

**Pedro Miguel Gonzalez de Abreu Ribeiro**

# **2MPspe**

**Um Modelo de Melhoria do Processo de  
desenvolvimento de Software para Pequenas Empresas**

Universidade do Minho

2004

**Pedro Miguel Gonzalez de Abreu Ribeiro**

# **2MPspe**

## **Um Modelo de Melhoria do Processo de desenvolvimento de Software para Pequenas Empresas**

Tese submetida à Universidade do Minho para a obtenção do grau de Doutor em Tecnologias e Sistemas de Informação, área de conhecimento Engenharia da Programação e dos Sistemas Informáticos.

Universidade do Minho  
Departamento de Sistemas de Informação  
Escola de Engenharia  
Setembro de 2004

*À Nani,*

*ao Pedro, ao António e à Ângela*

## Agradecimentos

---

Agradeço a todas as pessoas que tornaram possível a realização deste trabalho. Gostaria de expressar a minha gratidão de modo particular:

Ao Prof. Doutor Alexandre Rodrigues, orientador deste trabalho, pelo apoio, partilha de experiências e qualidade das sugestões.

Ao Departamento de Sistemas de Informação, em particular ao Prof. Doutor João Álvaro pela disponibilidade sempre demonstrada.

À Universidade do Minho e à Escola de Engenharia por acolherem este trabalho.

Ao João Oliveira, Miguel Bacelar e Luís Lima pela colaboração nos estudos realizados nas empresas.

À Prof. Doutora Helena Moreira pelo despertar para a área de MPS.

A todos os colegas e amigos do Departamento de Sistemas de Informação em particular à Ana Alice, ao Miguel Brito e Rui Dinis pelo apoio constante.

Ao José Luis, amigo e colega sempre disponível.

À minha família, em particular aos meus pais, irmãos e irmã pelos incentivos e colaboração constante.

À D. Alcina, pela paciência e carinho para com o Pedro, o António e a Ângela.

À Nani, pela enorme ajuda, pelos incentivos, por acreditar sempre e pela compreensão.

Aos meus filhos, Pedro, António e Ângela, por acharem que o Pai estava com o “vírus do trabalho”, mas perceberem...

## Resumo

---

Os sistemas de software são cada vez mais o centro de qualquer produto ou serviço moderno e, conseqüentemente é fundamental assegurar a sua qualidade. Os modelos de melhoria do processo de software têm origem no movimento *Total Quality Management* (TQM) assentando na premissa que o processo para desenvolver um produto tem um grande impacto na qualidade final do produto. Neste contexto o paradigma *benchmarking* é o mais utilizado e envolve a medição da maturidade de uma organização de software através da identificação de um conjunto de melhores práticas.

A questão inicialmente colocada, no âmbito deste trabalho, referiu-se à seguinte problemática: há uma evidência significativa que mostra que um projecto de Melhoria do Processo de desenvolvimento de Software (MPS) com sucesso resultará em melhorias na qualidade do produto e no valor do negócio, mas há também evidência que grande parte das iniciativas de MPS falha.

A partir de um trabalho prático sobre a problemática e de uma revisão exaustiva da literatura, definiu-se como finalidade da presente investigação desenvolver um modelo de MPS adaptado a pequenas empresas de software na área de sistemas de informação de gestão que trate explicitamente os factores contextuais da gestão do conhecimento e da cultura organizacional.

O modelo proposto denomina-se 2MPspe (Modelo de Melhoria do Processo de Software para Pequenas Empresas de software) e engloba três componentes principais. O Modelo Conceptual do 2MPspe ( $M_c2MP$ ) que define um referencial tridimensional constituído pelas dimensões do processo, da capacidade e do conhecimento. O Modelo de Avaliação ( $M_a2MP$ ), que define os indicadores que irão permitir avaliar os atributos do processo. Finalmente, o Método de Implementação ( $M_i2MP$ ) que apresenta as actividades, técnicas e ferramentas a utilizar numa iniciativa de melhoria do processo de desenvolvimento de software. A ferramenta principal do modelo desenvolvido é o questionário 2MPspe, utilizado como suporte para a realização do diagnóstico à empresa de software.

## Abstract

---

Software systems are becoming more and more the core of any modern product or service. Therefore, ensuring its quality is fundamental for business success. Software Process Improvement (SPI) models have their origin in the Total Quality Management (TQM) movement, and derive from the premise that the quality of a working process that produces a certain product has a significant impact on the final quality of that product. Within this context, the benchmarking paradigm is the more widely used and involves measuring the maturity of an organization, through the identification of a set of best practices.

The starting question for the scope of this research project addresses the following issues: there is significant evidence suggesting that a successful SPI project will result in product quality and business value improvements; however, there is also evidence that many SPI projects fail.

Based on some practical work and on a bibliographical review, the purpose of the research work here presented was defined as: developing a SPI model adapted for small software organizations operating in the Management Information Systems area; this model should also explicitly address the following contextual aspects: knowledge management and organizational culture.

The proposed model, named SPI Model for Small Software Organization (SPIM2so), includes three major components. The Conceptual Model ( $M_c$ SPIM) that defines a three-dimensional framework composed by the process, capacity and knowledge dimensions. The Assessment Model ( $M_a$ SPIM) defines the indicators that will allow process attributes to be measured. Finally, the Implementation Method ( $M_i$ SPIM), which presents activities, tools and techniques addressing a SPI project. The main tool within the proposed model is the SPIM2so questionnaire, which is the basis of the assessment phase.

# Índice

---

Agradecimentos .....	i
Resumo .....	ii
Abstract.....	iii
Índice .....	iv
Índice de figuras.....	viii
Índice de tabelas.....	xii
Siglas.....	xiii
<b>Capítulo 1.....</b>	<b>1</b>
1. Introdução .....	1
1.1. A Problemática .....	2
1.2. Finalidade e Objectivos .....	4
1.3. Metodologia de Investigação .....	6
1.4. Organização da Tese .....	8
<b>Capítulo 2.....</b>	<b>10</b>
2. Engenharia de Software .....	10
2.1. Conceitos e Definições .....	10
2.2. Processo de Engenharia de Software .....	15
2.2.1. Modelos do Referencial do Ciclo de Vida.....	17
2.3. Qualidade .....	25
2.3.1. Conceitos sobre Qualidade .....	25
2.3.2. <i>Total Quality Management (TQM)</i> .....	27
2.3.3. Modelos de Qualidade do Software.....	31
<b>Capítulo 3.....</b>	<b>47</b>
3. Melhoria do Processo de Desenvolvimento de Software .....	47
3.1. Processos do Ciclo de Vida do Software (ISO/IEC 12207) .....	49
3.2. O Contexto da Melhoria do Processo de Software .....	50

3.3. Modelos de Melhoria do Processo de Software.....	52
3.3.1. Paradigma <i>Benchmarking</i> .....	53
3.3.2. Paradigma Analítico .....	78
3.3.3. Metodologias Ágeis .....	84
<b>Capítulo 4</b> .....	<b>89</b>
4. Aplicação dos Modelos de Melhoria do Processo de Software.....	89
4.1. Casos de Sucesso e Insucesso dos Modelos de Melhoria.....	90
4.2. Limitações, Problemas e Factores Críticos das Iniciativas de MPS .....	96
4.2.1. Aplicação dos Modelos de MPS nas Pequenas Empresas de Software.....	101
4.3. Aplicação Prática em Pequenas Empresas de Software .....	103
4.3.1. Descrição dos Casos Práticos .....	104
4.3.2. Conclusões Globais sobre os Projectos .....	107
<b>Capítulo 5</b> .....	<b>108</b>
5. Modelo de Melhoria do Processo de Software para Pequenas Empresas	
(2MPspe).....	108
5.1. O Problema .....	108
5.2. Descrição Geral do 2MPspe .....	110
<b>Capítulo 6</b> .....	<b>119</b>
6. Modelo Conceptual do 2MPspe (M <sub>c</sub> 2MP) .....	119
6.1. Introdução .....	119
6.2. Dimensão dos Processos (Modelo de Referência dos Processos) .....	124
6.2.1. Âmbito do MRP <sub>2MPspe</sub> .....	125
6.2.2. Estrutura do MRP <sub>2MPspe</sub> .....	129
6.2.3. Descrição dos Processos do MRP <sub>2MPspe</sub> .....	149
6.3. Dimensão da Capacidade (Referencial de Medição).....	167
6.3.1. Níveis de Capacidade.....	168
6.3.2. Atributos da Capacidade (ACs) .....	170
6.3.3. Escala de Classificação dos ACs. ....	175
6.4. Dimensão do Conhecimento .....	178
6.4.1. Níveis de Conhecimento .....	180
6.4.2. Atributos do Conhecimento .....	182
6.4.3. Escala de Classificação dos AKs .....	190



6.5 Representação no Sistema de Eixos Tridimensional .....	192
<b>Capítulo 7</b> .....	198
7. Modelo de Avaliação (M <sub>a</sub> 2MP).....	198
7.1. Introdução .....	198
7.2. Práticas de Base .....	201
7.2.1. Processos de Engenharia (ENG).....	201
7.2.2. Processos de Suporte (SUP) .....	205
7.2.3. Processos de Gestão (GES).....	207
7.2.4. Processos Organizacionais (ORG).....	209
7.3. Práticas Genéricas.....	212
7.3.1. Nível 1: Processo Executado .....	213
7.3.2. Nível 1.5: Processo Valorizado .....	214
7.3.3. Nível 2: Processo Gerido .....	214
7.3.4. Nível 3: Processo Estabelecido.....	216
7.4. Práticas Transversais.....	217
7.4.1. AK1.1 – Atributo de Gestão do Conhecimento.....	218
7.4.2. AK1.2 – Atributo de Gestão da Cultura Organizacional.....	218
7.5. Aplicação do M <sub>a</sub> 2MP.....	219
7.5.1. Questionário do 2MPspe.....	221
7.5.2. Representação dos Resultados .....	225
<b>Capítulo 8</b> .....	229
8. Método de Implementação do 2MPspe (M <sub>i</sub> 2MP).....	229
8.1. Introdução .....	229
8.2. Descrição do M <sub>i</sub> 2MP .....	237
8.2.1. Etapa 1 – Planear Iniciativa .....	240
8.2.2. Etapa 2 – Realizar Diagnóstico.....	243
8.2.3. Etapa 3 – Planear Acções de Melhoria.....	245
8.2.4. Etapa 4 – Executar Acções de Melhoria.....	253
8.3. Utilização Prática do M <sub>i</sub> 2MP .....	255
<b>Capítulo 9</b> .....	257
9. Conclusões.....	257
9.1. Síntese do Trabalho Realizado .....	257

9.2. Contribuições Principais .....	261
9.3. Trabalho Futuro .....	264
9.4. Considerações Finais .....	266
Bibliografia .....	268
Anexos .....	278

## Índice de figuras

---

Figura 2.1– Camadas da engenharia de software [Pressman 1997] .....	11
Figura 2.2 – Modelo em cascata do processo de desenvolvimento de software [Royce 1970]. .....	19
Figura 2.3– Modelo em espiral (adaptado de Pressman [1997]). .....	20
Figura 2.4 – Rational Unified Process (RUP) [Ambler 2000]. .....	22
Figura 2.5 – Ciclo de Deming. ....	27
Figura 2.6 - Factores de qualidade do software de McCall et al. [1997]. .....	34
Figura 2.7 - A estratégia GQM. ....	36
Figura 2.8 - Relacionamentos chave no modelo informação da medição (adaptado de ISO/IEC [2001]). .....	40
Figura 2.9 – Relacionamento entre as normas ISO/IEC 9126 e 14598 (adaptado de ISO/IEC [1999]). .....	41
Figura 2.10 – Qualidade no ciclo de vida do software (adaptado de ISO/IEC [1999]). ..	42
Figura 2.11 – Qualidade interna e externa na norma ISO/IEC 9126. ....	43
Figura 2.12 – Modelo de qualidade para a qualidade “em uso” [ISO/IEC 1999]. .....	44
Figura 2.13 – Processo de avaliação (adaptado de ISO/IEC [1998]). .....	45
Figura 3.1 – Visão holística dos processos de software. ....	48
Figura 3.2 – Processos do ciclo de vida do software: papéis, perspectivas e relacionamentos (adaptado de ISO/IEC [1995]). .....	50
Figura 3.3 – Relacionamento entre o processo e os seus resultados (baseado em SWEBOK [2001]). .....	51
Figura 3.4 – Referencial para o ambiente de melhoria do processo (adaptado de Zahran [1998]). .....	52
Figura 3.5 – Os cinco níveis de capacidade do CMM. ....	54

Figura 3.6– Estrutura interna do CMM [Paulk et al. 1995].	57
Figura 3.7 – Perfil de maturidade de avaliações de 1999 até Dez. 2003 [SEI 2004].	59
Figura 3.8 – Tendência no perfil de maturidade [SEI 2004].	60
Figura 3.9 – Componentes do Relatório Técnico ISO/IEC TR 15504: 1998.	62
Figura 3.10 – Componentes da norma ISO/IEC 15504.	63
Figura 3.11 – Relacionamento entre os elementos da ISO/IEC 15504.	65
Figura 3.12– Modelo de um sistema de gestão da qualidade baseado em processos.	69
Figura 3.13 – Arquitectura do modelo Trillium [Bell 1994].	72
Figura 3.14 – Passos na MPS (SPIRE [1998]).	77
Figura 3.15 – Correspondência entre a curva normal e o gráfico de controlo (adaptado de Fryman [2002]).	79
Figura 3.16 – Crescimento da população (adaptado de Richardson e Pugh [1981]).	82
Figura 3.17 – Evolução do processo no PSP (adaptado de Humphrey [1996]).	83
Figura 3.18 – Processo ágil (adaptado de Schwaber [2001]).	85
Figura 4.1 – Resultados quantitativos do estudo empírico (adaptado de Kolmel e Eisenbiegler [2000]).	92
Figura 5.1 – Conhecimento versus esforço (adaptado de Orr [2002]).	111
Figura 5.2 – Contexto do 2MPspe.	112
Figura 5.3 – 2MPspe versus modelos MPS analisados.	114
Figura 5.4 – Visão global do 2MPspe.	114
Figura 5.5 – Visão detalhada do 2MPspe.	116
Figura 6.1 – Modelo unidimensional proposto pelo CMM.	120
Figura 6.2 – Modelo bidimensional proposto pela norma ISO/IEC 15504.	120
Figura 6.3 – Modelo tridimensional proposto pelo 2MPspe.	121
Figura 6.4 – Enquadramento do referencial tridimensional na área de avaliação e melhoria do processo de software.	122
Figura 6.5 – Modelo Conceptual do 2MPspe.	123

Figura 6.6 – Organigrama da empresa XPTO1. ....	127
Figura 6.7 – Organigrama da empresa XPTO2. ....	128
Figura 6.8 – Modelo de Referência dos processos proposto pela norma ISO/IEC 12207: 1995 / Amd1 [ISO/IEC 2002b].....	131
Figura 6.9 – Modelo de Referência dos Processos proposto pelo CMM. ....	132
Figura 6.10 – Modelo de Referência dos Processo do BOOTSTRAP. ....	134
Figura 6.11 – Modelo de Referência dos Processo do 2MPspe.....	135
Figura 6.12 – Referencial de Qualidade do QMIM [Balla et al. 2001]. ....	143
Figura 6.13 – Referencial de medição [ISO/IEC 2004b].....	167
Figura 6.14 – Níveis de capacidade do 2MPspe (adaptado de SPIRE [1998]). ....	169
Figura 6.15 – Detalhe da dimensão do conhecimento. ....	179
Figura 6.16 – Referencial Conhecimento × Capacidade.....	180
Figura 6.17 – Exemplo do nível de conhecimento de um conjunto de processos. ....	182
Figura 6.18 – Modelo de gestão de conhecimento de Nonaka e Takeuchi [1995]. ....	184
Figura 6.19 – Referencial proposto pelo 2MPspe.....	192
Figura 6.20 – Cenário 1: Processos (P) × Capacidade (C).....	193
Figura 6.21 – Cenário2: Processos (P) × Conhecimento (K).....	194
Figura 6.22 – Cenário 3: Processo × Capacidade × Conhecimento. ....	194
Figura 6.23 – Cenário 3: Representação em duas dimensões.....	195
Figura 6.24 – Capacidade e conhecimento das categorias dos processos. ....	196
Figura 6.25 – Maturidade da Organização.....	196
Figura 7.1 – Enquadramento do $M_a2MP$ . ....	199
Figura 7.2 – Contexto da determinação da capacidade do processo (adaptado de [ISO/IEC 2004a]).....	220
Figura 7.3 – Representação dos resultados na perspectiva global.....	225
Figura 7.4 – Representação dos resultados na perspectiva detalhada.....	226
Figura 8.1 – Actividades no processo de avaliação (adaptado de ISO/IEC [2004b]). .	230

Figura 8.2 – Enquadramento do M <sub>i</sub> 2MP.....	231
Figura 8.3 – Fases genéricas do processo de avaliação (adaptado de Zahran [1998]).	232
Figura 8.4 – Abordagem IDEAL. ....	233
Figura 8.5 – Actividades do método CBA-IPI. ....	235
Figura 8.6 – Método de implementação do Tapistry.....	234
Figura 8.7 – Método de melhoria da ISO/IEC 15504.....	236
Figura 8.8 – Etapas do M <sub>i</sub> 2MP. ....	237
Figura 8.9 – Tarefas do M <sub>i</sub> 2MP.....	238
Figura 8.10 – Intervenção dos participantes no programa de melhoria.....	239
Figura 8.11 – Etapa 1: “Planear Iniciativa” .....	240
Figura 8.12 – Etapa 2: “Realizar Diagnóstico” .....	243
Figura 8.13 – Etapa 3: “Planear Acções de Melhoria” .....	245
Figura 8.14 – Desenvolvimento do Plano de Acção.....	248
Figura 8.15 – Exemplo do documento “Prioridades dos processos”.....	248
Figura 8.16 – Perfis alvo e avaliado dos processos. ....	249
Figura 8.18 – Prioridades teóricas do 2MPspe. ....	253
Figura 8.19 – Etapa 4: “Executar acções de melhoria” .....	254
Figura 8.20 – Diagrama de Gantt da iniciativa de Melhoria. ....	255

## Índice de tabelas

---

Tabela 2.1 - Relacionamento entre Objectivos / Princípios / Atributos. ....	38
Tabela 3.1 – Desafios chave / níveis de maturidade [Paulk et al. 1995]. ....	55
Tabela 4.1 – Benefícios da MPS [Herbsleb et al. 1994]. ....	91
Tabela 4.2 – Benefícios estimados da maturidade [Harter et al. 2000]. ....	94
Tabela 4.3 – Parâmetros dos projectos versus nível CMM [King e Diaz 2002]. ....	94
Tabela 4.4 – ROI obtido com a progressão do nível CMM [King e Diaz 2002]. ....	95
Tabela 6.1 – Correspondência dos roadmaps com a ISO 12207. ....	142
Tabela 6.2 – Tabela comparativa entre os MRPs seleccionados. ....	147
Tabela 6.3 – Níveis de capacidade e respectivos atributos da capacidade. ....	175
Tabela 6.4 – Níveis derivados das classificações dos ACs (adaptado de ISO/IEC 2004b). ....	178
Tabela 6.5 – Abordagem ao conhecimento e os processos de conhecimento (adaptado de Marr et al. [2003]). ....	186
Tabela 6.6 – Referencial dos valores em competição [Quinn e McGrath 1985]. ....	189
Tabela 6.7 – Níveis de conhecimento versus Atributos de conhecimento. ....	191
Tabela 7.1 – Detalhe do questionário 2MPspe. ....	222
Tabela 8.1 – Esforço de uma avaliação Rapid. ....	234
Tabela 8.2 – Exemplo da “Matriz de impactos” (adaptado de SPIRE [1998]). ....	241
Tabela 8.3 – Exemplo do documento “Capacidade alvo”. ....	242
Tabela 8.4 – Exemplo do documento “Prioridades dos processos”. ....	246
Tabela 8.5 – Lacunas nos atributos (adaptado de ISO/IEC [2004d]). ....	250
Tabela 8.6 – Risco associado às lacunas nos ACs. ....	250
Tabela 8.7 – Risco associado ao contexto. ....	251

## Siglas

---

Neste documento são utilizadas diversas siglas que representam abreviaturas de designações utilizadas. As siglas são:

2MPspe .....	Modelo de Melhoria do Processo de Software para Pequenas Empresas
AC .....	Atributo da Capacidade
AK .....	Atributo de Conhecimento
AP .....	Atributo do Processo
CE .....	Comunidade Europeia
CMM .....	<i>Capability Maturity Model for software, v1.1</i>
DSDM .....	<i>Dynamic Systems Development Method</i>
ESI .....	<i>European Software Institute</i>
FCS .....	Factores Críticos de Sucesso
GC .....	Gestão de Configurações
GQM .....	<i>Goal / Question / Metric</i>
GRH .....	Gestão de Recursos Humanos
IDEAL .....	Início, Diagnóstico, Estabelecimento, Agir, <i>Leveraging</i>
IEC .....	<i>International Electrotechnical Commission</i>
IEEE .....	<i>Institute of Electrical and Electronics Engineers</i>
IPM .....	Impulsionador interno do Processo de Melhoria
ISO .....	<i>International Organization for Standardization</i>
KPA .....	<i>Key Process Area</i> (Área Chave do Processo)
M <sub>a</sub> 2MP .....	Modelo de Avaliação do 2MPspe
M <sub>c</sub> 2MP .....	Modelo Conceptual do 2MPspe
M <sub>i</sub> 2MP .....	Método de Implementação do 2MPspe
MPS .....	Melhoria do Processo de Software
MRP .....	Modelo de Referência dos Processos
PDCA .....	<i>Plan / Do / Check / Act</i> (Planear / Executar / Verificar / Agir)
PES .....	Pequenas Empresas de desenvolvimento de Software



PIEs.....	<i>Process Improvement Experiments</i>
PMEs.....	Pequenas e Médias Empresas
PSP.....	<i>Personal Software Process</i>
QMIM.....	<i>Quality through Managed Improvement and Measurement</i>
RAD.....	<i>Rapid Application Development</i>
RAPID.....	<i>Rapid Assessment for Process Improvement for software Development</i>
ROI.....	Retorno dO Investimento ( <i>Return Of Investment</i> )
RUP.....	<i>Rational Unified Process</i>
SEI.....	<i>Software Engineering Institute</i>
SGQ.....	Sistemas de Gestão de Qualidade
SIG.....	Sistemas de Informação de Gestão
SPICE.....	<i>Software Process Improvement and Capability dEtermination</i>
SPIRE.....	<i>Software Process Improvement in Regions of Europe</i>
SQA.....	<i>Software Quality Assurance</i> (Garantia da Qualidade do Software)
SWEBOK.....	<i>Software Engineering Body of Knowledge</i>
TAPISTRY.....	<i>Tailored Application of Software Improvement Techniques for Small Enterprises</i>
TIs.....	Tecnologias de Informação
TQM.....	<i>Total Quality Management</i> (Gestão pela Qualidade Total)
XP.....	<i>eXtreme Programming</i>

# Capítulo 1

---

## 1. Introdução

Os sistemas de software são cada vez mais o centro de qualquer produto ou serviço moderno e, conseqüentemente é fundamental assegurar a sua qualidade. Os dados sobre o sucesso dos projectos de desenvolvimento de software não são “agradáveis”. Estudos realizados mostram que cerca de 15% dos projectos são cancelados antes do seu fim e 80% são considerados sem sucesso [Standish Group 1995], [Glass 1998] e [Lindberg 1999]. Este facto é ainda mais crítico devido ao fraco desempenho de alguns projectos com grande visibilidade nos media, por exemplo, Airbus A320 [Randell 1989] e Arienne 5 [Lions 1997].

Para melhorar a qualidade do software, as organizações têm “consciência” que é necessário gerir o processo de desenvolvimento de software de uma forma efectiva [Yamamura 1999], [Pitterman 2000] e [Jiang e al. 2004]. Neste contexto, a Melhoria do Processo de desenvolvimento do Software (MPS) sobressai como a abordagem mais utilizada [Conradi e Fuggetta 2002] e [Niazi et al. 2005].

A MPS tem origem no movimento Gestão pela Qualidade Total<sup>1</sup> e assume uma correlação positiva entre o processo utilizado no desenvolvimento do produto e a qualidade final do produto produzido [Deming 1982] e [Humphrey 1989]. No contexto da MPS, o paradigma *benchmarking* é o mais utilizado e envolve a medição da

---

<sup>1</sup> *Total Quality Management* – TQM.

maturidade de uma organização de software através da identificação de um conjunto de boas práticas. Como afirmaram Combelles e De Marco [1998], “os aspectos que a MPS levanta são importantes, se não mesmo vitais para o futuro das organizações de software”.

No paradigma *benchmarking*, os dois modelos mais divulgados são o CMM (Capability Maturity Model) [Paulk et al. 1995] desenvolvido pelo Software Engineering Institute (SEI) e a norma recentemente publicada ISO/IEC 15504<sup>2</sup>. Embora os modelos de MPS proliferem, há ainda muitas questões em aberto<sup>3</sup> [Fayad e Laitinen 1997], [Emmerich et al. 1999], [Evans 2002] e [Rainer e Hall 2002]. Conradi e Fuggetta [2002] afirmam mesmo que “é surpreendente que o número de iniciativas e projectos relacionados com métodos de MPS inovadores seja tão limitado”.

## 1.1. A Problemática

Todas as organizações que desenvolvem software têm um processo de desenvolvimento de software mais ou menos definido. Verifica-se contudo que na prática, os processos: não são cumpridos, não são alterados para acompanharem novas tecnologias ou necessidades organizacionais, são definidos externamente e são frequentemente incompletos [Evans 2002]. A MPS pretende então construir uma infra-estrutura e cultura que suporte métodos, práticas e metodologias efectivas [Aaen 2003]. Mas, como foi referido, a área da MPS tem ainda muitas questões em aberto, por exemplo:

“Quanta disciplina é suficiente, quanta flexibilidade é suficiente?”

[Boehm e Port 2001]

“A MPS é em si mesmo sobre mudança organizacional.”

[Rainer e Hall 2001]

---

<sup>2</sup> as partes 2, 3 e 4 foram publicadas no 1º semestre de 2004, a parte 1 encontra-se no esboço final (FDIS) e a parte 5 no esboço do comité internacional (CD).

<sup>3</sup> por exemplo o caso dos telemóveis: Motorola parcialmente no nível 5 CMM; Ericsson parcialmente no nível 3 CMM e Nokia no nível 1, mas a cota de mercado não reflecte esta maturidade.

“As empresas desenvolvem projectos, em média, com duração de 3 a 6 meses e incrementos de 3 a 6 semanas. Por outro lado, sabemos que demora 4 a 5 anos a passar do nível 1 para o nível 3 do CMM.”

[Conradi e Fuggetta 2002]

“A taxa de insucesso das iniciativas MPS está perto dos 70%.”

[Ngwenyama e Nielsen 2003]

Embora com preocupações mais abrangentes, a questão inicialmente colocada, no âmbito deste trabalho, referiu-se à seguinte problemática: há evidências significativas que mostram que uma iniciativa de melhoria do processo de desenvolvimento de software com sucesso resultará em melhorias na qualidade do produto e no valor do negócio, mas há também evidência que grande parte dos projectos de MPS falha.

Verifica-se também que os modelos de MPS mais divulgados surgiram ligados a grandes empresas e projectos complexos e por esta razão não são directamente aplicáveis a pequenas empresas de software [SPIRE 1998], [Orci 2000] e [Rout et al. 2000]. Contudo, o processo de desenvolvimento de software é tão crítico para a qualidade do produto final numa pequena empresa de software como é para uma grande empresa [Kuvaja et al. 1999], [Russ e McGregor 2000] e [Horvat et al. 2000].

O estudo realizado sobre a problemática permitiu concluir que:

- (1) As organizações que implementam e institucionalizam uma MPS com sucesso obtêm benefícios significativos na produtividade, *rework*, precisão nas estimativas e no ROI (Retorno do Investimento)
- (2) Há uma grande taxa de insucesso na aplicação dos modelos de melhoria do processo de desenvolvimento de software

## 1.2. Finalidade e Objectivos

De acordo com as conclusões referidas, foi definido o pressuposto da investigação. Um projecto de melhoria do processo de desenvolvimento do software com sucesso traz benefícios substanciais para a qualidade do produto e para o valor do negócio, contudo há uma grande parte das iniciativas que falha.

O trabalho efectuado sobre a problemática enunciada salientou dois factores importantes para o sucesso de uma iniciativa de MPS nas pequenas empresas de software, ou seja: (1) abordagem explícita à gestão do conhecimento [Gasston e Halloran 1999], [Mathiassen e Pourkomeylian 2003] e [Ravichandran e Rai 2003] e (2) gestão da cultura organizacional [Paulk et al. 1997], [Nielsen e Norbjerg 2001] e [Spence 2002]. Tomando em consideração estes factores e o pressuposto referido definiu-se a finalidade da investigação: desenvolver um modelo de MPS adaptado às pequenas empresas de software na área de sistemas de informação de gestão que trate explicitamente os factores contextuais da cultura organizacional e da gestão do conhecimento.

Para atingir o propósito deste trabalho definiu-se um conjunto de tarefas a executar, cada uma cumprindo um objectivo parcial e produzindo resultados identificados.

Em primeiro lugar pretendeu-se contextualizar a área da MPS e clarificar conceitos relacionados. Nesta perspectiva desenvolveu-se uma revisão da literatura nos domínios da engenharia de software e da qualidade de um produto genérico. Como resultado desta tarefa enquadrou-se a temática da investigação nas áreas de conhecimento do processo de engenharia de software e da qualidade do software [SWEBOK 2001].

O segundo objectivo consistiu em aprofundar a área específica deste trabalho, realizando uma revisão bibliográfica sobre os modelos de melhoria do processo de desenvolvimento de software. Com esta intenção analisaram-se os paradigmas *benchmarking* e analítico, procurando enquadrá-los no ambiente específico das

pequenas empresas de software. Como resultado, sistematizaram-se as principais abordagens à MPS e definiram-se os seus fundamentos.

