisamos melhorar ainda mais a qualidade dos processos de software e, conseqüentemente, o produto final que é próprio software. Melhorar a qualidade significa diminuir o número de erros, o tempo de desenvolvimento, o custo ou até mesmo atender melhor as expectativas do cliente.

Para melhorar a qualidade dos processos de desenvolvimento de software diversos modelos de processos foram desenvolvidos. Porém, a simples adoção de um modelo de processo de software não garante que a qualidade do mesmo aumentará. Para medir a qualidade do processo de desenvolvimento de software da organização é preciso realizar uma avaliação nos seus processos.

A constante mudança nos requisitos do mercado vem obrigando às empresas a definirem processos baseados em diversos modelos de processo. Dessa maneira, a empresa tem a necessidade de realizar a avaliação de seus processos baseando-se em diversos modelos de processo.

A necessidade de mobilidade e acesso, por parte dos usuários, à aplicação, a partir de diferentes locais e a necessidade de uma ferramenta que possibilite uma interação maior entre a organização e a instituição avaliadora se faz cada vez mais presente.

Para automatizar esta atividade, esse trabalho propõe o desenvolvimento de uma ferramenta Web para avaliação de processos de software. Essa ferramenta deve ser genérica, ou seja, possibilitar a avaliação a partir de qualquer modelo de processo. Essa ferramenta pode ser utilizada para conhecer os pontos fortes e fracos do processo de uma organização e, a partir daí, traçar um plano de ação com o objetivo de melhorar o processo da mesma.

## **1.3 Objetivos**

A proposta da ferramenta ProEvaluator é automatizar o processo de avaliação proposto pela norma ISO 15504 [ISSO/IEC 15504-1, 2004]. A ferramenta deve cobrir as fases do processo proposto pela norma a fim de que possa ser implementada uma ferramenta de avaliação genérica (pode utilizar qualquer modelo de processo), visto que a norma ISO 15504 foi criada como um padrão de avaliação de processos e se caracteriza como um framework para criação de métodos de avaliação.

O ProEvaluator foi implementado sobre uma plataforma Web, pois, dessa maneira, o usuário pode acessar a ferramenta de qualquer lugar que possua internet. Essa arquitetura também possibilitará uma maior interação entre a

organização e a instituição avaliadora. Essa ferramenta pode ser utilizada para as organizações se auto-avaliarem ou pode ser utilizada pelas instituições avaliadoras para executar a avaliação oficial.

Outro importante objetivo desse trabalho é integrar o ProEvaluator ao ambiente de implementação de processos, o ImPProS, possibilitando a implementação do ciclo de melhoria contínua.

Baseado no estudo dos requisitos das principais ferramentas de avaliação de processos do mercado, nas premissas propostas pelo ImPProS e as possibilidades de melhoria propostas em [Moura, 2007] foram extraídos os seguintes requisitos para ferramenta ProEvaluator:

- Gratuidade – A ferramenta deve ser gratuita e disponível para uso público;
- Ferramenta Web – A ferramenta deverá ser implementada em uma arquitetura web de forma a possibilitar o acesso dos dados a partir de qualquer usuário, localidade, dispositivo ou sistema operacional;
- Ferramenta genérica – A ferramenta deve suportar qualquer modelo de processos e seu respectivo método de avaliação na realização da avaliação. Isso é uma premissa para o funcionamento integrado ao ImPProS;
- Possuir o ciclo de avaliação da norma ISO 15504: A ferramenta deve suportar as fases de planejamento, coleta de dados, validação de dados e pontuação de atributos de processo;
- Manter organização: A ferramenta deve possibilitar o cadastro, remoção e listagem das organizações a serem avaliadas;
- Manter projeto: A ferramenta deve possibilitar o cadastro, remoção e listagem dos projetos das organizações;
- Manter equipes dos projetos: A ferramenta deve possibilitar o cadastro, remoção e listagem dos integrantes dos projetos;

- Manter modelos: A ferramenta deve possibilitar o cadastro, remoção e listagem de modelos e normas de qualidade e seus respectivos componentes;
- Manter método de avaliação: A ferramenta deve possibilitar o cadastro, remoção e listagem de métodos de avaliação;
- Manter regras para atribuição de notas: A ferramenta deve permitir o cadastro, remoção e listagem de regras para atribuição de notas aos componentes do modelo. Essas regras são baseadas nos métodos de avaliação de cada modelo e são utilizadas pela ferramenta para geração de notas;
- Suporte a atribuição de notas ou conceitos: A ferramenta deve possibilitar a atribuição de notas para os componentes dos modelos de acordo com a análise das evidências da avaliação. A ferramenta também deve prever a nota ou conceito com base nas regras cadastradas;
- Associar evidências - A ferramenta deve possibilitar a associação de evidências aos componentes do modelo utilizado na avaliação;
- Manter avaliação: A ferramenta deve possibilitar o cadastro, remoção, listagem e execução de avaliação;
- Mapear modelos: A ferramenta deve possibilitar o mapeamento entre os componentes dos modelos de forma que o usuário possa obter uma previsão de uma avaliação em um modelo a partir de uma avaliação utilizando um outro modelo previamente mapeado;
- Manter avaliadores – A ferramenta deve possibilitar o cadastro, remoção e listagem de avaliadores. Cada avaliador deverá ter um perfil de acesso;
- Controlar acesso – A ferramenta deve gerenciar perfil dos usuários e realizar a autenticação dos mesmos, de forma a evitar que um usuário execute alguma atividade sem permissão;
- Integração com o ambiente ImPProS – A ferramenta deverá ser integrado ao ambiente ImPPros, podendo ser executada a partir do

mesmo, utilizar informações de sua base e alimentá-la com os resultados das avaliações;

A relevância e originalidade do trabalho estão no fato de não existirem ferramentas para avaliação de processos de software genérica, a qual possibilite o cadastro de modelos e métodos de qualidade e executar mapeamentos entre eles. Outro ponto é que, partir dos requisitos identificados, essa ferramenta propõe uma interface única entre a instituição avaliadora e a organização a ser avaliada, possibilitando uma maior interação entre os mesmos. Outro ponto é que as ferramentas existentes não estão integradas a um ambiente de implementação de processo, impossibilitando um ciclo de melhoria contínua automatizado.

## **1.4 Metodologia de Trabalho**

Como metodologia de pesquisa usada para reunir informações foi a pesquisa Bibliográfica, a fim de conhecer os conceitos e características dos principais modelos e normas de qualidade e seus respectivos métodos de avaliação. Essa metodologia também foi utilizada para o entendimento do ambiente ImPProS (Implementação de Processo de Software) [OLIVEIRA, 2005] para a possibilidade de integração entre as ferramentas.

A partir da pesquisa bibliográfica, foi possível implementar um modelo de avaliação genérico para a ferramenta o qual visa estar de acordo com qualquer outro modelo.

## **1.5 Estrutura do Documento**

Além deste capítulo introdutório, esta dissertação está organizada nos capítulos a seguir: O capítulo 2 descreve o estado da arte em avaliação de processos, mostrando alguns conceitos em relação ao assunto, uma descrição do processo de avaliação, os principais modelos, normas e métodos de avaliação de processos; O capítulo 3 descreve o ambiente de implementação progressiva de processos, o ImPProS; O capítulo 4 apresenta as principais ferramentas de avaliação de processos existentes; O capítulo 5 descreve a ferramenta de

avaliação ProEvaluator e sua integração com o ImPProS; Finalmente, o capítulo 6 apresenta uma conclusão acerca do trabalho desenvolvido, resumindo as principais contribuições do mesmo, bem como uma proposta de possíveis trabalhos futuros.

# Modelos para Melhoria e Avaliação de Processos de Software

*Segundo [ROCHA, 2001], qualidade de software é a área de conhecimento da Engenharia de Software que objetiva garantir a melhoria de software através da definição e normatização de processos de desenvolvimento. Atualmente, o grande desafio das empresas de software consiste justamente em saber como definir adequadamente o processo de desenvolvimento de software em relação às necessidades da organização e do mercado e como garantir uma melhoria contínua desses processos.*

*Este capítulo tem por finalidades apresentar alguns fundamentos relacionados à processo de software, apresentar os conceitos referentes ao processo de avaliação de processos de Software e descrever os principais modelos e normas de qualidade e seus respectivos métodos de avaliação utilizados.*

*A Seção 2.1 apresenta uma introdução ao processo de software; A Seção 2.2 apresenta uma introdução do processo de avaliação de software; A seção 2.3 apresenta algumas definições de conceitos relacionados ao tema e relata os principais modelos e normas de referência a processos de software; A Seção 2.4 apresenta os métodos de avaliação referente aos modelos e normas descritos na seção 2.3; Por fim, a Seção 2.5 apresenta as considerações finais sobre o capítulo.*

## 2.1 Processo de Software

O processo de software é definido por [Humphrey, 1989] como o conjunto de tarefas de engenharia de software necessárias para transformar os requisitos dos usuários em software. Segundo [Sommerville, 2000], um processo de software é um conjunto de atividades e resultados associados que geram um produto de software. Além das atividades a serem realizadas e seus resultados associados, na definição de um processo de software também devem ser consideradas as seguintes informações: atividades a serem realizadas, recursos utilizados, artefatos consumidos e gerados, procedimentos adotados, paradigma e tecnologia adotados, e o modelo de ciclo de vida utilizado [Falbo, 1998].

Uma **atividade** do processo é um passo que produz mudanças de estado visíveis externamente no produto de software. Uma atividade possui ainda recursos a serem alocados (por exemplo, máquinas e orçamento), é escalonada, monitorada e atribuída a desenvolvedores, que podem utilizar ferramentas para executá-la [Reis, 2003]. Um **agente** está relacionado com as atividades de um processo, e pode ser uma pessoa ou uma ferramenta automatizada [Oliveira, 2005]. Um **artefato** é um produto criado ou modificado por uma atividade, durante um processo, podendo ser reutilizado pela mesma ou por outra, para fabricação de novos produtos. Desta forma, uma atividade pode consumir artefatos (de entrada), e gerar novos artefatos (de saída) [Oliveira, 2005].

Um processo de software explora os seguintes conceitos [Fuggetta, 2000]:

- Tecnologia de desenvolvimento de software: suporte tecnológico usado no processo. Para realizar as atividades de desenvolvimento de software são necessárias ferramentas, infra-estrutura e ambientes;
- Métodos e técnicas de desenvolvimento de software: diretrizes para uso de tecnologia e realização das atividades de desenvolvimento de software;

- Comportamento organizacional: o conhecimento sistematizado das organizações e pessoas;
- Mercado e economia: como qualquer outro produto, o software deve atender às necessidades reais dos clientes em uma situação de mercado específica.

Segundo Acuña e Ferré [Acuña, 2001], um modelo de processo de software é uma representação abstrata da arquitetura, projeto ou definição do processo de software. As entidades descritas acima, bem como seus relacionamentos e restrições formam os componentes básicos que definem um modelo de processo de software.

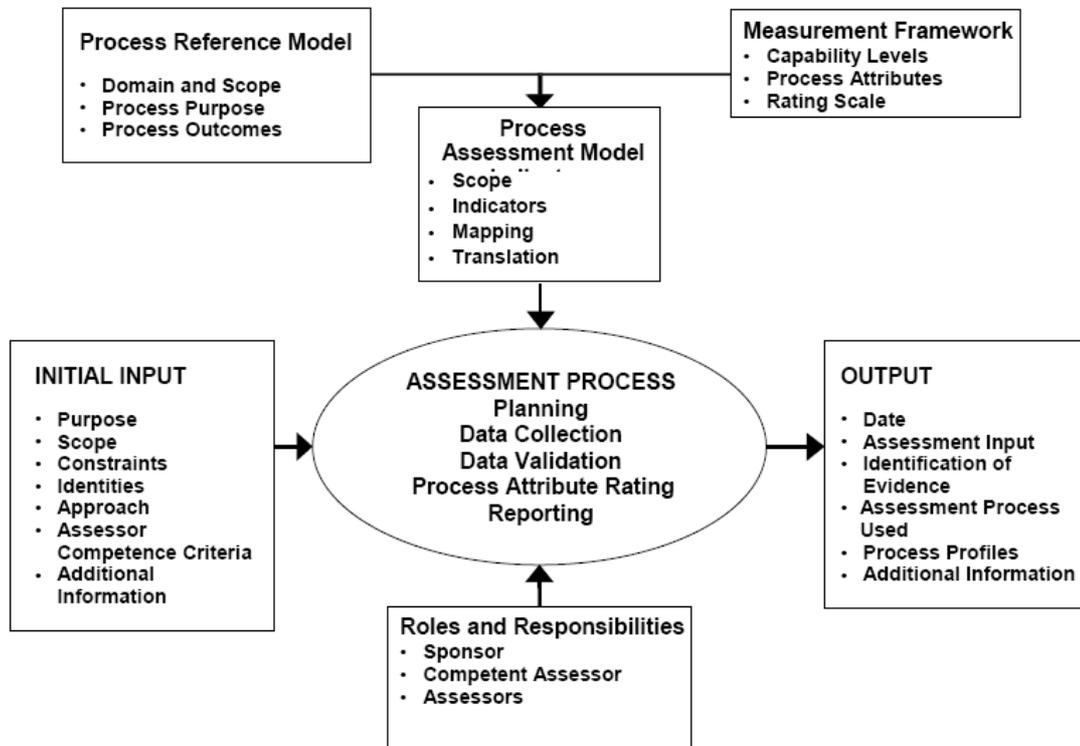
## 2.2 Avaliação de Processo de Software

Modelos de Processos definem os passos do processo de desenvolvimento e contêm uma representação abstrata da arquitetura, desenho ou definição de um processo, onde estão descritos, em diversos níveis de detalhe, os elementos de um processo [MCChesney, 1995]. Logo, através desses modelos é possível definir um processo de software para uma determinada organização, respeitando sempre as características da mesma.

Para saber se um processo foi adequadamente definido é preciso avaliá-lo. Uma avaliação de processo é uma avaliação disciplinada de processos usados por uma organização contra o modelo de avaliação de processo compatível com o(s) modelo(s) de referência com o objetivo de determinar a capacidade desses processos ou a maturidade da unidade organizacional para funcionar dentro das metas de qualidade, custo e cronograma e encontrar pontos fortes e fracos do processo de modo que ele possa ser constantemente melhorado [WANGENHEIM, 2005].

Um modelo de avaliação de processo forma a base para a coleta de evidência e pontuação da capacidade de processo e/ou maturidade da unidade organizacional, baseado em um modelo de referência de processo e um framework de medição. O processo de avaliação se baseia no modelo de avaliação de processo para gerar as saídas necessárias. Esse processo de

avaliação é conduzido por uma equipe de avaliadores, os quais possuem determinados papéis e responsabilidades [ISO/IEC 15504-1, 2004]. Esse fluxo pode ser visto na figura 2.1.



**Figura 2.1 – Principais elementos de um processo de avaliação [ISO/IEC 15504-1, 2004]**

O processo de avaliação pode ser feita baseada em dois tipos de representações: Avaliações de processo para representações contínuas e avaliações de processo para representações em estágio [WANGENHEIM, 2005].

Avaliações de processo para representações contínuas são baseadas em um modelo de avaliação de processo bidimensional composto de uma dimensão de processo e de uma dimensão de capacidade. Na dimensão de processo, essas representações descrevem um conjunto de entidades de processo relacionadas ao(s) modelo(s) de referência de processo, descrevendo processos de ciclo de vida, baseado em princípios de boas práticas na engenharia e gerência de processo. Na dimensão de capacidade, o modelo de avaliação de processo descreve capacidades relacionadas aos níveis de capacidade de processo, indicando a habilidade de um processo em alcançar ou contribuir para um

objetivo requerido baseado em uma estrutura de medição definida. A estrutura de medição provê um esquema para caracterizar a capacidade de um processo implementado a respeito ao modelo de avaliação de processo. O modelo de avaliação de processo é baseado em um conjunto de indicadores usado para demonstrar quanto do nível de capacidade foi atingido em relação ao modelo de avaliação de processo [ISO/IEC 15504-1, 2004].

Avaliações de processo para representações em estágio são baseadas em um modelo de avaliação de processo que organiza as áreas de processo em níveis de maturidade. Cada nível de maturidade engloba implicitamente aspectos relacionados que são específicos tanto às áreas de processo quanto da capacidade.

Para realizar uma avaliação, um método de avaliação específico precisa ser seguido, o qual descreve o processo de avaliação. De acordo com a [ISO/IEC 15504-1, 2004], o processo de avaliação contém pelo menos cinco atividades especificadas: desenvolver e documentar um plano para a avaliação, coletar os dados requeridos dentro do escopo da avaliação, analisar e validar os dados coletados, atribuir uma nota aos atributos de processo baseado nos dados validados e documentar e reportar os resultados da avaliação. Como resultado de um processo de avaliação, são determinadas informações e dados que caracterizam o processo avaliado, assim como a extensão da capacidade dos processos (ou maturidade da uma unidade organizacional).

Na seção 2.3, serão descritos os principais modelos de referência para processos de software e na seção 2.4 os seus respectivos métodos de avaliação.

## **2.3 Modelos e Normas de Qualidade de Software**

Nos últimos anos, diversas normas e modelos de referência têm sido desenvolvidos com o objetivo de definir, avaliar e melhorar a qualidade dos processos de software. Com o advento de padrões internacionais e a maior preocupação com a qualidade de software, as organizações passaram a adotar as referidas normas e modelos na definição de processos de software, buscando, assim, melhorar a qualidade de seus produtos, aumentar a produtividade de suas

equipes, e reduzir os custos e os riscos associados com o desenvolvimento de sistemas.

Normas de Qualidade surgiram não apenas da necessidade de se padronizar os processos de desenvolvimento de software, mas também pelos inúmeros prejuízos que as empresas estavam amargando devido ao fracasso de seus projetos de desenvolvimento de sistemas [STANDISH, 1995].

De acordo com a IEEE [IEEE, 2004], um Modelo de Referência de Processos de Software apresenta o que é reconhecido como boas práticas no desenvolvimento de produtos de software. Estas práticas podem estar somente relacionadas a atividades técnicas de engenharia de software, ou ainda referir também a atividades de gerência, engenharia de sistemas e gerência de recursos humanos.

A idéia principal destes modelos é definir um conjunto de melhores práticas de Engenharia de Software que devem ser aplicadas de forma sistemática em projetos de desenvolvimento, com objetivo de se atingir um padrão estabelecido previamente de qualidade de software. Estes modelos e normas de qualidade são utilizados como recurso fundamental para a melhoria contínua de processos de software em organizações. Não define como o processo deve ser implementado, mas prescreve suas características estruturais e semânticas em termos de objetivos e do grau de qualidade com que o trabalho deva ser realizado.

Um modelo de processo de software pode ser definido através de níveis de maturidade e/ou capacidade. O conceito de maturidade e capacidade de software surgiram inicialmente em 1986, quando o Instituto de Engenharia de Software (SEI) com assistência da Mitre Corporation começou a desenvolver uma estrutura de maturidade do processo que ajudaria organizações e melhorar seus processos de software [Humphrey, 1987a].

Segundo Paulk *et al.* [Paulk ET AL, 1995], capacidade do processo de software é o intervalo de resultados esperados que podem ser alcançados com a aplicação do processo de software. A capacidade do processo de software de uma organização fornece um meio de se prever os resultados mais prováveis a serem esperados no próximo projeto a ser empreendido pela organização.

Desempenho do processo de software representa os resultados reais alcançados seguindo-se o processo de software. Dessa forma, o desempenho do processo de software foca nos resultados alcançados, enquanto que a capacidade do mesmo foca nos resultados esperados [Paulk ET AL., 1995].

Maturidade do processo de software é a extensão para a qual um processo específico é explicitamente definido, gerenciado, medido, controlado e efetivado. A maturidade representa o potencial de crescimento de capacidade e indica a riqueza do processo de software da organização e a consistência com que o mesmo é aplicado em todos os seus projetos [Paulk ET AL., 1995].

Nesta seção serão apresentados os modelos de referência de processos de software da ISO/IEC 12207 [ISO/IEC 12207, 1995], da ISO/IEC 15504 [ISO/IEC 15504-3, 2004], do CMMI [CMU/SEI, 2002b] e ainda o modelo de referência do MR-MPS [MPS.BR, 2007a].

### **2.3.1 ISO/IEC 12207**

A ISO/IEC 12207 [ISO/IEC 12207, 1995] é uma norma de referência, e não de certificação, definida pela International Organization for Standardization (ISO) [ISSO, 2009] em conjunto com o International Electrotechnical Commission (IEC) [IEC, 2009].

O objetivo principal dessa norma é estabelecer uma estrutura comum para os processos de ciclo de vida de software visando ajudar as organizações a compreenderem todos os componentes presentes na aquisição e fornecimento de software e, assim, conseguirem firmar contratos e executarem projetos de forma mais eficaz [ISO/IEC 12207, 1995].

A primeira versão da ISO/IEC 12207 [ISO/IEC 12207, 1995] foi publicada internacionalmente em 1995. Três anos depois foi publicada nacionalmente pela ABNT [ABNT, 2009] – Associação Brasileira de Normas Técnicas. Em 2002, ela sofreu uma atualização em forma de emenda [ISO/IEC 12207:1995/Amd 1, 2002]. Segundo Pimentel [Pimentel, 2003], essa atualização foi feita para acrescentar as evoluções da engenharia de software, as necessidades encontradas pelos seus usuários e para se ajustar à norma ISO/IEC 15504 [ISO/IEC 15504-1, 2004]. Em 2004, foi publicada a segunda emenda

[ISO/IEC 12207:1995/Amd 2, 2002] da norma. O trabalho de mapeamento realizado utilizou a versão publicada em 1998 pela ABNT [ABNT, 2009] e as duas emendas.

A norma apresenta uma definição abrangente em relação aos processos, e orienta a adaptação para sua utilização nos projetos de software implementados numa organização. Ela também estabelece processos, atividades e tarefas que podem ser aplicadas na aquisição, no fornecimento, no desenvolvimento, na operação e na manutenção de software. Cada processo é composto por atividades, e cada uma destas possui um grupo de tarefas associadas. [ISO/IEC 12207, 1995]

A ISO/IEC 12207 [ISO/IEC 12207, 1995] apresenta as seguintes limitações:

- Embora descreva a arquitetura dos processos de ciclo de vida de software, a Norma não especifica detalhes de como implementar ou executar as atividades e tarefas incluídas nos processos;
- O usuário da Norma é responsável por definir o formato e o conteúdo da documentação a ser produzida;
- A Norma não define um modelo de ciclo de vida a ser utilizado, nem um método de desenvolvimento de software a ser aplicado. Tal escolha será baseada nas características do projeto e da equipe envolvida. Após a seleção de um modelo de ciclo de vida, a Norma recomenda que seja realizado um mapeamento dos processos, atividades e tarefas para o modelo.