O terceiro objectivo consistiu na validação do pressuposto da investigação. Para cumprir este objectivo foi realizada uma revisão bibliográfica sobre a problemática e foram desenvolvidos três casos práticos em pequenas empresas de software. Além desta validação, pretendeu-se identificar os factores críticos de sucesso de uma iniciativa de MPS. Esta identificação permitiu definir um conjunto de características alvo, a serem tratadas pelo 2MPspe (Modelo de Melhoria do Processo de Software para Pequenas Empresas de software).

O quarto objectivo está ligado à concepção da primeira versão do 2MPspe. Nesta perspectiva definiu-se uma arquitectura com três componentes, seguindo a proposta da norma ISO/IEC 15504-2 [ISO/IEC 2004b]. O modelo conceptual do 2MPspe ( $M_c2MP$ ) que caracteriza o referencial utilizado na avaliação e melhoria do processo de software. O modelo de avaliação ( $M_a2MP$ ), que define os indicadores que permitirão avaliar os atributos do processo. Finalmente, o método de implementação ( $M_i2MP$ ) que apresenta as actividades, técnicas e ferramentas a utilizar numa iniciativa de melhoria do processo de desenvolvimento de software. A ferramenta principal do modelo desenvolvido é o questionário 2MPspe, utilizado como suporte para a realização do diagnóstico da empresa de software.

O quinto objectivo prende-se com a validação do questionário 2MPspe. Para cumprir este objectivo foi realizada a revisão do questionário com a colaboração de um painel de *experts* do meio académico e do meio empresarial. A utilização de um painel de especialistas para revisão é comum na área de MPS [El Emam e Madhavji 1996] e [Dyba 2000]. Este processo incluiu duas iterações, resultando na versão 3 do questionário 2MPspe.

Finalmente, o sexto objectivo está ligado à evolução do 2MPspe e à promoção da utilização deste modelo nas pequenas empresas de software. Como vimos, mesmo para as pequenas empresas em ambientes emergentes, é fundamental a aposta na melhoria do processo de desenvolvimento de software. O 2MPspe trata especificamente

estes ambientes ao enquadrar no seu referencial os factores contextuais, gestão do conhecimento e gestão da cultura organizacional.

### 1.3. Metodologia de Investigação

Este trabalho de doutoramento insere-se na área de conhecimento da Engenharia da Programação e dos Sistemas Informáticos<sup>4</sup>. A finalidade da investigação é propor um novo método (o 2MPspe) para aplicar nas iniciativas de MPS nas pequenas empresas. Este método pretende resolver os problemas identificados e enunciados no pressuposto da investigação.

A utilização de uma metodologia de investigação servirá para orientar o trabalho de concepção do método e para demonstrar a validade do 2MPspe. Pretende-se então utilizar um modelo experimental para determinar a efectividade do 2MPspe<sup>5</sup>. No enquadramento deste trabalho foi seguida uma abordagem predominante de engenharia, isto é, desenvolver e testar uma solução para um determinado problema [Zelkowitz e Wallace 1998]. Na validação tecnológica, a experimentação permite a indução de teorias, uma vez que elas são formuladas através da observação [Tichy 1998].

Easterby-Smith et al. [1991] sugerem que é um bom ponto de partida para a concepção de um projecto de investigação analisar as duas filosofias extremas da investigação em gestão: positivismo e fenomenologia. A diferença fundamental é que o positivismo considera a realidade externa e independente dos desejos e interpretações do investigador. A fenomenologia, por outro lado, foca na dimensão humana da realidade.

Neste trabalho foi adoptada uma abordagem mista entre positivismo e fenomenologia.

---

<sup>4</sup> seguindo a categorização utilizada na UM.

<sup>5</sup> neste documento descreve-se a primeira versão do 2MPspe.

Segundo Zelkowitz e Wallace [1998], os modelos de validação na área de engenharia de software podem agrupar-se em três categorias genéricas:

- Baseados na observação. Recolha de dados relevantes através do acompanhamento de um projecto em execução.
- Históricos. Recolha de dados de projectos terminados.
- Controlados. Fornece múltiplas instâncias de uma observação para validação estatística dos resultados.

Os mesmos autores propõem uma taxonomia para a experimentação na engenharia de software, englobando 12 abordagens experimentais<sup>6</sup>: monitorização de projecto, estudo de caso, asserção, estudo de campo, pesquisa bibliográfica, legado, lições aprendidas, análise estática, replicação, ambiente sintético, análise dinâmica e simulação.

Para a orientação e validação deste projecto de investigação foram escolhidos quatro métodos experimentais, característicos da engenharia de software: a pesquisa bibliográfica da categoria “históricos”, a asserção e o estudo de casos da categoria “baseados na observação” e a experiência em ambiente sintético da categoria “controlados”.

A revisão bibliográfica requer que o investigador analise artigos e outros documentos publicados. Este método é útil para confirmar a hipótese e na perspectiva do enquadramento da área de investigação. Tem como principal vantagem a possibilidade de analisar um elevado número de estudos publicados. Em relação ao desenvolvimento do 2MPspe foi utilizado na revisão do estado da arte e na validação do pressuposto.

A asserção normalmente serve como base para a validação preliminar da tecnologia desenvolvida. Neste método experimental, o investigador é simultaneamente avaliador e responsável pelo desenvolvimento. A asserção não requer a satisfação de normas científicas rigorosas.

---

<sup>6</sup> os autores referem que não é uma lista final.



No estudo de casos, o investigador acompanha o projecto e recolhe dados. É considerado um método activo devido à influência que o observador pode ter no próprio processo de desenvolvimento.

Finalmente, as experiências em ambiente sintético, replica a utilização do método num ambiente artificial. O objectivo deste método experimental é avaliar um conjunto de variáveis pré-definido. No caso do 2MPspe, o questionário foi preenchido por um painel de *experts* da área empresarial e da área académica. Este processo teve duas iterações e resultou na versão 3 do questionário. Considerou-se não ser viável a recolha de uma amostra significativa, optando-se pela opinião de peritos.

#### **1.4. Organização da Tese**

A estrutura da tese reflecte a forma como o trabalho foi desenvolvido para atingir os objectivos apresentados. Assim, este primeiro capítulo tem início com uma breve descrição do enquadramento desta tese. Na segunda secção define-se a problemática que originou o projecto e seguidamente enuncia-se o propósito e as contribuições da investigação. Finalmente, na última secção, apresenta-se a abordagem metodológica que orientou o trabalho e permitiu validar os resultados.

O capítulo 2 apresenta o enquadramento da área de melhoria do processo de desenvolvimento de software no âmbito da engenharia de software. Este trabalho de investigação insere-se nas áreas de conhecimento: do processo de engenharia de software e da qualidade do software [SWEBOK 2001]. Descrevem-se as diferentes perspectivas relacionadas com o processo de desenvolvimento de software e apresentam-se alguns modelos.

O capítulo 3 apresenta a descrição detalhada dos modelos de MPS, com especial incidência no paradigma *benchmarking*. São descritos os modelos mais divulgados e os trabalhos centrados no contexto das pequenas empresas de software.

O capítulo 4 faz a validação do pressuposto descrevendo a revisão bibliográfica realizada e apresentando os casos práticos desenvolvidos em pequenas empresas de desenvolvimento de software.

O capítulo 5 baseia-se nos dados das secções anteriores e sistematiza a problemática, o pressuposto e o propósito da investigação. São descritas as características desejáveis para o 2MPspe. Em seguida é apresentada a descrição da arquitectura proposta para o 2MPspe e é desenvolvida uma descrição global de cada componente do modelo.

No capítulo 6 descreve-se o modelo conceptual do 2MPspe, designado  $M_c2MP$ . O  $M_c2MP$  define um referencial tridimensional constituído pela dimensão dos processos, da capacidade e do conhecimento. O capítulo descreve cada uma das dimensões e termina com a descrição da representação no sistema de eixos tridimensional.

O capítulo 7 apresenta o modelo de avaliação do 2MPspe ( $M_a2MP$ ). O  $M_a2MP$  fornece o conjunto de indicadores a partir dos quais são avaliados os atributos dos processos. O capítulo termina com a descrição do questionário 2MPspe, utilizado na realização do diagnóstico dos processos da organização.

O capítulo 8 define o método de implementação do 2MPspe, designado  $M_i2MP$ . Este método apresenta as actividades, técnicas e ferramentas a utilizar numa iniciativa de melhoria do processo de desenvolvimento de software.

Finalmente, no capítulo 9, são apresentadas as conclusões. Inicialmente desenvolve-se uma síntese do trabalho realizado e apresentam-se os resultados e contribuições principais do trabalho de investigação. Em seguida apresentam-se propostas de trabalho futuro. O capítulo termina com o desenvolvimento de algumas considerações finais.

Este documento inclui ainda quatro anexos com informação complementar.

## Capítulo 2

---

## 2. Engenharia de Software

Neste capítulo faz-se a contextualização da área de melhoria do processo de desenvolvimento de software no âmbito da engenharia de software. Em relação à classificação das áreas de conhecimento propostas pelo SWEBOK<sup>1</sup> [2001] será feita uma revisão de conceitos e definições sobre o processo da engenharia de software e sobre a qualidade do software. Com foco nestas duas áreas de conhecimento, apresentam-se as diferentes perspectivas do processo de software e alguns modelos propostos. Finalmente, define-se a qualidade do processo e relaciona-se esta abordagem com a Gestão pela Qualidade Total<sup>2</sup>.

### 2.1. Conceitos e Definições

Não se pode falar em processo de software<sup>3</sup> sem definir e analisar o seu relacionamento com a engenharia de software. A definição de engenharia de software e o próprio termo, suscitam alguma controvérsia na comunidade informática. Considera-se contudo que o desenvolvimento de software é uma disciplina de engenharia pela sua natureza inerente, embora o não seja pela sua prática corrente [Brooks 1987] e [McDermid 1991]. É consensual que a engenharia de software é realizada por pessoal

---

<sup>1</sup> SWEBOK – *Software Engineering Body of Knowledge*.

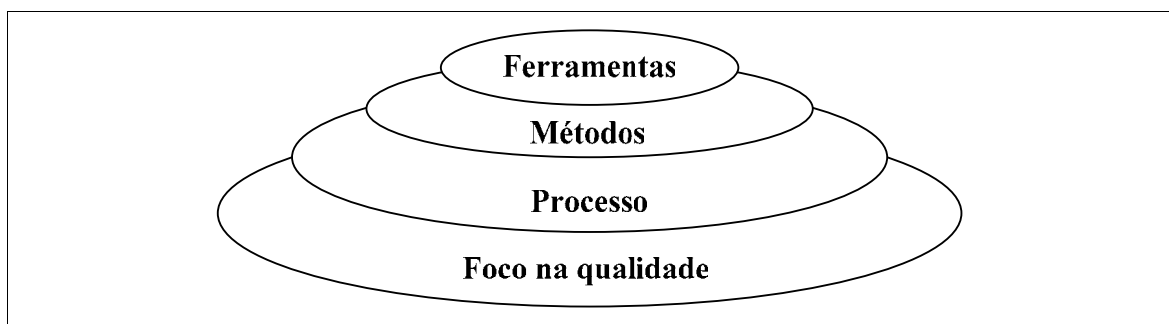
<sup>2</sup> TQM – *Total Quality Management*.

<sup>3</sup> neste relatório utiliza-se “processo de software” para referir o “processo de desenvolvimento de software” seguindo a expressão inglesa “*software process*”.

técnico e criativo que deve trabalhar dentro de um processo de software maduro e definido.

Para Pfleeger [1998] “a engenharia de software é a aplicação dos computadores como ferramentas para resolver problemas”. Esta definição obtém-se pela análise de outras disciplinas de engenharia, isto é, considera-se as ciências de computação responsáveis pela investigação e definição de teorias e ferramentas que serão aplicadas pela engenharia de software para resolver problemas. Se considerarmos as características específicas do software, analisado como um qualquer produto e a definição de engenharia, pode-se definir engenharia de software como o estabelecimento e utilização de princípios de engenharia “seguros” para obter software de forma económica, que seja fiável e trabalhe com eficiência em máquinas reais.

Pode considerar-se a engenharia de software como uma tecnologia em camadas (ver Figura 2.1). Qualquer abordagem de engenharia (incluindo a engenharia de software) tem que se apoiar na obrigação organizacional para a qualidade. A Gestão pela Qualidade Total e outras filosofias similares criam uma cultura de melhoria contínua do processo. Os alicerces da engenharia de software são o foco na qualidade [Pressman 1997].



*Figura 2.1– Camadas da engenharia de software [Pressman 1997]*

A fundação para a engenharia de software é a camada do processo, ou seja, é a cola que segura as camadas tecnológicas e permite o desenvolvimento de software de uma forma racional e cumprindo prazos. O processo define a estrutura para um conjunto de áreas chave do processo que tem que ser estabelecida para uma utilização efectiva da tecnologia da engenharia de software. As áreas chave do processo formam a base para o controlo da gestão de projectos de software e estabelecem o contexto em que os

métodos técnicos são aplicados, os produtos do trabalho são produzidos, os pontos de verificação são estabelecidos, a qualidade é assegurada e as alterações são convenientemente geridas [Paulk 1995].

Os métodos da engenharia de software fornecem os “como” técnicos para construir software. As ferramentas de engenharia de software fornecem suporte automático ou semi-automático para o processo e para os métodos [Pressman 1997].

No conjunto de normas do IEEE, a engenharia de software é definida como:

“(1) A aplicação de uma abordagem sistemática, disciplinada e quantificável ao desenvolvimento, operação e manutenção do software, isto é, a aplicação da engenharia ao software.

(2) O estudo de abordagens tal como em (1).”

[IEEE 1991]

Da definição anterior ressalta o facto de que o trabalho relacionado com a engenharia de software pode ser categorizado genericamente em três fases [Redmill 1997]:

1. Fase de Definição
2. Fase de Desenvolvimento
3. Fase da Manutenção

A primeira fase refere-se à determinação e representação do sistema pretendido pelo cliente e necessita de ferramentas de representação simples, de modo a facilitar o diálogo com o cliente. O resultado desta primeira fase deve ser uma descrição completa do sistema e do seu ambiente, isto é, dos requisitos do cliente. A segunda fase centra-se no “como” e refere-se à concepção, construção e teste da solução. Utilizando a definição de Taylor [1959], referindo-se ao desenho de um produto genérico, “é o processo de aplicar várias técnicas e princípios com o objectivo de definir um dispositivo, um processo ou um sistema com detalhe suficiente para permitir a sua realização física”. Ainda na fase do desenvolvimento, em seguida, os documentos da

concepção são traduzidos<sup>4</sup> para uma linguagem de programação e finalmente o produto é verificado e validado.

Finalmente, a terceira fase é a actividade de manutenção necessária em qualquer produto, definindo-se como o conjunto de actividades necessárias para manter e melhorar o produto, depois deste ser aceite. No caso do software, normalmente considera-se dividida em quatro tipos: correctiva, perfectiva, adaptativa e preventiva.

O SWEBOK [2001] pretende<sup>5</sup>, por um lado, clarificar as fronteiras da engenharia de software em relação a outras disciplinas e caracterizar os conteúdos da disciplina de engenharia de software. O referido documento está organizado em dez áreas de conhecimento: requisitos do software, concepção do software, construção do software, teste do software, manutenção do software, gestão de configurações do software, gestão da engenharia de software, processo de engenharia de software, métodos e ferramentas de engenharia de software e qualidade do software. Este guia apresenta sete disciplinas que se relacionam com a engenharia de software, a saber: factores humanos e ciências cognitivas; engenharia de computadores; ciências da computação; gestão e ciências da gestão; matemática; gestão de projectos e engenharia de sistemas.

Considerou-se importante caracterizar sucintamente cada uma das áreas de conhecimento propostas pelo SWEBOK [2001] numa perspectiva de contextualização deste trabalho.

A área dos requisitos do software trata da aquisição, análise, especificação, validação e gestão dos requisitos do software. Um requisito pode definir-se como uma propriedade que tem que ser exibida pelo produto para resolver um problema do mundo real [Pfleeger 1998] e [Sommerville 2001]. A concepção do software trata da “definição da arquitectura, componentes, interfaces e outras características do sistema ou componente” [IEEE 1991]. A área do conhecimento de construção do software trata da obtenção de um software a trabalhar e com significado, através da combinação das

---

<sup>4</sup> no caso do desenvolvimento de software.

<sup>5</sup> entre outros objectivos.

tarefas de codificação, validação e teste<sup>6</sup>. O teste do software consiste “na verificação dinâmica do comportamento do programa, num conjunto finito de casos de teste, em relação ao comportamento esperado especificado” [SWEBOOK 2001]. A área de conhecimento de manutenção do software trata das modificações realizadas num produto de software depois da sua entrega. Engloba alterações para corrigir erros, para melhorar o desempenho ou outros atributos e para adaptar a novos ambientes. A gestão de configurações refere-se às actividades de “identificação e documentação das características físicas e funcionais dos itens de configuração, controlo das alterações a essas características, registar e relatar o processamento de alterações e o estado da implementação e verificar a conformidade com os requisitos especificados” [IEEE 1991]. A área de gestão da engenharia do software trata da gestão do desenvolvimento do software e da sua medição e modelação. O processo de engenharia de software preocupa-se com a definição, implementação, medição, gestão, alteração e melhoria do processo. A área de conhecimento dos métodos e ferramentas da engenharia de software trata das ferramentas automáticas utilizadas para apoiar o processo de desenvolvimento e também métodos que cubram mais do que uma fase do processo. Finalmente, a área de conhecimento da qualidade do software trata os aspectos da qualidade que transcendem os processos do ciclo de vida.

Em relação a esta categorização das áreas de conhecimento da engenharia de software, este trabalho centra-se, principalmente, nas áreas do processo da engenharia de software e da qualidade do software.

Para o estabelecimento de uma qualquer disciplina é necessário definir um conjunto de princípios fundamentais reconhecidos universalmente. Neste aspecto, a disciplina de engenharia de software, apresenta algumas carências reconhecidas. O trabalho<sup>7</sup> da equipa liderada por Bourque [Bourque e al. 2002], na definição dos princípios fundamentais é um marco importante.

---

<sup>6</sup> teste unitário.

<sup>7</sup> contribuições de autores de renome, por exemplo, Fred Brooks, Roger Pressman, Carlo Ghezzi, Steve McConnell, Shari Pfleeger, Robert Charrette e outros.

A lista de princípios fundamentais resultante de dois estudos Delphi e de um *survey* a membros do IEEE é a seguinte [Bourque e al. 2002]:

- Aplicar e utilizar medições quantitativas na tomada de decisão
- Construir com e para a reutilização
- Controlar a complexidade com perspectivas múltiplas e vários níveis de abstracção
- Definir os artefactos de software rigorosamente
- Estabelecer um processo de software que forneça flexibilidade
- Implementar uma abordagem disciplinada e melhorá-la continuamente
- Investir na compreensão do problema
- Gerir a qualidade formalmente durante o ciclo de vida
- Minimizar a interacção entre as componentes de software
- Produzir o software de uma forma incremental
- Estabelecer objectivos de qualidade para cada produto entregue
- Uma vez que a mudança é inerente ao software, planeá-la e geri-la
- Uma vez que *tradeoffs* são inerentes à engenharia de software, torná-los explícitos e documentá-los
- Para melhorar a concepção, estudar soluções anteriores para problemas similares
- A incerteza é inevitável na engenharia de software. É necessário identificá-la e geri-la

## 2.2. Processo de Engenharia de Software

As definições de “processo” são variadas cobrindo várias perspectivas, como por exemplo:

“Uma série de etapas no fabrico ou noutra operação”

(Concise Oxford Dictionary)



“Uma organização lógica de pessoas, materiais, energia, equipamento e procedimentos em actividades de trabalho desenhadas para produzir um resultado específico”

[Pall 1987]

“Uma sequência de passos realizados com determinado objectivo, por exemplo, o processo de desenvolvimento de software.”

[IEEE 1991]

“Um conjunto de recursos e actividades inter-relacionadas que transformam entradas em saídas.

Nota: recursos podem incluir pessoas, equipamento, tecnologia e metodologia.”

[ISO/IEC 1994]

As definições anteriores variam no seu foco. Dão mais ênfase a certos aspectos e ignoram outros. A definição seguinte é uma perspectiva holística da definição do processo. Um processo é mais do que apenas um conjunto de procedimentos documentados. Um ambiente de processo efectivo tem três aspectos essenciais [Zahran 1998]:

- **Definição do processo.** Normalmente toma a forma de um documento (papel ou electrónico) que especifica as actividades e procedimentos para o processo.
- **Aprendizagem do processo.** Transferência do conhecimento do processo para aqueles que o executam.
- **Monitorização e cumprimento do processo.** Refere-se aos resultados do processo que se manifestam pelos produtos produzidos como resultado da execução das actividades do processo.

Mas afinal porquê começar a pensar no processo? Vejamos a opinião de alguns autores. Para Hammer e Champy [1993] “muitas pessoas e organizações não estão orientadas aos processos: estão focadas nas tarefas, nos trabalhos, nas estruturas mas não nos processos. O pensamento baseado em tarefas – a fragmentação do trabalho nos seus componentes mais simples e as suas atribuições aos trabalhadores especializados – influenciou o desenho organizacional das empresas nos últimos duzentos anos”.

Segundo Harrington [1991] “temos que parar de pensar na estrutura organizacional e focarmos nos processos que controlam os interfaces com os clientes. Ocorre um padrão de pensamento completamente diferente quando focamos a nossa ênfase no processo”. Finalmente, Humphrey [1989] afirma que “um primeiro passo importante para tratar dos problemas do software é considerar a tarefa global do desenvolvimento do software como um processo que pode ser controlado, medido e melhorado”.

O objectivo deve ser chegar a um processo maduro, isto é, um processo definido, treinado, cumprido e seguido. Watts Humphrey acrescenta que um processo maduro é continuamente melhorado.

Os processos de engenharia de software são definidos de diferentes maneiras por razões diversas, por exemplo [Curtis et al. 1992] e [Huff 1996]:

- Facilitar a compreensão e comunicação humana, sendo necessário que a equipa partilhe um formato de representação comum
- Suportar o aperfeiçoamento do processo, o que requer uma base para definir e analisar os processos
- Suportar a gestão do processo, o que requer um processo definido a partir do qual possa ser comparado o comportamento dos projectos actuais
- Definir e suportar as tarefas necessárias à realização do processo
- Definir suportes automáticos para determinadas tarefas

### **2.2.1. Modelos do Referencial do Ciclo de Vida<sup>8</sup>**

Os modelos do referencial do ciclo de vida apresentam uma definição de alto nível das fases que ocorrem durante o desenvolvimento. Não apresentam definições detalhadas das tarefas, mas antes uma perspectiva de alto nível das actividades e dos relacionamentos entre as actividades. Nestas actividades estarão englobadas as três fases genéricas da engenharia de software, isto é, definição, desenvolvimento e manutenção.

---

<sup>8</sup> denominação proposta no SWEBOK [2001].

Estes modelos são uma tentativa para colocar ordem numa actividade caótica por inerência. Raccoon [1995] sugere mesmo um “Modelo do Caos”<sup>9</sup> para descrever “o desenvolvimento de software como contínuo do utilizador para o profissional de desenvolvimento e para a tecnologia”. Há diversos modelos propostos na literatura, por exemplo [Moneymaker 1995] e [Pressman 1997], modelo em cascata, prototipagem, modelo incremental, *Rapid Application Development* (RAD).

Apresenta-se em seguida uma breve descrição de três modelos do referencial do ciclo de vida: modelo em cascata, modelo em espiral e o RUP<sup>10</sup>. O modelo em cascata foi seleccionado pela sua importância histórica<sup>11</sup>, o modelo em espiral foi escolhido pelo seu impacto no desenvolvimento de software e, finalmente o RUP é descrito devido à sua actualidade e aceitação na comunidade de software.

#### 2.2.1.1. Modelo em Cascata

O modelo em cascata (*waterfall*) ou modelo convencional do ciclo de vida de desenvolvimento de software (ver figura 2.2), foi apresentado por Royce [1970], mas é conhecido principalmente com a estrutura proposta por Boehm [1976]. É o modelo do processo de desenvolvimento mais difundido e historicamente o primeiro modelo consensual do processo de desenvolvimento de software.

Cada fase do modelo tem entradas e saídas bem definidas e as saídas de uma fase são frequentemente utilizadas como entradas da sucessora. A progressão de uma fase para a seguinte é apenas permitida depois de determinados critérios serem satisfeitos. Embora seja o modelo do processo mais utilizado é normalmente modificado pelas organizações que o utilizam para compensar as suas fraquezas.

---

<sup>9</sup> as metodologias ágeis reconhecem também a inerência do caos na engenharia de software.

<sup>10</sup> *Rational Unified Process*.

<sup>11</sup> foi a primeira tentativa de trazer ordem ao desenvolvimento de software.

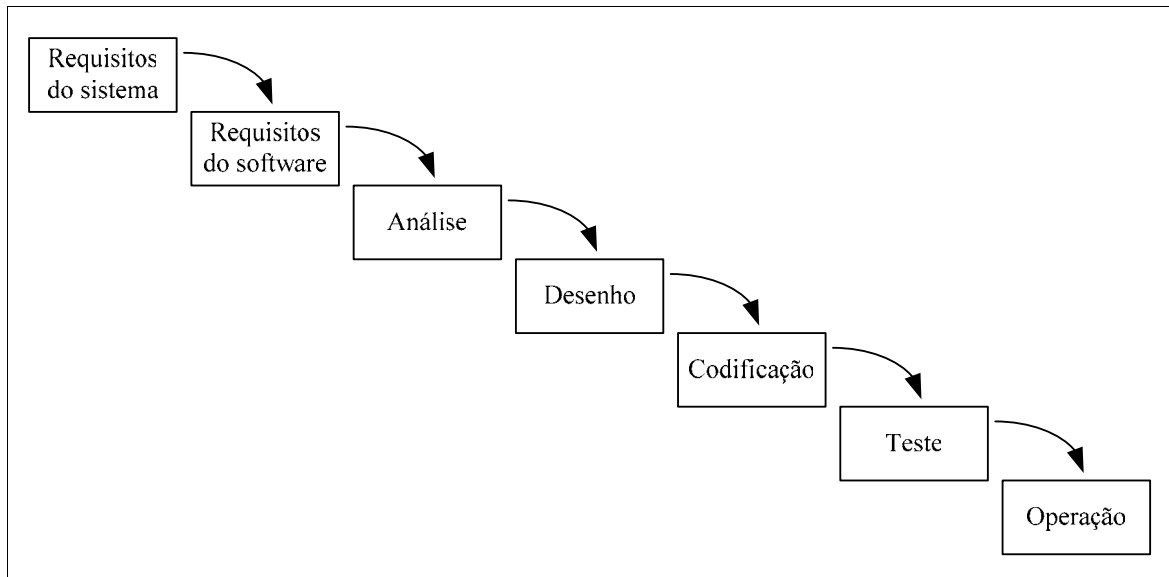


Figura 2.2 – Modelo em cascata do processo de desenvolvimento de software [Royce 1970].

A maior vantagem da utilização deste modelo situa-se nas áreas de organização e controlo do projecto de software [Sage e Palmer 1990]. É apropriado quando há uma definição correcta, completa e consistente dos requisitos. Este modelo apresenta também outros aspectos importantes [Davis et al. 1988], nomeadamente:

- Encoraja a concepção do software antes da sua implementação
- Permite aos gestores do projecto um acompanhamento preciso dos seus progressos
- Aconselha a criação de um conjunto de documentos ao longo de todo o processo que serão utilizados nas etapas finais
- Permite uma melhor estruturação e gestão de todo o processo de desenvolvimento

Apesar da sua importância e validade, o modelo em cascata apresenta algumas dificuldades [Humphrey 1989] e [Moneymaker 1995]:

- Não acompanha adequadamente as mudanças
- Assume uma sequência relativamente uniforme e ordenada de etapas de desenvolvimento
- Não permite uma entrega mais cedo de funcionalidades mais críticas para o cliente

### 2.2.1.2. Modelo em Espiral

Actualmente o modelo em espiral, apresentado por Boehm [1988] é o modelo do processo de desenvolvimento de software mais “rico”, uma vez que capta as melhores características do modelo em cascata, da prototipagem e acrescenta-lhe a análise do risco [Money maker 1995] e [Pressman 1997].

O modelo em espiral divide-se num número de actividades de estrutura, também chamadas regiões de tarefas. Tipicamente existem entre três a seis regiões de tarefas. A figura 2.3 apresenta um modelo em espiral contendo cinco regiões de tarefas.

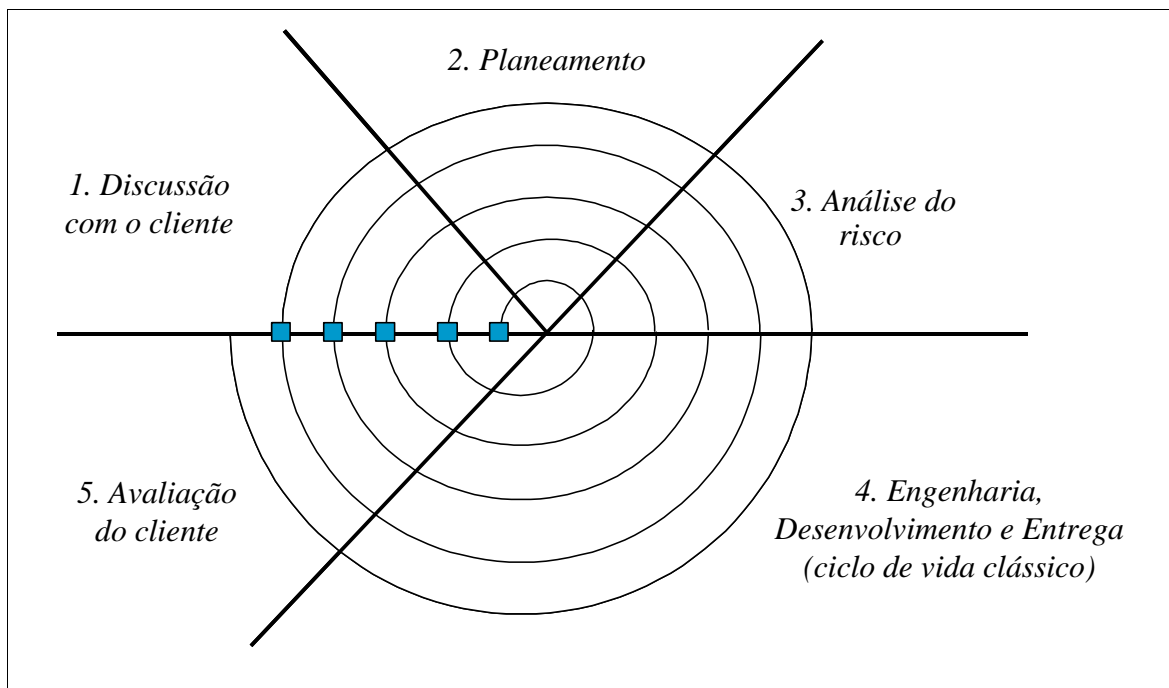


Figura 2.3– Modelo em espiral (adaptado de Pressman [1997]).

A primeira região refere-se às tarefas necessárias para o estabelecimento de uma comunicação efectiva entre a equipa de desenvolvimento e o cliente. As tarefas de planeamento relacionam-se com a definição de recursos, tempos e outra informação relacionada com o projecto. As actividades da região três centram-se na avaliação tanto dos riscos técnicos como de gestão. As tarefas de engenharia são as actividades necessárias para a construção da aplicação e finalmente, a última região refere-se à obtenção de *feedback* do cliente.

Cada região possui uma série de tarefas de trabalho que devem ser adaptadas às características específicas do projecto a ser desenvolvido. O modelo em espiral é uma abordagem realista ao desenvolvimento de sistemas e software em grande escala. Uma vez que o software evolui conforme o processo progride, torna-se possível à equipa de desenvolvimento e ao cliente perceberem e reagirem de forma adequada ao risco em cada nível de evolução.

O modelo em espiral apresenta como vantagens principais: a utilização das melhores características dos outros modelos do processo; considerar explicitamente a análise do risco e adaptação a projectos em que os requisitos não são bem conhecidos no início. As dificuldades principais deste modelo referem-se à necessidade de experiência na gestão de riscos e à tendência de não completar o ciclo, resultando em faltas importantes na documentação [Moneymaker 1995].

### **2.2.1.3. Rational Unified Process (RUP)**

O *Rational Unified Process* (ver figura 2.4) é um processo iterativo. Para sistemas simples é adequado utilizar uma abordagem sequencial para a resolução do problema. Contudo os sistemas actuais são normalmente complexos. A abordagem iterativa defendida pelo RUP, incentiva uma compreensão crescente do problema através de refinamentos sucessivos e um crescimento incremental de uma solução efectiva através de ciclos múltiplos. Incorporada na abordagem iterativa está a flexibilidade de acomodar novos requisitos ou “mudanças táticas” nos objectivos do negócio.

O desenvolvimento com o RUP é centrado na arquitectura. O processo centra-se no desenvolvimento inicial de uma arquitectura de base do software. A existência de uma arquitectura robusta “no lugar”, facilita o desenvolvimento paralelo, minimiza o *rework* e aumenta a probabilidade de reutilização de componentes. O RUP suporta as técnicas orientadas aos objectos e é um processo configurável. Verifica-se também que o RUP encoraja um controlo da qualidade e uma gestão de riscos progressiva e objectiva [Booch et al. 1999].

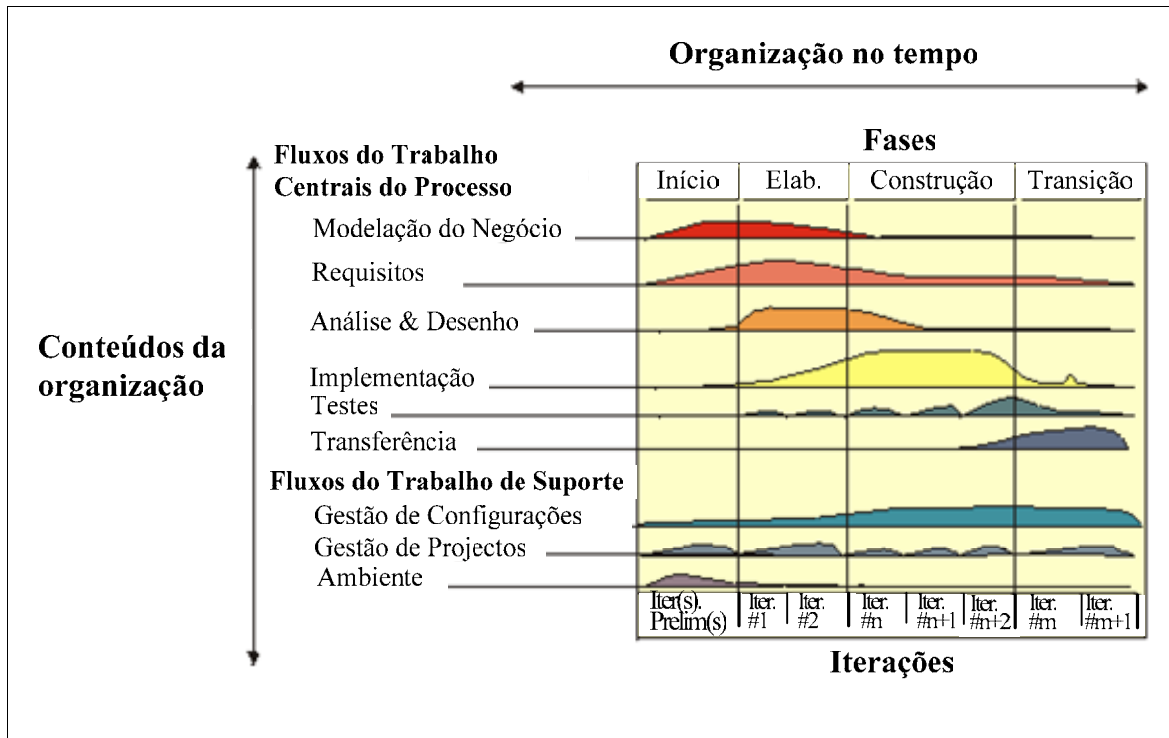


Figura 2.4 – Rational Unified Process (RUP) [Ambler 2000].

O RUP possui quatro fases em série e nove fluxos de trabalho “centrais”. Os ciclos de desenvolvimento são organizados no que o RUP chama iterações, permitindo reduzir o risco do projecto através da melhoria da comunicação entre a equipa de desenvolvimento e os clientes.

No RUP, uma fase é o espaço de tempo entre dois *milestones* principais do processo, em que um conjunto bem definido de objectivos é atingido, os artefactos correspondentes são terminados e são tomadas as decisões sobre a continuação para a fase seguinte. As quatro fases do RUP são:

- **Início:** Durante esta fase é estabelecido o caso de negócio do sistema e é delimitado o âmbito do projecto. O caso de negócio inclui critérios de sucesso, avaliação dos riscos, estimativas dos recursos necessários e um plano das fases mostrando o escalonamento dos *milestones* principais.
- **Elaboração:** Os objectivos da fase de elaboração são analisar o domínio do problema, estabelecer uma fundação “segura” para a arquitectura, desenvolver o plano do projecto e eliminar os elementos de maior risco do projecto.
- **Construção:** Durante esta fase, desenvolve-se iterativamente e incrementalmente um produto completo, pronto para a transição para a

comunidade de utilizadores. Isto implica a descrição dos requisitos em falta e o critério de aceitação, desenvolver a concepção e completar a implementação e teste do software.

- **Transição:** Durante a fase de transição o software é entregue à comunidade de utilizadores. Quando o sistema é passado aos utilizadores finais surgem sempre alguns aspectos que requerem um desenvolvimento adicional para ajustar o sistema, corrigir alguns erros não detectados ou terminar algumas características adiadas.

A passagem através das quatro fases é chamada um ciclo de desenvolvimento, resultando numa geração de software. A primeira passagem através das quatro fases é chamada o ciclo de desenvolvimento inicial. A não ser que a “vida” do produto termine, o software existente evoluirá para a próxima geração repetindo a mesma sequência de fases: início, elaboração, construção e transição.

O RUP consiste em nove processos de fluxos de trabalho [Booch et al. 1999]:

1. Modelação do negócio. Descreve a estrutura e dinâmica da organização.
2. Requisitos. Descreve o método baseado em casos de uso que permitirá elicitar os requisitos.
3. Análise e concepção. Descreve múltiplas vistas da arquitectura.
4. Implementação. Toma em consideração o desenvolvimento do software, o teste da unidade e a integração.
5. Teste. Descreve os casos de teste, procedimentos e métricas de acompanhamento de defeitos.
6. Transferência. Cobre a configuração do sistema entregue.
7. Gestão de configurações. Controla as alterações e mantém a integridade dos artefactos do projecto.
8. Gestão do projecto. Descreve várias estratégias de trabalho num processo iterativo.
9. Ambiente. Cobre as infra-estruturas necessárias para o desenvolvimento do sistema.

Com cada processo do fluxo de trabalho desenvolve-se um conjunto de actividades e artefactos relacionados. Define-se um artefacto como um documento,



relatório ou executável, produzido, manipulado ou consumido. Uma actividade descreve as tarefas – passos do pensamento, passos para a execução e passos de revisão – executadas pela equipa de desenvolvimento para criar ou modificar os artefactos, junto com as técnicas e orientações para executar as tarefas, possivelmente incluindo a utilização de ferramentas para ajudar a automatizar algumas tarefas.

As principais vantagens do RUP são: base em princípios “seguros” de engenharia de software, tais como, abordagem iterativa, conduzida pelos requisitos e baseada na arquitectura; fornece vários mecanismos, por exemplo, o protótipo no fim de cada iteração e o ponto de decisão sobre prosseguir ou não no fim de cada fase que dão visibilidade à gestão sobre o processo de desenvolvimento; a *Rational* tem feito um grande esforço no seu produto de suporte ao RUP.

Como principais desvantagens do RUP podem apontar-se [Ambler 2000]:

- É apenas um processo do desenvolvimento, isto é, não cobre o processo de software “completo”. Falta-lhe os conceitos de manutenção e suporte.
- Não suporta explicitamente uma estrutura multi-projecto do esforço do desenvolvimento, tal como, a modelação da arquitectura global da organização, perdendo a oportunidade de reutilização em grande escala.
- É actualmente fraco em áreas como a gestão de métricas, gestão da reutilização, gestão dos recursos humanos e testes.

Esta secção foi dedicada à descrição dos modelos do referencial do ciclo de vida principais, bem como uma apresentação dos pontos fortes e fracos de cada modelo do processo. Na próxima secção faz-se a “ponte” entre o processo do software e a gestão da qualidade, procurando-se as raízes para a melhoria contínua do processo.

## 2.3. Qualidade

Nesta secção faz-se uma breve apresentação sobre a qualidade, os seus conceitos fundamentais e a abordagem *Total Quality Management*<sup>12</sup> (TQM). O objectivo é analisar os “alicerces” dos modelos de melhoria do processo de software e nesta perspectiva descrevem-se as principais escolas de qualidade. No fim da secção faz-se a ligação para a qualidade do produto software, definindo e formalizando os conceitos e áreas principais.

### 2.3.1. Conceitos sobre Qualidade

A qualidade é um termo subjectivo que muitas vezes tem significados diferentes, para pessoas diferentes em situações diversas. Segundo o dicionário Houaiss<sup>13</sup>, qualidade é “propriedade que determina a essência ou a natureza de um ser ou coisa”. Os dicionários Webster’s e Oxford definem qualidade como o “grau de excelência”. Verificando-se ainda um conjunto de definições alternativas, vindas da indústria e dos meios académicos: “aptidão para uso”, “conformidade com os requisitos” ou “ausência de falhas no produto”. A *International Organization for Standardization* (ISO) define a qualidade de um dado produto como sendo “a totalidade das características (do produto ou serviço) que sustentam a capacidade de satisfazer as necessidades especificadas ou implícitas” [ISO/IEC 1994] ou mais recentemente, “o grau com que um conjunto de características inerentes ao produto satisfaz os requisitos” [ISO/IEC 2000b].

Associada à definição de qualidade há um conjunto de conceitos independentes do tipo de produto desenvolvido<sup>14</sup>, tais como: gestão da qualidade, manual da qualidade, sistema de gestão da qualidade, política da qualidade, garantia da qualidade. Em relação à qualidade, considera-se a definição proposta por Broh [1982] que inclui a perspectiva

---

<sup>12</sup> Gestão pela Qualidade Total.

<sup>13</sup> Dicionário Houaiss da Língua Portuguesa (2003), elaborado no Instituto de Antônio Houaiss de Lexicografia Portugal

<sup>14</sup> utilizam-se as categorias propostas pela norma ISO 9001:2000, que apresenta quatro categorias genéricas de produtos, nomeadamente, hardware, software, serviços e materiais processados.

do cliente e do produto, “qualidade é o grau de excelência a um preço aceitável e o controlo da variabilidade a um custo suportável”.

Naturalmente para satisfazer clientes cada vez mais exigentes, as organizações sentem necessidade de implementar um sistema de gestão da qualidade, de forma a controlar e acompanhar a qualidade de todas as fases do processo de produção e provar a conformidade aos clientes finais. O sistema de gestão da qualidade é documentado no manual da qualidade, que por sua vez é detalhado em tarefas e métodos específicos através dos procedimentos de qualidade.

As tarefas de garantia da qualidade referem-se à área da gestão da qualidade que demonstra que os requisitos de qualidade foram cumpridos. Numa perspectiva mais operacional<sup>15</sup> a garantia da qualidade abrange as técnicas necessárias à satisfação dos requisitos da qualidade durante o ciclo produtivo.

Em relação às características do produto, denomina-se defeito a um estado de não conformidade com os requisitos do cliente. As estratégias de tratamento dos defeitos podem ser várias, nomeadamente, a detecção, que consiste na identificação e correcção de um produto não conforme<sup>16</sup>; a prevenção, que pretende detectar e corrigir o defeito antes da sua produção<sup>17</sup>; a inspecção, conjunto de actividades de medição, análise, observação e teste às características do produto e do processo, com o objectivo de comparar os resultados com os objectivos de qualidade especificados e definir o grau de conformidade e a auditoria, “exame sistemático e independente para determinar se as actividades e os resultados relativos à qualidade satisfazem as disposições pré-estabelecidas” [ISO/IEC 1994].

---

<sup>15</sup> acções no processo de engenharia do produto.

<sup>16</sup> estratégia reactiva.

<sup>17</sup> estratégia proactiva.

### 2.3.2. Total Quality Management (TQM)

TQM é uma abordagem muito popular, ligada normalmente à introdução das preocupações da qualidade numa organização. O TQM ou gestão pela qualidade total, tem a sua origem na transferência da abordagem japonesa à qualidade para o ocidente. Podemos pensar a gestão pela qualidade total como sendo “um conjunto coeso de ferramentas de gestão, filosofias de gestão e métodos de melhoria, incluindo: orientação ao cliente, *empowerment* dos funcionários, gestão participativa, decisões baseadas em dados, melhoria contínua, orientação aos processos e um conjunto de ferramentas quantitativas para a melhoria do processo” [Klein 1991].

No mundo da “qualidade”, destacam-se normalmente as contribuições de Edwards Deming, Joseph Juran e Philip Crosby, como as personalidades fulcrais para a definição e divulgação da ciência da qualidade.

#### 2.3.2.1. Edwards Deming

É considerado o iniciador do movimento da qualidade, através dos seus trabalhos na indústria japonesa nos anos 50. O ciclo de Deming (ver figura 2.5) ou ciclo PDCA (Plan / Do / Check / Act), permite modelar o processo de melhoria contínua da qualidade, tendo como alicerce uma repetição sistemática realimentada.

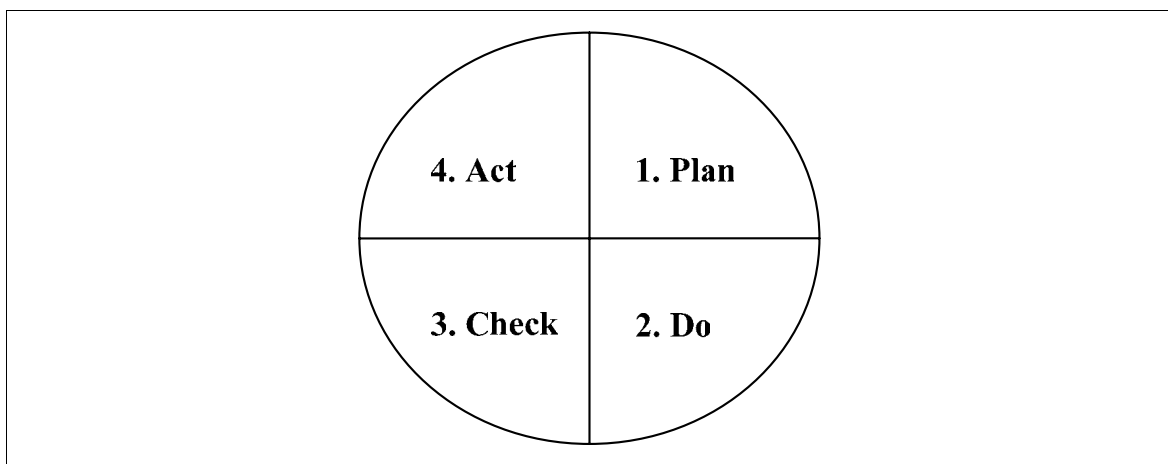


Figura 2.5 – Ciclo de Deming.

O processo de melhoria contínua, proposto por Deming, baseia-se em quatro etapas:

1. **Planear.** Esta etapa consiste na decisão e definição sobre a iniciativa mais benéfica para a melhoria da qualidade.
2. **Executar.** Consiste na implementação dos planos elaborados na etapa anterior.
3. **Verificar.** Medição dos resultados para verificar se os objectivos propostos foram alcançados.
4. **Agir.** Esta etapa engloba o estudo e aprendizagem do que foi feito, numa perspectiva de desenvolver conhecimento e transferir esse conhecimento para a obtenção de um produto / serviço melhor.

A filosofia de Deming[1982] pode ser descrita através dos seus catorze pontos, denominados obrigações da gestão. Estes pontos são apresentados sucintamente em seguida: criar constância de propósitos para a melhoria de produtos e serviços; adoptar uma nova filosofia que permita ultrapassar problemas crónicos<sup>18</sup> das organizações; deixar de depender das inspecções para atingir a qualidade; terminar com a prática de negociar com base no preço; melhorar constantemente o sistema de produção e serviço<sup>19</sup>; desenvolver a formação no local de trabalho; institucionalizar métodos modernos de liderança; eliminar o medo do local de trabalho, potenciando o debate de novas ideias e a apresentação de problemas e preocupações; remover as barreiras entre os departamentos; eliminar slogans; eliminar cotas numéricas; eliminar as barreiras que embaraçam os colaboradores; institucionalizar um programa constante de formação e auto-desenvolvimento e finalmente colocar todas as pessoas da organização a trabalhar para cumprir a transformação.

Um elemento chave da abordagem de Deming é o controlo estatístico do processo, como forma de melhorar a qualidade. Os passos são: inicialmente estabilizar o processo para colocá-lo sobre controlo estatístico e depois diminuir a sua variação. Verifica-se que um processo controlado estatisticamente tem uma determinada capacidade, isto é, o seu desempenho<sup>20</sup> é previsível.

---

<sup>18</sup> atrasos significativos, orçamentos ultrapassados, fraca qualidade dos produtos e serviços e fraco nível de serviço.

<sup>19</sup> através do estudo e compreensão detalhada dos processos.

<sup>20</sup> custo, qualidade e produtividade.

Este paradigma é construído com base em princípios e não em modelos ou métodos, verificando-se a necessidade de contratar profissionais com experiência e aptidões para a sua aplicação. Esta constatação deve-se ao facto das técnicas não estarem detalhadas<sup>21</sup> e à sua forte base em princípios que embora compreensíveis, exigem a participação de especialistas.

### **2.3.2.2. Joseph Juran**

Juran define qualidade como sendo “adaptação para o uso” e toda a sua abordagem gira em torno deste requisito universal dos produtos / serviços. As suas ideias são bem apresentadas através do conceito de “trilogia da qualidade” [Juran 1988], em que o autor apresenta os pilares da qualidade como sendo, o planeamento da qualidade, o controlo da qualidade e a melhoria da qualidade.

- Planeamento da qualidade
  - Identificar o cliente
  - Determinar os requisitos do cliente
  - Desenvolver um produto que satisfaça o cliente
  - Desenvolver os processos
  - Determinar o controlo dos processos
- Controlo da qualidade
  - Avaliar o desempenho real
  - Comparar o desempenho real com os objectivos de qualidade
  - Agir (melhorar o processo) com base nos desvios
- Melhoria da qualidade
  - Mostrar a necessidade de melhorias
  - Realizar o diagnóstico das causas
  - Desenvolver soluções
  - Provar a efectividade das soluções
  - Estabelecer o controlo necessário para manter os ganhos

---

<sup>21</sup> apenas o controlo estatístico do processo é descrito.

Com o objectivo de minimizar o custo da qualidade, Juran defendeu o desenvolvimento de um sistema contabilístico capaz de acompanhar / registar os custos da qualidade. Nesta perspectiva dividiu os custos da qualidade em custos de falhas internas, custos de falhas externas<sup>22</sup>, custos de avaliação<sup>23</sup> e custos de prevenção.

Em síntese, a qualidade para Juran obriga a que toda a organização esteja fortemente envolvida nesta filosofia global.

### 2.3.2.3. Philip Crosby

Este especialista da qualidade é mundialmente conhecido pelos seus livros “Quality without tears” e “Quality is free” e defende que a produção de bens e serviços com qualidade não traz custos adicionais, mas antes, os grandes custos estão associados a produtos com fraca qualidade. Crosby defende também a utilização sistemática de técnicas de prevenção em contraposição às técnicas reactivas<sup>24</sup>, mais difundidas.

Crosby desenvolveu um conjunto de técnicas para a construção da qualidade [Crosby 1979]. Neste contexto destaca-se claramente a “grelha de maturidade da gestão” que apresenta áreas de potenciais melhorias e consiste em cinco níveis:

1. Incerteza – não há compreensão sobre a importância da qualidade como uma ferramenta de gestão.
2. Despertar – constatação da importância da qualidade.
3. *Enlightenment* – estabelecimento do programa de qualidade.
4. *Wisdom* – utilização e estabelecimento das técnicas de prevenção.
5. Certeza – melhoria contínua e poucos problemas em relação à qualidade.

É de destacar a definição de uma nova atitude para a gestão, centrada na melhoria contínua e no estabelecimento de *standards* elevados de desempenho para atingir o ponto filosófico de zero defeitos.

---

<sup>22</sup> custos verificados depois dos produtos saírem da organização.

<sup>23</sup> custos associados com a procura das falhas antes dos produtos saírem da organização.

<sup>24</sup> por exemplo, a inspecção.

Como síntese desta secção, pode-se afirmar que o TQM envolve o conjunto global dos intervenientes no processo, desde os funcionários, aos clientes, fornecedores e qualquer colaborador, baseando-se numa estratégia obrigatória de melhoria contínua. Externamente, pretende-se satisfazer os níveis de qualidade e preços esperados pelo cliente e as exigências do mercado e internamente pretende-se minimizar os custos através de uma organização eficiente e eficaz.

Pode-se então definir TQM como “uma forma de melhorar continuamente o desempenho em todos os níveis de operação, em todas as áreas funcionais de uma organização, utilizando os recursos humanos e financeiros disponíveis. A melhoria é procurada através da satisfação de metas globais, tais como, custo, qualidade, quota de mercado, tempos e crescimento.” [Brocka e Brocka 1992]. Na definição anterior é de referir o carácter abrangente da melhoria, que vai muito além da mera produção dos bens, pretendendo alcançar os diversos aspectos do negócio.

Na próxima secção retrata-se os aspectos específicos da qualidade quando aplicada ao software, não só na perspectiva dos atributos do produto, mas também, no próprio processo de desenvolvimento.

### **2.3.3. Modelos de Qualidade do Software**

Como vimos na secção anterior, a qualidade pode definir-se como “a totalidade dos aspectos e características de um produto ou serviço que suportam a sua capacidade de satisfazer as necessidades explícitas e implícitas” [ISO/IEC 1994]. Na perspectiva do desenvolvimento do software a definição proposta por Pressman [1997] é mais específica, descrevendo a qualidade do software como “a conformidade com os requisitos de desempenho, os requisitos funcionais explicitamente declarados, as normas de desenvolvimento explicitamente documentadas e finalmente as características implícitas esperadas em todo o software desenvolvido de uma forma profissional”.



A definição anterior aponta três aspectos importantes na qualidade do software, a saber: (1) os requisitos explícitos são a “origem incontroversa” da medição das características de qualidade do produto final; (2) as normas documentadas permitem definir orientações e critérios de validação de todo o desenvolvimento, permitindo avaliar o desempenho do processo e (3) os requisitos não explicitamente declarados têm um papel muitas vezes esquecido na satisfação do cliente e por isso devem também entrar na medição das características de qualidade do produto.

Nesta breve discussão sobre qualidade do software interessa também ponderar as diferentes perspectivas dos grupos profissionais [Nance e Arthur 2002]. Para o engenheiro de software<sup>25</sup>, as características de qualidade referem-se normalmente a atributos associados a componentes individuais do software, por exemplo, a independência entre os módulos ou a coesão do código. Para o gestor do projecto, as características de qualidade a atingir centram-se em objectivos no nível do projecto, por exemplo, facilidade de manutenção ou correcção. Se pensarmos nas características de qualidade específicas dos utilizadores finais, verificamos que se referem a aspectos como a facilidade de utilização ou a facilidade de configuração.

É importante apontar as preocupações actuais de alguns autores conceituados no que se refere à qualidade do software.

“As organizações de software devem observar e analisar os seus produtos e processos de desenvolvimento. Devem sintetizar e “empacotar” o que aprenderam de modo a compreender o que tem que ser feito para melhorar os seus esforços de desenvolvimento de software...”

[Basili 2004]

“... Finalmente movemo-nos para o nível 3 de percepção interna da qualidade em que todos os interessados fornecem representações CRACK (Colaborativas,

---

<sup>25</sup> profissional que irá executar as tarefas de engenharia do produto.

Representativas, Autorizadas, Comprometidas e Conhecidas<sup>26</sup>) para negociar um conjunto de factores de qualidade satisfatórios e em que todos ganham...”

[Boehm 2004]

“... Qualidade é nem mais nem menos do que satisfazer as necessidades do cliente (estejam ou não os requisitos devidamente documentados) ...”

[Davis 2004]

“Se estamos sérios sobre a qualidade, eu recomendo que você:

- Faça com que os profissionais de desenvolvimento definam a forma correcta de realizar cada trabalho...
- Requeira planos detalhados e realistas do trabalho...
- Faça a medição e gestão da qualidade...
- Estabeleça como meta organizacional que cada produto tenha melhor qualidade que o anterior
- Planeie para fazer trabalho de qualidade.”

[Humphrey 2004]

“As organizações têm que parar de seguir as últimas modas. A qualidade é atingida incorporando-a desde o início, utilizando métodos e processos adaptados ao contexto de desenvolvimento e da aplicação e tornando claro aos profissionais do desenvolvimento que a qualidade tem prioridade máxima.”

[Leveson 2004]

“Temos que tratar os requisitos de qualidade como requisitos de primeira classe”

[Mead 2004]

“Resolvamos manter toda a documentação da concepção do programa, completa, precisa e actualizada”

[Parnas 2004]

---

<sup>26</sup> *Knowledgeable.*

“Perguntar a nós mesmos porque utilizamos um determinado conjunto de práticas de qualidade... Procurar evidências internas, matemáticas e baseadas em opiniões de especialistas, sobre as nossas práticas de qualidade e depois ver o corpo global de evidência e avaliar a credibilidade de cada parte”

[Pfleeger 2004]

Há diversos modelos de qualidade do software que propõem um conjunto de características ou atributos de qualidade, incluindo naturalmente as escalas para medir o grau com que o software os atinge.

### 2.3.3.1. Factores de Qualidade de McCall

McCall e os seus colegas [McCall et al. 1977] definiram um conjunto de factores que afectam a qualidade do software. Apresentaram três tipos de características de qualidade que focam na capacidade do produto sofrer alterações, na facilidade de se adaptar a novos ambientes e também nas suas características operacionais (ver figura 2.6).

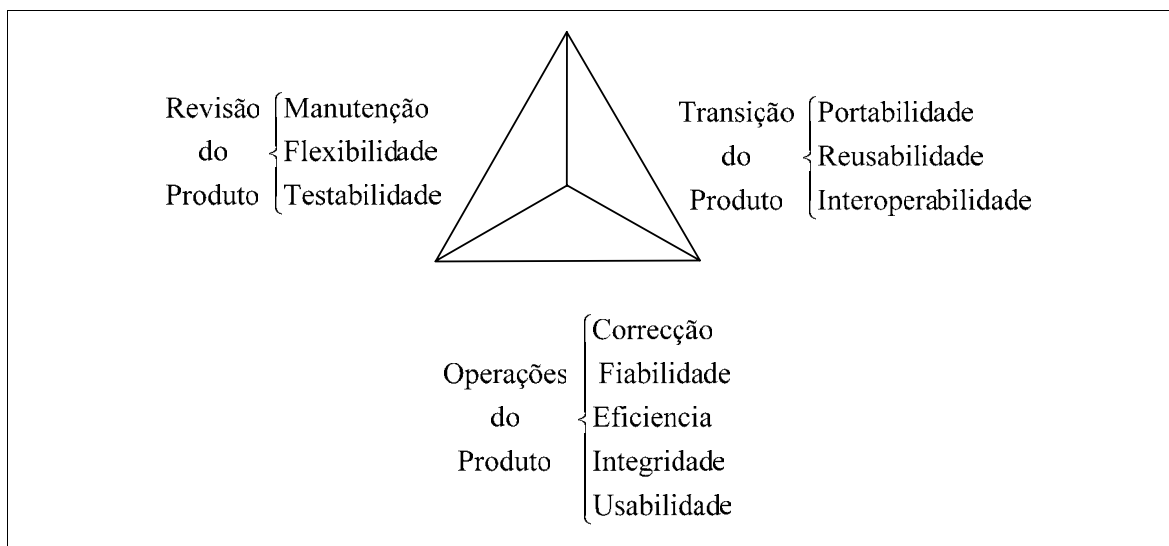


Figura 2.6 - Factores de qualidade do software de McCall et al. [1997]

Este modelo baseia-se na identificação dos factores de qualidade, que descrevem a perspectiva externa do software, tal como é percebida pelos utilizadores. Discute o

relacionamento destes factores com as características internas do produto (critérios de construção) e apresenta as métricas necessárias para definir um método de medição.