Os processos estão reunidos em três grupos: processos fundamentais, processos de apoio e processos organizacionais, como pode ser visto na figura 2.2.

<b>Processos Fundamentais</b>	<b>Processos de Apoio</b>
Aquisição Fornecimento Desenvolvimento Operação Manutenção	Documentação Gerência de Configuração Garantia da Qualidade Verificação Validação Revisão Conjunta Auditoria Resolução de Problemas
<b>Processos Organizacionais</b>	
Gerência Infra-estrutura Melhoria Treinamento	

**Figura 2.2 - Processos da norma ISO/IEC 12207.**

O primeiro grupo engloba os processos essenciais que regulam o relacionamento adquirente-fornecedor, como também define as ações do desenvolvedor, do operador e do mantenedor de software. Os processos de apoio fornecem suporte aos outros processos, contribuindo para o sucesso e a qualidade do projeto de software. Os processos do último grupo são utilizados por uma organização para definir uma estrutura de processos de ciclo de vida e pessoal associados, que devem ser continuamente melhorados [ISO/IEC 12207, 1995].

### **2.3.2 ISO/IEC 15504**

A grande frustração por parte dos usuários de softwares com relação aos produtos entregues e o surgimento, nos anos 80, de iniciativas americanas e inglesas, com o objetivo de melhorar o processo de seleção dos seus fornecedores de software, possibilitou uma grande proliferação de modelos de processos, como o SW-CMM [CMU/SEIc, 1993], Bootstrap [KUVAJA, 1994] e

Trillium [FRANÇOIS, 1994], entre outros. Cada um desses modelos com suas características particulares e seus métodos de avaliação.

Em 1993, a ISO [ISO, 2009] realizou um estudo sobre as necessidades e os requisitos para avaliação de processos de software e concluiu que havia um consenso sobre a necessidade de um padrão universal único para avaliação de processos de software com o objetivo de harmonizar os modelos já existentes e servir como referência para criação de modelos de avaliação de processos. Essa necessidade não se dava apenas pela facilidade de se trabalhar com um método amplamente aceito, como também, pela possibilidade de se comparar resultados.

Outro objetivo do projeto foi produzir um relatório Técnico que fosse mais geral e abrangente que os modelos existentes e mais específico que a norma ISO 9001 [SALVIANO, 2001].

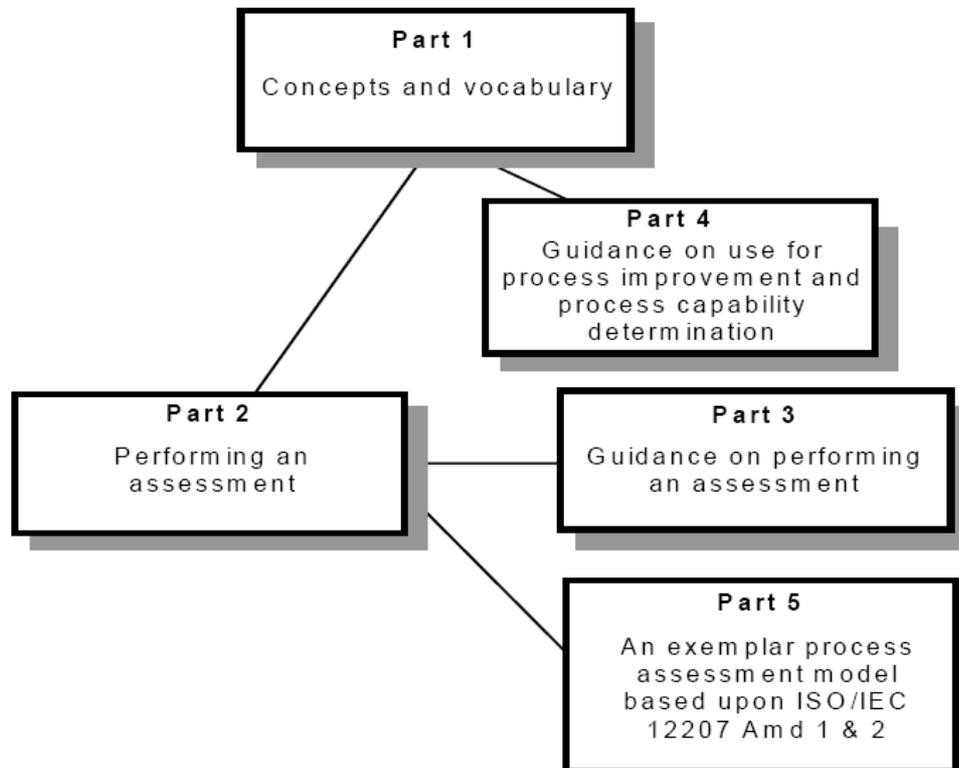
O resultado disso foi a criação do projeto SPICE (Software Process Improvement Capability dEtermination) cujo objetivo era desenvolver uma Norma Internacional para Avaliação de Processos de Software e realizar “trials” antes da publicação como norma [ROUT, 1995]. Daí, surgiu a norma ISO 15504 [ISO/IEC 15504-1, 2004], oficialmente publicada pela ISO em 2003.

A ISO 15504 é um padrão para avaliação de processo. Ela é um *framework* para construção de modelos de avaliação e melhoria para processos de software [ISO/IEC 15504-1, 2004].

Essa norma pode ser usada em dois contextos: melhoria contínua, identificando oportunidades de melhoria no processo da própria organização através dos resultados da avaliação e determinação da capacidade, identificando o nível de capacidade dos processos dos fornecedores da organização.

A ISO/IEC 15504 foi dividida em seis partes: Parte 1, Possui conceitos e vocabulário (informativo) [ISO/IEC 15504-1, 2004]; Parte 2, Realização de uma Avaliação (normativo). Informa como usar os componentes normativos [ISO/IEC 15504-2, 2004]; Parte 3, Guia para Realização de uma Avaliação (informativo). Guia para aplicação da parte 2 [ISO/IEC 15504-3, 2004]; Parte 4, Guia para Utilização dos Resultados de uma Avaliação (informativo). Provê um guia detalhado sobre o processo de melhoria e determinação da capacidade

[ISO/IEC 15504-4, 2004]; Parte 5, Um Exemplo de Modelo de Avaliação de Processo (informativo) compatível com a ISO/IEC 15504 [ISO/IEC 15504-5, 2006]. A figura 2.3 mostra o relacionamento dessas partes.



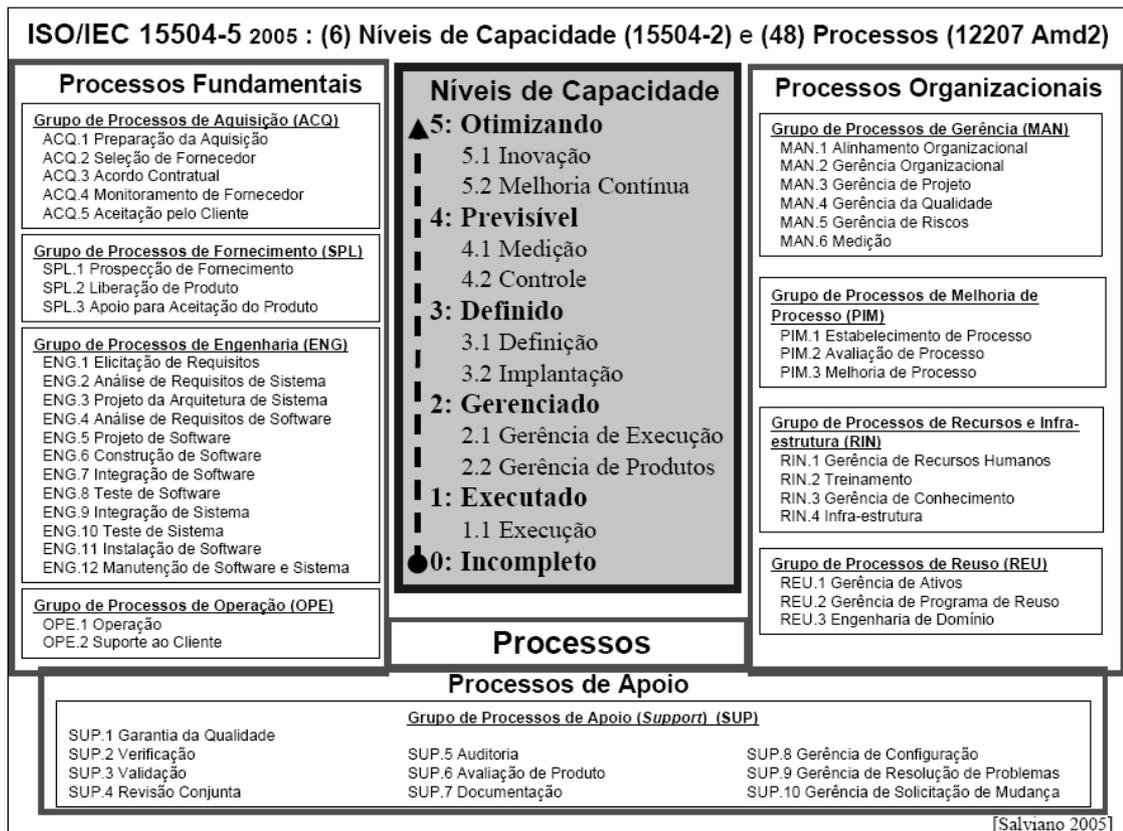
**Figura 2.3 - Componentes da ISO/IEC 1550**

A norma ISO/IEC 15504 é baseada em um ou mais modelos externos de referência de processo. A parte normativa da ISO/IEC 15504 define apenas os requisitos para o modelo de referência de processo. Por exemplo, para o domínio de engenharia de software, a parte 5 [ISO/IEC 15504-5, 2006] da norma ISO/IEC 15504 contém um modelo exemplo de avaliação de processo baseado no modelo de referência de processo definido na ISO/IEC 12207 Emenda 1/2 [ISO/IEC 12207:1995/Amd 1, 2002].

Esse modelo de referência define a dimensão de processo, que corresponde à definição de um conjunto de processos considerados universais e fundamentais para a boa prática da engenharia de software e a dimensão de capacidade, que corresponde à definição de um modelo de medição com base na identificação de um conjunto de atributos que permite determinar a capacidade

de um processo para atingir seus propósitos, gerando os produtos de trabalho e os resultados estabelecidos [Moreau, 2003].

O modelo de referência de processo utilizado na parte 5 [ISO/IEC 15504-5, 2006] possui quarenta e oito processos, agrupados em três categorias (apoio, fundamentais e organizacionais). Esses processos e seus respectivos agrupamentos podem ser vistos na figura 2.4.



**Figura 2.4 – Estrutura do modelo de referência utilizado na parte 5 da ISO/IEC 15504 [ISO/IEC 15504-5, 2006]**

As descrições dos processos contém o propósito do processo, o qual descreve os objetivos gerais para executar o processo, junto com um conjunto de resultados que demonstram o alcance bem-sucedido do propósito de processo. Um modelo de avaliação de processo se baseia sobre modelo(s) de referência de processo, adicionando a definição e uso de indicadores.

Um indicador de avaliação de processo é um atributo ou característica objetiva de uma prática ou produto de trabalho que suporta o julgamento do desempenho ou capacidade de um processo implementado. Indicadores são ou

indicadores de desempenho de processo ou indicadores de atributos de processo [ISO/IEC 15504-2, 2004].

Indicadores de desempenho de processo são usados para medir o grau de atendimento do atributo de desempenho de processo para o processo avaliado. Indicadores de desempenho de processo são práticas base e produtos de trabalho. A prática base é uma atividade que deve atingir os resultados básicos que refletem o propósito do processo. Produtos de trabalho, os quais são identificáveis e usáveis para atingir o propósito do processo, são produzidos pela execução de um processo. Evidências do desempenho das práticas base e da presença de produtos de trabalho com suas características esperadas, provêm evidência objetiva do alcance do propósito dos processos de software.

A norma ISO/IEC 15504 [ISO/IEC 15504-2, 2004] define 6 níveis de capacidade de processos seqüenciais e cumulativos. Os níveis podem ser usados para avaliar como uma organização está realizando um determinado processo ou como um guia para a melhoria de processos.

Segundo o modelo de referência da norma [ISO/IEC 15504-3, 2004], os níveis são definidos da seguinte forma:

- Nível 0 (Processo Incompleto) - Há uma falha geral na satisfação do propósito do processo. Produtos de trabalho ou resultados de processos são poucos ou difíceis de serem identificados.
- Nível 1 (Processo Executado) - O propósito do processo é geralmente alcançado, porém de uma forma não planejada e acompanhada. Existem produtos de trabalho e eles evidenciam a satisfação do propósito do processo.
- Nível 2 (Processo Gerenciado) - O processo gera produtos de trabalho de acordo com procedimentos planejados e acompanhados. A execução do processo passa a construir produtos de trabalho que satisfazem os requisitos de qualidade especificados, dentro do cronograma de tempo e dos recursos necessários.
- Nível 3 (Processo Estabelecido) - O processo utiliza um processo padrão que é capaz de atingir seus resultados definidos. A

implantação de um processo usa uma versão customizada e aprovada de um processo padrão.

- Nível 4 (Processo Previsível) - O processo passa a ser executado consistentemente dentro de limites definidos para atingir seus resultados. A qualidade dos produtos é conhecida de forma quantitativa.
- Nível 5 (Processo Otimizando) - O processo definido e padrão é alterado e adaptado para atingir de forma efetiva os objetivos correntes e futuros do negócio. A otimização contínua do processo envolve experiências de idéias e tecnologias inovativas.

Na escala de avaliação dos atributos de processo definida pela norma ISO/IEC 15504 [ISO/IEC 15504-3, 2004] existem quatro resultados possíveis: totalmente, parcialmente, largamente ou não atingido.

Cada nível contém um ou mais atributos de processo associados. O processo só atinge um determinado nível de capacidade se os resultados esperados dos seus atributos de processo correspondentes forem totalmente ou largamente atingidos. Fora isso, todos os atributos de processo dos níveis inferiores devem ter seus resultados avaliados como totalmente atingidos.

Conforme pode ser visto na figura 2.5, segue um digrama de classes o qual representa o modelo conceitual cobrindo objetos de modelo e seus atributos e tipos [WANGENHEIM, 2005].

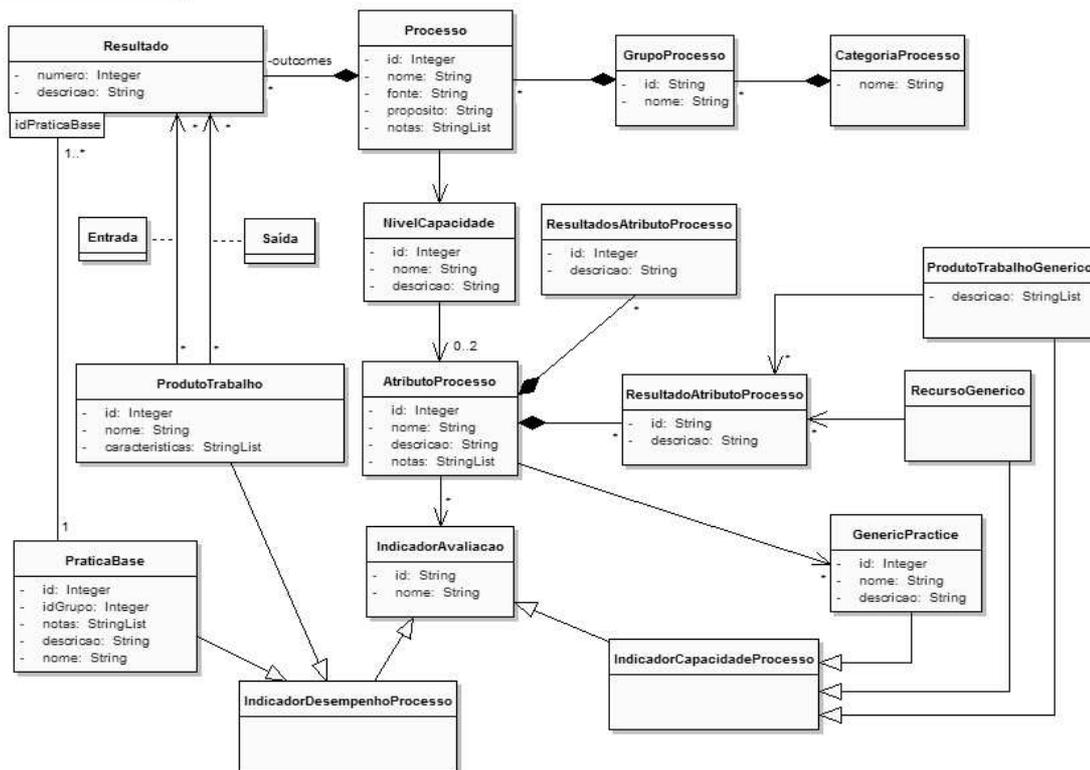


Figura 2.5- Diagrama de classes da ISO/IEC 15504 [WANGENHEIM, 2005]

### 2.3.3 CMMI – Capability Maturity Model Integration

Desde 1991, diversos modelos CMM [SEI, 1995] vinham sendo desenvolvidos para abordar diversas disciplinas, como: engenharia de sistemas, engenharia de software, gerenciamento e desenvolvimento de recursos humanos, etc. Dentre os modelos CMM [SEI, 1995], os seguintes eram os mais notáveis e foram utilizados como base para a criação do CMMI [CMU/SEI, 2006]:

- The Capability Maturity Model for Software (SW-CMM) [SEI, 1997b], foco na avaliação da maturidade da organização em relação a processos de engenharia de software;
- Systems Engineering CMM (SE-CMM) [EIA, 1998], foco na avaliação da maturidade da organização em relação a processos de engenharia de sistemas;
- Integrated Product Development CMM (IPD-CMM) [SEI, 1997a], com abrangência maior que o SE-CMM, inclui também outros

processos necessários à produção e suporte ao produto, tais como suporte ao usuário, processos de fabricação, etc.

- People CMM (P-CMM) [CURTIS, 2002] , utilizado para avaliar a maturidade da organização em seus processos de administração de recursos humanos.

Apesar dos modelos do CMM [SEI, 1995] terem sido bastante úteis para diversas organizações, o uso de múltiplos modelos causou bastante problema. Várias organizações visam concentrar seus esforços de melhoria entre várias disciplinas. Porém, as diferenças entre os modelos específicos de cada uma delas, em termos de arquitetura, conteúdo e abordagem limitaram a facilidade com que as organizações pudessem alcançar um processo de melhoria bem sucedido. Além disso, a aplicação de múltiplos modelos, que não estão integrados, aumenta o custo para a organização em termos de treinamento, avaliação e atividades de melhoria [CMU/SEI, 2006].

A partir daí, surgiu a necessidade de um modelo que resolvesse o problema em relação ao uso de diversos modelos CMM [SEI, 1995]. Dessa maneira, o CMMI [CMU/SEI, 2006] integra os diversos modelos CMM [SEI, 1995] numa estrutura única, todos com a mesma terminologia, processos de avaliação e estrutura. Também teve como objetivo, a integração das práticas, de forma que, organizações que almejem melhorar seus processos nas diferentes disciplinas, tenham a disposição um único modelo consistente.

O projeto também se preocupou em tornar o CMM [SEI, 1995] compatível com a norma ISO/IEC 15504 [ISO/IEC 15504-3, 2004], de modo que as avaliações em um modelo fossem reconhecidas como equivalentes aos do outro [CMU/SEI, 2002].

O CMMI [CMU/SEI, 2006] é um modelo de referência para avaliação e melhoria da maturidade dos processos de uma organização. Ele fornece às organizações de TI um guia de como obter controle em seus processos para desenvolver e manter seus sistemas, e como evoluir em direção a uma cultura de excelência em gestão. Além disso, ele direciona as organizações a selecionar uma estratégia de melhoria, determinar a maturidade atual do processo e

identificar as questões mais críticas para a qualidade e melhoria de processos [CMU/SEIb, 2002].

Sendo um modelo de referência de processos, o CMMI [CMU/SEI, 2006] não define como o processo deve ser implementado, mas prescreve suas características estruturais e semânticas em termos de objetivos e do grau de qualidade com que o trabalho deva ser realizado.

O modelo CMMI [CMU/SEI, 2006] oferece 2 (duas) abordagens de avaliação: por estágios (como o antigo SW-CMM [SEI, 1997b]) e contínua (baseada na ISO/IEC 15504) [ISO/IEC 15504-3, 2004]. Estas abordagens podem ser visualizadas na figura 2.6.

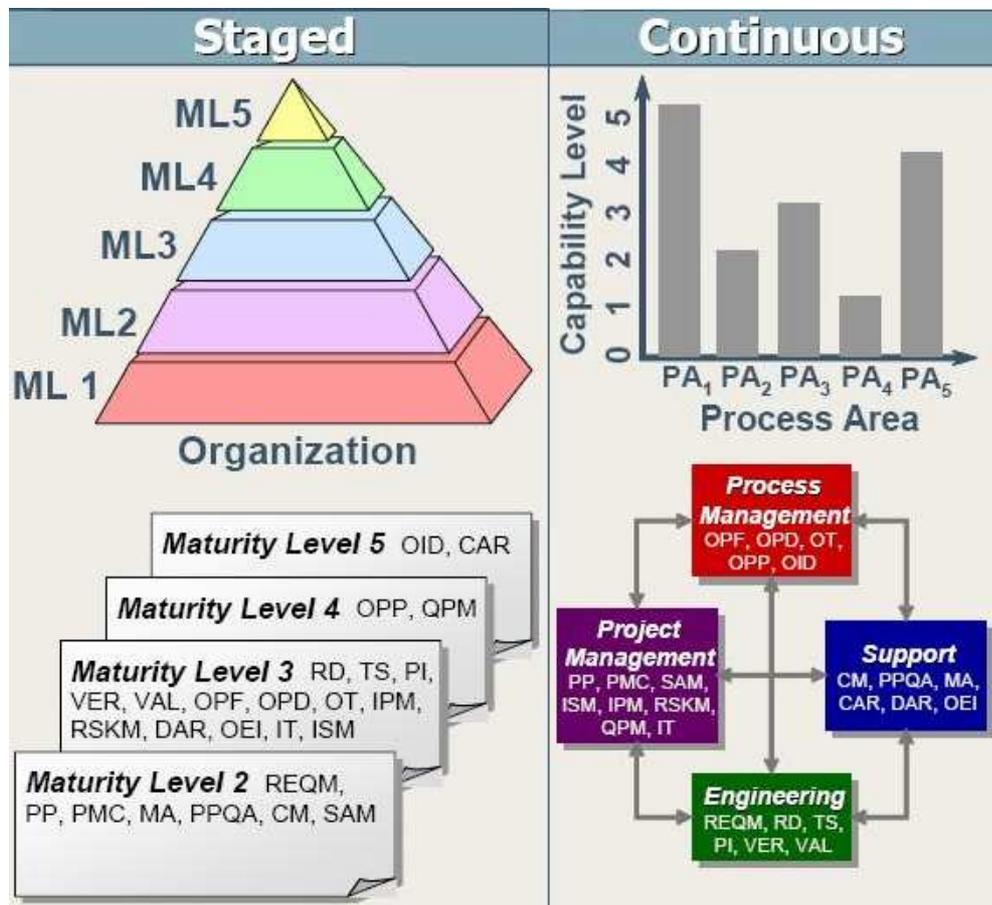


Figura 2.6 – Abordagem estagiada e contínua do modelo CMMI [CMU/SEI, 2006].