Uma vez que é difícil medir directamente os factores de qualidade, utiliza-se uma medição indirecta baseada num conjunto de métricas e na expressão

$$F_q = c_1 \times m_1 + c_2 \times m_2 \dots c_n \times m_n$$

em que  $F_q$  é o factor de qualidade,  $c_n$  são coeficientes de regressão e  $m_n$  são as métricas que afectam o factor de qualidade.

Infelizmente, muitas das métricas propostas por McCall apenas podem ser medidas de uma forma subjectiva e é o próprio McCall [Cavano e McCall 1978] que discute esta situação “... A subjectividade e a especialização também se aplicam à determinação da qualidade do software bem como uma forma de derivar medições quantitativas da qualidade do software para uma análise objectiva...”

### **2.3.3.2. Goal / Question / Metric [Basili e Rombach 1988]**

É possível medir muitas características dos produtos, projectos e processos do software. A dificuldade, muitas vezes, refere-se à utilização de métricas realmente úteis para os profissionais de software [Wiegers 1999a]. O GQM<sup>27</sup> é uma estratégia que se baseia na utilização de metas para a medição de produtos e processos do software.

Para Rombach e Ulery [1989], o GQM é útil porque facilita a identificação, não apenas das métricas objectivas necessárias, mas também das razões porque os dados estão a ser recolhidos. O “porquê” é importante uma vez que formaliza o modo como as medições devem ser interpretadas e permite a reutilização dos planos e métodos de medição em projectos futuros.

---

<sup>27</sup> Goal / Question / Metric.

O GQM permite uma definição *top-down* das métricas, a partir das metas do “mundo real” e uma análise *bottom-up* das medições recolhidas (ver figura 2.7)

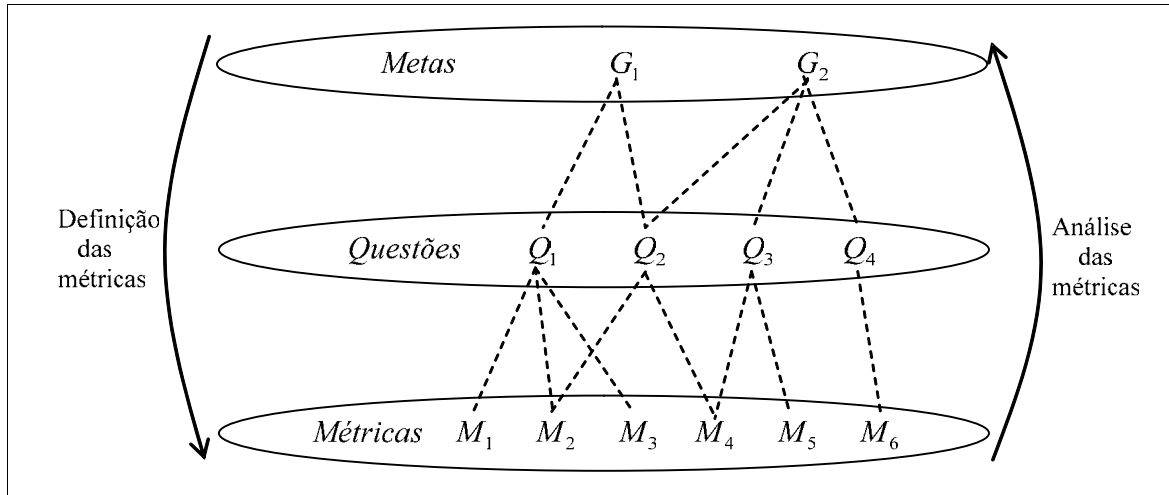


Figura 2.7 - A estratégia GQM.

O GQM inicia-se com a selecção de um conjunto de metas do nível organizacional ou de projecto. Estas metas devem ser preferencialmente descritas de forma quantitativa e mensurável. Por exemplo:

- G1: Dentro de um ano reduzir a duração do teste em 10%, mantendo o nível de defeitos.
- G2: Aumentar a taxa de reutilização de componentes em 30%, dentro de 3 meses.
- G3: Diminuir o custo da manutenção adaptativa em 15% no próximo projecto.

O passo seguinte consistirá na definição de um conjunto de questões que emergem de cada meta e que devem permitir saber se estamos a atingir a meta ou não. Se a nossa meta fosse, por exemplo, G1 “Dentro de um ano reduzir a duração dos testes em 10%, mantendo o nível de defeitos”, um conjunto de questões apropriadas poderia ser:

- Q1: Qual é a duração média dos testes?
- Q2: Quanto tempo é gasto nos testes dos componentes, nos testes de integração e nos testes finais?

Q3: Qual é o nível de defeitos actuais?

Finalmente, serão identificadas as métricas que emergem de cada questão e que devem permitir responder de forma quantitativa e objectiva à questão. No caso da questão 1, “Qual é a duração média dos testes?”, utilizaríamos a métrica M1, “duração média dos testes”, que seria calculada, de forma indirecta, através da soma das métricas: M2, “duração média dos testes dos componentes”; M3, “duração média dos testes de integração” e M4, “duração média dos testes finais”.

Refira-se também que a importância do GQM, ultrapassa o facto de ser um dos primeiros trabalhos a tentar definir um conjunto de métricas úteis para a avaliação da qualidade do software. A sua integração no Paradigma de Melhoria da Qualidade<sup>28</sup> como motor e suporte do programa de melhoria faz da abordagem GQM um marco histórico na área de melhoria do processo do software.

O Paradigma de Melhoria da Qualidade baseia-se em três passos:

- Executar uma auditoria a um projecto e seu ambiente de modo a definir objectivos de qualidade úteis
- Acompanhar o projecto e recolher dados relacionados com os objectivos de qualidade
- Analisar os dados recolhidos e desenvolver um conjunto de recomendações relacionadas com melhorias futuras

O GQM foi aplicado com sucesso em diversas empresas [Daskalantonakis 1992] e [Broekers et al. 1996] e como resultado da definição e utilização de métricas significativas verificou-se uma melhoria na compreensão da qualidade do produto e no próprio processo de desenvolvimento.

---

<sup>28</sup> QIP – Quality Improvement Program.

### 2.3.3.3. O Referencial Objectivos / Princípios / Atributos [Nance e Arthur 2002]

Esta abordagem apresentada por Arthur e Nance [1990] desenvolve as métricas para o produto através da ligação entre objectivos, definidos no nível do projecto, princípios sugeridos pela engenharia de software e atributos desejáveis no produto final (ver tabela 2.1).

Nível	Designação	Instâncias
<b>Projecto</b>	Objectivos	Facilidade de manutenção Correcção Reutilização Facilidade de Teste Fiabilidade Portabilidade Adaptabilidade
<b>Processo</b>	Princípios	Decomposição Hierarquia Decomposição Funcional <i>Information hiding</i> Refinamento passo a passo Programação Estruturada Verificação do ciclo de vida Documentação Concorrente
<b>Produto</b>	Atributos	Acoplamento Reduzido Coesão Melhorada Complexidade reduzida Interfaces bem definidos Fiabilidade Facilidade de Mudança Ligação Visibilidade do comportamento Detecção precoce de erros

Tabela 2.1 - Relacionamento entre Objectivos / Princípios / Atributos.

Espera-se então que através da medição dos atributos do produto seja possível avaliar o processo de desenvolvimento e assim verificar os objectivos do projecto.

Esta abordagem difere da estratégia proposta por McCall<sup>29</sup> e do GQM<sup>30</sup> uma vez que todas as métricas propostas estão ligadas a objectivos no nível do projecto, através dos princípios que orientam o processo de desenvolvimento. Através dos pares atributo/propriedade e da ligação entre atributos e princípios para objectivos, este referencial suporta uma abordagem sistemática e bem definida para examinar a qualidade do produto e do processo [Nance e Arthur 2002].

#### **2.3.3.4. ISO/IEC FCD 15939: Engenharia de Software - Referencial para o Processo de Medição do Software**

Esta norma identifica as actividades e tarefas necessárias para gerir com sucesso o processo de medição do software. Não pretende catalogar nem propor métricas para a qualidade do software. Propõe quatro actividades principais: (1) Estabelecer e sustentar os compromissos necessários para o programa de medições; (2) Planear o processo de medição; (3) Executar o processo de medição e (4) Avaliar as medições. Cada uma das tarefas referidas é composta por um conjunto de actividades especificadas pela norma.

A norma ISO/IEC [2001] apresenta também o modelo de informação da medição (ver figura 2.8), que é uma estrutura que liga as necessidades de informação com as entidades relevantes e os atributos de interesse. Este modelo descreve o modo como os atributos são quantificados e convertidos para métricas que darão o suporte necessário para o processo de tomada de decisão.

Na figura seguinte, a entidade é um objecto (por exemplo, processo, produto, projecto ou recurso) que se pretende caracterizar medindo os seus atributos. Um indicador é um valor quantificado que permite avaliar um determinado atributo e será a base para a análise do conceito mensurável.

---

<sup>29</sup> secção 2.3.3.1.

<sup>30</sup> secção 2.3.3.2.



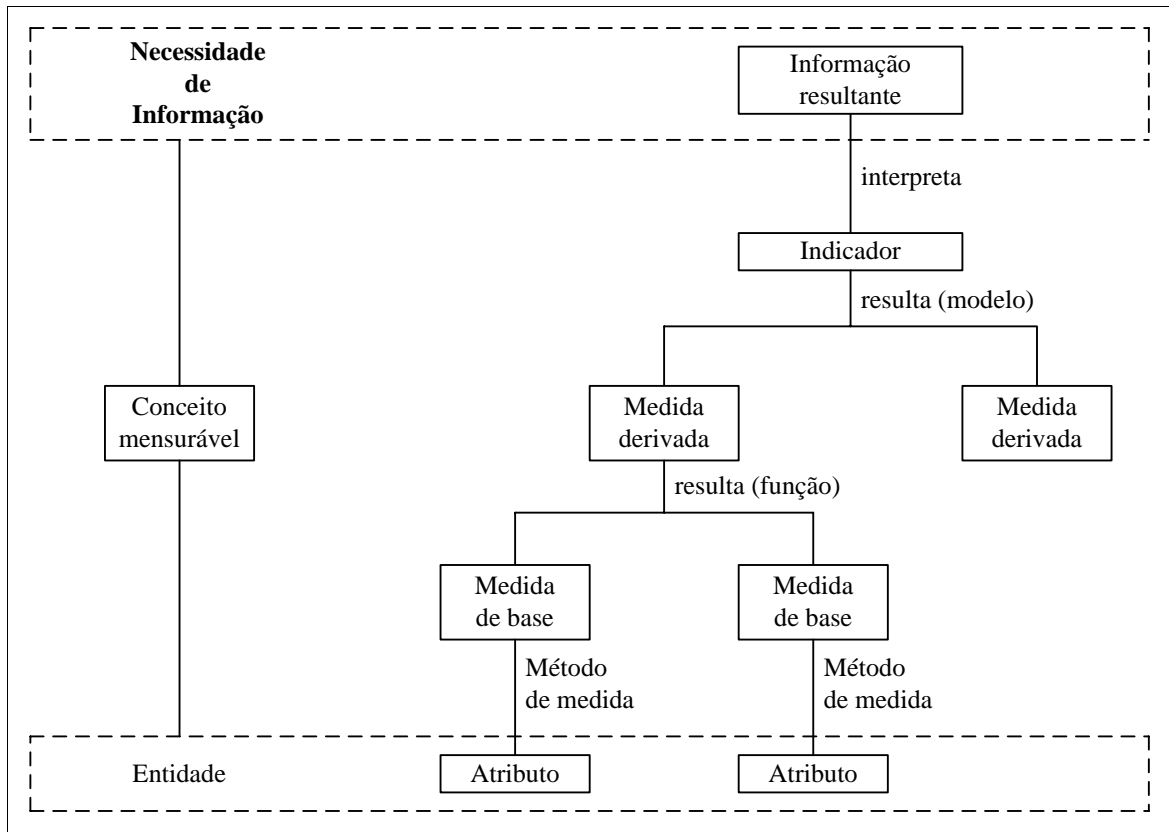


Figura 2.8 - Relacionamentos chave no modelo informação da medição (adaptado de ISO/IEC [2001]).

### 2.3.3.5. ISO/IEC 9126 e ISO/IEC 14598

A avaliação do produto software é um factor considerado chave para assegurar uma qualidade adequada do produto final. Considera-se que a avaliação só pode ser efectiva se for utilizado um modelo de qualidade e simultaneamente for seguido um processo de avaliação bem definido e estruturado. Nesta perspectiva a ISO procedeu à separação da norma ISO/IEC 9126: 1991, em duas normas, ISO/IEC 9126: 2001 e ISO/IEC 14598: 1999. A família ISO/IEC 9126, organizada em 4 partes, especifica um modelo de qualidade do produto software. A família ISO/IEC 14598, organizada em 6 partes, especifica o processo de avaliação, encontrando-se dividida segundo as perspectivas da avaliação (profissionais do desenvolvimento, comprador e avaliador independente). Na figura 2.9 ilustra-se o relacionamento entre as duas normas.

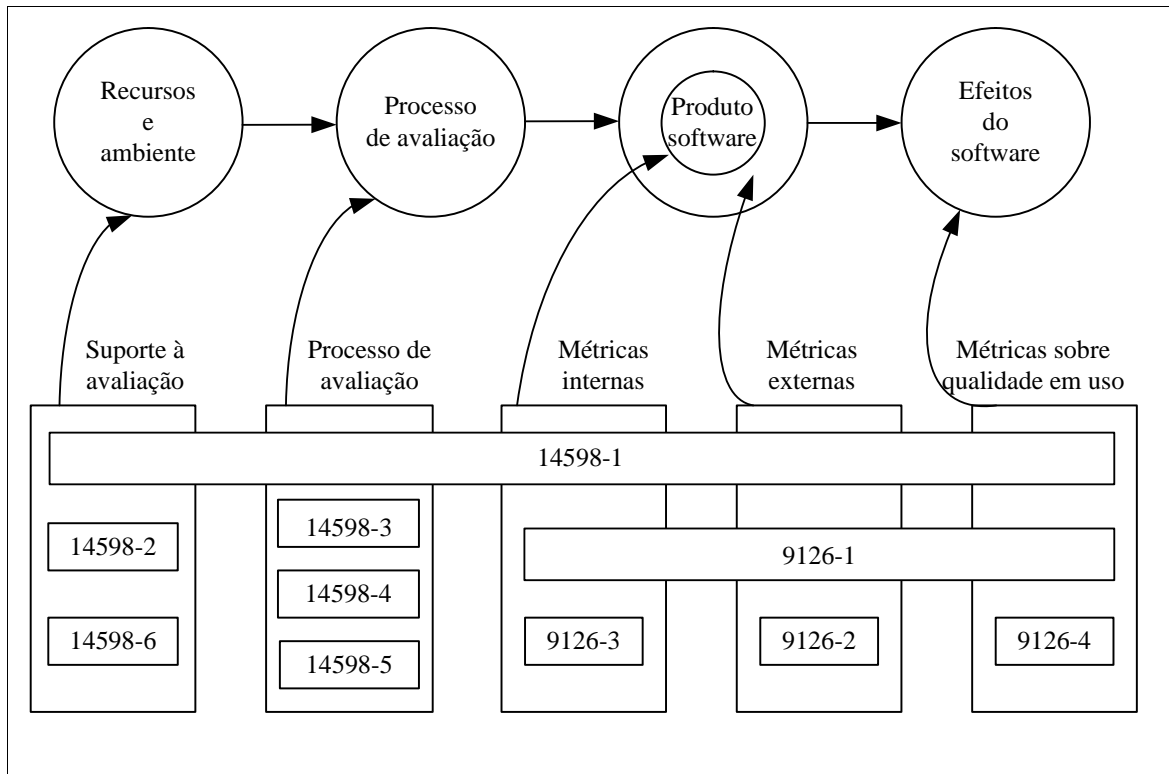


Figura 2.9 – Relacionamento entre as normas ISO/IEC 9126 e 14598 (adaptado de ISO/IEC [1999]).

A norma ISO/IEC 9126-1, define um modelo de qualidade, consistindo num conjunto de características de qualidade de alto nível, em que cada uma é decomposta de uma forma hierárquica em subcaracterísticas. A capacidade do software em relação a cada subcaracterística é determinada por um conjunto de atributos, que pode ser medido.

A qualidade do produto software pode então ser avaliada medindo os atributos internos ou os atributos externos ou finalmente, os atributos “em uso” (ver figura 2.10).

A qualidade interna é definida [ISO/IEC 1999] como o conjunto global das características do produto software, visto na sua perspectiva interna. A qualidade externa representa a qualidade quando o software é executado, normalmente num ambiente simulado e com dados fictícios. Finalmente, a qualidade “em uso” representa a perspectiva do utilizador final sobre a qualidade do produto software, quando utilizado num ambiente determinado.

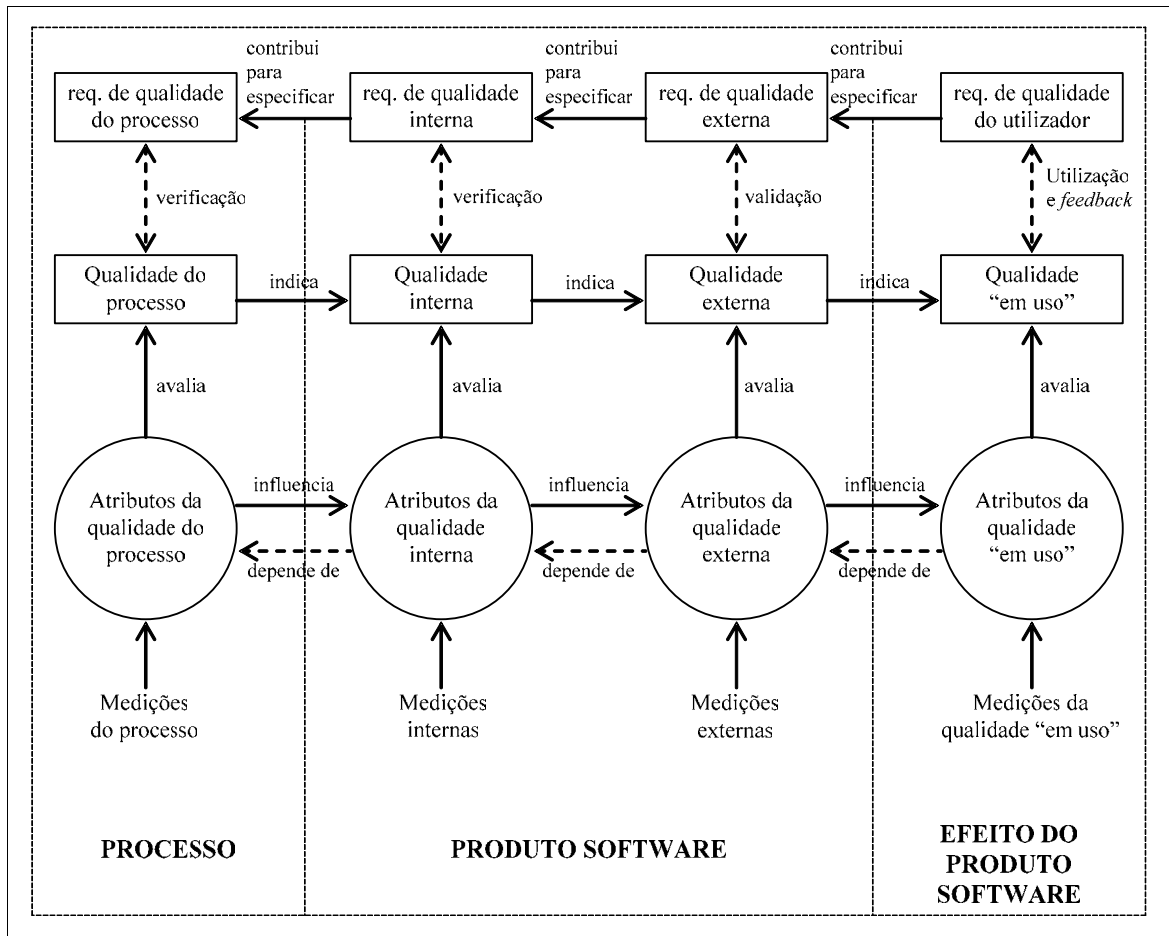


Figura 2.10 – Qualidade no ciclo de vida do software (adaptado de ISO/IEC [1999]).

No que se refere à qualidade interna e externa, o modelo de qualidade da ISO/IEC 9126-1, categoriza os atributos de qualidade do software em seis características:

Funcionalidade – a capacidade do produto software satisfazer as necessidades explícitas e implícitas.

Fiabilidade – a capacidade do software manter um determinado nível de desempenho quando sujeito a determinadas condições.

Facilidade de utilização – a capacidade do produto software ser compreendido e utilizado pelo utilizador.

Eficiência – a capacidade do software fornecer um determinado nível de serviços quando restringido a um conjunto de recursos definido.

Facilidade de manutenção – a capacidade do produto software ser alterado, corrigido, melhorado ou adaptado.

Portabilidade – a capacidade do software ser transferido para outro ambiente.

Cada uma das características definidas é ainda dividida em subcaracterísticas (ver figura 2.11). Repare-se que a subcaracterística “conformidade” está presente em todas as características, avaliando a capacidade do produto software seguir a legislação, convenções e normas relativas a cada característica.

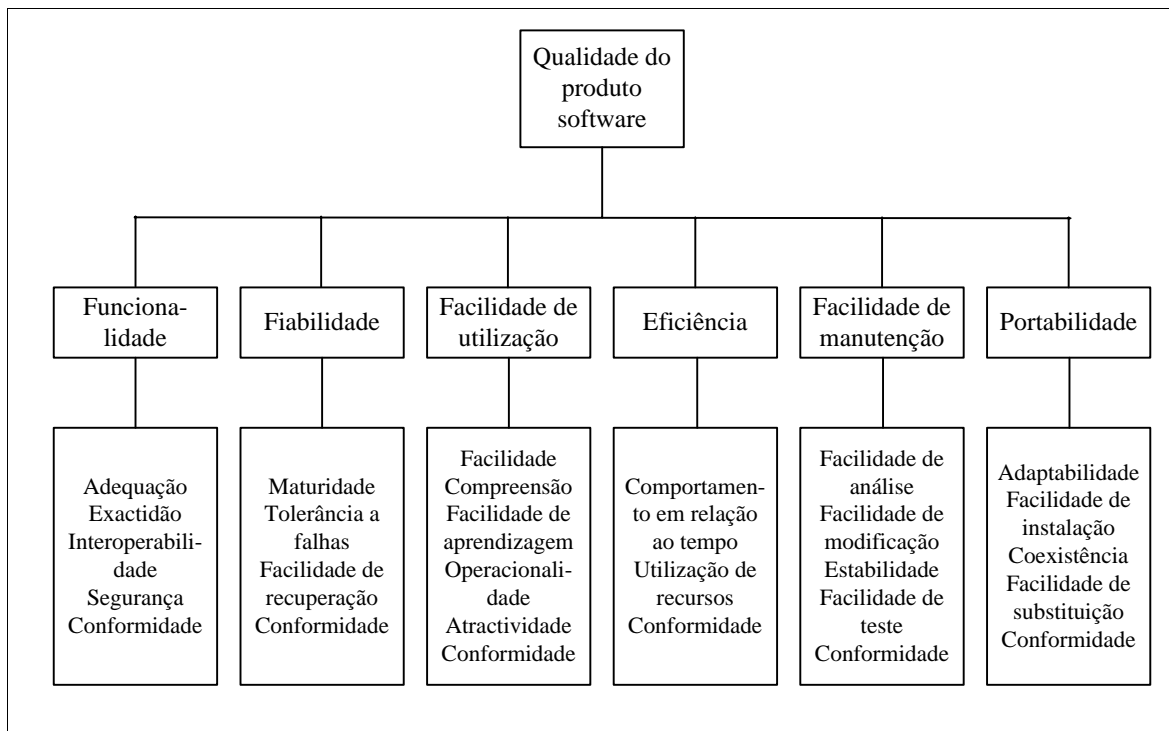


Figura 2.11 – Qualidade interna e externa na norma ISO/IEC 9126.

Em relação à qualidade “em uso”, os atributos de qualidade são categorizados em quatro características (ver figura 2.12): efectividade, produtividade, segurança e satisfação. A efectividade avalia a capacidade do produto software permitir que os utilizadores atinjam determinados objectivos com exactidão e de forma completa. A produtividade avalia a capacidade do software permitir que os utilizadores gastem um valor apropriado de recursos em relação à eficiência atingida. A segurança avalia a capacidade do software atingir níveis aceitáveis de risco. Finalmente, satisfação avalia a capacidade do software satisfazer as necessidades dos utilizadores.

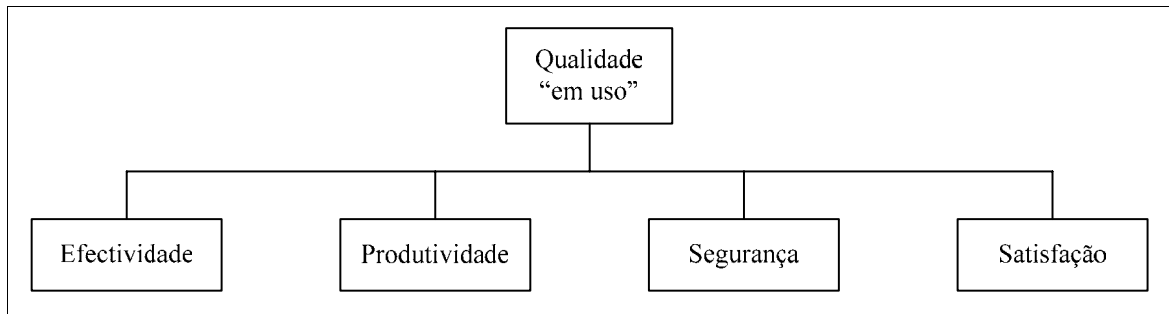


Figura 2.12 – Modelo de qualidade para a qualidade “em uso” [ISO/IEC 1999].

A qualidade “em uso” é a perspectiva de qualidade do utilizador, num ambiente contendo o produto software e é medida através dos resultados da utilização do software nesse ambiente e não pelas propriedades do produto software. Como já vimos (figura 2.10) a obtenção da qualidade “em uso” depende do software atingir a qualidade externa necessária, que por sua vez depende do nível de qualidade interna atingido pelo produto software.

A norma 9126 pode ser utilizada em diversos contextos, nomeadamente

- Identificação dos requisitos
- Identificação de critérios de aceitação para o produto final
- Identificar critérios para a garantia de qualidade
- Definir objectivos para os testes do software

A norma ISO/IEC 14598 apresenta uma perspectiva geral dos processos de avaliação do produto software, fornecendo também orientações e requisitos da avaliação em três situações diferentes: (1) empresas que desenvolvem software e pretendem melhorar a qualidade do seu próprio produto; (2) empresas que compram software e pretendem avaliar a qualidade do software que irão adquirir e (3) empresas que fazem a certificação e que irão emitir um parecer independente sobre a qualidade de um determinado produto software.

Esta norma permite uma avaliação padronizada das características de qualidade do software, complementando a norma ISO/IEC 9126. Refira-se que a norma 14598 detalha de uma forma pormenorizada o processo de avaliação, apresentando modelos para os relatórios da avaliação, técnicas para a medição das características de qualidade e fases do processo de avaliação (ver figura 2.13).

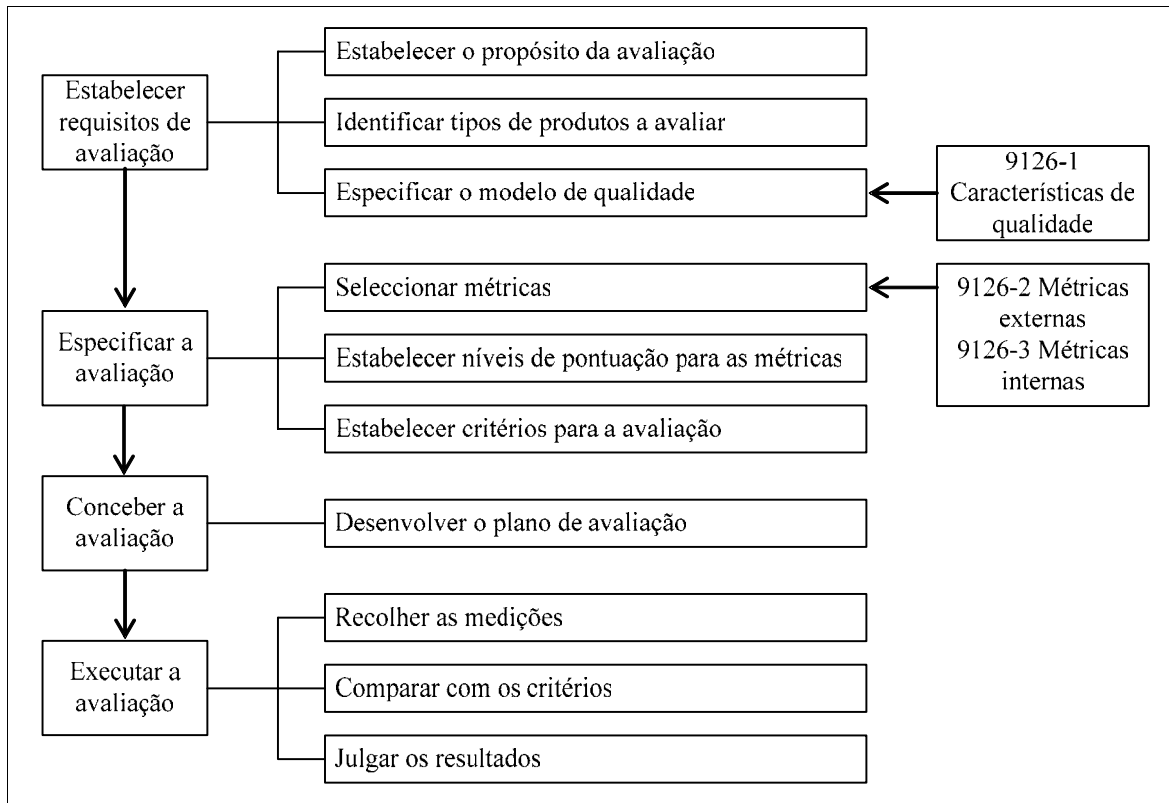


Figura 2.13 – Processo de avaliação (adaptado de ISO/IEC [1998]).