A abordagem em estágios oferece uma seqüência comprovada de melhorias, começando com práticas básicas de gerenciamento e progredindo por um caminho pré-definido e comprovado de níveis sucessivos, cada um servindo

como base para o próximo, além de oferecer uma classificação única que resume os resultados de avaliações e permite comparação entre organizações.

Cada nível de maturidade possui um conjunto de áreas de processo que indicam qual deve ser o foco da organização para a melhoria do processo organizacional [CMU/SEI, 2006]. Alcançando cada nível assegura-se que uma base de melhoria adequada foi estabelecida para o próximo nível, minimizando o investimento da organização num processo de melhoria e maximizando os benefícios trazidos para a organização.

Essa abordagem possui cinco níveis, conforme descrito abaixo em ordem crescente de maturidade [CMU/SEI, 2006]:

- Nível 1 (Executado) - Associado a um processo caótico e ao fato da organização não definir uma estrutura estável de apoio aos processos. Sucesso dependente de iniciativas individuais e os resultados normalmente são entregues com orçamentos e calendários estourados. [CHRISISS, 2004]
- Nível 2 (Gerenciado) – Habilidade de gerenciar os requisitos e planejar, executar, medir e controlar seus processos. Além disso, os artefatos produzidos satisfazem à descrição de processos, padrões e procedimentos. [CHRISISS, 2004]
- Nível 3 (Definido) – Entendimento de forma mais clara sobre o relacionamento entre os processos e possibilidade de gerenciá-los de forma pró-ativa. Os processos estão descritos em padrões, procedimentos, ferramentas e métodos de forma mais rigorosa. As funções e responsabilidades são bem entendidas.
- Nível 4 (Quantitativamente gerenciado) - Estatística e outras técnicas quantitativas são utilizadas. O resultado da aplicação delas serve para analisar se os objetivos de desempenho de qualidade e de processo, definidos pela organização e seus projetos, estão sendo atingidos. Esses objetivos estão de acordo principalmente com as necessidades dos clientes, da organização e daqueles que implementam o processo. [CHRISISS, 2004]

- Nível 5 (Otimizado) - Objetivos quantitativos de melhoria de processo são definidos e a organização atinge o estágio de melhoria contínua através de avanços incrementais e inovadores em tecnologia e processos. A organização se preocupa em resolver causas de variação do processo e mudá-lo para atingir os objetivos. [CHRISISS, 2004]

Dentre as vantagens de se utilizar a abordagem estagiada, pode-se citar: permite a comparação de forma direta entre as organizações; provê um caminho de melhoria bem definido para as organizações; estrutura familiar para os que estão migrando do SW-CMM [SEI, 1997b]; possui diversos estudos de caso e dados que comprovam o seu ROI [CMU/SEI, 2006].

Já a abordagem contínua oferece uma abordagem flexível para a melhoria do processo. Ela é destinada a organizações que desejam melhorar uma área de processo particular ou um conjunto de processos, com base nos aspectos críticos da organização ou processos que estão ligados às metas de negócio da organização. Esta abordagem possibilita comparações dentro e entre organizações em termos de áreas de processo [CMU/SEIb, 2002].

Assim como o modelo por estágios, a representação continua também possui áreas de processo e praticas. No entanto as praticas de uma área de processo estão organizadas de maneira que permite o crescimento e melhoria individual de cada área de processo. As praticas genéricas estão agrupadas em níveis de capacidade. Os níveis de capacidade de todas as áreas de processo em conjunto determinam a melhoria da organização.

A estrutura contínua possui seis níveis de capacidade, conforme descrito abaixo em ordem crescente de capacidade [CMU/SEI, 2006]:

- Nível 0 (Incompleto) - Um processo incompleto significa um processo não executado ou parcialmente executado. Ou seja, uma ou mais metas específicas da área de processo não foram satisfeitas.
- Nível 1 (Executado) - Um processo no nível 1 de capacidade é considerado um processo executado. Isto significa que todas as

metas específicas da área de processo foram satisfeitas, conforme está descrito na meta genérica deste nível.

- Nível 2 (Gerenciado) - Caracteriza um processo gerenciado. Ou seja, o processo que além de executado é planejado e controlado de acordo com uma política estabelecida pela organização.
- Nível 3 (Definido) - É caracterizado como um processo definido. Um processo definido é um processo gerenciado que também foi adequado ao conjunto de processos padrões da organização de acordo com as diretrizes definidas pela organização para adequação de processos.
- Nível 4 (Quantitativamente gerenciado) - Caracteriza o processo como “quantitativamente gerenciado”, que é considerado um processo definido e controlado através de técnicas de estatística ou outras técnicas quantitativas.
- Nível 5 (Otimizado) - Um processo otimizado é um processo quantitativamente gerenciado que está sempre mudando e se adaptando para atender objetivos de negócios correntes. Um processo otimizado foca na melhoria contínua da execução do processo através melhorias tecnológicas inovadoras e incrementais.

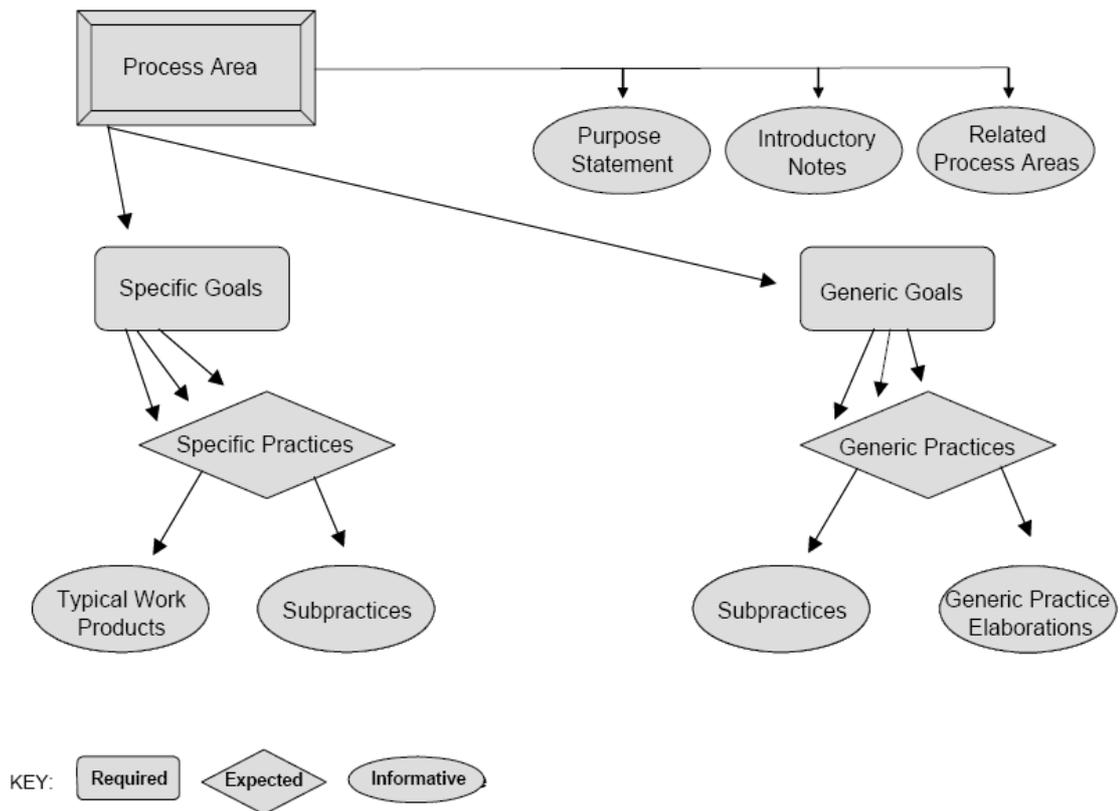
Dentre as vantagens de se utilizar a abordagem contínua, pode-se citar: fornece maior flexibilidade focando em áreas de processo específicas de acordo com metas e objetivos de negócio; permite a comparação de áreas de processo iguais entre diferentes organizações; foco bem definido nos riscos específicos de cada área de processo; estrutura compatível com o padrão ISO/IEC 15504.

A figura 2.7 representa um mapeamento entre a representação contínua e a estagiada. Essa equivalência possibilita à organização obter seu nível de maturidade a partir de uma avaliação na representação contínua ou obter seu nível de capacidade a partir de uma representação estagiada.

Model	Process Area	Acronym	ML	CL1	CL2	CL3	CL4	CL5
CMMI 1.1	Project Planning	PP	2	Orange	Yellow			
CMMI 1.1	Project Monitoring and Control	PMC	2					
CMMI 1.1	Supplier Agreement Management	SAM	2					
CMMI 1.1	Requirements Management	REQM	2					
CMMI 1.1	Configuration Management	CM	2					
CMMI 1.1	Process and Product Quality Assurance	PPQA	2					
CMMI 1.1	Measurement and Analysis	MA	2					
CMMI 1.1	Organizational Training	OT	3	Yellow				
CMMI 1.1	Organizational Process Focus	OPF	3					
CMMI 1.1	Organizational Process Definition	OPD	3					
CMMI 1.1	Integrated Project Management (IPPD)	IPM	3					
CMMI 1.1	Risk Management	RSKM	3					
CMMI 1.1	Integrated Teaming	IT	3					
CMMI 1.1	Integrated Supplier Management	ISM	3					
CMMI 1.1	Requirements Development	RD	3					
CMMI 1.1	Technical Solution	TTS	3					
CMMI 1.1	Product Integration	PI	3					
CMMI 1.1	Verification	VER	3					
CMMI 1.1	Validation	VAL	3					
CMMI 1.1	Decision Analysis and Resolution	DAR	3					
CMMI 1.1	Organizational Environment for Integration	OEI	3					
CMMI 1.1	Organizational Process Performance	OPP	4					Green
CMMI 1.1	Quantitative Project Management	QPM	4					
CMMI 1.1	Organizational Innovation and Deployment	OID	5	Blue				
CMMI 1.1	Causal Analysis and Resolution	CAR	5					

**Figura 2.7 - Equivalência entre a representação contínua e estagiada do CMMI**

Os elementos do modelo são divididos em três categorias. Os componentes necessários são os componentes que devem ser alcançados por uma organização que possui um processo planejado e implementado (Ex: metas genéricas e específicas) [CMU/SEI, 2006]. Os componentes esperados são os componentes que descrevem o que uma organização normalmente implementa para alcançar um determinado componente necessário (Ex: Práticas específicas e genéricas) [CMU/SEI, 2006]. Já os informativos (Sub-práticas, produtos típicos de trabalho, elaborações de práticas genéricas, título e notas de práticas e metas, referências, etc.) são considerados componentes informativos, os quais ajudam a entender metas e práticas e como alcançá-las. A figura 2.8 ilustra os relacionamentos desses componentes [CMU/SEI, 2006].



**Figura 2.8 - Componentes do CMMI [CMU/SEI, 2006]**

Uma área de processo é agrupamento de práticas relacionadas que, quando implementadas coletivamente satisfazem um conjunto de objetivos considerados importantes para realizar uma melhoria significativa nesta área [Chrissis, 2004].

Os objetivos específicos são obrigatórios e descrevem uma única característica que deve estar presente para que uma área de processo esteja satisfeita. Objetivos genéricos, também obrigatórios, são objetivos que estão distribuídos em várias áreas de processo e descrevem características que devem estar presentes para institucionalizar os processos que implementam determinada área de processo [Chrissis, 2004].

As práticas genéricas e específicas, não são obrigatórias, mas desejáveis, e descrevem uma atividade considerada importante para atingir determinado objetivo. As sub-práticas fornecem uma descrição detalhada para interpretar e implementar uma prática. As elaborações de práticas genéricas oferecem um tutorial de como as práticas genéricas deveriam ser aplicadas a uma área de

processo. Produtos típicos de trabalho são apenas informativos, e fornecem uma lista de possíveis resultados de uma prática [Chrissis, 2004]. Amplificações de disciplina provêm a orientação para interpretar a informação do modelo em disciplinas específicas (por exemplo, IPPD, engenharia de sistemas ou engenharia de software). Cada área de processo possui ainda a descrição de seu propósito, notas introdutórias e áreas de processo relacionadas, os quais podem ser identificadas como itens informativos.

A modelagem da estrutura do modelo de avaliação de processo do CMMI é apresentada na figura 2.9 [WANGENHEIM, 2005], onde um digrama de classe mostra o modelo conceitual cobrindo os objetos de modelo com seus atributos e tipos. Este diagrama cobre ambas representações CMMI, por estágios e contínua.

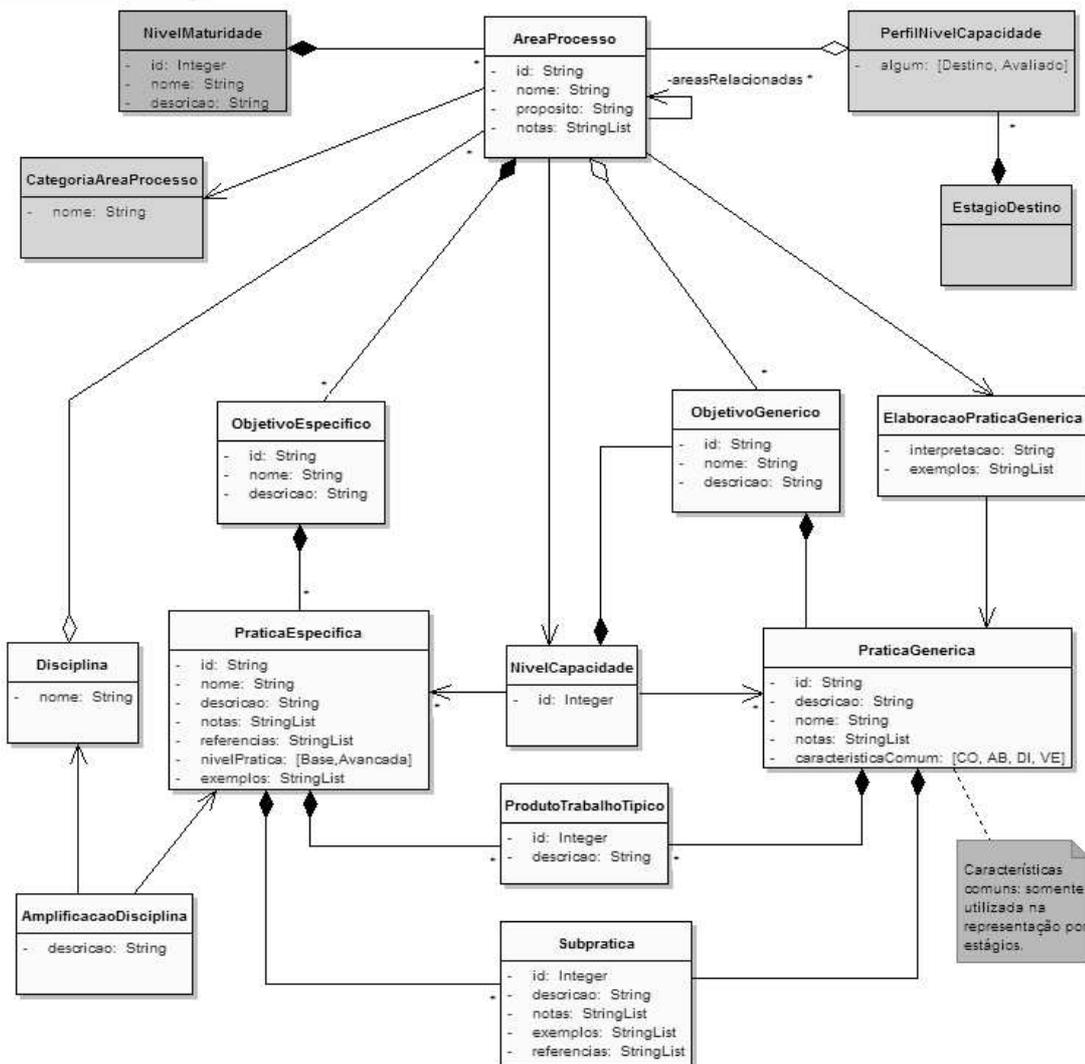


Figura 2.9 - Diagrama de classes do CMMI [WANGENHEIM, 2005]

### **2.3.4 MPS.BR - Programa de Melhoria do Processo de Software Brasileiro**

Segundo [MBI, 2007], as empresas brasileiras de software quase não atuam no mercado externo. O maior desafio dessas empresas está em desenvolver uma estratégia clara e criar as condições para crescer em mercados internacionais nos quais o nível de exigência é maior. Para alcançar esse objetivo de exportar software com qualidade as empresas precisam aumentar a qualidade dos seus processos.

No entanto, existe uma grande dificuldade das empresas de softwares brasileiras com relação a quantidade de recursos financeiros disponíveis para definição, gerenciamento e implantação de melhoria contínua de seus processos.

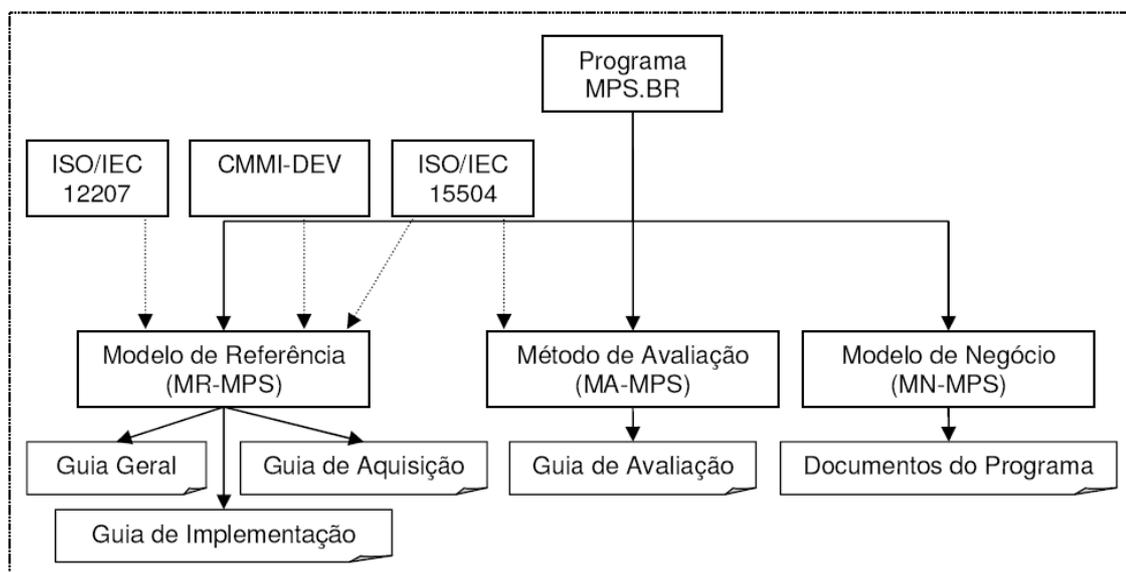
Visando resolver essa deficiência, foi criado, em dezembro de 2003, o Programa para Melhoria de Processo de Software Brasileiro, o MPS.BR. O MPS.BR surgiu a finalidade de melhorar a qualidade do software que é produzido pelas empresas brasileiras, principalmente pequenas e médias, a um custo mais acessível.

Este programa é coordenado pela Associação para Promoção da Excelência do Software Brasileiro (SOFTEX), contando com apoio do Ministério da Ciência e Tecnologia (MCT), da Financiadora de Estudos e Projetos (FINEP) e do Banco Interamericano de Desenvolvimento (BID) e ainda conta com o apoio da indústria representada COPPE/UFRJ (Programa de Engenharia de Sistemas e Computação da Universidade Federal do Rio de Janeiro), CESAR (Centro de Estudos e Sistemas Avançados de Recife), CenPRA (Centro de Pesquisas Renato Archer) e organizações integrantes do Sistema SOFTEX do Rio de Janeiro e Campinas.

O MPS.BR é compatível com os padrões de qualidade aceitos internacionalmente e tem como pressuposto o aproveitamento de toda competência existente nos padrões e modelos de melhoria de processo já disponíveis. Ele é baseado nas abordagens internacionais para definição, avaliação e melhoria de processos de software. A base técnica para a construção e aprimoramento deste modelo de melhoria e avaliação de processo de software é composta pelas normas NBR ISO/IEC 12207 – Processo de Ciclo de Vida de

Software [NBR ISO/IEC 12207, 1998], pelas emendas 1 [ISO/IEC 12207:1995/Amd 1, 2002] e 2 [ISO/IEC 12207:1995/Amd 2, 2004] da norma internacional ISO/IEC 12207 [ISO/IEC 12207, 1995] e pela ISO/IEC 15504 – Avaliação de Processo. A base técnica do MPS.BR é composta pelas seguintes normas: ISO/IEC 12207 [ISO/IEC 12207, 1995] e suas emendas 1 e 2, ISO/IEC 15504 – SPICE e o CMMI-SE/SW [CMU/SEIa, 2002].

O programa é constituído de 3 (três) componentes: a) Modelo de Referência (MR-MPS) [MPS.BR, 2007a], b) Método de Avaliação (MA-MPS) [MPS.BR, 2007b] e o c) Modelo de Negócio (MN-MPS) [MPS.BR, 2007c], conforme pode ser visualizado na **Erro! Fonte de referência não encontrada.**:



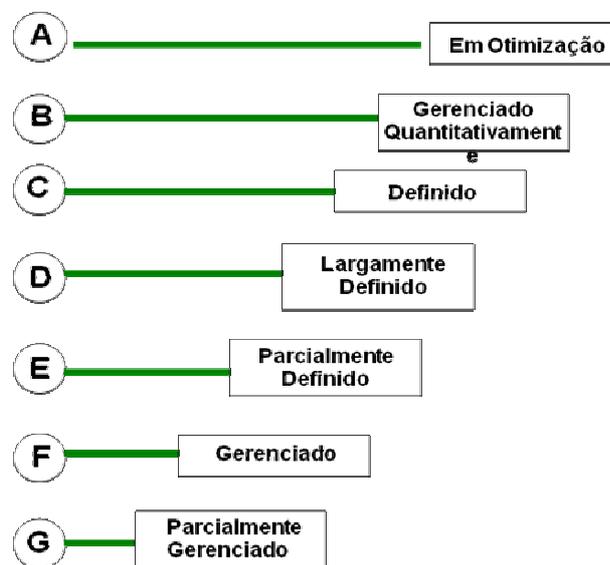
**Figura 2.10 - Componentes do programa MPS.BR [MPS.BR, 2007a]**

O Modelo de Referência (MR-MPS) [MPS.BR, 2007a] representa o modelo ideal para uma organização, isto é, ele indica quais são as metas a que uma organização deve visar para atingir um certo grau de qualidade. O MR-MPS está em conformidade com os requisitos de modelos de referência de processo da norma ISO/IEC 15504-2 [ISO/IEC 15504-2, 2003].

Já o Método de Avaliação (MA-MPS) [MPS.BR, 2007b], institucionaliza a forma como se deve proceder a avaliação de uma empresa, verificando assim se a mesma está de acordo com as normas especificadas pelo MR-MPS. O processo e o método de avaliação MA-MPS estão em conformidade com a norma ISO/IEC 15504-2 [ISO/IEC 15504-2, 2003].

O Modelo de Negócio (MN-MPS) [MPS.BR, 2007c] descreve as regras de negócio para implementação do MR-MPS e para avaliação seguindo o MA-MPS; a forma de organização de grupos de empresas para implementação do MR-MPS e avaliação MA-MPS; as regras de certificação para consultores de aquisição, e os programas anuais de treinamento por meio de cursos, provas e workshops MPS.BR.

O MR-MPS [MPS.BR, 2007a] segue a abordagem em estágios baseada no modelo CMMI, contudo existem mais níveis, de forma que as organizações podem ser avaliadas de forma mais gradual, diminuindo assim o custo da avaliação. Os níveis previstos pelo MPS.BR podem ser visualizados na figura 2.11:



**Figura 2.11 – Níveis de maturidade do MR-MPS [MPS.BR, 2007a]**

A escala de maturidade começa no nível G e atinge o ápice de maturidade no nível A. Cada nível possui seus processos e seus atributos de processos necessários, conforme pode ser verificado na figura 2.12. É importante observar que os atributos dos processos são cumulativos em relação aos níveis, isto é, um nível de maturidade mais elevado possui os seus atributos necessários e acumula todos os atributos dos níveis mais baixos, ou seja, para que uma organização consiga obter o nível F, por exemplo, necessita primeiro obter o nível G.

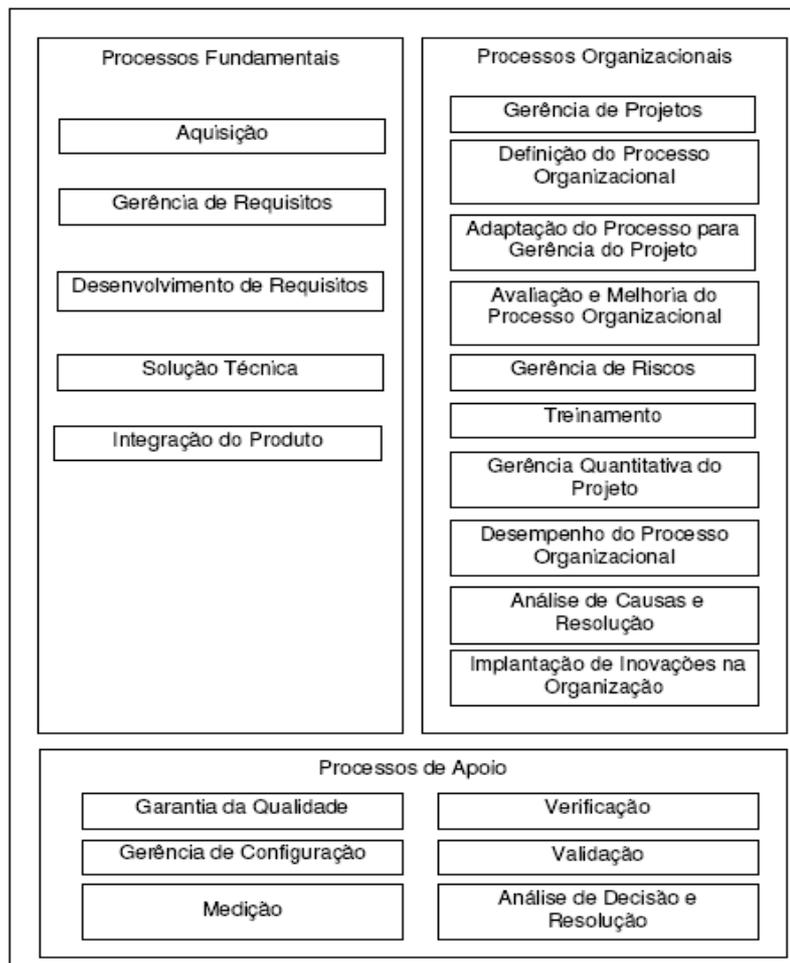
Nível	Processos	Atributos de Processo
<b>A</b>	Análise de Causas de Problemas e Resolução – ACP	AP 1.1, AP 2.1, AP 2.2, AP 3.1, AP3.2, AP 4.1, AP 4.2 , AP 5.1 e AP 5.2
<b>B</b>	Gerência de Projetos – GPR (evolução)	AP 1.1, AP 2.1, AP 2.2, AP 3.1 e AP3.2, AP 4.1 e AP 4.2
<b>C</b>	Gerência de Riscos – GRI	AP 1.1, AP 2.1, AP 2.2, AP 3.1 e AP3.2
	Desenvolvimento para Reutilização – DRU	
	Análise de Decisão e Resolução – ADR	
	Gerência de Reutilização – GRU (evolução)	
<b>D</b>	Verificação – VER	AP 1.1, AP 2.1, AP 2.2, AP 3.1 e AP3.2
	Validação – VAL	
	Projeto e Construção do Produto – PCP	
	Integração do Produto – ITP	
	Desenvolvimento de Requisitos – DRE	
<b>E</b>	Gerência de Projetos – GPR (evolução)	AP 1.1, AP 2.1, AP 2.2, AP 3.1 e AP3.2
	Gerência de Reutilização – GRU	
	Gerência de Recursos Humanos – GRH	
	Definição do Processo Organizacional – DFP	
	Avaliação e Melhoria do Processo Organizacional – AMP	
<b>F</b>	Medição – MED	AP 1.1, AP 2.1 e AP 2.2
	Garantia da Qualidade – GQA	
	Gerência de Configuração – GCO	
	Aquisição – AQU	
<b>G</b>	Gerência de Requisitos – GRE	AP 1.1 e AP 2.1
	Gerência de Projetos – GPR	

**Figura 2.12 - Estrutura do MPS.BR**

Os processos são descritos em termos de propósito, resultados e informações adicionais. O propósito descreve o objetivo geral da execução do processo. Os resultados esperados estabelecem as metas a serem atingidas com a efetiva implementação do processo [MPS.BR, 2007a]. As informações adicionais são referências que podem ajudar na definição do processo pela organização. Elas fornecem descrições de atividades, tarefas e melhores

práticas que podem apoiar a definição e implementação do processo nas organizações.

Conforme descrito pelo MR-MPS [MPS.BR, 2007a], os processos do MPS.BR são descritos e classificados de maneira semelhante à norma ISO/IEC 15504 [ISO/IEC 15504-3, 2004] e à norma ISO/IEC 12207 [ISO/IEC 12207, 1995]. Eles foram divididos em Processos Fundamentais, Processos Organizacionais e Processos de Apoio, como pode ser verificado na figura 2.13. Esse agrupamento foi feito por uma questão de organização, conforme o seu objetivo principal no ciclo de vida de software.



**Figura 2.13 - Processos do MPS.BR [MPS.BR, 2007a]**

Os Processos fundamentais atendem o início e a execução do desenvolvimento, operação ou manutenção dos produtos de software e serviços correlatos durante o ciclo de vida de software. Já os Processos de apoio auxiliam um outro processo e contribuem para o sucesso e qualidade do projeto

de software. E, finalmente, os Processos organizacionais são empregados em nível corporativo para estabelecer, implementar e melhorar um processo do ciclo de vida.

Capacidade do processo é uma caracterização da habilidade do processo atingir os objetivos de negócio atuais ou futuros [ISO/IEC 15504-1, 2004]. Ela também expressa o grau de refinamento e institucionalização com que o processo é executado na organização/unidade organizacional [MPS.BR, 2007a]. A capacidade do processo no MPS.BR [MPS.BR, 2007a] é representada por um conjunto de atributos de processo descrito em termos de resultados esperados.

À medida que a organização/unidade organizacional evolui nos níveis de maturidade, um maior nível de capacidade deve ser atingido pela organização para executar o processo, ou seja, novos atributos de processo e resultados devem ser satisfeitos conforme se alcança novos níveis de maturidade.

O atendimento aos atributos do processo (AP), pelo atendimento aos resultados esperados dos atributos do processo (RAP) é requerido para todos os processos no nível correspondente ao nível de maturidade, embora eles não sejam detalhados dentro de cada processo.

A capacidade de cada processo no MPS.BR possui nove atributos de processos (AP). Cada atributo está detalhado em termos de resultados esperados do atributo de processo (RAP) para alcance completo do atributo de processo. O MR-MPS [MPS.BR, 2007a] define os atributos de processo como: AP 1.1 – O processo é executado; AP 2.1 – O processo é gerenciado; AP 2.2 – Os produtos de trabalho do processo são gerenciados; AP 3.1 – O processo é definido; AP 3.2 – O processo está implementado; AP 4.1 – O processo é medido; AP 4.2 – O processo é controlado; AP 5.1 – O processo é objeto de inovações; AP 5.2 – O processo é otimizado continuamente.

O diagrama de classes do MR-MPS é apresentada na figura 2.14 [WANGENHEIM, 2005], onde mostra o modelo conceitual cobrindo os objetos de modelo com seus atributos e tipos. Este diagrama é mais simples que aqueles apresentados para a norma 15504 e para o modelo CMMI porque não apresenta as informações adicionais para a implementação (onde são indicados outros modelos).

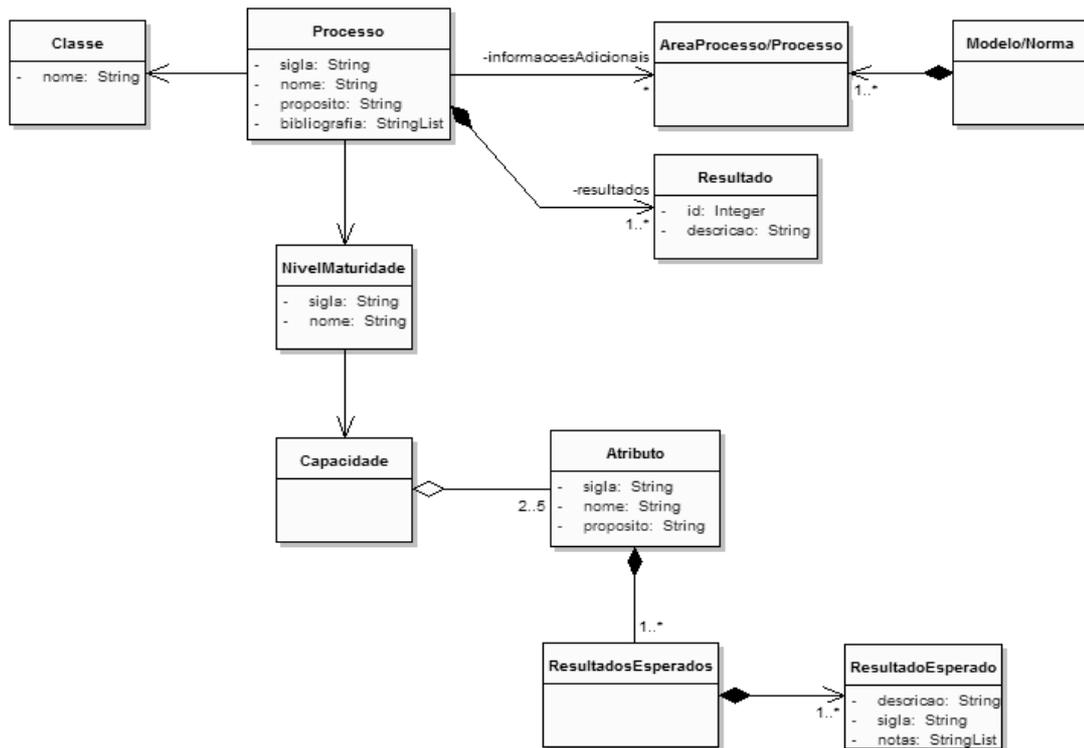


Figura 2.14 - Diagrama de classes do MPS.BR [WANGENHEIM, 2005]

## 2.4 Métodos de Avaliação de processos de software

Um modelo de avaliação de processo forma a base para a coleta de evidência e pontuação da capacidade de processo e/ou maturidade da unidade organizacional, baseado em um modelo de referência de processo e um framework de medição.

De acordo com a [ISO/IEC 15504-3, 2004], para realizar uma avaliação de processo de software, um método de avaliação específico precisa ser seguido, o qual descreve o processo de avaliação.

Nessa seção serão descritos os métodos de avaliação dos modelos de referência descritos anteriormente. Serão analisados o método de avaliação da ISO/IEC 15504, o SCAMPI relativo ao CMMI e o MA-MPS relativo ao MPS.BR.

## 2.4.1 ISO/IEC 15504

Os processos do modelo de referência da ISO/IEC 15504 são descritos e classificados de maneira semelhante à norma [ISO/IEC 12207, 1995]. A norma internacional ISO/IEC 12207 tem como objetivo principal estabelecer uma estrutura comum para os processos de ciclo de vida de software visando ajudar as organizações a compreenderem todos os componentes presentes na aquisição e fornecimento de software e, assim, conseguirem firmar contratos e executarem projetos de forma mais eficaz.

Essa norma define um conjunto de requisitos para um Modelo de Avaliação e para um Método de Avaliação. Uma avaliação que esteja de acordo com estes requisitos é referenciada como uma avaliação em conformidade com a avaliação ISO/IEC 15504 (2003). Ela não define um método de avaliação explícito, definindo apenas os requisitos necessários. Isto significa que as empresas podem desenvolver os seus próprios métodos de avaliação em conformidade com a ISO/IEC 15504 (2003).

O objetivo da norma ISO/IEC 15504 com relação à avaliação de processos é definir um conjunto de requisitos genéricos para realização de uma avaliação, os quais são agrupados dentro das seguintes atividades: desenvolver e documentar um plano para a avaliação, coletar os dados requeridos dentro do escopo da avaliação, analisar e validar os dados coletados, atribuir uma nota aos atributos de processo baseado nos dados validados e documentar e reportar os resultados da avaliação. Como resultado de um processo de avaliação, são determinadas informações e dados que caracterizam o processo avaliado, assim como a extensão da capacidade dos processos (ou maturidade da uma unidade organizacional). Esse ciclo pode ser visto na figura 2.15.



**Figura 2.15 - Ciclo para avaliação de processo na ISO/IEC 15504**

Além da definição dos requisitos genéricos, a norma considera três tipos de elementos como importantes para sua realização: um modelo de avaliação;

um método de avaliação; um ou mais avaliadores competentes. Ela também estabelece as responsabilidades do patrocinador e do auditor líder da avaliação e ainda todas as entradas necessárias para realização da avaliação tais como: modelo de avaliação que será utilizado, escopo da avaliação, restrições para realização da avaliação, a abordagem, identidade e papéis da equipe de avaliação e alguma informação adicional que seja relevante para realização da avaliação.

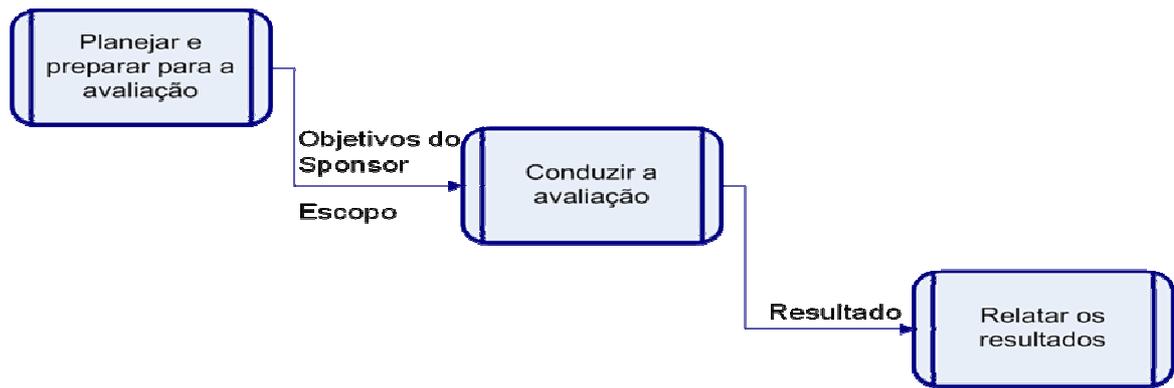
#### **2.4.2 SCAMPI – Standard CMMI Appraisal Method for Process Improvement**

O SCAMPI é o método oficial do SEI para avaliar organizações que implantaram o modelo de referência CMMI. O seu objetivo é identificar as forças e fraquezas dos processos revelando riscos de desenvolvimento/aquisição e determinando a capacidade e o nível de maturidade [CMU/SEIa, 2002].

É representado pelo conjunto de documentos: a) ARC – Appraisal Requirements for CMMI [CMU/SEIc, 2001], b) MDD – Method Definition Document [CMU/SEId, 2001] e c) Guidance for Government Source Selection and Contract Process Monitoring [CMU/SEIe, 2002].

O ARC define os requisitos básicos e critérios para os métodos baseados nos modelos CMMI, além de distinguir as 3 classes de avaliação: A,B e C. O MDD define o detalhamento dos requisitos do ARC através das atividades e práticas associadas a cada processo do método SCAMPI. E o Guidance for Government Source Selection and Contract Process Monitoring fornece um guia para aquisição focado em governo.

De acordo com o MDD, os processos de avaliação consistem em: planejar e preparar para a avaliação, conduzir a avaliação e relatar os resultados. Estes processos podem ser visualizados na **Erro! Fonte de referência não encontrada.6:**



**Figura 2.16 - Processos de avaliação do MDD [CMU/SEId, 2001]**

#### **2.4.2.1 Detalhamento das etapas:**

Para o planejamento e preparação, as seguintes atividades devem ser realizadas:

- Identificar o escopo da avaliação – onde ocorre o levantamento das necessidades de negócio da unidade organizacional sendo avaliada;
- Desenvolver o plano da avaliação – onde ficam registrados os requisitos do plano de avaliação, acordos, estimativas, riscos, métodos de adaptação e considerações práticas associadas à avaliação;
- Selecionar e preparar a equipe de avaliação – uma equipe treinada, experiente e apropriadamente qualificada é selecionada para conduzir o processo de avaliação;
- Obter e analisar as evidências iniciais - obtém informações que identifiquem áreas potencialmente problemáticas ou falhas na implementação das práticas;
- Preparar para a coleta de evidências – consiste em planejar e documentar a coleta de dados incluindo as fontes de dados, ferramentas e técnicas a serem usadas e contingências para gerenciar o risco da falta de dados.

Para conduzir uma avaliação no local de trabalho, as seguintes atividades devem ser realizadas:

- Examinar as evidências – que compreende coletar as informações a respeito das práticas implementadas na unidade organizacional e relacionar os dados coletados ao modelo de referência;
- Verificar e validar as evidências – consiste em verificar a implementação das práticas nas unidades organizacionais para cada instanciação e validar os resultados da implementação descrevendo as falhas na implementação das práticas do modelo;
- Documentar as evidências – registra as informações obtidas identificando e consolidando os dados e transformando-os em registros que documentem a implementação das práticas, assim como suas forças e fraquezas;
- Gerar os resultados da apresentação - Mede a satisfação dos objetivos baseado na extensão da implementação da prática através da unidade organizacional. A extensão da implementação da prática é determinada baseada nos dados validados, coletados de toda a amostra das unidades organizacionais. A medida do nível de capacidade ou nível de maturidade é guiada algoritmicamente pela medida de satisfação do objetivo.

Quanto à apresentação dos resultados, as seguintes atividades são realizadas:

- Apresentar os resultados da avaliação – Provê resultados da avaliação que podem ser usados para guiar ações de melhoria. As forças e as fraquezas dos processos em uso também são apresentadas. Além disso, determina, se planejado, qual o nível de capacidade ou o nível de maturidade dos processos em uso.
- Empacotar e arquivar os resultados da avaliação – guarda registros e dados importantes da avaliação e disponibiliza o material selecionado de maneira apropriada.

#### **2.4.2.2 Classes de Avaliação**

De acordo com o ARC (), o CMMI pode ser avaliado por 3 classes de avaliação, que se baseiam nas diferenças entre a confidencialidade dos dados, o custo e a longevidade da avaliação e quanto a geração de resultados:

Classe A: satisfazem todos os requisitos estabelecidos pelo ARC. O método SCAMPI representa a classe A e precisa estar aderente à ISO/IEC 15504. Dois tipos de evidência objetiva tem que ser apresentadas neste tipo de avaliação e esta classe apresenta graduação quanto a maturidade.

Classe B: satisfazem um subconjunto de requisitos estabelecidos pelo ARC. Também como na classe acima, é muito importante a constatação de duas evidências objetivas. Mas não fornece graduação quanto a maturidade e não é obrigatoriamente aderente à ISO/IEC 15540. Esta classe de avaliação é recomendada para organizações que estão iniciando as atividades de melhoria de processo aderentes ao CMMI.

Classe C: contempla um subconjunto de requisitos estabelecidos pela classe B. Apenas uma evidência objetiva é necessária. Este tipo de avaliação pode ser utilizado quando deseja-se um diagnostico rápido sobre os processos de uma organização.