A norma 14598, distingue claramente os propósitos da avaliação dos produtos intermédios e finais. Assim, um produto intermédio pode ser avaliado em relação à sua qualidade para:

- Decidir sobre a aceitação de um produto intermédio desenvolvido por uma empresa subcontratada
- Decidir sobre o encerramento de uma tarefa intermédia
- Prever a qualidade do produto final
- Recolher e tratar informação sobre os produtos intermédios para gerir e controlar o processo

Em relação ao produto final, este pode ser avaliado em relação à qualidade para:

- Decidir sobre a aceitação de um produto
- Decidir quando considerar uma versão pronta
- Comparar o produto com produtos da competição
- Seleccionar entre produtos alternativos
- Decidir sobre a necessidade de melhorar ou substituir um determinado produto

Pode então afirmar-se que a norma ISO/IEC 14598 complementa a norma ISO/IEC 9126 definindo um modelo de avaliação da qualidade do software.

O próximo capítulo faz a ponte entre a perspectiva da qualidade do produto e da qualidade do processo, definindo os processos do ciclo de vida do software e caracterizando a área da melhoria do processo de software.

## Capítulo 3

---

### 3. Melhoria do Processo de Desenvolvimento de Software

A qualidade do produto software é uma forte preocupação encontrada em muitos níveis da sociedade moderna. Esta preocupação relaciona-se principalmente com a crescente dependência da informação, demonstrada por estas instituições. Partindo do pressuposto<sup>1</sup> que o processo para desenvolver o software, tem um grande impacto sobre os resultados<sup>2</sup> um dos cenários preferidos é melhorar o processo para obter melhorias significativas na qualidade do produto final.

Na abordagem à melhoria do processo de software, interessa considerar uma visão holística dos processos do software (ver figura 3.1), definindo três aspectos igualmente importantes, nomeadamente a definição do processo, a aprendizagem do processo e os seus *outputs*.

O primeiro aspecto está ligado à definição do processo que consistirá na sua documentação, especificando actividades, procedimentos, infra-estrutura e recursos necessários para o processo. O segundo aspecto refere-se à institucionalização do processo, ou seja, a transferência do conhecimento (aprendizagem) para os recursos encarregues da execução. Tem como responsabilidade principal o controlo do processo,

---

<sup>1</sup> largamente reconhecido na comunidade científica [Zahran 1998] e [SPIRE 1998].

<sup>2</sup> é mais fácil ver a qualidade de um prato de comida, analisando o processo do que provando.



numa perspectiva de manter os resultados dentro das fronteiras desejadas. Finalmente a visão dos resultados, manifesta-se no produto através da execução do processo. A responsabilidade chave nesta perspectiva é a medição, ou seja, recolher dados que irão permitir avaliar o desempenho do processo e serão o *input* principal para a elaboração das acções de melhoria.

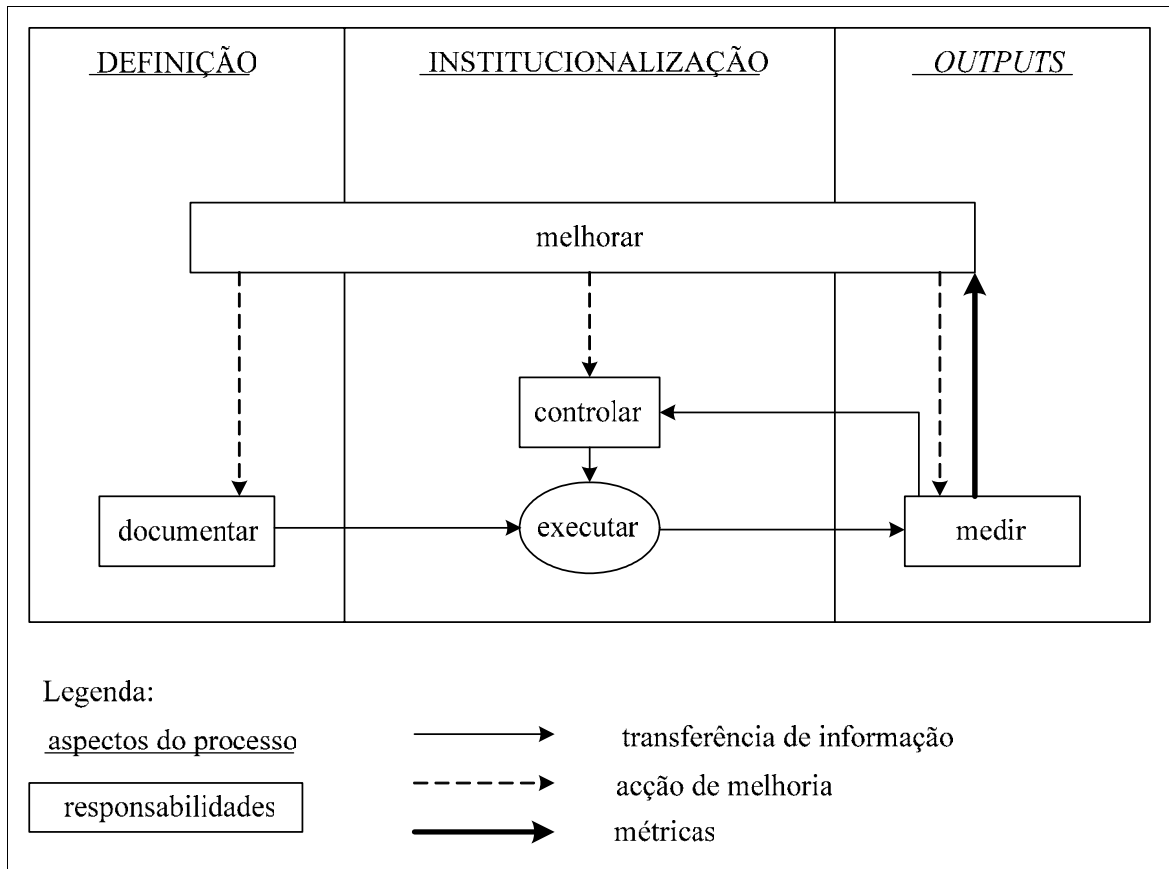


Figura 3.1 – Visão holística dos processos de software.

A responsabilidade de melhorar o processo é transversal às três perspectivas<sup>3</sup> e irá afectar a documentação, controlo, medição do processo e indirectamente a própria execução.

O ponto de partida na visão holística apresentada anteriormente é então a definição dos processos. Esta documentação e normalização permitirão uma descrição e transmissão da forma de trabalhar, através de toda a organização. Espera-se com esta

<sup>3</sup> ver secção 3.3.

estratégia, deixar de “reinventar” a roda e permitir que lições apreendidas sejam repetidas pelos profissionais encarregues de executar o processo.

Na secção seguinte apresenta-se a norma ISO/IEC 12207 que descreve os processos do ciclo de vida do software.

### **3.1. Processos do Ciclo de Vida do Software (ISO/IEC 12207)**

A norma internacional ISO/IEC 12207 fornece um referencial para o ciclo de vida do software. Foi a primeira norma internacional <sup>4</sup> a apresentar de uma forma detalhada os processos, tarefas e actividades do ciclo de vida do software. Apresenta uma arquitectura comum para o processo de software em relação à aquisição, fornecimento, desenvolvimento, operação e manutenção do software. Os processos propostos estão divididos em três classes: processos primários, processos de suporte e processos organizacionais (ver figura 3.2).

Os processos primários servem os intervenientes principais, durante o ciclo de vida do software. Os intervenientes principais que executam os processos primários são: o cliente, o fornecedor, o profissional do desenvolvimento, o operador e o profissional de manutenção. Os processos de suporte apoiam outros processos, como parte integrante destes e finalmente os processos organizacionais são utilizados globalmente pela organização e incluem as tarefas necessárias para gerir e melhorar os processos.

Refira-se que esta norma não prescreve a forma como realizar uma determinada actividade, isto está sempre muito dependente do projecto específico e do estado da tecnologia, a opção é então deixar “ao critério do utilizador da norma”. A norma detalha cada um dos processos<sup>5</sup>, definindo uma hierarquia entre processos, actividades e tarefas.

---

<sup>4</sup> publicada em 1 de Agosto de 1995, em 2002 foi publicada uma emenda para a alinhar com a norma ISO/IEC 15504.

<sup>5</sup> ver secção 6.2.

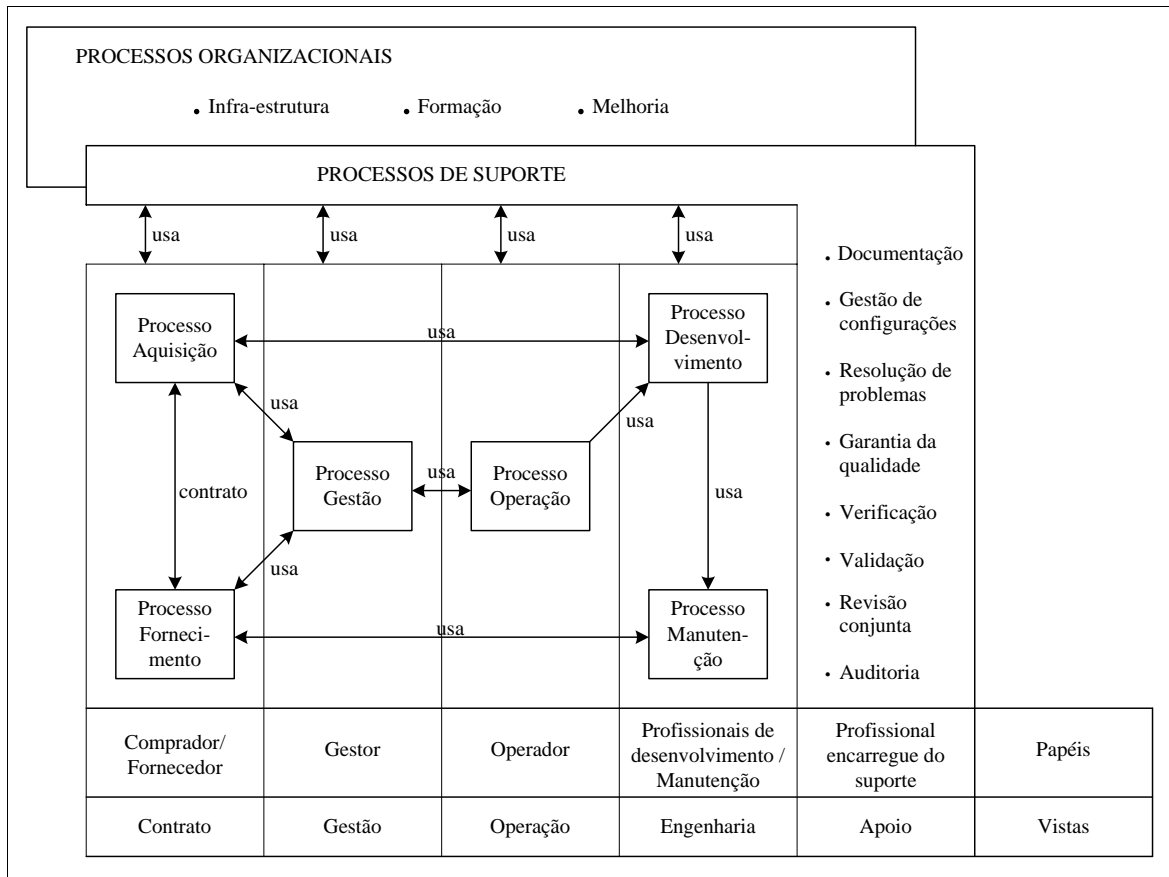


Figura 3.2 – Processos do ciclo de vida do software: papéis, perspectivas e relacionamentos (adaptado de ISO/IEC [1995]).

O objectivo principal é apresentar um conjunto de processos fundamentais para um projecto de software com qualidade e que cobrem as “melhores práticas” utilizadas com sucesso pelas empresas de software.

### 3.2. O Contexto da Melhoria do Processo de Software

Como já foi referido, a abordagem baseada no processo baseia-se no pressuposto [SWEBOK 2001] “que o processo tem um impacto nos resultados do processo” (ver figura 3.3).

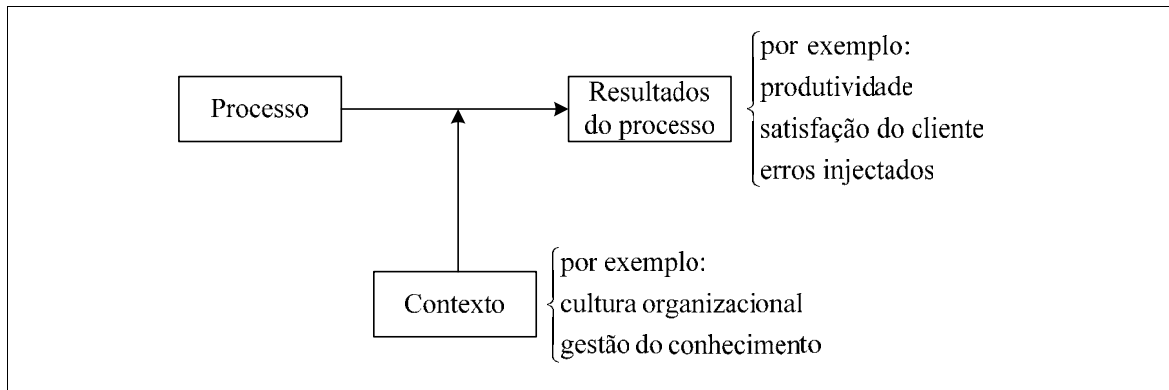


Figura 3.3 – Relacionamento entre o processo e os seus resultados (baseado em SWEBOK [2001]).

Este impacto resultante de alterações no processo pode ter um efeito negativo ou positivo nos resultados do processo. Refira-se também que podemos distinguir claramente duas áreas de intervenção relacionadas com a melhoria. Por um lado, a formalização das actividades de base de qualquer processo<sup>6</sup> e numa segunda fase a definição de acções de melhoria baseada num ROI quantitativamente justificada. Segundo Florac e Carleton [1999] os objectivos da melhoria do processo numa organização, são:

- Aprofundar o conhecimento dos processos existentes e definir os factores que afectam a capacidade
- Definir, avaliar e executar acções específicas de melhoria de modo a alinhar os resultados dos processos com as necessidades do negócio
- Analisar os custos e benefícios das mudanças realizadas nos processos

Em termos genéricos um ambiente de melhoria do processo efectivo tem que definir quatro componentes principais [Zahran 1998], a saber: a infra-estrutura do processo, o modelo de melhoria do processo, o método de avaliação do processo e o plano para a melhoria do processo (ver figura 3.4)

A infra-estrutura do processo divide-se em duas áreas. Por um lado a definição de papéis e responsabilidades, denominada infra-estrutura de gestão, a outra vertente está ligada à infra-estrutura técnica, relacionada com o ambiente e ferramentas necessárias para suportar as actividades do processo.

<sup>6</sup> a documentação, medição e controlo.

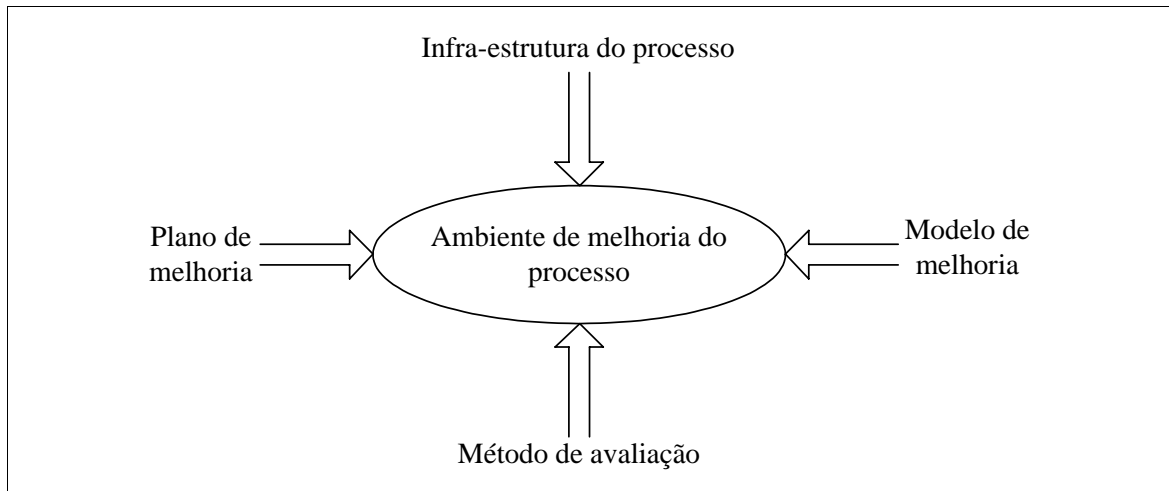


Figura 3.4 – Referencial para o ambiente de melhoria do processo (adaptado de Zahran [1998]).

O modelo de melhoria do processo deve caracterizar o processo do software e definir níveis de maturidade, bem como os parâmetros a alcançar para atingir esses níveis. Baseado neste modelo do processo, o método de avaliação deverá permitir avaliar as práticas actuais da organização e assim identificar pontos fortes e fracos dos processos, orientando as estratégias de melhoria.

Finalmente, o plano de melhoria define o modo como as conclusões obtidas através da avaliação serão aplicadas na prática. Espera-se que este plano defina um conjunto de fases e uma estratégia de “alto nível” para a aplicação consistente das acções de melhoria específicas.

A secção seguinte irá apresentar de uma forma sucinta os modelos de melhoria mais reconhecidos em termos da comunidade científica. Estes modelos serão um dos pontos de partida para as componentes do 2MPspe.

### 3.3. Modelos de Melhoria do Processo de Software

Como vimos, a melhoria do processo de software tem origem no movimento TQM, assentando na premissa que o processo para desenvolver um produto tem um grande impacto na qualidade final do produto [Humphrey 1989].

Os modelos de melhoria do processo de software mais divulgados, incluem um modelo para avaliação do processo de software e um *roadmap* que especifica as etapas de melhoria e indica o caminho a seguir. A avaliação do processo de software permite ver qual é o estado das práticas de uma organização. A famosa frase de Humphrey explica a necessidade da avaliação, “se não sabemos onde estamos, um mapa não vai ajudar”.

A medição do processo surge, então, como ponto de partida de um projecto de melhoria do processo de software. Card [1991] descreve dois paradigmas genéricos para caracterizar o tipo de medição do processo. O denominado paradigma analítico que se baseia na evidência quantitativa para determinar onde as melhorias são necessárias e também para verificar o sucesso de uma iniciativa de melhoria. O paradigma *benchmarking* baseia-se na identificação de organizações excelentes, numa área, e na documentação das suas “boas práticas”. Como seria de esperar, é possível utilizar ambos os paradigmas num projecto de melhoria do processo de software.

Nas próximas secções apresentam-se alguns modelos de melhoria do processo mais divulgados, com uma ênfase especial no paradigma *benchmarking*.

### 3.3.1. Paradigma *Benchmarking*

Como vimos, este paradigma envolve a avaliação da maturidade da organização com base na medição da capacidade dos processos. O modelo de avaliação utiliza as “boas práticas” identificadas para classificar os processos da organização.

Há um elevado número de modelos de melhoria do processo propostos na literatura. Nesta secção faz-se uma descrição completa do *Capability Maturity Model*<sup>7</sup> (CMM), SPICE e ISO/IEC 15504, SPIRE e da ISO 9001:2000. A última secção apresenta uma breve descrição de outros modelos.

---

<sup>7</sup> neste documento o CMM refere-se ao SW – CMM v1.1(*Capability Maturity Model for software*)

### 3.3.1.1. Capability Maturity Model (CMM)

O CMM foi desenvolvido pelo *Software Engineering Institute* (SEI) com o patrocínio do Departamento de Defesa dos EUA e baseia-se nos trabalhos desenvolvidos por Watts Humphrey sobre a caracterização do processo de software [Humphrey 1988]. Paulk et al. [1995] definem formalmente o CMM como “a descrição das etapas por onde passam as organizações de software, conforme definem, implementam, medem, controlam e melhoram os seus processos de software”

O CMM baseia-se num referencial de maturidade que fornece orientações efectivas para um programa de melhoria do processo. Para a avaliação dos processos é utilizado um questionário de maturidade<sup>8</sup> que permite caracterizar a capacidade da organização. A melhoria do processo de software é feita em pequenos passos evolutivos e não em inovações revolucionárias. O CMM fornece o referencial para organizar estes passos evolutivos em níveis de maturidade que dão as fundações sucessivas para a melhoria contínua do processo.

Os cinco níveis de maturidade (ver figura 3.5) apresentam prioridades claras sobre a forma como aumentar a capacidade do processo. Quanto maior o nível de maturidade da organização, menor é o risco associado e maior a produtividade e qualidade.

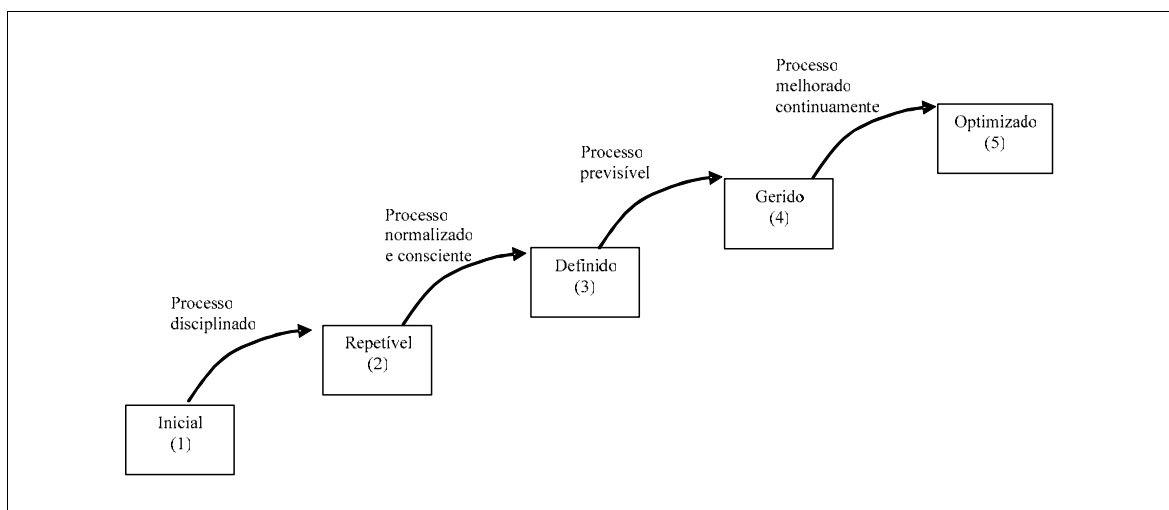


Figura 3.5 – Os cinco níveis de capacidade do CMM.

<sup>8</sup> 101 perguntas de sim ou não.

Cada nível de maturidade tem os seus próprios desafios, que a organização deve focar para aumentar a sua maturidade (ver tabela 3.1).

Nível	Características	Desafios Chave	Produtividade Qualidade
5 Optimizado	(Melhorado continuamente) Feedback quantitativo dos processos	- Processo baseado nos recursos humanos - Manter o nível otimizado	
4 Gerido	(Quantitativo) Processo medido	- Mudanças tecnológicas - Análises de problemas - Prevenção de problemas	
3 Definido	(Qualitativo) Processo definido institucionalizado	- Medição do processo - Análise do processo - Planos quantitativos de qualidade	
2 Repetível	(Intuitivo) Processo dependente dos indivíduos	- Formação, testes, práticas técnicas e revisões. - Foco no processo, normas e processos	
1 Inicial	(Ad-hoc) Caótico	- Planeamento e gestão do projecto - Gestão de configurações - Garantia da qualidade	Risco

Tabela 3.1 – Desafios chave / níveis de maturidade [Paulk et al. 1995].

### 3.3.1.1.1. Caracterização dos Níveis de Maturidade

Apresenta-se em seguida uma breve descrição de cada um dos níveis de maturidade:

- 1. Inicial.** O processo de software pode caracterizar-se como ad-hoc ou mesmo caótico. Poucos processos estão definidos e o sucesso depende dos esforços individuais. Pode afirmar-se que a capacidade é imprevisível uma vez que o processo está constantemente a ser modificado conforme o trabalho progride.
- 2. Repetível.** Os processos de gestão de projectos estão estabelecidos para acompanhar os custos, prazos e funcionalidade. O processo é



suficientemente estável para repetir sucessos anteriores em projectos de aplicações similares.

- 3. Definido.** O processo de software para as actividades de engenharia e gestão está documentado, normalizado e integrado num processo standard para a organização. Todos os projectos utilizam uma versão aprovada e colocada “à medida” do processo standard.
- 4. Gestão.** O processo é medido e opera dentro de limites mensuráveis. São recolhidas medições detalhadas sobre o processo de software e sobre a qualidade do produto. O processo de software e os produtos são compreendidos e controlados quantitativamente
- 5. Optimizado.** A organização tem agora os alicerces para a melhoria contínua e optimização do processo. A melhoria ocorre com aperfeiçoamentos incrementais no processo existente e através da inovação, utilizando novos métodos e tecnologias.

Paulk et al. [1995] definem a capacidade do processo como “a amplitude de resultados esperados que podem ser atingidos, seguindo o processo”. Quanto maior a maturidade do processo de uma organização, mais previsíveis são os seus projectos e maior a qualidade e previsibilidade dos recursos humanos.

#### 3.3.1.1.2. Estrutura do CMM

Os níveis de maturidade são constituídos por várias Áreas Chave do Processo<sup>9</sup>, mas cada área chave do processo pertence a um e um só nível de maturidade. Cada uma das áreas é organizada em cinco secções, denominadas características comuns e estas especificam as práticas chave que, quando executadas em conjunto completam os objectivos de determinada área chave do processo (ver figura 3.6).

---

<sup>9</sup> KPA - Key Process Area.

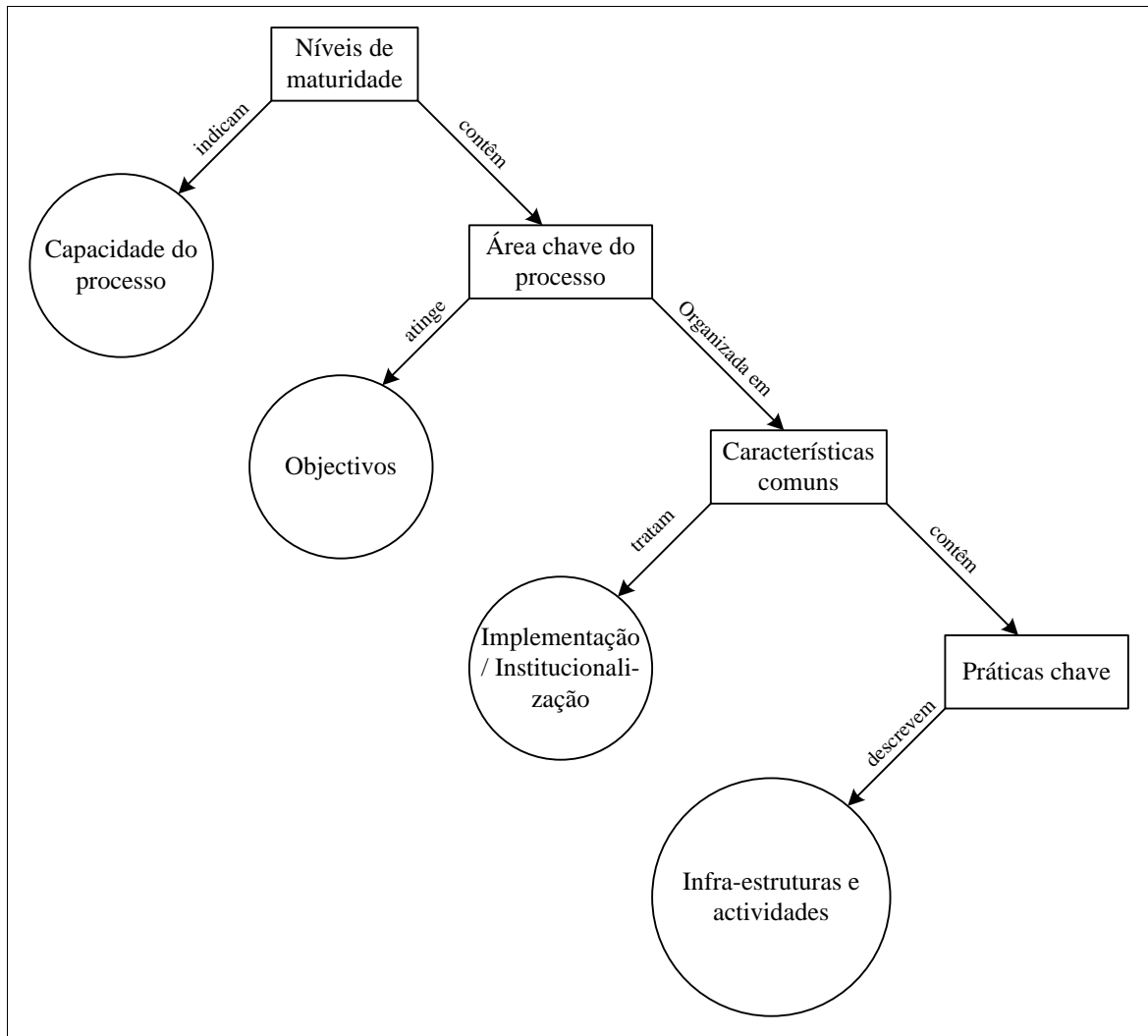


Figura 3.6– Estrutura interna do CMM [Paulk et al. 1995].