Outras diferenças entre as classes de avaliação podem ser visualizadas na tabela abaixo:

Características	Classe A	Classe B	Classe C
Quantidade relativa de evidências de objetivos coletadas.	Alta	Média	Baixa
Geração de Graduação	Sim	Não	Não
Quantidade relativa de recursos necessários.	Alta	Média	Baixa
Dimensão relativa da equipe.	Grande	Média	Pequena
Requisitos para liderar a avaliação.	Requer líder de avaliação.	Requer líder de avaliação ou pessoa treinada e experiente.	Requer uma pessoa treinada e experiente.

### 2.4.3 MA-MPS

O MA-MPS [MPS.BRb, 2007] é o método de avaliação do programa MPS.BR, o qual visa verificar a maturidade da organização na execução de seu processo de software.

De acordo com o MA-MPS [MPS.BRb, 2007], os resultados da execução de uma avaliação MPS são: obtenção de dados e informações que caracterizam os processos de software da organização, determinação do grau em que os resultados esperados são alcançados e os processos atingem seu propósito e atribuição de um nível de maturidade do MR-MPS [MPS.BRa, 2007] para a organização.

O MA-MPS [MPS.BRb, 2007] estabelece os processos e atividades para realização da avaliação no programa MPS. Os processos de avaliação são: contratar a avaliação, preparar para a realização da avaliação, realizar a avaliação e documentar os resultados da avaliação. Estes estão detalhados nas atividades apresentadas na **Erro! Fonte de referência não encontrada.7**



Figura 2.17 - Processos MA-MPS [MPS.BRb, 2007]

O processo “Contratar Avaliação” tem como resultado final a celebração de um contrato entre a organização a ser auditada (contratante) e a instituição avaliadora (IA).

Após a assinatura do contrato é iniciado o processo “Preparar a Realização da Avaliação” que consiste nas atividades de planejamento da avaliação, preparação da documentação inicial por parte da organização, condução de avaliação inicial em conjunto com a organização. Nessa atividade, planilhas são preenchidas com evidências, posteriormente utilizadas para indicar a implementação dos processos. A partir dessa planilha, é executada a avaliação inicial, onde são informadas as falhas no processo que deverão ser ajustadas até a avaliação final.

O processo de “Realizar a Avaliação” tem como principais propósitos o treinamento da equipe, a condução da avaliação e a comunicação de seus resultados a organização avaliada. Durante a condução da avaliação são analisadas as evidências diretas e indiretas, realizadas as entrevistas e atribuídos os conceitos aos resultados esperados dos processos. A partir disso o nível organizacional é obtido e reportado a organização.

A última etapa é o processo “Documentar os resultados” que finaliza o ciclo da avaliação com o relatório final para posterior divulgação do Softex.

## **2.5 Considerações finais:**

Nas últimas décadas, várias abordagens associadas à melhoria de processo têm ganhado importância na comunidade de software. Os conceitos, métodos, e práticas englobam uma maneira de pensar, de agir e de entender os dados gerados pelos processos que, coletivamente, resultam em melhoria da qualidade, aumento da produtividade e competitividade dos produtos de software.

No entanto, em todas estas abordagens, fragilidades no processo de melhoria ainda são visíveis. Por exemplo: os modelos não estarem adequados a realidade das empresas, excesso de documentação e frustração globalizada, entre outras. Vistas as dificuldades relatadas acima, a abordagem de avaliação de processo apresentada neste trabalho pretende combinar algumas características das abordagens apresentadas neste capítulo.



# ImPPros - Um Ambiente de Implementação Progressiva de Processo de Software

*Este capítulo tem por objetivos apresentar uma visão geral de processo de software e apresentar o ambiente ImPPros – Um Ambiente de Implementação Progressiva de Processo de Software – contexto no qual a ferramenta proposta nesse trabalho está inserida.*

*O capítulo está dividido da seguinte forma: Na Seção 3.1 é apresentado o contexto no qual o ImPProS se insere; A Seção 3.2 define um ambiente de definição de processos, mostra sua evolução e descreve o ambiente ImPProS, seus objetivos, arquitetura e o seu ciclo de vida de processo; Por fim, a Seção 3.3 apresenta as considerações finais sobre o capítulo.*

## 3.1 Contextualização

Os processos de software podem apresentar grande complexidade e possibilitar diversas alternativas de execução de suas atividades. Desta forma, um processo de software definido permite que profissionais de engenharia de software possam trabalhar de forma ordenada, possibilitando um melhor entendimento do seu trabalho, bem como de outras atividades executadas por outros membros da mesma equipe [Humphrey, 1989].

Assim, durante a definição de um processo de software torna-se necessário adequar o processo a diversos fatores como: tecnologias envolvidas, tipo de software em questão, domínio de aplicação, grau de capacitação ou maturidade da equipe em engenharia de software, características próprias da organização, características do projeto e da equipe.

No entanto, não existe um processo de software que possa ser genericamente aplicado a diversos projetos, visto que nenhum projeto é idêntico ao outro. Variações nas políticas e procedimentos organizacionais, métodos e estratégias de aquisição, tamanho e complexidade do projeto, requisitos e métodos de desenvolvimento do sistema, entre outros fatores, influenciam na forma como um produto de software é adquirido, desenvolvido, operado e mantido [ISO, 1997].

Dessa maneira, melhorar a definição e posterior execução dos processos de software tornou-se um desafio e a partir destes foram criados os ambientes de implementação de processo de software automatizado, mantendo a realidade da organização, com seus domínios específicos variando de projeto a projeto, sendo possível sua simulação a partir dos parâmetros de configuração (prazo, custos, recursos, entre outros) [Oliveira, 2007].

Dentro deste contexto é que se encontram os PSEE( Process-centred Software Engineering Environment) que suportam além do desenvolvimento de software, outras funções como gerência e garantia da qualidade durante o ciclo de vida do software, sendo um apoio automatizado para execução das atividades.

## **3.2 Ambientes de Desenvolvimento de Software Orientados a Processo**

Conforme discutido anteriormente, grande parte das organizações de desenvolvimento de software possui dificuldade em entregar produtos de qualidade, dentro dos prazos e custos definidos. Tais problemas estão diretamente relacionados à baixa qualidade do processo de desenvolvimento de software, e não do produto em si [Paula, 2003].

Com o objetivo de solucionar este problema, várias tecnologias têm surgido, visando disciplinar o processo de desenvolvimento, estabelecendo etapas bem definidas, fornecendo um mecanismo de controle do processo de software. Inicialmente, a tecnologia CASE (Computer Aided Software Engineering) se destacou com a proposta de se utilizar software para auxiliar na produção de software. A partir da integração de ferramentas CASE, aliada à definição de uma estrutura unificadora de serviços, onde várias ferramentas, de diferentes métodos podem ser integradas, permitindo a comunicação e cooperação entre as mesmas, surgiram os ambientes de desenvolvimento de software (ADS) [Reis, 2000].

A evolução dos ADS continuou com a utilização de outras tecnologias, tais como inteligência artificial, sistemas distribuídos, banco de dados não convencionais, dentre outras [Oliveira, 2005]. Porém, a definição de ambientes centrados em processo representa uma evolução significativa nos ADS. A partir da automação do processo de software, incorporada aos ADS mais recentes, surgiram os ambientes de desenvolvimento de software orientados a processo, também conhecidos na literatura como PSEE (Process-Centered Software Engineering Environment) [Gimenez, 1994].

Os PSEEs surgiram como uma abordagem para disciplinar um processo de software, a partir da definição de mecanismos para controle, acompanhamento, automatização e melhoria contínua do processo. Segundo Reis [Reis, 2000], esses ambientes fornecem desde ferramentas integradas, para prover apoio ao desenvolvimento dos projetos de software, até mecanismos de implementação e controle das fases de um ciclo de vida do processo [Humphrey 1989]: definição, simulação, execução e avaliação do processo de software.

Reis, no trabalho [Reis, 2000], define as principais funções genéricas que podem ser suportadas por PSEEs:

- Engenharia de Processos: definição e manutenção dos modelos de processo, provendo facilidade nas atividades de definição, análise e simulação dos processos de software;

- Engenharia de Software: desenvolvimento e manutenção de um produto de software através do seguimento de um processo de software;
- Gerência de Projetos: coordenação e monitoramento das atividades de engenharia de software, a fim de garantir que o processo está sendo seguido.

A mudança de um processo em uma organização não é uma tarefa simples. Daí a necessidade de prover mecanismos que auxiliem a organização na implementação de seus processos de acordo com sua realidade.

### **3.2.1 O Ambiente ImPProS**

O ambiente ImPProS – Ambiente de Implementação Progressiva do Processo de Software [Oliveira, 2005] - é um PSEE de implementação progressiva, o qual tem como objetivo principal fornecer um apoio automatizado para implementação dos processos de uma organização. O termo progressiva é decorrente do fato de que a implementação do processo é aperfeiçoada com as experiências aprendidas nas suas etapas de definição, simulação, execução e avaliação.

O ImPProS é um projeto de iniciativa do Centro de Informática da UFPE – Universidade Federal de Pernambuco com a parceria da UNAMA – Universidade da Amazônia, financiado pelo CNPq – Conselho Nacional de Desenvolvimento Científico e Tecnológico, que visa a criação de um ambiente de apoio à implementação de um processo de software em uma organização de forma progressiva [Oliveira, 2005].

Segundo Oliveira [Oliveira, 2007] o ImPProS – Implementação Progressiva de Processo de Software está sendo concebido com os seguintes objetivos:

- Especificar um meta-modelo de processo de software a fim de definir uma terminologia única entre os vários modelos de qualidade de processo de software existentes, para uso do ambiente em seus serviços providos;

- Apoiar a definição de um processo de software para organização;
- Permitir a modelagem e instanciação deste processo;
- Permitir a simulação do processo a partir das características instanciadas para um projeto específico;
- Dar apoio à execução do processo de software tomando como base uma máquina de inferência;
- Possibilitar a avaliação dos critérios do processo de software;
- Apoiar a melhoria contínua do processo de software e o reuso através da realimentação e coleta das experiências aprendidas.

Vale salientar que todos os objetivos listados acima foram adaptados a partir da estrutura que compõe o meta-processo de software descrito em [Reis, 2003], das características propostas para a implementação de um processo de software [Balduino, 2002] e do ciclo de vida para melhoria contínua de processo definido pelo Modelo IDEAL [Mcfeeley, 1996].

### **3.2.1.1 Arquitetura**

Para alcançar os objetivos descritos anteriormente, o ambiente foi concebido para adotar a arquitetura apresentada na Figura 3.1. Pode-se notar que a arquitetura contempla quatro tipos de usuários para interação com o Ambiente:

- **Projetista do Processo:** responsável pela definição do processo e coleta de experiência sobre a execução de projetos. Este tipo de usuário interage com o ambiente recebendo orientações e identificando melhorias para processos existentes ou em concepção;
- **Gerente de Processo:** acompanha a simulação e a avaliação do processo a fim de prover conhecimentos formal e informal para possibilitar o reuso e a melhoria contínua dos processos;
- **Gerente de Projetos:** este usuário atua nas fases de instanciação do processo para um projeto específico, acompanhando a execução do processo e a sua avaliação para posterior coleta de experiências;

- Equipe de Desenvolvimento: agrupa todos os perfis relacionados à execução de um projeto de software (Gerentes, Analistas, Engenheiros de Software, Arquitetos, etc.).

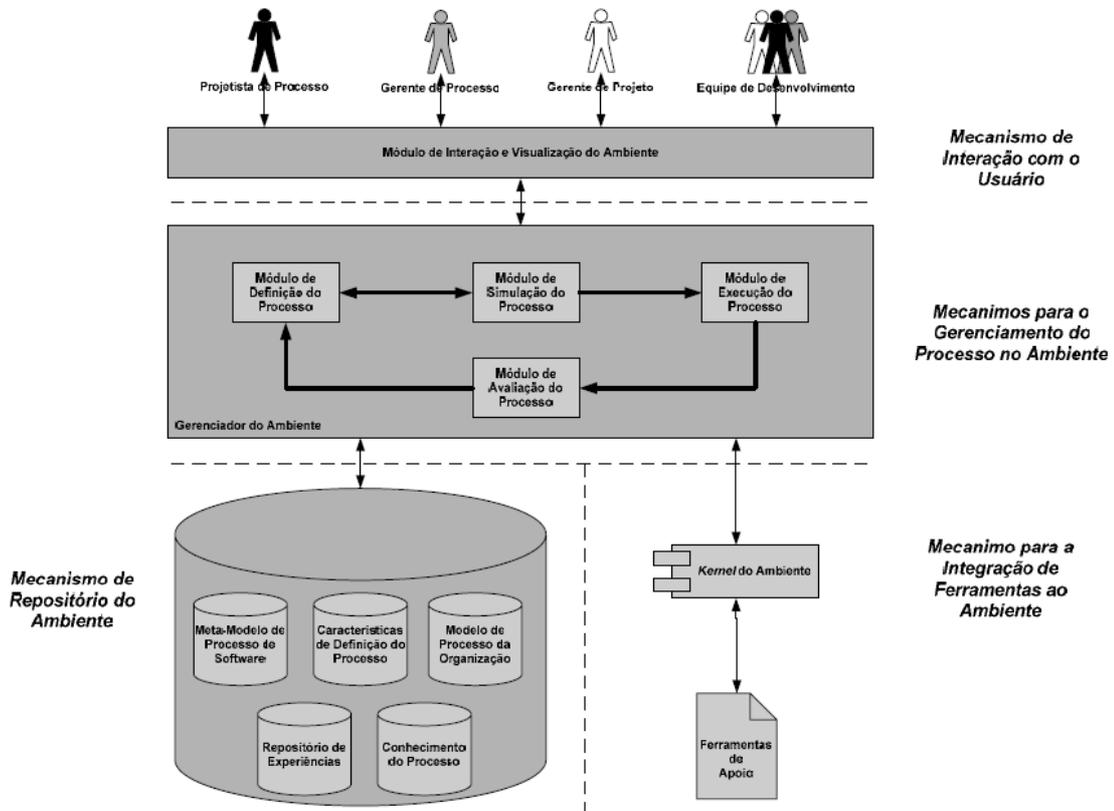


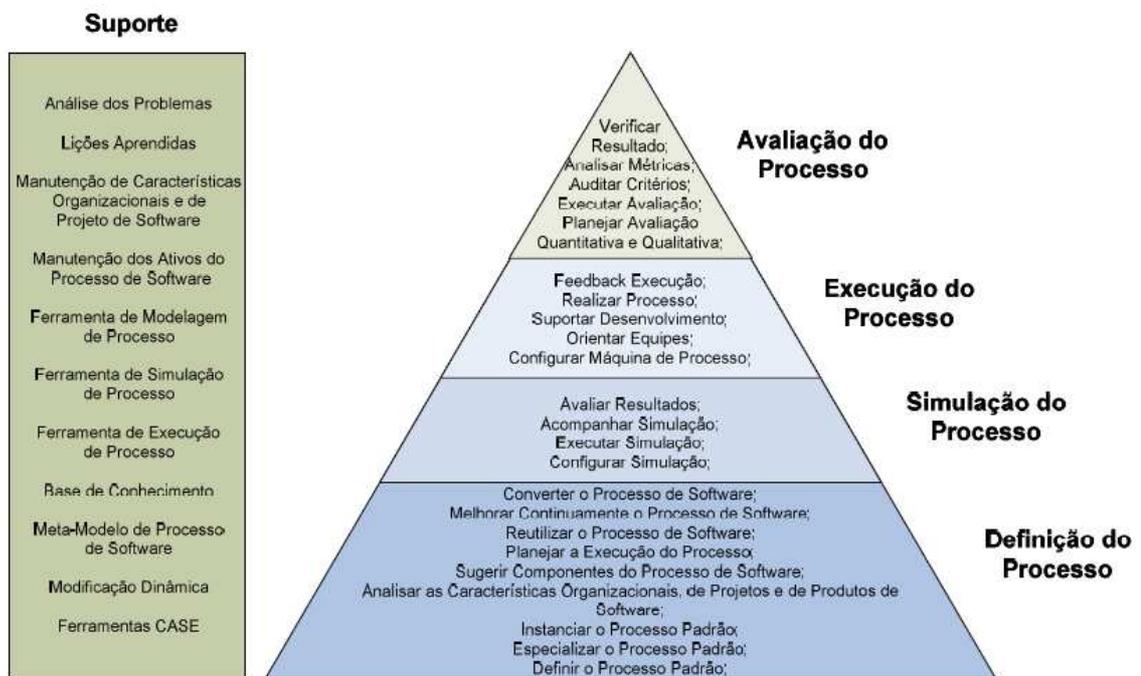
Figura 3.1 - Arquitetura do ImPPros [Oliveira, 2005]

Outro ponto importante nessa arquitetura são os mecanismos propostos por [Oliveira, 2007]: Mecanismo de interação com o usuário; Mecanismo para o gerenciamento do processo no ambiente; Mecanismo de repositório do ambiente e o Mecanismo para a integração de ferramentas ao ambiente.

O módulo de interação e visualização do ambiente consiste no mecanismo de interação com o usuário. Este tem como principal foco prover diferentes visões da mesma informação. A informação é definida e especificada, podendo ser apresentada de forma diferente perante os diversos usuários que estão acessando ao mesmo tempo o ambiente. Esta camada trabalha questões relacionadas à usabilidade do ambiente.

A camada posterior, o gerenciador do ambiente, tem como finalidade prover os principais serviços (módulos de definição, simulação, execução e

avaliação do processo de software) especificados ao ambiente de forma automatizada, ou seja, possibilitar que os usuários do ambiente executem suas funções tendo como referencial um guia. Ele é composto por quatro módulos (Módulo de Definição do Processo, Módulo de Simulação do Processo, Módulo de Execução do Processo, Módulo de Avaliação do Processo), onde alguns destes possuem suas funções executadas por ferramentas de apoio integradas ao ambiente. Nesse caso, o gerenciador do ambiente tem a finalidade de gerenciar apenas o uso destas ferramentas e em alguns casos agregar novos serviços que contemplem as funções do ambiente. A figura 3.2 apresenta o relacionamento desses módulos no ambiente ImPProS de forma Botton-up. Na próxima seção, o relacionamento entre esses módulos serão descritos mais detalhadamente.



**Figura 3.2 - Relacionamento dos módulos de Gerenciamento do ImPProS**

O mecanismo de repositório do ambiente gerencia todos os objetos a partir de bases de dados que controlam toda a execução e manutenção dos componentes de processo. Esta camada contém as bases de meta-modelo de processo de software, características de definição do processo, modelo de processo da organização, repositório de experiências e conhecimento do processo.

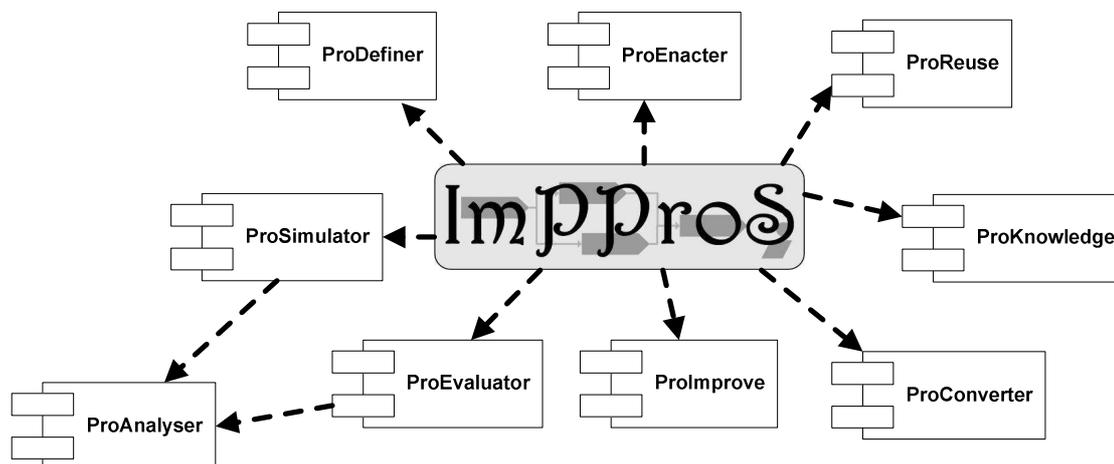
A última camada é composta pelos mecanismos de integração das ferramentas ao ambiente. Este mecanismo provê a integração do ambiente com outras ferramentas de apoio ao processo de software e à execução do projeto de software, possibilitando a automação de atividades definidas no processo e a execução de alguns módulos do ambiente.

De acordo com o que foi proposto por [Oliveira, 2007], o ambiente foi concebido para ser composto de um ambiente cooperativo, formado por nove ferramentas de apoio principais:

- ProDefiner: provê a definição do processo de software a partir da análise de características específicas e aprendizado adquirido com outras definições;
- ProSimulator: possibilita a simulação do processo de software instanciado a partir de um plano de execução do processo e assim antever problemas;
- ProEnacter: permite a execução automatizada e acompanhamento do processo de software pela equipe do projeto;
- ProEvaluator: provê a avaliação da execução do processo de software a partir da análise de critérios qualitativos e quantitativos;
- ProImprove: possibilita a execução sistemática das atividades de melhoria do processo de software, a partir do modelo IDEAL;
- ProAnalyser: permite a análise e tomada de decisão acerca da avaliação de itens que compõe o processo de software;
- ProReuse: provê a execução do reuso de processo de software a partir da definição do escopo do projeto e sua adaptação ao contexto de uso;
- ProKnowledge: possibilita a coleta, análise e uso de conhecimentos aprendidos ao longo da execução do processo de software;

- ProConverter: provê a conversão dos componentes do processo de software a partir das estruturas de normas/modelos de qualidade e seus mapeamentos.

A figura 3.3 apresenta o conjunto de ferramentas que compõem o ambiente ImPProS e o macro-relacionamento entre as mesmas.