As áreas chave do processo indicam as áreas que uma organização deve focar para melhorar o seu processo de software. O nível inicial não tem nenhuma área chave do processo. Para os níveis seguintes as áreas são:

#### Nível 2: Repetível

- Gestão de configurações
- Garantia da qualidade do software
- Gestão de subcontratações
- Acompanhamento do projecto de software
- Planeamento do projecto de software
- Gestão de requisitos

#### Nível 3: Definido

- Foco no processo organizacional

- Definição do processo organizacional
- Programa de formação
- Gestão integrada do software
- Engenharia do produto
- Coordenação intergrupos
- Revisão por pares

#### Nível 4: Gerido

- Gestão da qualidade do software
- Gestão quantitativa do processo

#### Nível 5: Optimizado

- Gestão da mudança do processo
- Gestão da mudança da tecnologia
- Prevenção de defeitos

Cada área chave do processo tem um conjunto de objectivos que definem o âmbito, fronteiras e intenção. Em relação às características comuns, representam as práticas que descrevem as áreas chave do processo. Há cinco características comuns:

- Compromisso para executar - Declarações da política e liderança.
- Capacidade para executar - Estruturas organizacionais, recursos e fundos, formação, orientação, pré-requisitos.
- Actividades executadas - Actividades, papéis e procedimentos e o seu estabelecimento.
- Medições e análise - Medições básicas e outras medições.
- Verificação da implementação - Vigilância da gestão sénior, vigilância da gestão de projectos e garantia da qualidade.

Como abordagem à melhoria do processo, o SEI aconselha a utilização da abordagem denominada IDEAL<sup>10</sup>, que engloba cinco fases: início, diagnóstico, estabelecimento, acção e *leveraging*. Esta abordagem representa um referencial de alto nível que descreve as fases, actividades e recursos necessários para o sucesso de um programa de melhoria do processo de software. O CMM é utilizado na fase de

---

<sup>10</sup> *Initiation, Diagnosing, Establishing, Acting e Leveraging.*

diagnóstico, permitindo também estabelecer as prioridades e definir o plano de acção [McFeely 1996]. Em relação à avaliação da maturidade das organizações, o SEI desenvolveu dois métodos: avaliação do processo de software e aferição<sup>11</sup> da capacidade do software. Um projecto de avaliação do processo de software é utilizado para determinar o estado actual do processo do software de uma organização, para determinar os aspectos mais prioritários em relação aos processos e para obter o apoio organizacional para a iniciativa de Melhoria do Processo de Software (MPS). Um projecto de aferição da capacidade do software é utilizado para identificar os fornecedores de software mais qualificados para o fazer e também para monitorizar o estado do processo de software utilizado num projecto a decorrer.

### 3.3.1.1.3. Perfil de Maturidade

Sobre o perfil do software em termos mundiais, apresentado no último relatório do SEI [2004], incluindo 1593 organizações e 8185 projectos (ver figura 3.7), nota-se uma clara melhoria na maturidade das organizações que relataram a avaliação realizada.

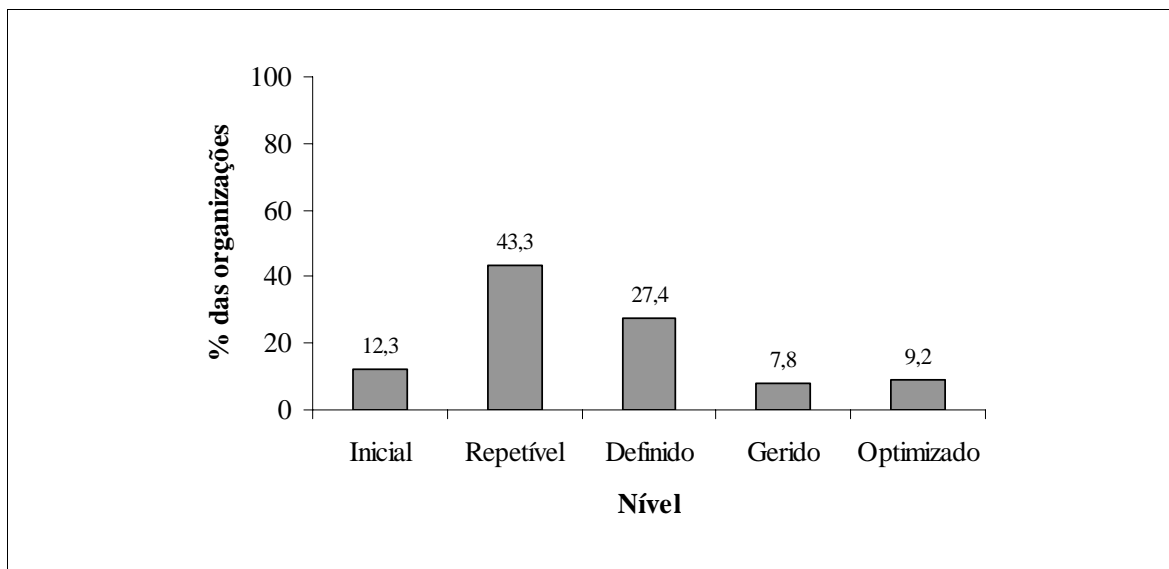


Figura 3.7 – Perfil de maturidade de avaliações<sup>12</sup> de 1999 até Dez. 2003 [SEI 2004].

<sup>11</sup> Evaluation.

<sup>12</sup> avaliações refere-se a SPA (Software Process Assessment), CBA – IPI (CMM – Based Appraisal for Internal Process Improvement) e SCAMPI (Standard CMMI Appraisal Method for Process Improvement).

Refira-se que os dados da figura 3.7 incluem 54,7% de empresas fora dos EUA e 51,3% de empresas com menos de 100 funcionários. Estes dados têm que ser analisados com algum cuidado, devido ao facto de as organizações relatarem os dados de livre vontade, sendo de esperar que as empresas com maior nível sintam maior “interesse” em comunicar esse resultado. Noutra perspectiva, uma empresa que realiza uma avaliação, tem obrigatoriamente um nível de preocupações dificilmente encontrado em Portugal.

Em relação a dados recolhidos desde 1987, o SEI refere 2401 organizações e 13266 projectos em 48 países espalhados pelos 5 continentes. A figura 3.8 apresenta a evolução verificada (valores acumulados).

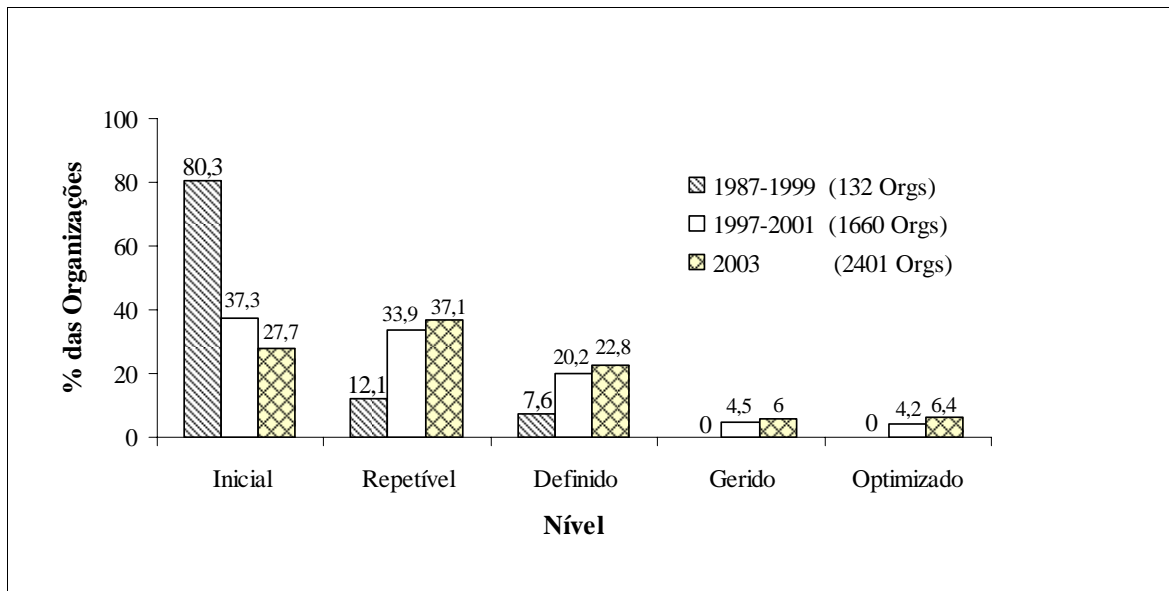


Figura 3.8 – Tendência no perfil de maturidade [SEI 2004].

Em relação às tendências nas organizações, verifica-se que a área chave do processo do nível 2, menos satisfeita nas organizações avaliadas no nível 1 é a Garantia da Qualidade do Software. Nas organizações no nível 2, a área chave do processo do nível 3 em que se verifica mais dificuldade é a Gestão Integrada do Software. Sobre o tempo para evoluir entre níveis, verifica-se os valores seguintes<sup>13</sup>:

⇒ Nível 1 para 2: 22 meses

⇒ Nível 2 para 3: 19 meses

<sup>13</sup> organizações que iniciaram o seu esforço de MPS em 1992 ou mais tarde.

- ⇒ Nível 3 para 4: 25 meses
- ⇒ Nível 4 para 5: 13 meses

### 3.3.1.2. Norma ISO/IEC 15504

No início dos anos 90 foram desenvolvidos diversos métodos para a avaliação da capacidade e melhoria do processo. O foco<sup>14</sup>, o âmbito<sup>15</sup>, a origem<sup>16</sup> e as restrições<sup>17</sup> à utilização eram variados, mas teve início um grande consenso internacional sobre a necessidade urgente de uma norma de domínio público, para a avaliação do processo de software. Em Junho de 1991, o JTC1<sup>18</sup>/SC7<sup>19</sup> do ISO/IEC aprovou a resolução com a recomendação para um novo grupo de trabalho (WG10) para desenvolver uma norma internacional sobre avaliação do processo de software. Esta recomendação foi aceite em Janeiro de 1993 e foi atribuído ao WG10 o projecto denominado SPICE (*Software Process Improvement and Capability dEtermination*).

A versão 1.0 do SPICE surgiu em Junho de 1995 e continha 9 partes. Dos *trials* realizados e da votação dos membros da ISO foram considerados 4 aspectos para a versão seguinte [Rout e Simms 1998].

1. Necessidade de um maior alinhamento com a norma ISO/IEC 12207
2. Necessidade de tornar a norma menos prescritiva
3. Necessidade de tornar mais clara a forma de migrar/harmonizar para as abordagens já existentes (especificar os requisitos para serem conformes)
4. Necessidade de simplificar a arquitectura apresentada, particularmente a parte referente à dimensão da capacidade.

Com a incorporação destas recomendações no SPICE, resultou o relatório técnico ISO/IEC TR 15504, publicado em 1998 e englobando 9 componentes (ver figura 3.9).

---

<sup>14</sup> avaliação da capacidade e/ou melhoria do processo.

<sup>15</sup> grandes organizações ou PMEs de software.

<sup>16</sup> normalmente CMM ou ISO 9000.

<sup>17</sup> domínio público ou proprietários.

<sup>18</sup> Comité Técnico Conjunto responsável pelas normas das Tecnologias da Informação.

<sup>19</sup> Sub-Comité responsável pelas normas da Engenharia de Software.

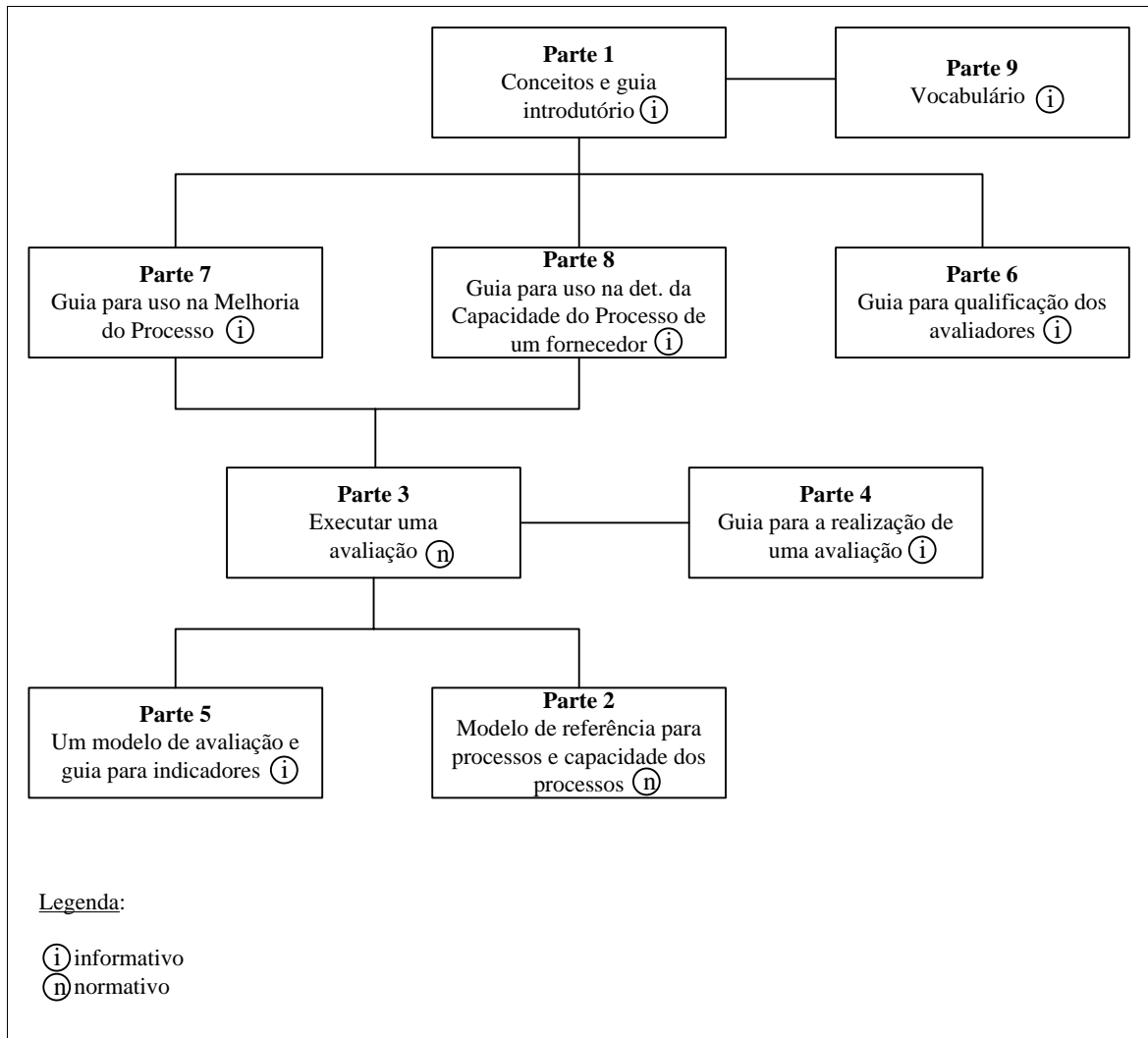


Figura 3.9 – Componentes do Relatório Técnico ISO/IEC TR 15504: 1998.

Finalmente, após 6 anos seguindo a evolução prevista no processo de normalização utilizado pela ISO/IEC<sup>20</sup>, a norma ISO/IEC 15504 encontra-se no estado final da sua publicação. As versões finais da parte 2, parte 3 e parte 4 foram publicadas no 1º semestre de 2004. A parte 1 encontra-se na fase FDIS (esboço final da norma internacional) e a parte 5 encontra-se ainda na fase CD (esboço do comité internacional).

A norma ISO/IEC 15504 fornece uma abordagem estruturada para a avaliação dos processos com um dos seguintes objectivos [ISO/IEC 2004a]:

- Compreender o estado dos processos para melhoria do processo

<sup>20</sup> fases do processo de normalização: TR (*Technical Report*); WD (*Working Draft*); CD (*Committee Draft*); DIS (*Draft International Standard*) e finalmente IS (*International Standard*).

- Determinar a adaptação dos processos para uma necessidade particular
- Determinar a adaptação dos processos de outra organização para um determinado contrato

A versão final da norma encontra-se organizada em cinco partes (ver figura 3.10). A parte 1 (informativa) descreve o modo como as componentes estão organizadas e encaixam, orientando a sua selecção e utilização.

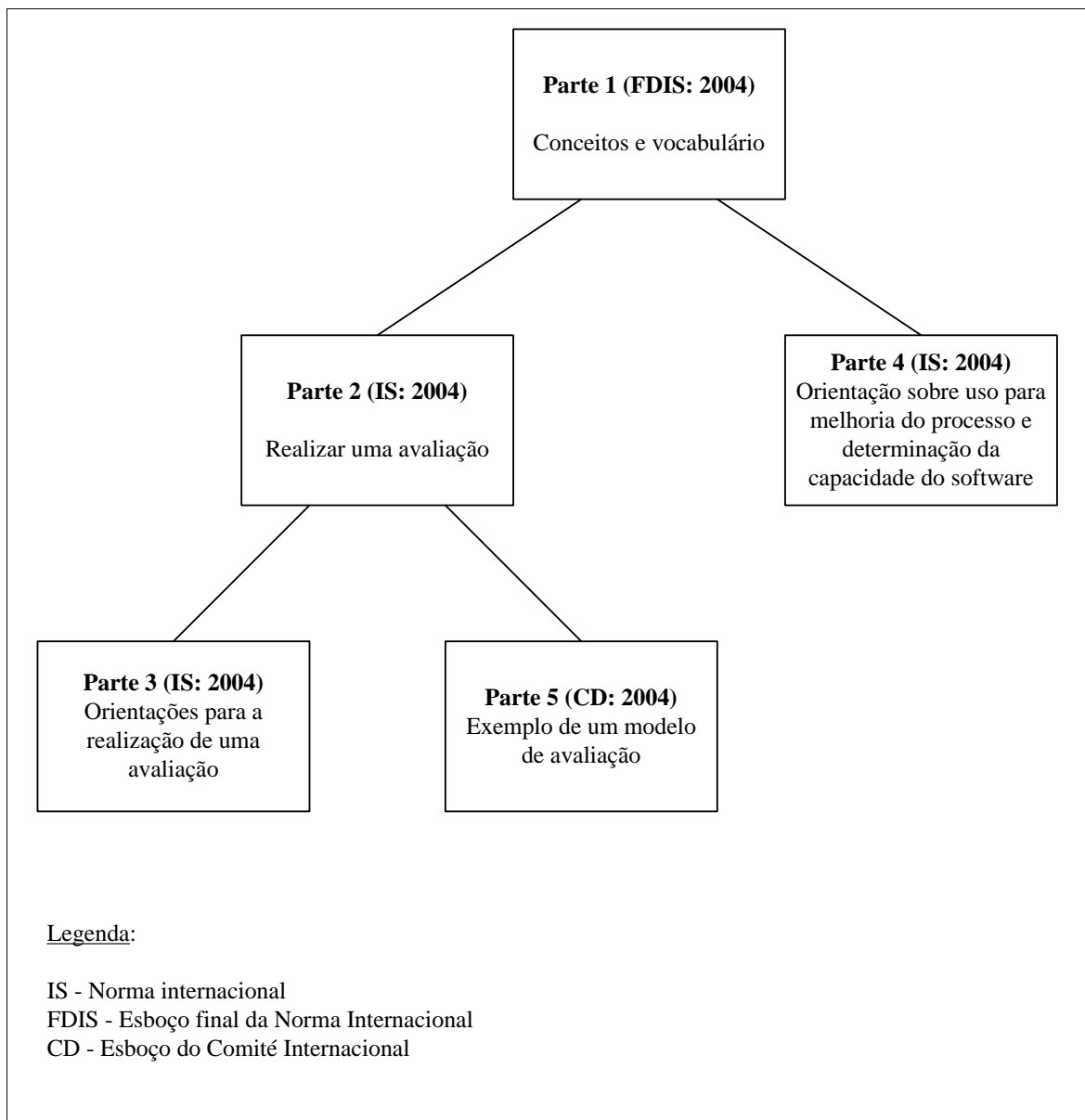


Figura 3.10 – Componentes da norma ISO/IEC 15504.



A parte 2 (normativa) apresenta um modelo bidimensional para a descrição dos processos e da capacidade dos processos. A descrição dos processos utiliza um modelo de referência dos processos, externo à norma. A dimensão da capacidade define seis níveis de capacidade, caracterizados por nove atributos do processo [ISO/IEC 2004b].

A parte 3 (informativa) explica e dá orientações sobre o cumprimento dos requisitos apresentados na parte 2. As orientações abrangem 5 áreas [ISO/IEC 2004c]:

- O processo de avaliação
- O referencial de medição
- O modelo de referência e o modelo de avaliação dos processos
- Instrumentos ou ferramentas para avaliação
- Competência dos avaliadores

A parte 4 (informativa) dá orientações sobre a utilização da norma no contexto da melhoria do processo e da determinação da capacidade. No caso da melhoria do processo, os conceitos e princípios são apropriados para diferentes necessidades do negócio, domínios de aplicação e tamanho da organização, de modo a serem utilizados por todos os tipos de organizações na orientação das suas actividades de melhoria. No caso da determinação da capacidade, o objectivo é ser aplicável no relacionamento cliente - fornecedor e na determinação da capacidade do processo [ISO/IEC 2004d].

Finalmente, a parte 5 (informativa) apresenta um modelo de avaliação baseado no modelo de referência ISO/IEC 12207 e conforme os requisitos definidos na 15504-2. Faz a descrição detalhada do modelo de avaliação, incluindo a dimensão dos processos e a dimensão da capacidade dos processos [ISO/IEC 2003].

A figura 3.11 mostra o relacionamento entre as diversas componentes da norma. A avaliação do processo tem dois contextos principais. No contexto da melhoria do processo, a avaliação permite caracterizar as práticas actuais. A partir da análise dos pontos fortes, pontos fracos e riscos dos processos é possível estabelecer prioridades de melhoria. No contexto da determinação da capacidade, a preocupação central é comparar a capacidade actual dos processos relacionados com um perfil de capacidade

alvo para identificar os riscos envolvidos na utilização do processo seleccionada para executar o projecto.

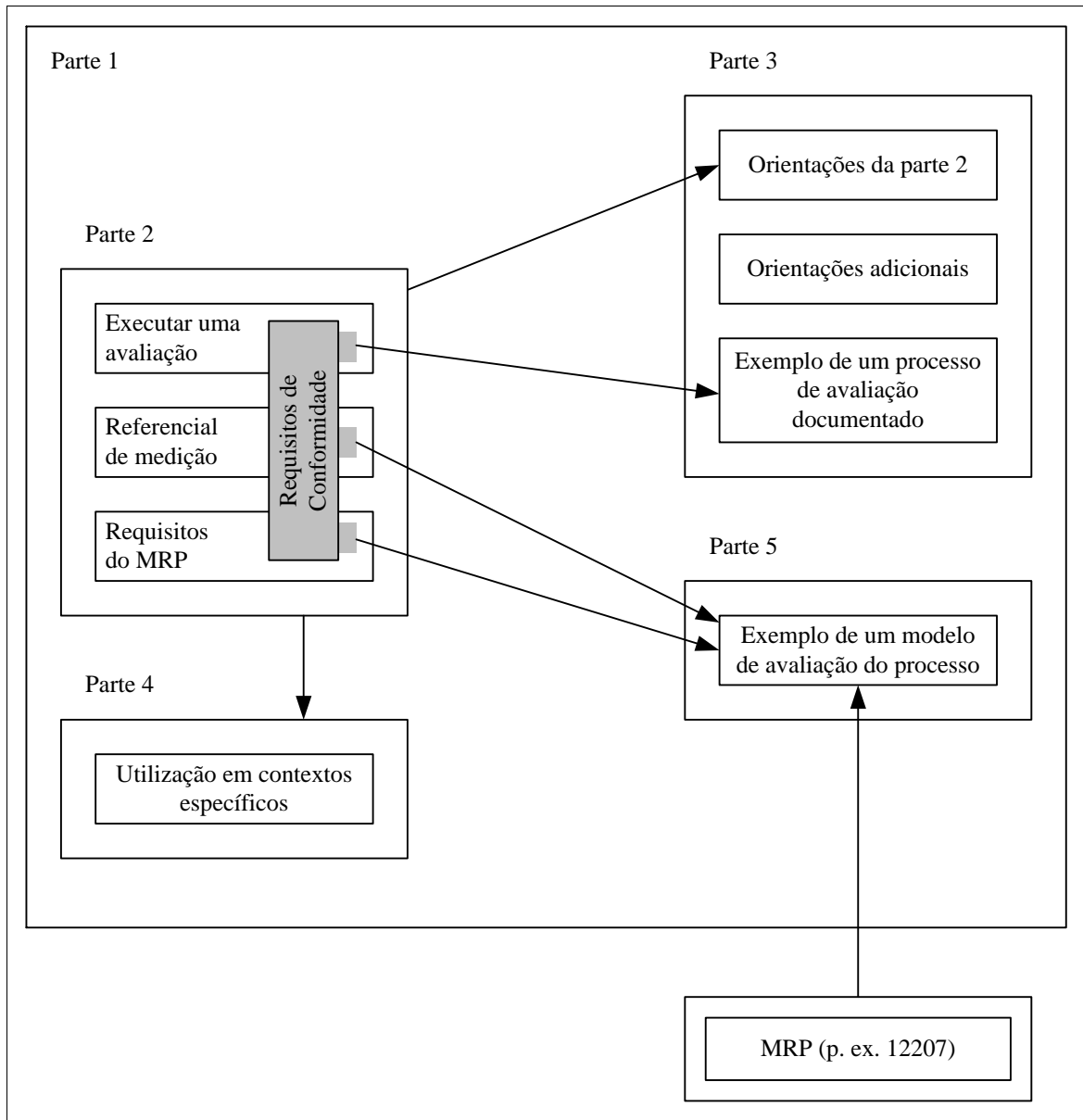


Figura 3.11 – Relacionamento entre os elementos da ISO/IEC 15504.

Segundo a parte 1, este referencial:

- Facilita a auto-avaliação
- Fornece uma base tanto para a melhoria do processo de software como para a determinação da capacidade do processo de software
- Toma em consideração o contexto onde os processos estão implementados
- Produz uma classificação dos processos

- Centra-se na aptidão do processo para atingir a sua meta
- É apropriado para todos os domínios de aplicação e tipos de organizações
- Fornece *benchmark* objectivo entre organizações

A avaliação do processo numa organização deve encorajar e estabelecer a cultura de melhoria constante, a engenharia dos processos e a optimização dos recursos.

Esta norma propõe um modelo com seis níveis de capacidade. Os níveis representam capacidade crescente do processo implementado e são os seguintes [SPIRE 1998]:

**Nível 0** – Incompleto. Caracteriza-se pelo caos.

**Nível 1** – Executado. O propósito do processo é geralmente atingido (fazer as próprias “coisas”).

**Nível 2** – Gerido. O processo entrega produtos de trabalho com uma qualidade aceitável e dentro do prazo.

**Nível 3** – Estabelecido. O processo é executado e gerido utilizando um processo normalizado.

**Nível 4** – Previsível. O processo definido é executado de forma consistente e dentro de limites de controlo definidos (gestão por números).

**Nível 5** – Optimização. O desempenho do processo é optimizado para cumprir as necessidades actuais e futuras do negócio.

A medição da capacidade dos processos é efectuada com base nos atributos do processo (AP). O nível 0 não possui nenhum AP. O nível 1 é caracterizado com o atributo de execução do processo. O nível 2 apresenta dois APs: atributo de gestão da execução e atributo de gestão dos produtos de trabalho. O nível 3 é medido através do atributo de definição do processo e do atributo de transferência<sup>21</sup> do processo. O nível 4 possui também dois atributos: atributo de medição do processo e atributo de controlo do processo. Finalmente, o nível 5 é medido com base no atributo de inovação do processo e no atributo de optimização do processo.

---

<sup>21</sup> *deployment*.

A norma ISO/IEC 15504 é a base do 2MPspe e por esta razão, a descrição detalhada da arquitectura da norma<sup>22</sup> é realizada nos capítulos seguintes<sup>23</sup>.

### 3.3.1.3. Norma ISO 9001: 2000

A família de normas ISO 9000 sofreu uma profunda remodelação no ano de 2000, ficando organizada em três normas:

**ISO 9000: 2000** Sistemas de Gestão da Qualidade (SGQs) – fundamentos e vocabulário. Descreve os fundamentos dos SGQs e especifica a terminologia para os SGQs. Substitui as normas ISO 8402: 1994, a ISO 9000-1: 1994 e a ISO 9000-3: 1997.

**ISO 9001: 2000** Sistemas de Gestão da Qualidade – requisitos. Especifica os requisitos dos sistemas de gestão da qualidade e substitui as normas ISO 9001:1994, ISO 9002: 1994 e ISO 9003: 1994.

**ISO 9004: 2000** Sistemas de Gestão da Qualidade – orientações para melhorias do desempenho. Dá orientações sobre os SGQs, incluindo os processos para a melhoria contínua, que irão contribuir para a satisfação dos clientes e outras partes da organização. Esta norma substitui a ISO 9004-1: 1994.

As raízes da norma ISO 9001: 2000 encontram-se na versão de 1994 da família ISO 9000. O processo de revisão teve início no último trimestre de 1997<sup>24</sup> e terminou em Dezembro de 2000, com a publicação da norma. A revisão da norma teve em consideração um conjunto de factores que emergiram de comentários espontâneos dos utilizadores e dos *surveys* realizados pela ISO. Entre os factores considerados, destacam-se [Tricker 2001]:

- Os problemas com o modelo de 20 elementos, claramente influenciado pelas organizações industriais

---

<sup>22</sup> apresentada sucintamente nesta secção.

<sup>23</sup> capítulos 6, 7 e 8.

<sup>24</sup> primeiro esboço de trabalho (WD1).

- A crescente utilização da ISO 9000 em indústrias regulamentadas (por exemplo, farmacêutica) e as consequentes alterações necessárias
- A proliferação de normas com orientações para interpretar a família ISO 9000: 1994
- Maior ênfase em cumprir as necessidades do cliente
- Dificuldades apresentadas pelas pequenas organizações em cumprir todos os requisitos da norma
- Necessidade de uma maior compatibilidade com outras normas na área de gestão de sistemas
- Tratamento de requisitos específicos de determinados sectores
- Adopção de uma gestão orientada ao processo

A família ISO 9000: 2000 sofreu uma profunda remodelação centrada na clarificação da abordagem ao processo, na melhoria contínua e na simplificação.

A nova estrutura com 4 processos principais é mais genérica, está ligada com a norma ISO 14001: 1996 e adopta uma gestão do processo (ver figura 3.12) actualmente bastante consensual.

A norma ISO 9001: 2000 declara explicitamente a “adopção de uma abordagem por processos quando se desenvolve, implementa e melhora a eficácia de um sistema de gestão de qualidade, para aumentar a satisfação do cliente indo ao encontro dos seus requisitos”.

A figura 3.12 apresenta o modelo do processo proposto pela norma ISO 9001: 2000, mostrando a interligação entre as quatro grandes secções (responsabilidade da gestão; gestão de recursos; realização do produto; medição, análise e melhoria) e o foco na melhoria contínua do processo.

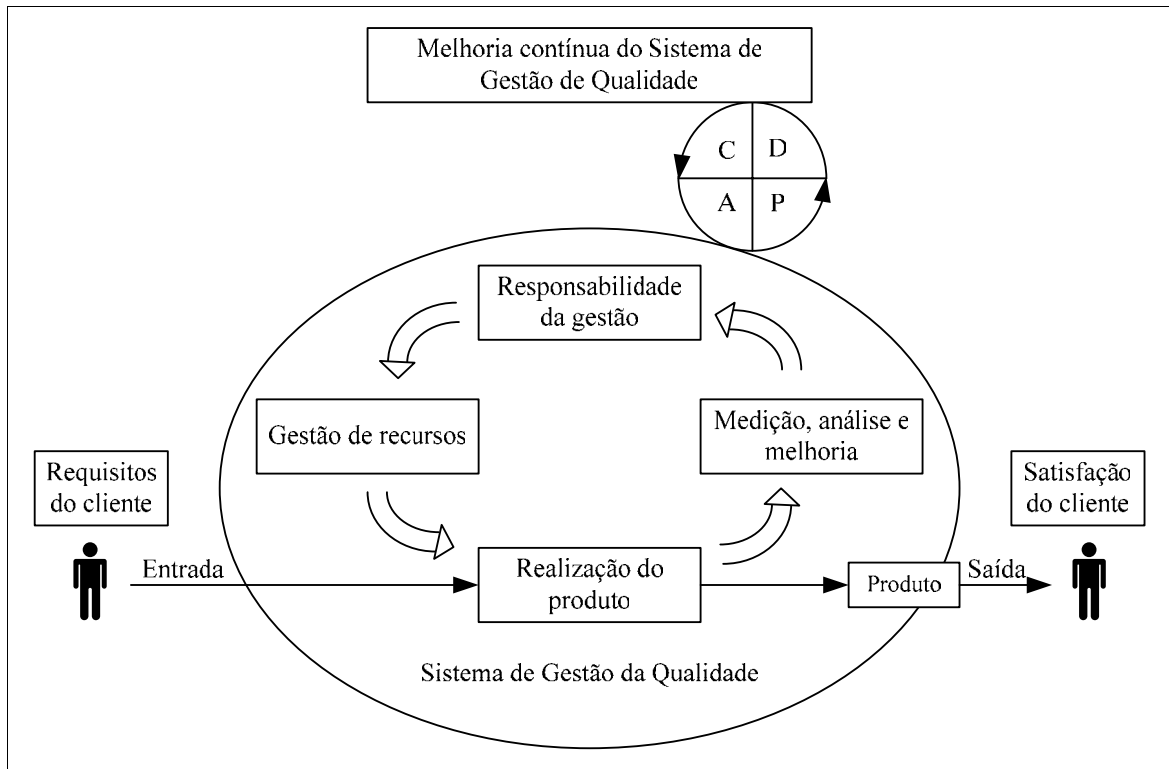


Figura 3.12– Modelo de um sistema de gestão da qualidade baseado em processos.

Na edição actual, a ISO 9001 e a ISO 9004 foram desenvolvidas como um par consistente de normas de sistemas de gestão da qualidade. A ISO 9001 especifica os requisitos para um sistema de gestão da qualidade (que pode ser utilizado para aplicação interna, para certificação ou para fins contratuais). A ISO 9004 dá orientações referentes a uma gama mais alargada de objectivos, particularmente para a melhoria contínua do desempenho e eficiência global de uma organização. Não é destinada para a certificação ou para fins contratuais. Ambas as práticas têm como base oito princípios de gestão da qualidade, que reflectem boas práticas de gestão. Os oito princípios são:

- Foco no cliente
- Liderança
- Envolvimento do pessoal
- Abordagem ao processo
- Abordagem sistémica à gestão
- Melhoria contínua
- Tomada de decisão baseada em factos
- Relacionamento com fornecedores mutuamente benéfico

A norma ISO 9001 declara que a organização tem que possuir um SGQ documentado, que defina os processos necessários para assegurar que o produto está conforme os requisitos do cliente. A organização tem que implementar, manter e melhorar continuamente o SGQ. Em relação à documentação do SGQ, a norma obriga a incluir: documentos sobre a política de qualidade e objectivos de qualidade; um manual de qualidade; procedimentos documentados; documentação sobre o planeamento, operação e controlo dos processos e registos de qualidade. A extensão da documentação do SGQ dependerá da dimensão da organização, da complexidade dos processos e da competência do pessoal.

Os processos de negócio genéricos são detalhados nas secções 5 a 8 da norma, apresentando-se em seguida os seus elementos:

- Secção 5 – Responsabilidade da gestão (compromisso da gestão; foco no cliente; política de qualidade; planeamento; responsabilidade; representante da gestão; revisão pela gestão)
- Secção 6 – Gestão de recursos (providenciar recursos; recursos humanos; infra-estrutura; ambiente de trabalho)
- Secção 7 – Realização do produto (planeamento dos processos de realização; processos relacionados com o cliente; planeamento da concepção e desenvolvimento; compra; operações para produção; propriedade do cliente; controlo da medição e monitorização de aparelhos)
- Secção 8 – Medição, análise e melhoria (planeamento; satisfação do cliente; auditorias internas; medição e monitorização dos processos e produto; não conformidade; análise de dados; melhoria; acção correctiva e preventiva).

Embora a norma ISO 9001: 2000 esteja agora centrada na melhoria contínua e numa abordagem focada nos processos, continua a apresentar a desvantagem de não ser específica para o desenvolvimento de software. Em relação ao panorama mundial, englobando indústria e serviços, a ISO registou 561747 certificações ISO 9000, espalhadas por 159 países [ISO/IEC 2002a].

### 3.3.1.4. Outros Modelos de Melhoria do Processo do Software

O número de modelos de melhoria do processo do software proposto na literatura é bastante elevado. Nesta secção apresenta-se uma descrição sucinta de quatro modelos: duas iniciativas da CE, a saber, SPIRE [1998] e BOOTSTRAP [Kuvaja 1999], um modelo específico para pequenas empresas de software, o RAPID [Rout et al. 2000] e um modelo com raízes abrangentes, TRILLIUM [Bell 1994].

#### 3.3.1.4.1. TRILLIUM

O modelo Trillium foi desenvolvido pela Bell [1994] e permite avaliar a capacidade de desenvolvimento do produto e respectivo suporte, em fornecedores de telecomunicações ou de produtos baseados em Tecnologias de Informação. Embora tenha uma orientação para as telecomunicações, tem uma importância elevada devido a três factores: (1) origem em normas conceituadas<sup>25</sup> na comunidade científica, nomeadamente, normas ISO<sup>26</sup>, Bellcore<sup>27</sup>, Malcom Baldrige<sup>28</sup>, e as normas IEEE e IEC; (2) foco no cliente e (3) perspectiva no produto.

A arquitectura proposta pelo Trillium engloba áreas de capacidade, *roadmaps* e práticas (ver Figura 6.12) e define uma escala composta por 5 níveis de maturidade progressivos:

- Nível 1. Não estruturado (risco alto)
- Nível 2. Repetível e orientado ao projecto (risco médio)
- Nível 3. Definido e orientado ao processo (risco baixo)
- Nível 4. Gerido e integrado (risco muito baixo)
- Nível 5. Completamente integrado (o risco ainda mais baixo)

---

<sup>25</sup> situação relativa ao início dos anos 90.

<sup>26</sup> ISO 9001 e ISO 9000-3.

<sup>27</sup> Bellcore's TR-NWT-000179 e TA-NWT-001315.

<sup>28</sup> critérios do Malcom Baldrige National Award.



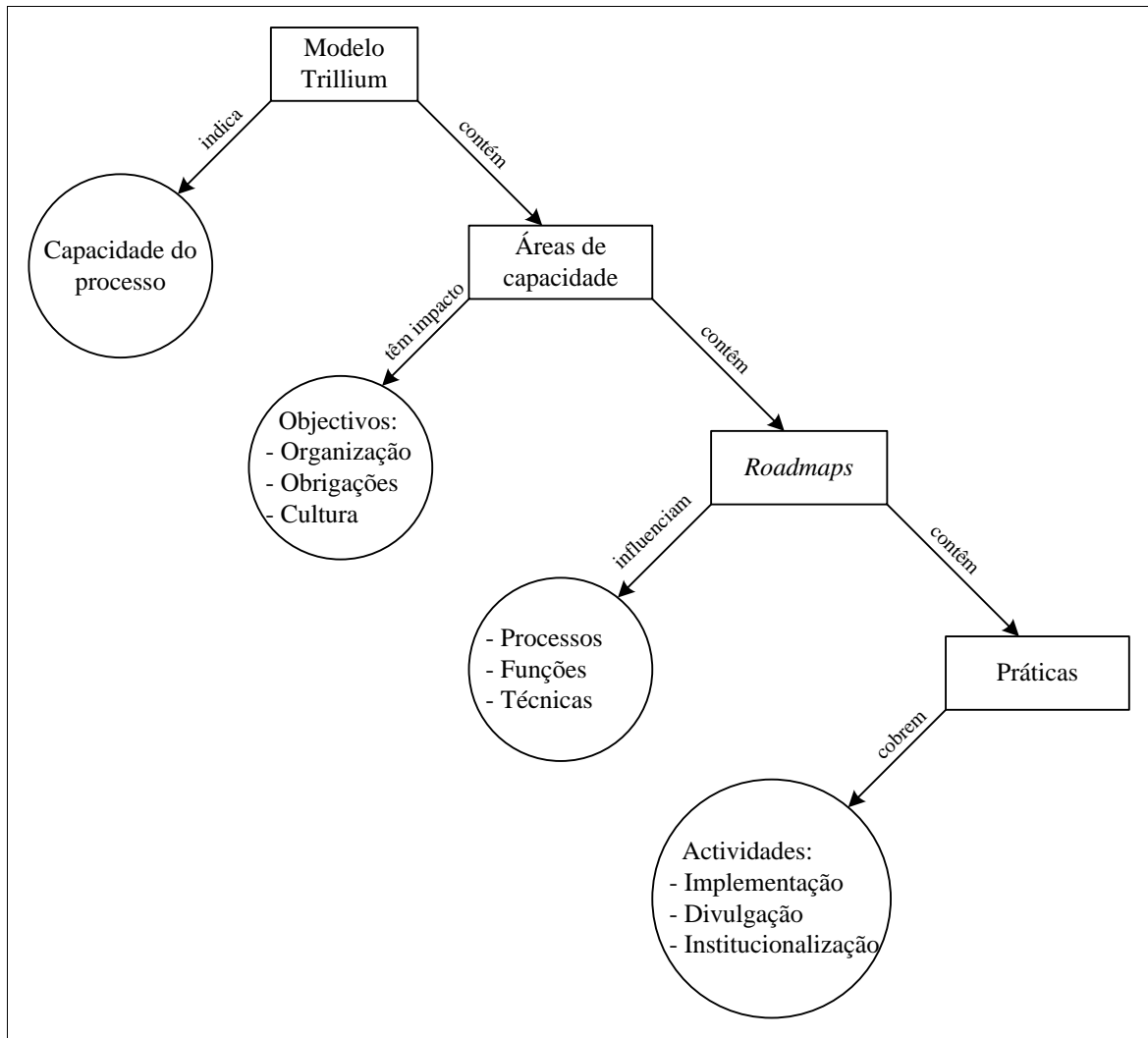


Figura 3.13 – Arquitectura do modelo Trillium [Bell 1994].

O modelo Trillium propõe oito áreas de capacidade. Cada área de capacidade tem um determinado número de *roadmaps* associado. Por sua vez, um *roadmap* é um conjunto de práticas relacionadas, que se encontram espalhadas por vários níveis de capacidade. As oito áreas de capacidade propostas pelo Trillium são [Bell 1994]:

- Qualidade do processo organizacional
- Gestão e desenvolvimento dos recursos humanos
- Processo
- Gestão
- Sistema de qualidade
- Práticas de desenvolvimento
- Ambiente de desenvolvimento
- Suporte ao cliente

O modelo Trillium fornece orientações para melhorar a capacidade de uma organização num ambiente competitivo. Neste contexto, o Trillium [Bell 1994] define capacidade como a aptidão da organização para entregar um produto de forma consistente, que vá ao encontro das expectativas do cliente, com um número mínimo de defeitos e no menor período de tempo.

#### **3.3.1.4.2. RAPID (Rapid Assessment for Process Improvement for Software Development)**

Este método de melhoria do processo de software foi desenvolvido na Universidade de Griffith, Austrália e aplicado em vários<sup>29</sup> projectos de melhoria em pequenas e médias empresas de desenvolvimento de software da região [Rout et al. 2000]. O método pretende ter uma aplicação rápida (as entrevistas consomem apenas 1 dia) e tem como objectivos, cobrir as necessidades das PME's de desenvolvimento de software e cumprir a conformidade com a norma ISO/IEC 15504. Coloca como restrição a inadequação à gestão de fornecedores.

O RAPID inclui apenas a avaliação de oito processos. Os processos principais do método são elicitação de requisitos e desenvolvimento do software. O processo de desenvolvimento de software é tratado de uma forma agregada. Os processos importantes para apoiar o desempenho dos processos e gerir os produtos de trabalho, ou seja, gestão de projectos, gestão de configurações, garantia da qualidade e resolução de problemas. Finalmente, os processos incluídos na perspectiva de suportar o alcance dos níveis de maturidade mais elevados, nomeadamente gestão do risco e estabelecimento do processo.

Os autores reconhecem algumas fraquezas no método, nomeadamente a falta de exactidão das classificações com dificuldades inerentes no *benchmarking*, não validação das entrevistas através de outras técnicas (por exemplo, inspecções aos produtos de trabalho) e a avaliação do desenvolvimento de forma agregada, não permitindo a

---

<sup>29</sup> o artigo relata a experiência realizada em 25 organizações, com o nº de colaboradores a variar entre 3 e 120.

granularidade adequada para as tarefas de engenharia. Em relação aos aspectos positivos, é referido o forte envolvimento dos profissionais do software e o *feedback* dos participantes referindo a concordância entre os resultados e a sua experiência.

### 3.3.1.4.3. BOOTSTRAP

A metodologia Bootstrap deve-se a um consórcio europeu desenvolvido com fundos da Comunidade Europeia. A missão escolhida na data<sup>30</sup> do arranque foi “O projecto Bootstrap deve fertilizar os ambientes para a introdução na indústria de moderna tecnologia de software. Pretende-se que o faça através da análise do estado actual da engenharia de software na indústria, identificação de mudanças potenciais e motivando a aceitação de novos contextos para a engenharia de software”.

Esta etapa inicial concluiu-se em 1992 com o lançamento do Bootstrap, versão 2.22 [Kuvaja et al. 1994]. Esta primeira versão inspirou-se directamente no CMM e na família ISO 9000. Após o fim do projecto ESPRIT, alguns dos parceiros decidiram explorar os resultados, estabelecendo uma organização internacional para continuar o desenvolvimento e utilização da metodologia. Fundaram o instituto BOOTSTRAP.

Foram estabelecidos os objectivos da metodologia BOOTSTRAP [Kuvaja 1999]:

- Fornecer suporte para a avaliação da capacidade do processo em relação a um conjunto reconhecido de melhores práticas de engenharia de software
- Utilizar normas na área de engenharia de software como origem das melhores práticas utilizadas
- Assegurar uma avaliação fiável e repetível
- Identificar pontos fortes e fracos nos processos da organização
- Fornecer dados de *benchmarking* para comparação com os resultados da avaliação
- Fazer com que os resultados sejam uma base fiável e adaptada para o plano de melhoria

---

<sup>30</sup> em 1989

- Planear as acções de melhoria alinhadas com os objectivos do negócio

Foi desenvolvida então a versão 3 da metodologia em conformidade com a norma emergente ISO/IEC 15504 e alinhada com a norma ISO/IEC 12207. Esta versão inclui as seguintes características:

- Método de avaliação. Engloba as fases de preparação, execução e planeamento da melhoria;
- Modelo do processo. Contém a dimensão do processo, a dimensão da capacidade e a dimensão de suporte à tecnologia.
- Níveis de capacidade. Alinhados com os requisitos da norma ISO/IEC 15504. Vão desde o nível 0, denominado incompleto até ao nível 5, designado optimizado.
- Escala, classificações e apresentação dos resultados. A escala e as classificações estão também em conformidade com a norma ISO/IEC 15504. Os resultados são produzidos no nível do projecto e no nível da unidade de produção de software<sup>31</sup>.
- Orientações para a melhoria do processo. A metodologia considera as necessidades e objectivos da organização, o perfil de capacidade dos processos e *benchmarks* da indústria como a base para as orientações da melhoria do processo.

A metodologia BOOTSTRAP apresenta como vantagens principais: constante evolução; alinhamento com a norma ISO/IEC 15504 e dados de *benchmark* para comparação de resultados.

#### 3.3.1.4.4. SPIRE<sup>32</sup>

O SPIRE é também um projecto financiado pela Comunidade Europeia e tem como base uma abordagem muito pragmática à Melhoria do Processo de Software

---

<sup>31</sup> Software Process Unit – SPU.

<sup>32</sup> Software Process Improvement in Regions of Europe.

(MPS). O objectivo do SPIRE é “ajudar os gestores e o pessoal nas pequenas<sup>33</sup> empresas de software (PES) a melhorar o seu negócio através da melhoria sistemática do processo de software” [SPIRE 1998].

Nesta perspectiva a MPS é abordada como “aprender a trabalhar de uma forma mais inteligente, de modo a fornecer software melhor, mais barato e mais rápido. Se fizermos isto de forma sistemática, com uma incidência clara nas necessidades de negócio, iremos melhorar o nosso negócio” [SPIRE 1998].

A filosofia do SPIRE baseia-se num número de proposições e crenças, importantes não só para as PES, mas também para qualquer organização:

- MPS faz sentido para o negócio mas requer investimento, não só monetário, mas também da gestão
- Para assegurar o retorno dum investimento na MPS é necessário um forte alinhamento com as necessidades do negócio
- A melhoria tem que se basear numa avaliação objectiva dos processos
- Para ter sucesso, um projecto de MPS deve ser gerido como outro projecto qualquer
- Os ganhos a longo prazo só serão obtidos através da melhoria contínua, revendo os resultados obtidos e iniciando novos projectos de MPS.
- Os esforços de MPS devem ser conduzidos com uma abordagem disciplinada e passo a passo
- Para o sucesso da MPS, as pessoas e a cultura são tão importantes como a tecnologia – se não mesmo mais
- A MPS normalmente falha se a gestão de topo não demonstrar um compromisso genuíno com o programa

O SPIRE tem como base a norma ISO/IEC 15504 e define a MPS de forma geral como englobando: (1) definir os processos correctos de preferência utilizando um modelo existente<sup>34</sup>; (2) seguir os processos definidos e refiná-los até atingir uma forma de operação efectiva e eficiente e (3) procurar continuamente áreas de melhoria, como filosofia do dia a dia do negócio.

---

<sup>33</sup> empresas com menos de 50 funcionários.

<sup>34</sup> CMM, ISO/IEC 15504 ou ISO 9001.

A figura 3.14 apresenta os passos de melhoria propostos no SPIRE. Em relação à avaliação do processo do software, o SPIRE utiliza os quatro primeiros níveis da norma ISO/IEC 15504, isto é: incompleto (0); executado (1); gerido (2) e estabelecido (3) e os cinco atributos do processo característicos destes níveis.

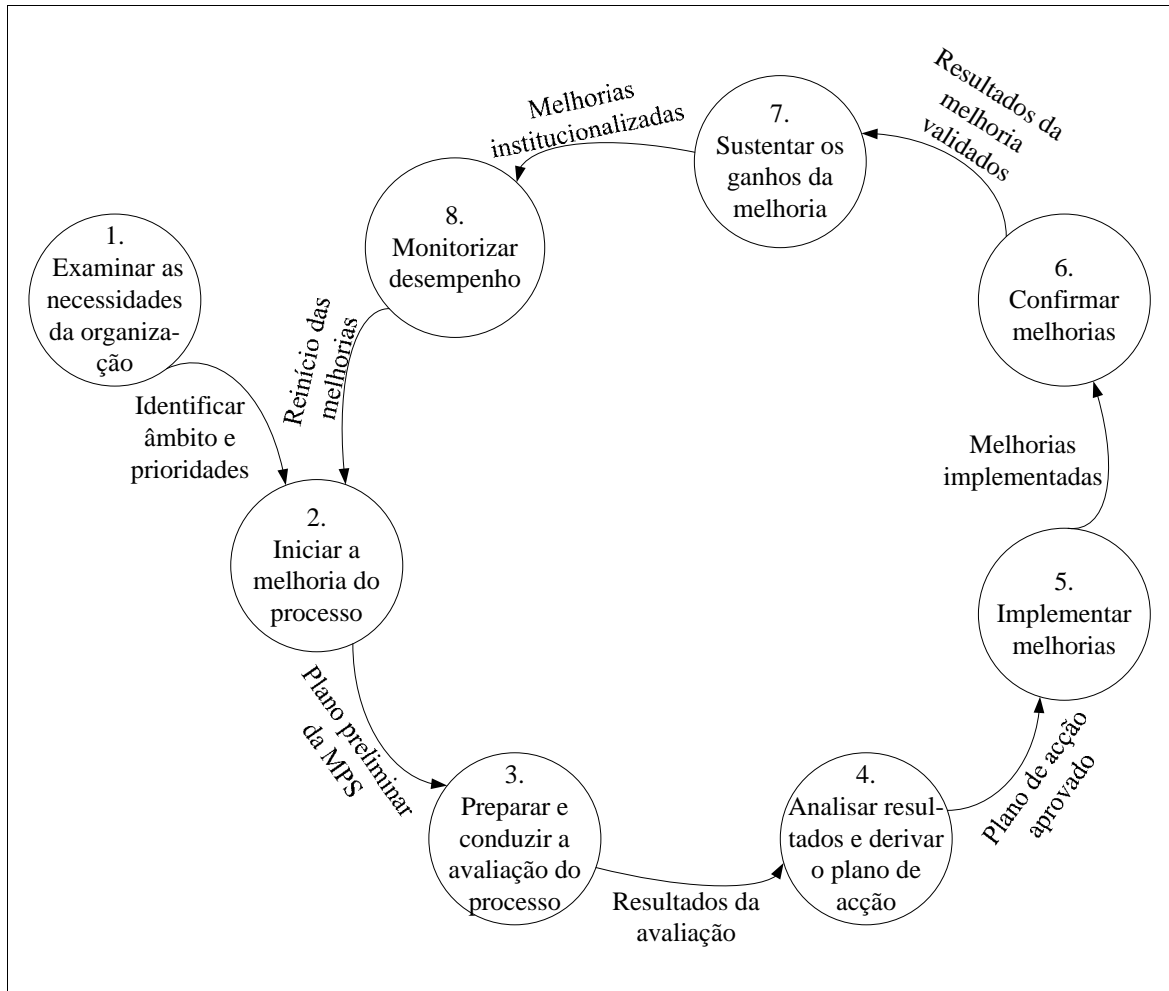


Figura 3.14 – Passos na MPS (SPIRE [1998]).

O SPIRE utiliza as categorias de processos e os processos propostos pela ISO/IEC 15504. Cada processo é categorizado por: “um nome; uma declaração do seu propósito; a motivação; as suas entradas, actividades e saídas; as melhores práticas da indústria e os resultados esperados” [SPIRE 1998].

Em resumo, o SPIRE tem a ver com a MPS nas PES, com foco especial no retorno de resultados e no alinhamento com os objectivos do negócio.

### 3.3.2. Paradigma Analítico

O paradigma analítico baseia-se na recolha de evidências quantitativas para determinar onde as melhorias são necessárias e avaliar se a iniciativa de melhoria foi ou não um sucesso. Nesta secção apresenta-se um pequeno conjunto de métodos característicos deste paradigma.

#### 3.3.2.1. Controlo Estatístico do Processo

O método de controlo estatístico do processo<sup>35</sup> é utilizado há muitos anos por diversos tipos de indústrias<sup>36</sup>, como base para a tomada de decisão sobre os processos e para prever o comportamento do processo [Florac e Carleton 1999]. Os gráficos<sup>37</sup> de controlo estatístico do processo foram desenvolvidos por Stewhart [1931]. No sentido mais puro, um gráfico de controlo é uma representação que mostra o que se está a passar num processo ao longo do tempo. Segundo Fryman [2002], os benefícios dos gráficos de controlo são:

- Obtenção de um conhecimento profundo do processo
- Monitorização do desempenho do processo
- Identificação das variações no desempenho e tipos de variações
- Monitorização e redução das variações
- Determinação da capacidade do processo
- Controlo do processo
- Identificação da presença de tendências e outros padrões no processo

Controlar um processo significa mantê-lo nas suas fronteiras normais de desempenho, ou seja, fazer com que tenha um comportamento consistente. Para atingir este objectivo é necessário [Florac e Carleton 1999]: (1) determinar se o processo está ou não controlado (é estável em relação à variabilidade inerente do desempenho

---

<sup>35</sup> *Statistical Process Control (SPC)*.

<sup>36</sup> manufactura, indústria química, petróleo e aeroespacial.

<sup>37</sup> também designados gráficos de controlo.

medido); (2) identificar variações no desempenho causadas por variações especiais e (3) eliminar as fontes das causas especiais, para estabilizar o processo.

Um gráfico de controlo baseia-se na curva normal (ver figura 3.15), onde  $\mu + 3\sigma$  é substituído pelo termo Limite Superior de Controlo (LSC) e  $\mu - 3\sigma$  é substituído pelo termo Limite Inferior de Controlo (LIC). Da mesma forma,  $\mu \pm 2\sigma$  são denominados limite superior e inferior de aviso.

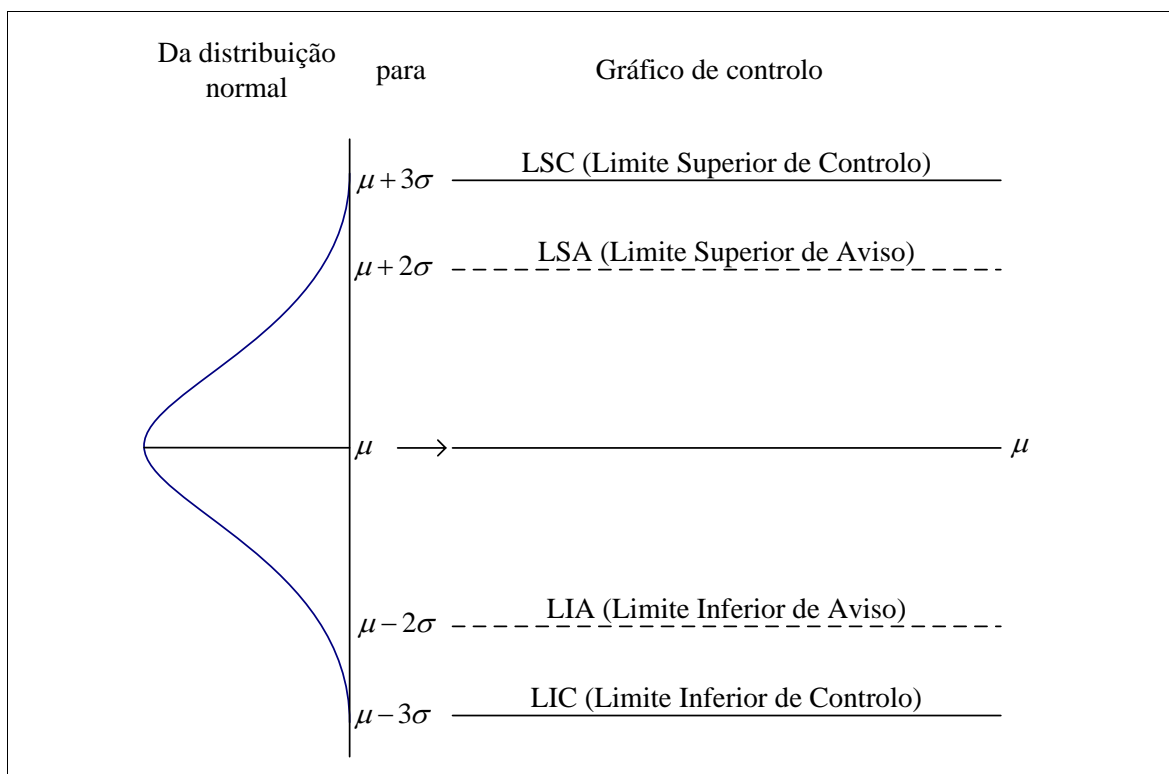


Figura 3.15 – Correspondência entre a curva normal e o gráfico de controlo (adaptado de Fryman [2002]).