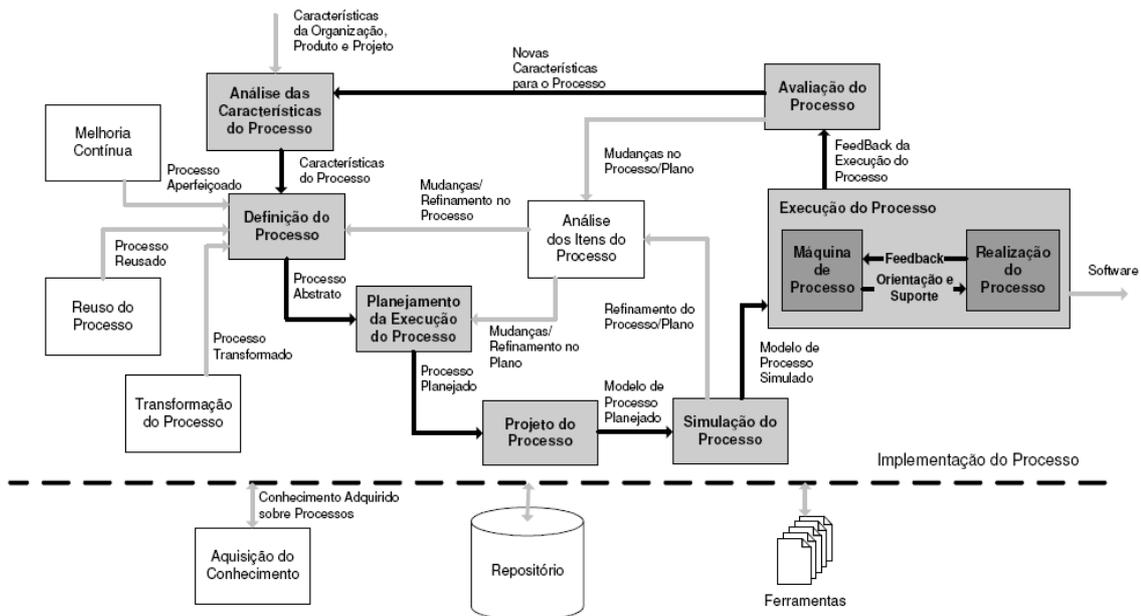


**Figura 3.3 - Ferramentas de Apoio ao Ambiente ImPProS**

### 3.2.1.2 Ciclo de Vida de Processo no Ambiente ImPProS

Um processo de software é definido segundo as necessidades iniciais da organização. Porém, uma vez definido, o processo se encontra em constante evolução, causada por mudanças planejadas ou não dentro da organização [Humphrey 1989]. Desta maneira, existe um ciclo de vida para processo de software, análogo ao ciclo de vida de um produto de software. As atividades do ciclo de vida do processo de software são chamadas meta-atividades, e o processo de desenvolvimento e evolução do processo de software é chamado metaprocesso.

O ciclo de vida de processo no ImPProS é uma adaptação do ciclo de vida proposto por [Reis.00], levando em consideração fases mais focadas ao tratamento dos processos de software no que tange aos modelos e normas da qualidade. Este ciclo de vida é apresentado na Figura 3.4, onde as caixas representam as fases do ciclo e as setas simbolizam o fluxo de informações sobre o objeto em tratamento.



**Figura 3.4 - Ciclo de Vida de Processo no ImPProS [Oliveira, 2005].**

Segundo [Oliveira, 2005], o ciclo de vida se inicia com a fase Análise das Características do Processo, a qual identifica as principais características da organização, de projetos e de produtos de software necessárias para a implementação do processo.

Posteriormente, a fase Definição do Processo guia a implementação do processo a partir da execução de tarefas de acordo com os níveis em que este processo é usado em função da análise das características. Esta fase pode ser auxiliada pelas fases de: Melhoria Contínua, provendo parâmetros de aperfeiçoamento do processo de software em definição; Reuso do Processo, fornecendo processos de software (alimentados no repositório mediante suas definições), ou partes destes, relevantes ao contexto atual; e Transformação do Processo, que recomenda a constituição de novos processos com base nos mapeamentos existentes entre as normas e modelos da qualidade em uso no ImPProS.

A fase de Planejamento da Execução do Processo recebe um processo abstrato e gera um processo planejado, ou seja, com recursos e agentes alocados, cronograma definido, métricas e estimativas inferidas e previsão inicial do custo. Esta etapa é contemplada com base nas informações obtidas

pela análise das características da organização e do projeto de software, e pelo conhecimento adquirido de processos anteriores.

A seguir, é possível que o processo passe pela fase de Simulação do Processo, onde a finalidade é identificar possíveis problemas no planejamento e/ou na definição do processo abstrato, discutido e proposto em [Souza, 2007]. Esta fase é auxiliada pela fase Análise dos Itens do Processo que identifica possíveis inconsistências no plano e/ou no processo abstrato, requerendo o retorno às fases anteriores para o refinamento dos itens produzidos. Isso é executado até que o modelo de processo planejado esteja pronto para ser executado.

A próxima fase trata da Execução do Processo, que é apoiada pela máquina de processos auxiliando na coordenação das atividades realizadas por pessoas (gerentes e equipe de desenvolvimento) e por ferramentas automatizadas, obtendo com isso um feedback da realização do processo, a partir da organização das informações constantes no modelo de processo simulado de acordo com a perspectiva de interesse na fase de execução.

Paralelamente à fase de execução ocorre a Avaliação do Processo que: analisa a necessidade de modificação dinâmica do processo, recebendo o auxílio da fase Análise de Itens do Processo a fim de identificar as mudanças no processo abstrato ou no plano, o que faz com que as fases de definição, planejamento, projeto e simulação possam ser realizadas em paralelo com a execução; e provê informações quantitativas e qualitativas descrevendo o desempenho de todo o processo em execução, e estabelecendo novas características para o processo.

Pode-se destacar, ainda, a fase de Aquisição de Conhecimento, que auxilia a análise, manutenção e disseminação das experiências na implementação do processo de software.

Algumas fases do ciclo de vida podem ser auxiliadas por Ferramentas de apoio ao processo e ao projeto de software e o Repositório contempla todas as informações do meta-modelo de processo, das características de definição do processo, dos modelos de processo das organizações, das experiências adquiridas e de todos os resultados ao longo da implementação do processo.

### **3.3 Considerações Finais**

Neste capítulo foi apresentado a ambiente de implementação progressiva de software, o ImPProS. Na primeira seção, foi discutido o contexto no qual o ImPProS está inserido. Posteriormente foi apresentado uma sucinta descrição sobre tecnologia CASE, ambiente de desenvolvimento de software (ADS), ambiente de desenvolvimento de software orientado a processo (PSEE) e como essas tecnologias evoluíram. Em seguida foi descrito o ambiente ImPProS, os seus objetivos principais, sua arquitetura e os módulos funcionais que o compõem incluindo seus componentes e as ferramentas de apoio. Por fim, um resumo sobre as fases que compõem o ciclo de vida de processo no ImPProS.

Esse trabalho propõe a implementação da ferramenta de apoio ProEvaluator, a qual está inserida no contexto do módulo de avaliação de processos de software.

# Ferramentas para Avaliação de Processos de Software

*Este capítulo visa apresentar as ferramentas de avaliação de processos de software encontradas e, através de um comparativo, mostrar suas respectivas qualidades e deficiências.*

*O capítulo está dividido da seguinte forma: A Seção 4.1 descreve a ferramenta Appraisal Assistant; A Seção 4.2 descreve a ferramenta ambiente de Apoio às Avaliações MPS.BR; A Seção 4.3 descreve a ferramenta SEAL; A Seção 4.4 descreve a ferramenta PISA; A Seção 4.5 descreve uma primeira versão da ferramenta ProEvaluator; A Seção 4.6 apresenta um comparativo sobre as ferramentas; Por fim, a Seção 4.7 apresenta as considerações finais sobre o capítulo.*

## 4.1 Appraisal Assistant

É uma aplicação *stand-alone* (aplicação desktop) desenvolvida pelo Software Quality Institute, da Griffith University, para dar suporte à avaliação da capacidade do processo ou maturidade organizacional. Sua abordagem é consistente com os requisitos do Processo de Avaliação da ISO/IEC 15504 e os requisitos de avaliação do CMMI. Diferente de outras ferramentas existentes, Appraisal Assistant possui uma abordagem dirigida a evidências para registrar a informação gerada durante a avaliação (Figura 4.1). Ela provê:

- Suporte a vários modelos de processos como o ISO/IEC 15504-5, Automotive SPICE, CMMI-DEV v.1.2, +SAFE, e CMMI SE/SW/IPPD/SS V 1.1;
- Suporte ao método de avaliação SCAMPI e ao método genérico definido pela Norma ISO/IEC 15504;
- Conversão do resultado de um framework para outro. Ou seja, é possível expressar resultados de uma avaliação CMMI como perfis de processo padrão ISO/IEC 15504;
- Suporte a relatórios da avaliação.

Essa ferramenta permite a edição modelos existentes e o cadastro de novos modelos baseados no CMMI e na ISO/IEC 15504. O mesmo se aplica para métodos de avaliação

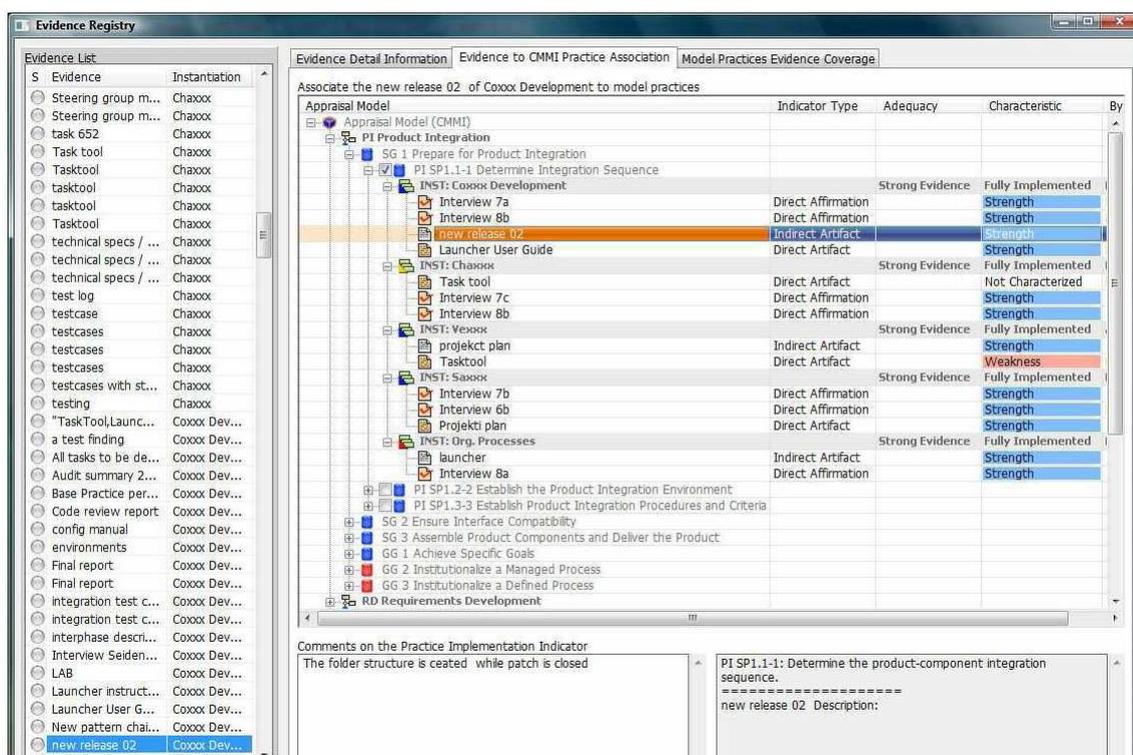


Figura 4.1 - Registro de Evidências CMMI

## 4.2 Ambiente de Apoio às Avaliações MPS.BR

O CORE-KM (Customizable Organizational Resources Environment with Knowledge Management) é um ambiente Web customizável para gerência de conhecimento em diferentes organizações, capaz de apoiar seus processos

organizacionais [GALOTTA, 2004]. A infra-estrutura base do CORE-KM traz uma série de ferramentas genéricas que podem ou não estar nos ambientes customizados, em função da necessidade de cada organização.

O Ambiente de Apoio a Avaliações MPS.BR [MURADAS, 2006] pode ser instanciado a partir do CORE-KM. Este ambiente tem o conhecimento comum MPS.BR e os templates de artefatos requeridos para a execução do processo [MURADAS, 2006]. A Instituição Avaliadora precisa então cadastrar suas próprias informações, bem como definir sua interface (Figura 4.2). A ferramenta é utilizada para apoiar uma Instituição Avaliadora na realização de avaliações MPS.BR. Os principais objetivos desse ambiente são:

- Apoio à execução das atividades de planejamento do processo de avaliação MPS.BR
- Gerência das atividades das Instituições Avaliadoras;
- Disseminação das experiências vividas pelos avaliadores durante a realização das avaliações segundo o método de avaliação MPS.BR.

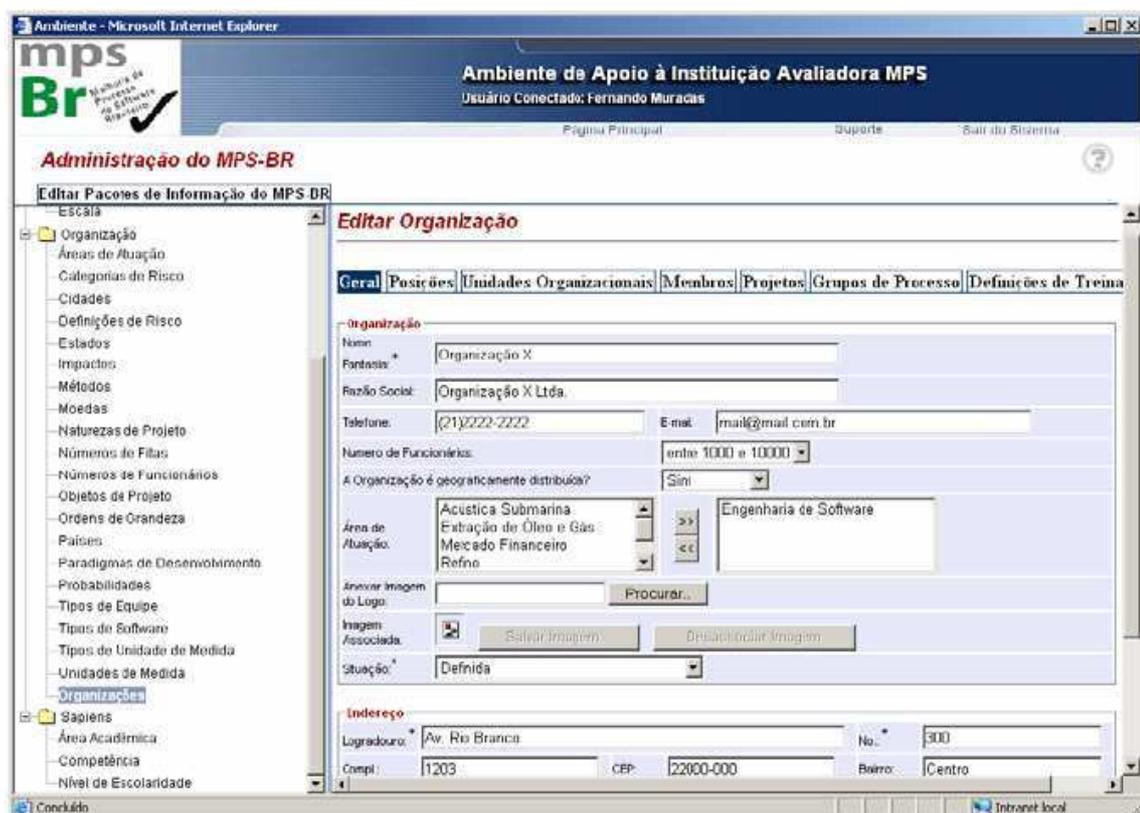


Figura 4.2 - Cadastro da organização avaliada

O Ambiente de Apoio a Avaliações MPS.BR é composto por 3 ferramentas específicas. A primeira trata do apoio ao processo de avaliação MPS.BR propriamente dito. A segunda trata do cadastro e consulta dos profissionais da Instituição Avaliadora com suas características. A última é uma agenda para se cadastrar atividades e consultar a disponibilidade dos profissionais para realizar as avaliações.

De acordo com o processo de avaliação do MPS.BR, essa ferramenta está focada basicamente na fase de planejamento da avaliação. Ela também não oferece suporte à outros modelos e métodos de avaliação.

### **4.3 SEAL**

A ferramenta para avaliação de processos SEAL [Walker, 1995] é uma ferramenta *Stand-Alone* e pode ser encontrada gratuitamente na internet. Ela provê uma abordagem sistemática para a realização de avaliações baseadas no modelo de avaliação definido pela norma ISO/IEC 15504. É possível criar várias instâncias de avaliação e para cada uma delas, por meio de uma interface gráfica, registrar o nível alcançado dos produtos de trabalho, práticas base e práticas de gerenciamento (Figura 4.3).

Uma vez que esses níveis são determinados, é possível fazer uma avaliação automática dos atributos de processo. É possível aumentar ou diminuir os atributos de processo, práticas base, práticas de gerenciamento e produtos de trabalho, por isso essa ferramenta é adaptável a pequenas evoluções da norma ISO/IEC 15504, versão relatório técnico, ou seja, ela ao suporta o cadastro de novos modelos e métodos de avaliação, apenas a atualização da norma ISO/IEC 15504.

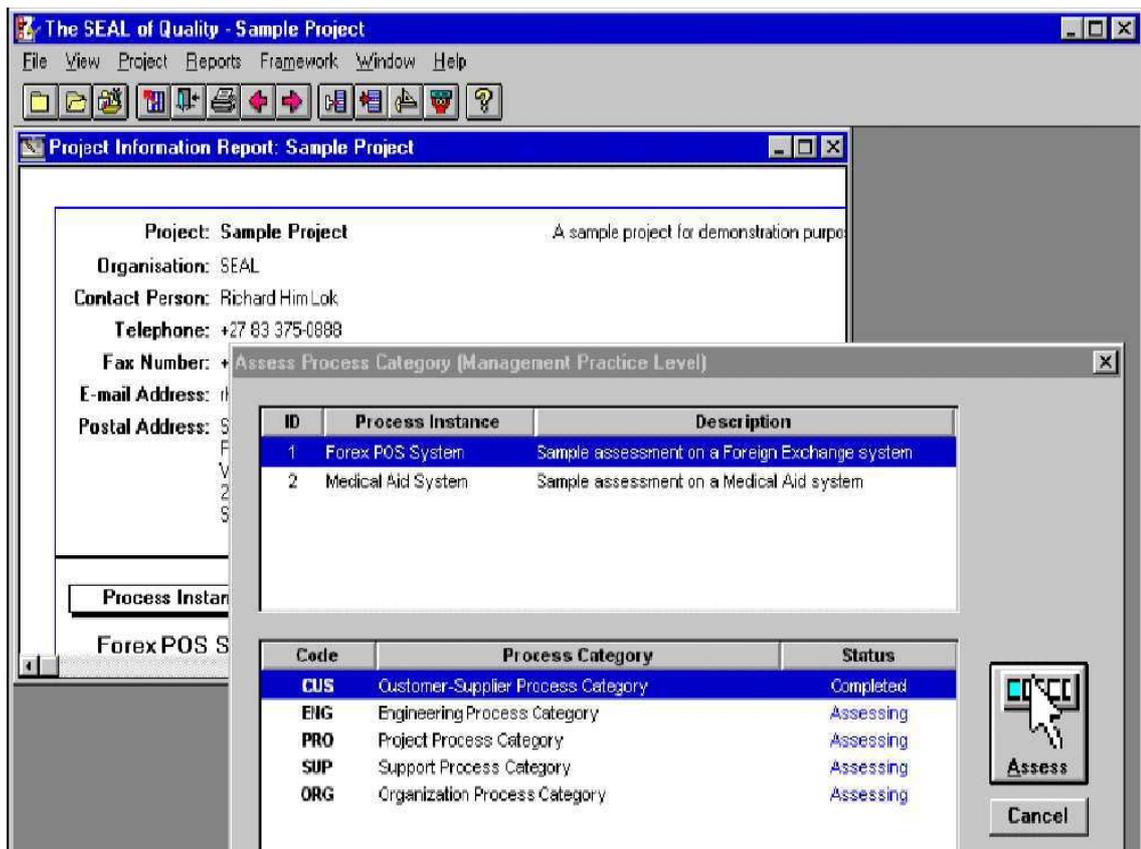


Figura 4.3 - Definição de processos na ferramenta Seal

## 4.4 PISA

A ferramenta PISA é uma ferramenta *Stand-Alone* a qual provê um suporte automático para as atividades de avaliação de processos de software baseadas na norma ISO/IEC 15504 [Fabbrini, 2003]. Ela oferece um apoio às atividades de coleta/validação de dados e fornecimento de notas aos atributos de processo (Figura 4.4). De acordo com a metodologia automatizada por essa ferramenta, a avaliação de processos é dividida em quatro partes: preparação, coleta de dados, avaliação e emissão de relatórios.

A fase de preparação inclui atividades de registro de dados da organização, configuração de pesos das características da avaliação e registro do escopo da avaliação. A fase de coleta de dados contém a atividade de preenchimento de checklists. A fase de avaliação inclui as atividades de análise de práticas base e práticas de gerenciamento automaticamente. A última fase consiste na emissão do relatório final de avaliação.

Essa ferramenta pode ser utilizada por organizações com diferentes domínios de aplicação e tamanho, assim como pode ser adaptável a pequenas evoluções da norma. Porém, ela só pode ser utilizada para avaliações baseadas na norma ISO/IEC 15504.

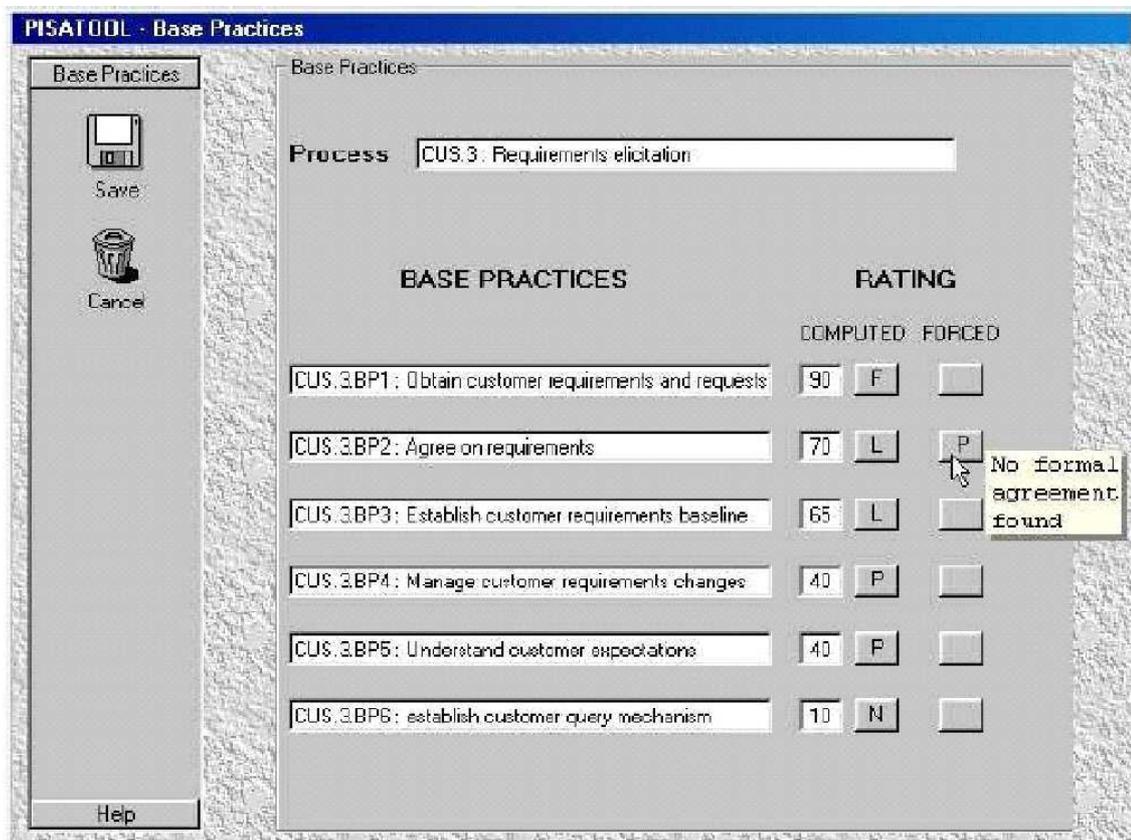


Figura 4.4 - Atribuição de notas da ferramenta PISA

## 4.5 ProEvaluator – MPS.BR

Segundo [MOURA, 2007], foi proposto uma versão do ProEvaluator para avaliação de processos de software, o qual dá apoio à automação do Método de Avaliação do MPS.BR, o MA-MPS. A ferramenta ProEvaluator – MPS.BR – é uma ferramenta *Stand-Alone* a qual visa automatizar algumas atividades do método MA-MPS [MPS.BR, 2006b], para ser usada pelas organizações durante uma avaliação oficial MPS.BR. Essa ferramenta também pode ser usada numa pré-avaliação para as organizações se auto-avaliarem, podendo identificar seus pontos fortes e oportunidades de melhoria.

Essa ferramenta visa automatizar as fases de “Realização da Avaliação” e “Documentação dos Resultados”, descritas no processo de avaliação do MPS.BR. Suas Principais funcionalidades são:

- Oferecer suporte ao processo de avaliação interna das organizações;
- Oferecer suporte à pré-avaliação realizada pelas organizações;
- Oferecer suporte à atividade de avaliação de processos de software definida no Método de Avaliação MPS.BR.



Figura 4.5 - Avaliação de Processo na ferramenta [MOURA, 2007].

Conforme proposto, essa ferramenta não oferece suporte a outros modelos e métodos de avaliação, apenas ao MPS.BR.

## 4.6 Comparativo entre as Ferramentas

A seguir encontra-se um comparativo em relação às principais características das ferramentas descritas. Figura 4.6.

<b>Ferramenta</b>	<b>Gratuita</b>	<b>Ambiente Web</b>	<b>Ambiente Multi-usuário (Perfil)</b>	<b>Suporta diferentes domínios</b>	<b>Suporta diferentes modelos e métodos</b>	<b>Maneamento entre modelos</b>	<b>Cadastro de Evidências</b>	<b>Cadastro de regras de agregação (notas)</b>	<b>Suporta atribuição de notas</b>
PISA				X	X				X
SEAL	X								X
Ambiente de Apoio às Avaliações MPS.BR	X	X		X					
Appraisal Assistant	X		X	X		X	X	X	X
ProEvaluator – MPS.BR	X			X			X		X

**Figura 4.6 – Comparativo das Ferramentas de Avaliação**

Como se pode observar, a ferramenta que possui uma maior cobertura em relação às características comuns de uma ferramenta de avaliação é a Appraisal Assistant. Apesar de ela ser *Stand-alone*, é uma ferramenta gratuita que pode ser utilizada em organizações de diferentes domínios. Ela também oferece suporte a cadastro de regras de agregação para geração de notas, cadastro de evidências e mapeamentos entre os modelos previstos.

No entanto, apesar de ela oferecer suporte a mais de um modelo, essa ferramenta não pode ser considerada como uma ferramenta genérica no que diz respeito a modelos e métodos de avaliação, pois ela apenas prevê apenas modelos baseados na ISO/IEC 15504 e CMMI e seus respectivos métodos de avaliação.

As ferramentas PISA e SEAL são ferramentas *stand-alone* e sem controle de usuários. Ambas são baseadas no modelo ISO 15504, e oferecem suporte à atribuição de notas de avaliação. Além disso, a ferramenta SEAL está disponível gratuitamente na Internet, porém, não é adaptável a diferentes

domínios e modelos. Já a ferramenta PISA é adaptável a diferentes domínios e pequenas evoluções do modelo ISO 15504.

O Ambiente de Apoio às Avaliações MPS.BR é uma ferramenta Web gratuita que suporta apenas o modelo MPS.BR. Além disso, apenas suporta a fase de planejamento do método MA-MPS, mas não contempla às fases de avaliação e geração de relatório final. Já a versão do ProEvaluator – MPS.BR – é uma ferramenta *stand-alone*, gratuita e oferece suporte apenas as fases de avaliação e geração de relatório final. Também só suporta o modelo MPS.BR.

## 4.7 Considerações Finais

Este capítulo apresentou diversas ferramentas de avaliação de processos de software. Como pode-se observar, a maioria delas são ferramentas *stand-alone*, mono-usuário e sem controle de perfil. Nenhuma delas pôde ser considerada genérica no que diz respeito a modelos e métodos de avaliação. Apenas o Appraisal Assistant oferece suporte a mais de um modelo e método de avaliação.

No próximo capítulo, será descrita a ferramenta para avaliação de processos de software desenvolvida neste trabalho, o ProEvaluator. O ProEvaluator é uma ferramenta de avaliação genérica, ou seja, permite o cadastro de qualquer modelo ou método de avaliação. Seus requisitos foram baseados nos conceitos e requisitos propostos pelo ImPProS [Oliveira, 2007], em conjunto com as possibilidades de melhorias do ProEvaluator – MPS.BR [MOURA, 2007].

Essa ferramenta também foi integrada ao ambiente ImPProS. No entanto, por ser uma ferramenta Web, algumas limitações da atual arquitetura do ImPProS impossibilitam uma integração mais efetiva. Dessa maneira, uma nova arquitetura do ambiente e, conseqüentemente, uma nova forma de comunicação entre as ferramentas serão propostos a fim de tornar o ambiente ImPProS mais eficiente e adaptado às novas necessidades das empresas.



# ProEvaluator: Uma Ferramenta para Avaliação de Processos de Software

*Conforme descrito no capítulo 2, para uma organização definir adequadamente seus processos de desenvolvimento de software e garantir uma melhoria contínua dos processos é necessário avaliá-los. Segundo [WANGENHEIM, 2005], uma avaliação de processos tem o objetivo de determinar a suas capacidades ou a maturidade da unidade organizacional para funcionar dentro das metas de qualidade, custo e cronograma e encontrar pontos fortes e fracos do processo de modo que ele possa ser constantemente melhorado.*

*Este capítulo apresenta uma ferramenta Web de avaliação de processo de software integrada ao ambiente ImPProS, o ProEvaluator. Tal ferramenta permite avaliar processos de software utilizando qualquer modelo ou método de avaliação conforme propõe o ImPProS [Oliveira, 2007].*

*O capítulo está dividido da seguinte forma: A Seção 5.1 descreve uma visão geral sobre a ferramenta; A Seção 5.2 descreve os requisitos da ferramenta; A Seção 5.3 descreve a arquitetura da ferramenta; A Seção 5.4 a integração com o ImPProS; Por fim, a Seção 5.5 apresenta as considerações finais sobre o capítulo.*

## 5.1 Visão Geral

O ProEvaluator é uma ferramenta Web a qual auxilia e automatiza o processo de avaliação de processos de software, integrada ao ambiente de implementação de processo de software, o ImPProS [OLIVEIRA, 2005].

O principal objetivo dessa ferramenta é possibilitar a avaliação utilizando qualquer modelo ou método de avaliação. Para isso, ela implementa um meta-modelo genérico para o processo de avaliação seguindo o ciclo proposto pelo ISO/IEC 15504. Isso se dá pelo fato dessa norma definir um conjunto de requisitos genéricos para realização de uma avaliação. Segundo [ISO/IEC 15504-3, 2004], a norma é um padrão para avaliação de processos. Ela provê um framework para construções de modelos de avaliação.

Dessa maneira, a ferramenta visa suportar as principais fases relativas ao ciclo do processo de avaliação proposto pela ISO/IEC 15504, conforme pode ser visto na figura 5.1. Em cada fase destacada, o ProEvaluator se insere visando automatizar as atividades requeridas para o processo de avaliação. A fase referente. Por questões de escopo e cronograma, a fase referente a documentação e relatórios de resultados não será contemplada nesse trabalho.



**Figura 5.1 – Fases contempladas pelo ProEvaluator**

O ProEvaluator tanto pode ser utilizado no processo de avaliação interna das organizações quanto no processo de avaliação oficial pela instituição avaliadora. No processo de avaliação interna, a organização visa obter um diagnóstico dos seus processos com o objetivo de melhorar a qualidade dos mesmos identificando os pontos fortes e fracos de seus processos. Já na avaliação oficial, a organização visa se certificar da qualidade de seus processos através de uma instituição avaliadora credenciada.

Essa proposta de utilização por diferentes organizações ao mesmo tempo unido a crescente mobilidade dos usuários, a necessidade de utilização partir de diferentes locais e em diversos dispositivos e conseqüentemente a necessidade

de interoperabilidade, a arquitetura Web foi proposta para o ProEvaluator. Dessa maneira, a ferramenta poderá ser utilizada por qualquer pessoa e em qualquer lugar que possua um *browser*. Essa arquitetura, além de possibilitar uma melhor plataforma multi-usuário e com controle de acesso, possibilita uma maior segurança no processo de avaliação, visto que o processo envolve diferentes organizações.

Algumas tecnologias foram utilizadas para obter uma melhor qualidade do código e uma conseqüente diminuição no custo de manutenção, um melhor desempenho e melhor produtividade no desenvolvimento. Entre elas o Apache Struts [Struts, 2009] e o Hibernate [Hibernate, 2009]. As páginas Web foram implementadas utilizando JSP [JSP, 2009] e o *servet container* Apache TomCat [Tomcat, 2009]. O banco de dados utilizado foi o MySql [MySql, 2009].

## 5.2 Requisitos

Os requisitos da ferramenta ProEvaluator foram identificados baseados na proposta do ImPProS [Oliveira, 2005], na análise das ferramentas de avaliação existentes feita no capítulo anterior onde foram identificados os pontos fracos e fortes de cada aplicação e avaliação e na real necessidade das empresas no que diz respeito a uma ferramenta para avaliação de processos.

Com base nessa análise, seguem os requisitos funcionais e não-funcionais do ProEvaluator:

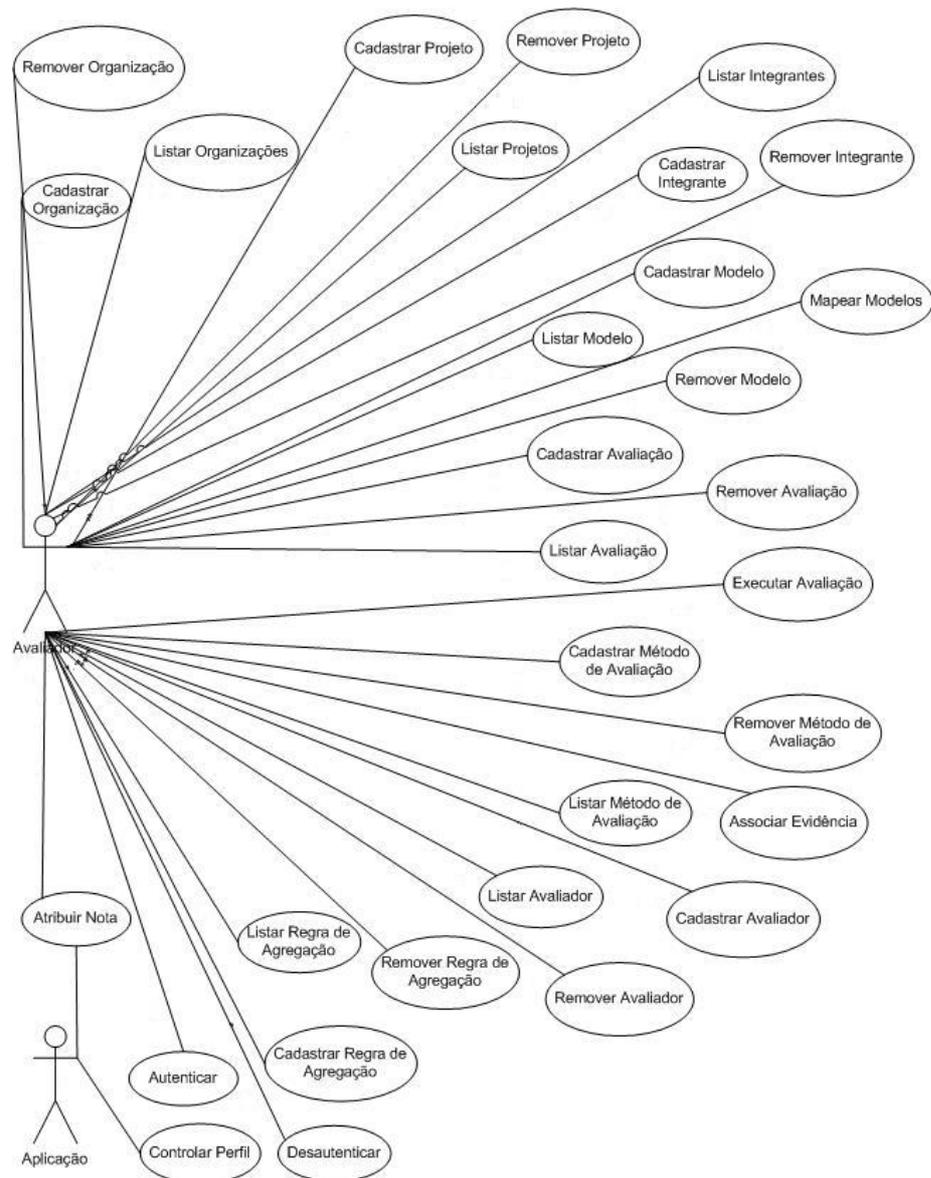
- Gratuidade – A ferramenta deve ser gratuita e disponível para uso público;
- Ferramenta Web – A ferramenta deverá ser implementada em uma arquitetura web de forma a possibilitar o acesso dos dados a partir de qualquer usuário, localidade, dispositivo ou sistema operacional;
- Ferramenta genérica – A ferramenta deve suportar qualquer modelo de processos e seu respectivo método de avaliação na realização da avaliação. Isso é uma premissa para o funcionamento integrado ao ImPProS;

- Possuir o ciclo de avaliação da norma ISO 15504: A ferramenta deve suportar as fases de planejamento, coleta de dados, validação de dados e pontuação de atributos de processo;
- Manter organização: A ferramenta deve possibilitar o cadastro, remoção e listagem das organizações a serem avaliadas;
- Manter projeto: A ferramenta deve possibilitar o cadastro, remoção e listagem dos projetos das organizações;
- Manter equipes dos projetos: A ferramenta deve possibilitar o cadastro, remoção e listagem dos integrantes dos projetos;
- Manter modelos: A ferramenta deve possibilitar o cadastro, remoção e listagem de modelos e normas de qualidade e seus respectivos componentes;
- Manter método de avaliação: A ferramenta deve possibilitar o cadastro, remoção e listagem de métodos de avaliação;
- Manter regras para atribuição de notas: A ferramenta deve permitir o cadastro, remoção e listagem de regras para atribuição de notas aos componentes do modelo. Essas regras são baseadas nos métodos de avaliação de cada modelo e são utilizadas pela ferramenta para geração de notas;
- Suporte a atribuição de notas ou conceitos: A ferramenta deve possibilitar a atribuição de notas para os componentes dos modelos de acordo com a análise das evidências da avaliação. A ferramenta também deve prever a nota ou conceito com base nas regras cadastradas;
- Associar evidências - A ferramenta deve possibilitar a associação de evidências aos componentes do modelo utilizado na avaliação;
- Manter avaliação: A ferramenta deve possibilitar o cadastro, remoção, listagem e execução de avaliação;
- Mapear modelos: A ferramenta deve possibilitar o mapeamento entre os componentes dos modelos de forma que o usuário possa

obter uma previsão de uma avaliação em um modelo a partir de uma avaliação utilizando um outro modelo previamente mapeado;

- Manter avaliadores – A ferramenta deve possibilitar o cadastro, remoção e listagem de avaliadores. Cada avaliador deverá ter um perfil de acesso;
- Controlar acesso – A ferramenta deve gerenciar perfil dos usuários e realizar a autenticação dos mesmos, de forma a evitar que um usuário execute alguma atividade sem permissão;
- Integração com o ambiente ImPProS – A ferramenta deverá ser integrado ao ambiente ImPProS, podendo ser executada a partir do mesmo, utilizar informações de sua base e alimentá-la com os resultados das avaliações;

A partir dos requisitos descritos, foram identificados os casos de uso da aplicação, os quais representam as funcionalidades da ferramenta. Com base nesses casos de uso, um diagrama de casos de uso foi modelado conforme pode ser visto na figura 5.2. Para realização de qualquer funcionalidade da aplicação, o usuário deverá estar autenticado no sistema. Por razões de simplificação do diagrama, o modelo não será desmembrado em seus componentes (atributos de processo, processo, resultados, práticas, etc.).



**Figura 5.2 - Diagrama de casos de uso**

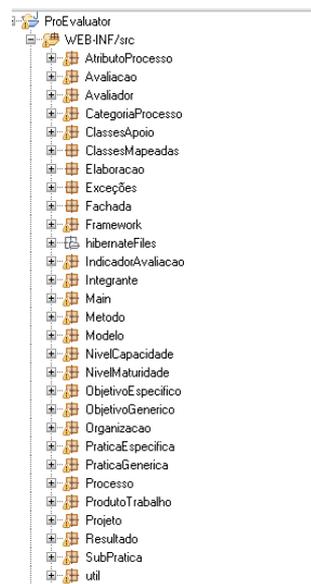
## 5.3 Arquitetura

Visando construir uma aplicação escalável, com baixo custo de manutenção e de fácil entendimento, o sistema está amplamente modularizado, estruturado em camadas distintas, independentes, fracamente acopladas e com papéis bem definidos [Sommerville, 2000]. São elas:

- Apresentação (GUI): Esta camada possui todas as interfaces com os usuários da aplicação. Todas as páginas Web e suas controladoras pertencentes ao Struts estão alocadas nessa camada;

- Negócio: Esta camada é responsável pelas regras de negócio do sistema. É nela que estão compreendidas todas as classes que implementam o domínio da aplicação;
- Dados: Esta camada é responsável pelas operações de armazenamento, manutenção e recuperação dos dados pertencentes ao sistema. Essa camada possui as classes de repositório e os mapeamentos do Hibernate.

O fluxo de dados ocorre da camada superior (apresentação) para a camada de dados, baseado nos conceitos da engenharia de software [Sommerville, 2000]. Para possibilitar a modularização e o baixo acoplamento, cada objeto de fronteira de uma camada possui uma interface para comunicação com a outra camada. A aplicação também está separada por pacotes, onde cada pacote agrupa as classes relacionadas, conforme a figura 5.3.



**Figura 5.3 - Pacotes do ProEvaluator**

### 5.3.1 Design Patterns

Visando a criação de uma arquitetura mais estável e melhor definida, alguns padrões de projeto foram adotados nessa ferramenta [Sommerville, 2000].

### **5.3.1.1 Padrões Estruturais**

Os padrões estruturais estão associados à maneira como classes e objetos são organizados estruturalmente. Eles provêm facilidades para a utilização de conceitos de orientação a objetos (herança e composição, por exemplo) e tratam da composição de classes e objetos para formar estruturas grandes e complexas. Dentre eles, os seguintes foram utilizados:

- Facade - O padrão Facade (Fachada) provê uma interface simples para facilitar a comunicação com um subsistema, ou seja, ele encapsula um subsistema e se torna o único acesso a este. O padrão facade normalmente é implementado utilizando também o padrão Singleton. Um objeto fachada será utilizado para encapsular todo o sistema, servindo como ponto único a este (por isso da implementação utilizando também o padrão Singleton);
- PDC – Persistent Data Collections - Padrão utilizado para separar as regras da coleção de negócio das regras da coleção de dados, respeitando a arquitetura de camadas adotada no projeto. Tem o objetivo de tornar independente, para a aplicação, a implementação da persistência.

### **5.3.1.2 Padrões Comportamentais**

São padrões que se preocupam com algoritmos e a atribuição de responsabilidades entre objetos. Dentre eles, os seguintes foram utilizados:

- Chain Of Responsibility - Esse padrão evita o acoplamento do remetente de uma solicitação ao seu receptor, ao dar a mais de um objeto a oportunidade de tratar uma solicitação. Encadeia os objetos receptores, passando a solicitação ao longo da cadeia até que um objeto a trate. Esse padrão é implementado na arquitetura do framework Struts;
- Observer - O Observer é um padrão de projeto de software que define uma dependência um-para-muitos entre objetos de modo que quando um objeto muda o estado, todos seus dependentes sejam notificados e atualizados automaticamente. Permite que objetos interessados sejam avisados da mudança de estado ou

outros eventos ocorrendo num outro objeto. É utilizado na camada de apresentação em alguns objetos na página Web.

### 5.3.1.3 Padrões Criacionais

Os padrões criacionais estão associados ao processo de criação de objetos. Dentre eles foi utilizado:

- Singleton - O padrão Singleton é um padrão criacional, utilizado quando há a necessidade de se garantir que apenas uma única instância de uma classe será usada e também prover um ponto de acesso global a essa instância. No ProEvaluator, a classe Fachada, além de implementar o padrão de projeto Facade, também implementa o padrão Singleton.

## 5.4 Integração com o ImPProS

O ImPProS é um ambiente de implementação de processo de software a partir do uso de ferramentas de suporte ao processo de software e de apoio ao desenvolvimento de software. Ele deve possibilitar em sua estrutura a integração de ferramentas, a partir da disponibilização por parte do fornecedor destas ferramentas, de um conector de comunicação entre o ImPProS e a ferramenta a ser integrada. Dessa maneira, o ImPProS deve ser capaz de ativar outras ferramentas e compartilhar funções de outras ferramentas [Oliveira, 2007], conforme pode ser visto na figura 5.4.

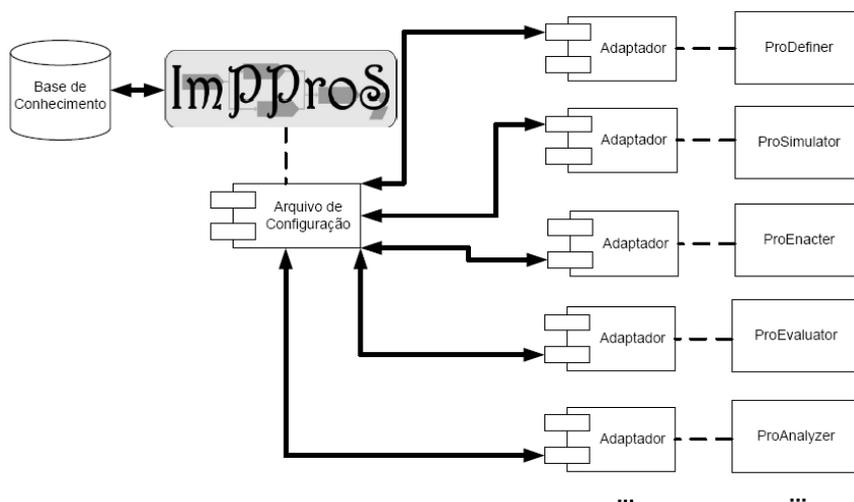


Figura 5.4 - Estrutura de Integração do ImPProS [Oliveira, 2007]

De forma geral, um conector deve ser implementado e integrado ao ambiente ImPProS, conforme os conceitos e requisitos descritos em [Oliveira, 2007]. Esse conector deve permitir ao ambiente ativar a execução de qualquer ferramenta de apoio e até mesmo obter acesso aos serviços da ferramenta. Maiores detalhes podem ser encontrados em [Oliveira, 2007].

Essa arquitetura de comunicação foi proposta por [Oliveira, 2007] considerando aplicações *Stand-alone*, como é o caso do ImPProS, ao contrário da ferramenta ProEvaluator, a qual propõe uma nova arquitetura, a Web. Dessa maneira, no contexto de compartilhamento de serviços das ferramentas, essa estrutura de comunicação se torna inviável em um ambiente Web. Uma arquitetura adequada para esse cenário poderia ser a arquitetura orientada a serviços, onde cada ferramenta disponibilizaria suas funcionalidades como serviços através de uma interface, o *Web-Services* [Barrett, 2006]. Dessa maneira, o acesso às funcionalidades das ferramentas em um ambiente Web seria mais viável, além do que possibilitaria uma maior modularização das ferramentas. Maiores informações em [Barrett, 2006].

Em se tratando da ativação das ferramentas de apoio e o compartilhamento de dados na base de conhecimento do ImPProS, foi desenvolvido um adaptador conforme proposto por [Oliveira, 2007] para a integração do ProEvaluator. Esse adaptador também é responsável por compartilhar e alimentar a base de conhecimento do ImPProS. Após sua implementação, esse adaptador foi cadastrado e ativado no ambiente. A partir daí, a ferramenta ProEvaluator pode ser ativada pelo ambiente, mesmo elas possuindo arquiteturas diferentes, como pode ser visto na figura 5.5.



Figura 5.5 – Execução do ProEvaluator através do ImPProS

## 5.5 Considerações Finais

Neste capítulo foi apresentada a ferramenta de avaliação de processos de software, o ProEvaluator. Seus requisitos funcionais e não funcionais foram identificados com bases nas premissas propostas pelo ImPProS, análise das ferramentas disponíveis e novas necessidades dos usuários. Os seus casos de uso também foram mostrados através do diagrama de casos de uso. Toda a arquitetura da aplicação foi detalhada, incluindo tecnologias, padrões de projeto, entre outros conceitos alinhados à engenharia de software.

Também foi descrito a estrutura de integração com o ImPProS, conforme proposto em [Oliveira, 2007]. Devido a diferença entre arquiteturas, a forma de comunicação proposta para execução de serviços se tornou inviável. Dessa maneira, uma nova arquitetura de comunicação se faz necessário como por exemplo uma arquitetura orientada a serviços, SOA [Barrett, 2006].



*A adoção de modelos e normas está sendo uma das principais saídas para a melhoria de processos. No entanto a dificuldade de definir adequadamente um processo e mantê-lo em constante melhoria é o grande desafio das empresas. Dessa maneira, é necessário que o processo adaptado seja avaliado para garantir que todas as boas práticas recomendadas pelos modelos estejam sendo implementadas.*

*A automatização dos processos de avaliação é uma necessidade não apenas da equipe de avaliação como também da própria organização. E os dados gerados através do resultado de muitas avaliações acaba não retornando para as organizações como ponto de melhoria.*

*Apesar de já existirem ferramentas de avaliação de processos no mercado, tornou-se claro através deste trabalho que existe a necessidade de padronização para um modelo genérico, de forma que a avaliação possa ser feita independente do modelo ou método de avaliação e a necessidade de acesso em à aplicação de diferentes lugares e por diferentes organizações, torna necessário a implementação da ferramenta com uma arquitetura Web. Outro ponto importante é a necessidade da integração dessa ferramenta a um ambiente de implementação de processos de software, o ImPProS.*

*Este capítulo apresenta as considerações finais sobre o trabalho desenvolvido. A seção 6.1 descreve as contribuições da pesquisa realizada. Na seção 6.2 é apresentada uma proposta de possíveis trabalhos futuros.*

## 6.1 Contribuições do Trabalho

O trabalho descrito nesta dissertação teve por objetivo o desenvolvimento de uma ferramenta para Avaliação de Processos de Software genérica, ou seja, pode ser efetuado o processo de avaliação baseado em qualquer modelo ou método de avaliação.

Essa ferramenta poderá ser utilizada para as organizações se auto-avaliarem, obtendo o seu nível de maturidade, além de identificar os pontos fracos e oportunidades de melhoria do seu processo. Outra abordagem interessante é utilizá-la durante a avaliação oficial pela instituição avaliadora.

Dessa maneira, as principais contribuições deste trabalho são:

- Implementação de uma ferramenta Web de avaliação de processos de software genérica, ou seja, com suporte a diferentes modelos e métodos de avaliação baseada no processo de avaliação proposto pela ISO 15504;
- Efetuado um estudo sobre a estrutura de comunicação proposta por [Oliveira, 2007] a fim de fazer uma análise crítica da possibilidade de implementação. Integração da ferramenta ao ambiente ImPProS e proposto uma nova arquitetura para o ambiente ImPProS e suas ferramentas de apoio, a arquitetura orientada a serviços (SOA) [Barrett, 2006].

Para alcançar esses resultados, foi necessária a revisão da literatura sobre os principais modelos, normas e métodos de avaliação de processos a fim de implementar uma ferramenta genérica baseada na norma ISO 15504.

Foi ressaltada a importância de uma organização definir um processo de desenvolvimento de software e utilizar a avaliação de processos para identificar pontos fortes e oportunidades de melhoria do mesmo, aumentando assim, a qualidade do seu processo e, conseqüentemente, a qualidade do software produzido.

Também foi realizado um estudo e apresentado uma comparação sobre as principais ferramentas de avaliação. Isso foi necessário para propor novos requisitos para a ferramenta. Outro ponto relevante foi a implementação do

mapeamento de modelos, o que possibilita a organização ter visões de avaliações em diferentes modelos.

## 6.2 Trabalhos Futuros

Apesar de muitas melhorias terem sido desenvolvidas em relação à primeira versão do ProEvaluator [Moura, 2007], alguns trabalhos futuros podem ser desenvolvidos:

- Importação/Exportação dos dados da auditoria para a ferramenta - A ferramenta deve permitir a exportação de dados da avaliação para uma planilha excel. Atualmente, as avaliações dos diversos modelos são realizadas através de planilhas. Para tornar a migração da planilha para a ferramenta ProEvaluator gradual, é importante que essa ferramenta suporte a importação de dados da planilha;
- Geração de gráficos – A ferramenta deve possibilitar a geração de gráficos sobre o resultado da avaliação;
- Geração de relatórios – A ferramenta deve possibilitar a geração dos relatórios sobre os resultados da avaliação. Isso faria com que a ferramenta atendesse a fase de documentar e reportar os resultados da ISO 15504;
- Melhoria em desempenho e usabilidade – Foi observado que há possibilidades de melhoria no que diz respeito à desempenho e usabilidade da aplicação. Novas interações e designs podem ser propostos a fim de deixar o sistema mais intuitivo.
- Implementação da nova arquitetura de integração com o ImPProS – Partindo da análise sobre a arquitetura de comunicação, foi proposto a implementação da arquitetura orientada a serviços no ambiente ImPPoS entre todas as ferramentas;
- Necessidade de estudo de caso – Após a implementação da ferramenta, é necessário um estudo de caso para avaliar a ferramenta em um ambiente real.

# REFERÊNCIAS BIBLIOGRÁFICAS

- [ACUÑA, 2001] ACUÑA, S. T.; FERRÉ, X. Software Process Modelling. In: WORLD MULTICONFERENCE ON SYSTEMICS, CIBERNETICS AND INFORMATICS, 5, 2001, Orlando, EUA. Proceedings... p. 1-6.
- [Barrett, 2006] Barrett, R. & Pahl, C. (2006), Model Driven Design of Distribution Patterns for Web Service Compositions, in 'Proc. International Conference on Web Services ICWS '06', pp. 887--888.
- [CMU/SEIa, 2002] STANDARD CMMI APPRAISAL METHOD FOR PROCESS IMPROVEMENT (SCAMPI), Version 1.1: Method Implementation Guidance for Government Source and Contract Process Monitoring, CMU/SEI-2002-HB-002, Pittsburgh, Software Engineering Institute, Carnegie Mellon University. Disponível em: <http://www.sei.cmu.edu>
- [CMU/SEIb, 2002] CAPABILITY MATURITY MODEL INTEGRATION (CMMI), Version 1.1 CMMI for Software Engineering (CMMI-SW, V1.1), Pittsburgh, Software Engineering Institute, Carnegie Mellon University. Disponível em: <http://www.sei.cmu.edu>
- [CMU/SEIc, 1993] CAPABILITY MATURITY MODEL FOR SOFTWARE, Version 1.1 (CMU/SEI-93-TR-024). Pittsburg, Software Engineering Institute, Carnegie Mellon University, 1993. Disponível em: <http://www.sei.cmu.edu>
- [CMU/SEI, 2006] CAPABILITY MATURITY MODEL FOR DEVELOPMENT, Version 1.2. Pittsburg, Software Engineering Institute, Carnegie Mellon University, August 2006. Disponível em: <http://www.sei.cmu.edu>
- [CMU/SEId, 2001] APPRAISAL REQUIREMENTS FOR CMMI, Version 1.1 (ARC, V1.1), (CMU/SEI-2001-TR-034). Pittsburgh, Software Engineering Institute, Carnegie Mellon University, 2001. Disponível em: <http://www.sei.cmu.edu>
- [CHRISSIS, 2004] CHRISSIS, M.B., KONRAD, M., SHRUM, S., 2004, "CMMI Guidelines for Process Integration and Product Improvement", SEI Series in Software Engineering, Ed. Addison Wesley
- [CURTIS, 2002] Curtis, Bill; Hefley, William E.; & Miller, Sally A. The People Capability Maturity Model Guidelines for Improving the Workforce. Boston: Addison-Wesley, 2002.

- [EIA, 1998] Electronic Industries Alliance. Systems Engineering Capability Model (EIA/IS-731). Washington, DC, 1998.
- [FABBRINI, 2003] F.FABBRINI, E. FANTINI, M.FUSANI, G.LAMI "Performing SPICE Assessments: Yet Another Tool". In Joint ESA - 3rd International SPICE Conference on Process Assessment and Improvement 17-21 March 2003 ESTEC, Noordwijk, The Netherlands.
- [Falbo, 1998] Falbo, Ricardo A., Integração de Conhecimento em um Ambiente de Desenvolvimento de Software. Tese de Doutorado, COPPE/UFRJ, Rio de Janeiro, Brasil, 1998.
- [Fuggetta, 2000] Fuggetta, A. (2000) "Software Process: A Roadmap", In: Future of Software Engineering, A Finkelstein (ed).
- [GALOTTA, 2004] GALOTTA, C., OLIVEIRA, K., ROCHA, A.R. 2004, "Apoio a Interação entre Processos de Negócio e de Software através de Gerência do Conhecimento." Simpósio Brasileiro de Qualidade de Software - 2004 - Brasília, DF, Brasil
- [Hibernate, 2009] Hibernate Framework. Disponível em: <https://www.hibernate.org/>. Último acesso: 26/06/2009.
- [HUMPHREY, 1989] HUMPHREY, W. S. Managing the Software Process. SEI Series in Software Engineering, Massachusetts: Addison-Wesley Longman Publishing Co., Inc. Boston, EUA, 1989.
- [HUNTER, 1997] HUNTER, R., ROBINSON, G., WOODMAN, IAN., "Tool Support for Software Process Assessment and Improvement". University of Strathclyde, Department of Computer Science, 1997. Disponível em: <https://www.interscience.wily.com>
- [IEEE, 2004] "SWEBOK – Guide to the Software Engineering Body of Knowledge" – A project of the IEEE Computer Society Professional Practices Committee
- [ISO 9000, 2000] NBR/ISO 9000, 2000, "Sistema de Gestão da Qualidade – Fundamentos e Vocabulário" Associação Brasileira de Normas Técnicas, Rio de Janeiro, Brasil
- [ISO/IEC 15504-2, 2003] International Organization for Standardization and International Electrotechnical Commission. ISO/IEC 15504-2: Information Technology - Process Assessment – Part 2 - Performing an Assessment, Geneva: ISO, 2003.

- [ISO/IEC 15504-3, 2004] The International Organization for Standardization and the International Electrotechnical Commission. *ISO/IEC 15504-3: Information Technology - Process Assessment - Part 3 - Guidance on Performing an Assessment*, Geneve: ISO, 2004.
- [ISO/IEC 15504-4, 2004] - the International Organization for Standardization and the International Electrotechnical Commission. *ISO/IEC 15504-4: Information Technology - Process Assessment – Part 4 - Guidance on use for Process Improvement and Process Capability Determination*, Geneve: ISO, 2004.
- [ISO/IEC 15504-4, 2004] [ISO/IEC 15504-4, 2004] - International Organization for Standardization and International Electrotechnical Commission. *ISO/IEC 15504-4: Information Technology - Process Assessment – Part 4 - Guidance on use for Process Improvement and Process Capability Determination*, Genebra: ISO, 2004.
- [ISO/IEC 15504-5, 2006] The International Organization for Standardization and the International Electrotechnical Commission. *ISO/IEC 15504-5: Information Technology - Process Assessment - Part 5: An exemplar Process Assessment Model*, Geneve: ISO, 2006.
- [ISO/IEC 12207, 1995] The International Organization for Standardization and the International Electrotechnical Commission. *ISO/IEC 12207 Information technology – Software life cycle processes*, Geneve: ISO, 1995.
- [ISO/IEC 12207:1995/Amd 1, 2002] International Organization for Standardization and International Electrotechnical Commission. *ISO/IEC 12207 Amendment: Information Technology - Amendment 1 to ISO/IEC 12207*, Genebra: ISO, 2002.
- [ISO/IEC 12207:1995/Amd 2, 2004] International Organization for Standardization and International Electrotechnical Commission. *ISO/IEC 12207 Amendment: Information Technology - Amendment 2 to ISO/IEC 12207*, Genebra: ISO, 2004.
- [JONES, 2000] JONES, C. “Software Assessments, Benchmarks, and Best Practices”. Estados Unidos: Addison-Wesley. 2000.
- [JSP, 2009] JavaServer Pages Technology. Disponível em: <http://java.sun.com/products/jsp/>. Último acesso: 26/06/2009.
- [MBI, 2007] PESQUISA DE MERCADO JUNTO ÀS 50 MAIORES EMPRESAS DE SOFTWARE DO BRASIL, MBI, ASSESPRO-SP, ITS. Disponível em: [www.assespro-sp.org.br](http://www.assespro-sp.org.br). Último acesso: 26/07/2007.

- [MEMBERS, 2002] MEMBERS OF THE ASSESSMENT METHOD INTEGRATED TEAM. “Standard CMMI Appraisal Method for Process Improvement (SCAMPI)”, Version 1.1, março 2002, Method Definition Document CMU/SEI-2001-HB-001. Estados Unidos: Carnegie Mellon University – Software Engineering Institute. 2001. Disponível em [www.sei.cmu.edu/pub/documents/01.reports/pdf/01hb001.pdf](http://www.sei.cmu.edu/pub/documents/01.reports/pdf/01hb001.pdf).
- [MOURA, 2007] MOURA, J., “PROEVALUATOR: UMA FERRAMENTA PARA AVALIAÇÃO DE PROCESSOS DE SOFTWARE”, DISSERTAÇÃO DE MESTRADO. CIN/UFPE, RECIFE-PE, 2007.
- [MPS.BR, 2007a] ASSOCIAÇÃO PARA PROMOÇÃO DA EXCELÊNCIA DO SOFTWARE BRASILEIRO – SOFTEX. MPS.BR - Guia Geral, versão 1.2, junho 2007. Disponível em: [www.softex.br](http://www.softex.br).
- [MPS.BR, 2007b] ASSOCIAÇÃO PARA PROMOÇÃO DA EXCELÊNCIA DO SOFTWARE BRASILEIRO – SOFTEX. MPS.BR – Guia de Avaliação, versão 1.1, junho 2007. Disponível em: [www.softex.br](http://www.softex.br).
- [MPS.BR, 2007c] ASSOCIAÇÃO PARA PROMOÇÃO DA EXCELÊNCIA DO SOFTWARE BRASILEIRO – SOFTEX. MPS.BR – Guia de Aquisição, versão 1.2, junho 2007. Disponível em: [www.softex.br](http://www.softex.br).
- [MURADAS, 2006] MURADAS, FERNANDO MARTINS “Processo de Avaliação MPS.BR: Definição e Ambiente de Apoio”, COPPE/UFRJ, 2006, Rio de Janeiro.
- [MySQL, 2009] MySQL Database. Disponível em: <http://www.mysql.com/>. Último acesso: 26/06/2009.
- [NBR ISO/IEC 12207, 1998] ASSOCIAÇÃO BRASILEIRA DE NORMAS TÉCNICAS. NBR ISO/IEC 12207:1998. TI – Processos de Ciclo de Vida do Software. Rio de Janeiro: ABNT, 1998.
- [OLIVEIRA, 2005] OLIVEIRA, S., VASCONCELOS, A., ROUILLER, A. C., “Uma Proposta de um Ambiente de Implementação de Processo de Software”, Artigo publicado na Revista InfoComp – Revista de Ciência da Computação da UFLA – vol. 4, n. 1, Lavras-MG, 2005.
- [Paula, 2003] Paula, W. P. Engenharia de software : fundamentos, métodos e padrões. Rio de Janeiro: LTC, 2003. 584p.
- [Paulk et al. 1995] PAULK, M.C., CURTIS, B., CHRISSIS, M. B., WEBER, C. V. The Capability Maturity Model: guidelines for

- improving software process. The SEI Series in Software Engineering, Addison-Wesley, 1995.
- [Pimentel, 2003] Pimentel, F. F. Uma Perspectiva de Mapeamento do CMMI ao MPS.BR. Trabalho de Graduação da Universidade Federal de Pernambuco. Publicada em Setembro de 2003. Disponível em: <<http://www.cin.ufpe.br/~tg>>. Acesso em: 03 out. 08.
- [REIS, 2000] REIS, R. Q. Reutilização de Processos de Software. 2000. Exame de Qualificação de Tese (Doutorado em Informática), Universidade Federal do Rio Grande do Sul, Porto Alegre.
- [REIS, 2003] REIS, C. A. L. Uma Abordagem Flexível para Execução de Processos de Software Evolutivos. 2003. 267 p. Tese (Doutorado em Informática), Universidade Federal do Rio Grande do Sul, Porto Alegre.
- [ROCHA, 2001] ROCHA, Ana Regina Cavalcanti da, MALDONADO, José Carlos & WEBER, Kival Chaves, Qualidade de Software: Teoria e Prática, São Paulo: Prentice Hall, 2001.
- [SEI, 1995] Software Engineering Institute. The Capability Maturity Model: Guidelines for Improving the Software Process. Reading, MA: Addison-Wesley, 1995.
- [SEI, 1997a] Integrated Product Development Capability Maturity Model, Draft Version 0.98. Pittsburgh, PA: Enterprise Process Improvement Collaboration and Software Engineering Institute, Carnegie Mellon University, July 1997.
- [SEI, 1997b] Software Engineering Institute. Software CMM, Version 2.0 (Draft C), October 22, 1997.
- [SEI, 2009] SEI – Software Engineering Institute – Carnegie Mellon. Centro que desenvolve pesquisas em engenharia de software. Disponível em: <<http://www.sei.cmu.edu>>. Acesso em: 04 jun. 00.
- [Sommerville, 2000] Sommerville, I. Engenharia de Software. Addison Wesley, 6ª edição, 2000.
- [Struts, 2009] Apache Struts. Disponível em: <http://struts.apache.org/>. Último acesso: 26/06/2009.
- [WALKER, 1995] WALKER, A.J., LOK, H.R., 1995, “SPICE Assessments using the SEAL assessment tool”, Software Engineering Applications Laboratory.

- [WOHLEN, 2000] WOHLIN, C., RUNESON, P., HOST, M., OHLSSON, M., REGNELL, B., WESSLEN, A., "Experimentation in Software Engineering: an introduction", Kluwer Academic Publishers, USA, 2000.
- [WANGENHEIM, 2005] Wangenheim, C. G. & Thiry, M. Analisando a Integração da ISO/IEC 15504, CMMI-SE/SW e MPS.BR. Relatório Técnico LQPS002.05P. LQPS Qualidade e Produtividade de Software. Disponível em: <<http://www.inf.ufsc.br/~gresse>>. Acesso em: 03 out. 07.