Quando consideramos a variabilidade que ocorre com um processo, há dois tipos de variação que podem ocorrer. Os dois tipos são: variação por causa comum e variação por causa especial<sup>38</sup>, verificando-se que estes tipos de variações são responsáveis por toda a variação exibida por qualquer processo, isto é,

$$[\text{Variação total}] = [\text{variação por causa comum}] + [\text{variação por causa especial}]$$

<sup>38</sup> também denominada por causa assinalável.



A variação por causa comum deve-se à interacção natural ou inerente entre os diversos componentes do processo (pessoas, máquinas, material, ambiente e métodos). Em relação ao gráfico de controlo é a região representada por  $\mu \pm 3\sigma$ . As variações por uma causa especial surgem de eventos que não fazem parte do processo normal e representam alterações anormais, persistentes ou repentinas, a um ou mais componentes do processo. Como afirma Deming [1993] “quando o gráfico de controlo indica que nenhuma causa especial está presente, o processo diz-se que está em controlo estatístico, ou estável. A média e os limites de variação são previsíveis com um grande grau de confiança”.

Dependendo do tipo de medições, há duas classes de gráficos de controlo a utilizar: gráficos de controlo de variações e gráficos de controlo de atributos. Uma descrição detalhada sobre a utilização de gráficos de controlo pode ser encontrada em Montgomery [1996] e Wheeler e Poling [1998].

Verifica-se que a comunidade de software olha para os gráficos de controlo como uma ferramenta de análise adaptada a outras indústrias, onde os processos se caracterizam pela sua repetibilidade e assim os princípios do controlo estatístico do processo são ainda pouco utilizados nas organizações de software.

### **3.3.2.2. Simulação do Processo – Dinâmica de Sistemas**

A modelação dinâmica e a simulação como ferramentas de melhoria do processo têm sido intensamente utilizadas na área de manufactura. A simulação é mais do que apenas uma tecnologia, uma vez que somos forçados a pensar em termos globais sobre o comportamento do sistema e sobre o facto de um sistema ser mais do que a soma dos seus componentes [Christie 1999].

A Dinâmica de Sistemas foi inicialmente introduzida por Forrester [1961] e surgiu como uma nova metodologia adaptada à modelação do comportamento de sistemas socio-económicos complexos. As raízes da dinâmica de sistemas encontram-se na teoria de controlo de *feedback* [Roberts 1964] e a abordagem baseia-se numa

perspectiva holística dos problemas de gestão focada nos aspectos humanos do comportamento do sistema [Rodrigues e Williams 1997].

Forrester [1961] define a dinâmica de sistemas como “... a investigação das características informação - *feedback* dos sistemas geridos e a utilização de modelos para o desenho de formas organizacionais e orientações políticas melhoradas”.

Para Coyle [1979], “Dinâmica de sistemas é o ramo da teoria de controlo que trata de sistemas socio-económicos e o ramo das ciências da gestão que trata dos problemas do controlo”.

Finalmente, Wolstenholme [1990] define dinâmica de sistemas como “um método rigoroso para a descrição qualitativa, exploração e análise de sistemas complexos em termos dos seus processos, informação, fronteiras organizacionais e estratégias que facilita a modelação e análise da simulação quantitativa, para a concepção da estrutura e comportamento do sistema”.

Os modelos da dinâmica de sistemas são apresentados em duas formas equivalentes: o diagrama de influência e sempre que necessário o modelo de simulação [Coyle 1996]. Estas duas formas tornam a comunicação com o cliente ou patrocinador, simples e eficaz.

A abordagem da dinâmica de sistemas a problemas complexos foca nos processos de *feedback*. A premissa é que o comportamento dinâmico é uma consequência da estrutura do sistema [Richardson e Pugh 1981]. Sucintamente, *feedback* é a transmissão e retorno de informação [Greenberger et al. 1976] e um ciclo *feedback* é uma sequência fechada de causas e efeitos, um caminho fechado de acção e informação (ver figura 3.16). O diagrama de influência mostra as influências a trabalhar no sistema e a interligação entre elas é a causa do comportamento dinâmico do sistema. Um diagrama de influência mostra um conjunto interligado de ciclos de *feedback*. Não são apenas a fundação a partir da qual os modelos quantitativos são construídos, mas são também uma ferramenta valiosa para compreender e descrever sistemas [Coyle 1996].

O modelo de simulação ou diagrama de fluxo permite obter um modelo quantitativo do sistema. Enquanto o diagrama de influência é desenvolvido e utilizado para uma análise qualitativa, o diagrama de fluxo permite criar um conjunto de equações que representam o sistema. A execução do conjunto de equações ao longo do tempo permite ver o comportamento previsto do sistema. A notação proposta (ver figura 3.16) para o diagrama de fluxo inclui níveis (acumulações), taxas (fluxos) e origem / destino dos materiais [Rubin et al. 1995].

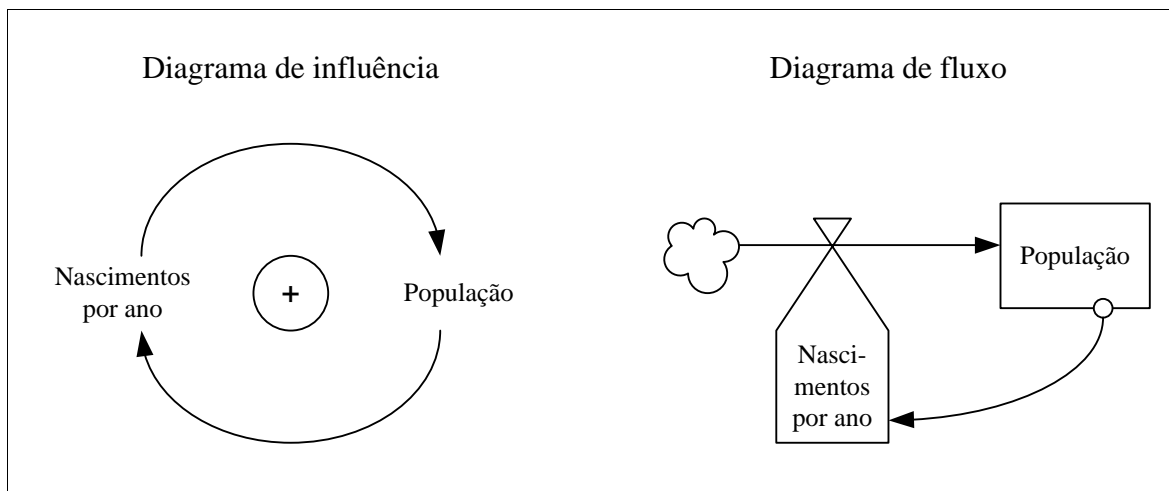


Figura 3.16 – Crescimento da população (adaptado de Richardson e Pugh [1981]).

Na literatura encontra-se um número significativo de aplicações da Dinâmica de Sistemas (DS) ao desenvolvimento de software. A primeira aplicação foi proposta por Abdel-Hamid e Madnick [1982], conduzindo mais tarde a um modelo genérico do processo de desenvolvimento de software [Abdel-Hamid e Madnick 1991]. Rodrigues [2000] desenvolveu uma metodologia para integrar a dinâmica de sistemas e os métodos de gestão de projectos tradicionais. A área de aplicação da metodologia integrada foi o desenvolvimento de software. Christie [1999] analisa a aplicação da dinâmica de sistemas à melhoria do processo de software. Outros trabalhos de aplicação da DS ao desenvolvimento de software devem-se a Rubin et al. [1995], Tvdet e Collofello [1995], Madachy [1996], Kellner et al. [1999] e Ruiz et al. [2002].

Os objectivos comuns dos modelos de simulação são fornecer mecanismos para experimentar, prever, compreender e responder a questões *what-if* [Ruiz et al. 2002]. Em relação ao processo de software, a dinâmica de sistemas pode ajudar em três

aspectos [Christie 1999]: (1) previsão dos custos de um projecto; (2) análise quantitativa do desempenho e (3) previsão do comportamento futuro de um projecto.

### 3.3.2.3. Personal Software Process (PSP)

O *Personal Software Process* (PSP) foi desenvolvido por Humphrey. Segundo o próprio, “foi concebido para ajudar os engenheiros do software a fazerem um bom trabalho” [Humphrey 1997]. Em qualquer profissão, para se fazer um trabalho competente é necessária a utilização disciplinada de práticas estabelecidas. O desenvolvimento de software não é diferente, não é disciplina ou criatividade, é antes trazer disciplina para as actividades de forma a criar um ambiente propício à criatividade [Ferguson et al. 1997]. O referencial proposto pelo PSP contém 7 passos (ver figura 3.17) e fornece ferramentas de medição e análise que permitem a compreensão e melhoria dos processos individuais [Humphrey 1996].

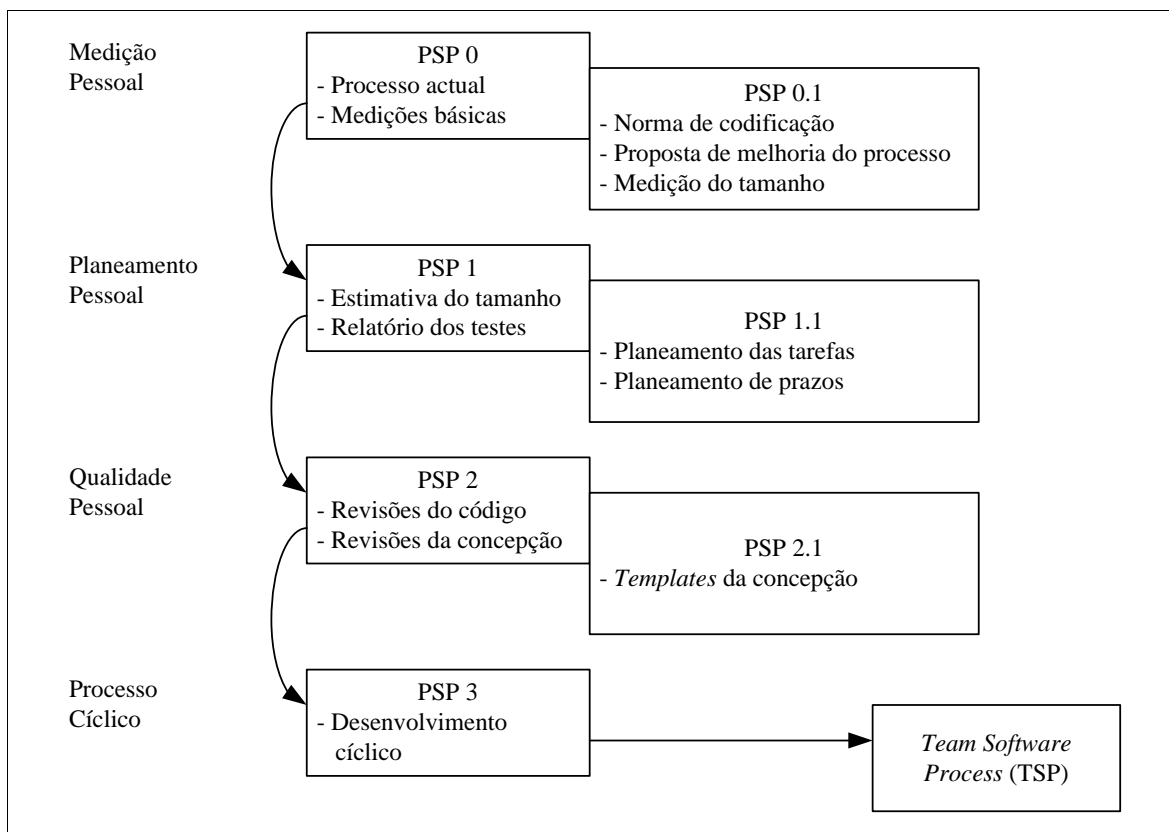


Figura 3.17 – Evolução do processo no PSP (adaptado de Humphrey [1996]).

As medições efectuadas nos cursos de PSP mostram uma melhoria de produtividade de cerca de 20% e um aumento de qualidade na ordem de 5 vezes (Ferguson et al. [1997]).

### 3.3.3. Metodologias Ágeis

No contexto da melhoria do processo do software, é importante situar as metodologias ágeis. Estas metodologias surgem nos finais dos anos 90 como resposta à complexidade crescente das novas tecnologias, a requisitos em constante mudança e a recursos humanos instáveis.

Em Fevereiro de 2001, um grupo de especialistas em metodologias de desenvolvimento de software reunidos numa convenção criaram o “Manifesto Ágil”. O manifesto apresenta os princípios dos processos ágeis, que engloba um conjunto crescente de novas abordagens ao desenvolvimento de sistemas. O manifesto ágil valoriza [Orr 2002]:

- Indivíduos e interacções em vez de processos e ferramentas
- Software a trabalhar em vez de documentação
- Colaboração com o cliente em vez de negociação do contrato
- Resposta à mudança em vez de seguir o plano

Para Highsmith [2002b] há três aspectos principais relacionados com as metodologias ágeis versus rigorosas: (a) a perspectiva do caos ou da ordem; (b) valores e princípios da colaboração e (c) “apenas a metodologia suficiente”. Os processos ágeis (ver figura 3.18) apresentam dois atributos chave para lidar com o desenvolvimento de sistemas [Schwaber 2001]:

1. Emergência. A arquitectura, concepção e funcionalidade do produto emergem através do desenvolvimento do projecto, a partir da “visão” do produto.
2. Auto-organização. A equipa de desenvolvimento responde à turbulência do ambiente através da auto adaptação ao trabalho a realizar.

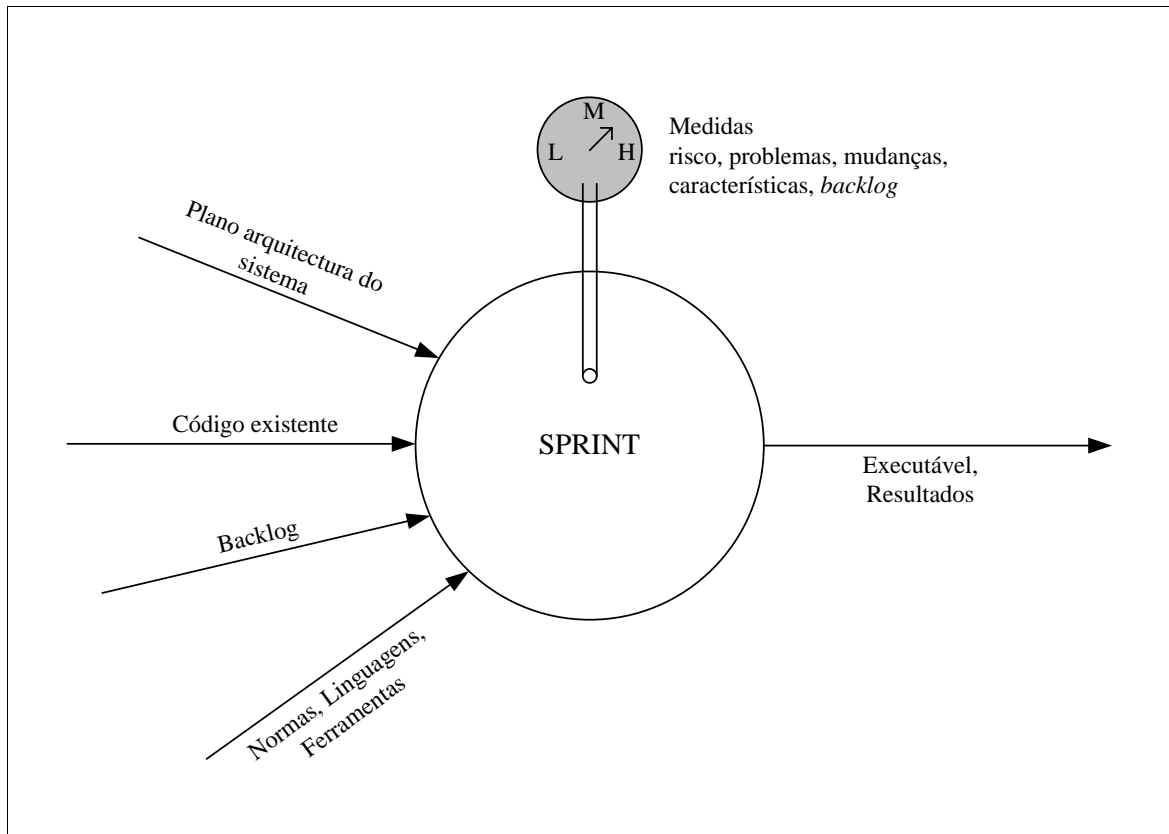


Figura 3.18 – Processo ágil (adaptado de Schwaber [2001]).

Como Fowler [2003] descreve “Os métodos leves são adaptativos em vez de previsíveis... Tentam adaptar-se e prosperar na mudança, até ao ponto de se mudarem a eles próprios” e “mudança é a famosa constante do software” [Jacobson 2002].

Os processos ágeis caracterizam-se então por serem executados por equipas auto-organizadas. A forma de desenvolver o trabalho está muito pouco definida, o que provoca que poucos produtos de trabalho sejam obrigatórios. Entregam o sistema a funcionar de uma forma incremental e iterativa, como resposta a requisitos emergentes e em mudança.

Apresenta-se em seguida uma breve descrição de algumas metodologias ágeis mais divulgadas.

## O eXtreme Programming (XP)

O XP foi desenvolvido por Kent Beck, Ward Cunningham e Ron Jeffries e é a abordagem ágil que gerou mais interesse na comunidade de software [Highsmith 2002a]. O XP defende os valores da comunidade, simplicidade, *feedback* e coragem e baseia-se em 12 práticas chave [Beck 2002]: jogo de planeamento, versões pequenas, metáfora, concepção simples, teste, *refactoring*, programação em pares, posse colectiva, integração contínua, semana de 40 horas, cliente nas instalações e normas de codificação. “O XP fornece um sistema de práticas dinâmicas cuja integridade como uma unidade holística foi provada” [Highsmith 2002a]. O XP apresenta a limitação de não ser escalável mas tem como ponto forte a criação de equipas motivadas para construir software de qualidade, de uma forma iterativa.

## Scrum

No Scrum [Schwaber e Beedle 2002] o desenvolvimento do software é executado com iterações de 1 mês, denominadas sprints. O Scrum é um processo para gerir e controlar o desenvolvimento. Os requisitos complexos e emergentes são tratados, considerando apenas um pequeno subconjunto em cada iteração. Durante cada sprint, a equipa mantém breves reuniões diárias e no fim do sprint é entregue um incremento do produto executável. O Scrum é ideal para projectos com requisitos em constante mudança como produtos para novos mercados ou projectos Web [Cohn e Ford 2003].

## Dynamic Systems Development Method (DSDM)

O DSDM é a formalização do RAD num processo iterativo e incremental. O DSDM não utiliza algumas práticas comuns às abordagens ágeis, por exemplo, artefactos mínimos, reuniões diárias e incrementos dos produtos em cada iteração. Mas inclui muitas ideias destas abordagens [Stapleton 1998]:

- Equipas incluindo profissionais de desenvolvimento e clientes
- As tarefas não estão definidas
- Todo o projecto é adaptativo
- Os produtos são avaliados no fim de cada iteração

Segundo Schwaber [2001], o DSDM é uma boa metodologia para organizações que pretendam iniciar a sua transição para os processos ágeis, contudo, o DSDM não engloba todos os benefícios das abordagens ágeis.

Apresentou-se nesta secção uma pequena amostra de abordagens ágeis, mas há outras metodologias também bastante referenciadas, por exemplo, FDD – *Feature Driven Development* [Coad et al. 1999], ASD – *Adaptative Software Development* [Highsmith 2000a], LD – *Lean Development* [Poppendieck 2001] e Crystal [Cockburn 2002a].

No contexto da concepção do modelo 2MPspe há 3 aspectos dos processos ágeis que foram considerados:

1. Como Paulk [2002] defende, “O CMM diz o que fazer em termos globais, mas não diz como fazê-lo; as metodologias ágeis fornecem um conjunto de boas práticas que contêm um modelo de implementação para um ambiente determinado”. Os modelos de melhoria do processo de software e os métodos ágeis não são incompatíveis. Uns centram a sua atenção no “como” executar o desenvolvimento, os outros focam a atenção no “que” tem que ser feito para desenvolver software. De qualquer forma, o 2MPspe deve facilitar a percepção da forma como a agilidade deve entrar no processo de desenvolvimento de software.
2. Segundo Cockburn [2002b], “... metade da agilidade surge de mudar alguns dos documentos escritos pela comunicação informal entre os membros da equipa, trocando a memória organizacional do grupo de externa, para conhecimento tácito”. A estratégia de gestão de conhecimento deve ser tratada explicitamente para balancear a flexibilidade e rigidez nos processos.
3. Schwaber [2001] afirma “Os processos de desenvolvimento de sistemas nunca são completamente ágeis nem completamente definidos. Alguns processos ágeis usam processos para gerir o risco”. Mesmo as metodologias ditas ágeis possuem processos definidos (rígidos).



Neste capítulo desenvolveu-se uma revisão sobre modelos de melhoria do processo e analisou-se as abordagens ágeis. No próximo capítulo estuda-se os resultados da aplicação prática dos modelos de melhoria e apresenta-se a problemática estudada.

## Capítulo 4

---

### **4. Aplicação dos Modelos de Melhoria do Processo de Software**

Nos dois capítulos anteriores apresentou-se o enquadramento da área de melhoria do processo de software na disciplina de engenharia de software e descreveram-se os modelos de melhoria de software mais reconhecidos pela comunidade de software.

A questão inicialmente colocada, no âmbito deste trabalho, refere-se à seguinte problemática: há uma evidência significativa que mostra que um projecto de melhoria do processo de software com sucesso resultará em melhorias na qualidade do produto e no valor do negócio, mas há também evidência que grande parte destes projectos falha.

Este capítulo apresenta o estudo realizado sobre esta problemática, incidindo numa revisão exaustiva da literatura e em trabalho de campo. Na primeira secção apresentam-se casos práticos – recolhidos na literatura – com foco nos resultados obtidos. Na secção seguinte, investigam-se os trabalhos realizados por outros autores sobre os factores de sucesso num projecto de melhoria do processo de software. A terceira secção descreve o trabalho de campo, centrado na aplicação do CMM e ISO/IEC 15504 em três empresas de desenvolvimento de software. Apresenta-se uma

breve descrição de cada caso prático e elaboram-se algumas “conclusões preliminares” sobre a aplicação dos modelos de melhoria do processo de software mais reconhecidos<sup>1</sup>.

#### 4.1. Casos de Sucesso e Insucesso dos Modelos de Melhoria

Nesta secção apresenta-se uma revisão dos resultados verificados nas iniciativas de melhoria do processo de software. Os resultados centram-se nos benefícios obtidos através da produtividade melhorada e da diminuição do *rework*, incluindo naturalmente dados sobre o ROI<sup>2</sup>.

Um primeiro aspecto a ter em conta é que medir apenas o ROI pode ser redutor na avaliação do efeito de um projecto de MPS, uma vez que há outros aspectos<sup>3</sup>, que são também importantes, por exemplo, a moral dos profissionais de software, a satisfação do cliente e a redução do risco. Assim, considerando esta perspectiva, procuram-se outras métricas relacionadas com o tempo de ciclo<sup>4</sup>, custos de manutenção, produtividade, satisfação dos clientes e efeitos nos profissionais do software.

O trabalho de Herbsleb et al. [1994] faz o tratamento estatístico dos relatos de 13 organizações envolvidas num projecto MPS baseado no CMM. Na tabela 4.1 apresentam-se as conclusões<sup>5</sup> dos autores.

O ROI é definido como o rácio dos benefícios medidos em relação aos custos medidos. Os benefícios incluem as poupanças devido ao aumento da produtividade e à diminuição dos defeitos; os custos incluem a equipa de MPS, a formação e as avaliações.

---

<sup>1</sup> CMM e ISO/IEC 15504.

<sup>2</sup> Retorno do Investimento.

<sup>3</sup> por vezes difíceis de medir.

<sup>4</sup> *cycle time* – tempo decorrido desde a definição dos requisitos do produto até à entrega do produto ao cliente.

<sup>5</sup> baseadas em organizações que passaram do nível 1 para o nível 2 ou do nível 2 para o nível 3 (CMM).

<b>Parâmetro medido</b>	<b>Amostra (nº de organizações)</b>	<b>Mediana</b>	<b>Min.</b>	<b>Max.</b>
Custo/Engenheiro de software/ano (\$)	5	\$1375	\$490	2004\$
Produtividade / ano (%)	4	35%	9%	67%
Tempo de Desenvolvimento (%)	2	19%	15%	23%
Defeitos após entrega (%)	5	39%	10%	94%
ROI	5	5 : 1	4,2 : 1	8,8 : 1

*Tabela 4.1 – Benefícios da MPS [Herbsleb et al. 1994].*

McGarry e Decker [2002] descrevem o caminho de melhoria do processo no centro SEAS<sup>6</sup>, durante o período entre 1994 e Novembro de 1998. Este esforço culminou na obtenção do nível 5 do CMM e simultaneamente da certificação ISO 9001. Os autores relatam as conclusões obtidas de cerca de 30 projectos activos durante o período, verificando-se uma melhoria de aproximadamente 10% por ano na produtividade e na qualidade. Os defeitos referem-se ao número de erros encontrados<sup>7</sup> antes da entrega do produto e que provocaram alterações no código executável. Não inclui erros encontrados nos testes às unidades e na documentação. São referidas também algumas melhorias<sup>8</sup> no tempo de ciclo e precisão das estimativas [McGarry 2001].

Kolmel e Eisenbiegler [2000] realizaram um estudo empírico envolvendo empresas europeias. O questionário que serviu de suporte ao estudo foi enviado a 180 empresas envolvidas nos PIEs<sup>9</sup> (*Process Improvement Experiments*) com uma taxa de retorno de 55%. Os resultados são apresentados na figura 4.1, verificando-se um número de resposta mais baixo (20) no parâmetro “tempo para o mercado” e mais elevado (45) na “melhoria da qualidade”. A “previsibilidade” refere-se à capacidade de cumprir os prazos e o orçamento. Os “benefícios no mercado” são avaliados pela quota de mercado, aumento das vendas, imagem da empresa e satisfação do cliente.

<sup>6</sup> System, Engineering and Analysis Support Center da Computer Sciences Corporation (CSC).

<sup>7</sup> por equipas independentes.

<sup>8</sup> não quantificadas.

<sup>9</sup> programas de MPS patrocinados pela CE no âmbito da iniciativa ESSI.

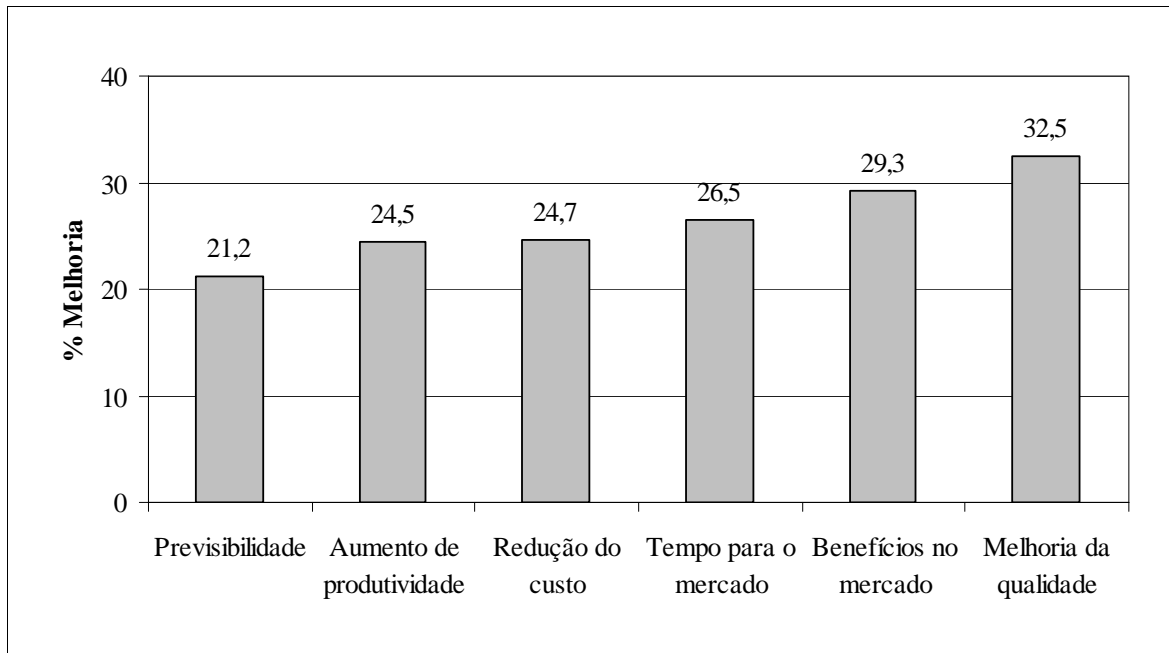


Figura 4.1 – Resultados quantitativos do estudo empírico (adaptado de Kolmel e Eisenbiegler [2000]).

Os resultados apresentam desvio padrão entre 16% e 24% indicando uma validade e fiabilidade elevadas da amostra. Os dados quantitativos sobre os resultados das iniciativas de MPS nas organizações europeias são escassos, deste facto deriva a importância deste estudo.

Os *trials* do SPICE [SPICE 1998] apresentam algumas observações a ter em conta: (1) aproximadamente 60% dos participantes “não acreditam que a avaliação realizada teve um grande impacto na organização”; (2) apenas 28% está em desacordo com a afirmação “mudou pouca coisa desde a avaliação” e (3) 79% acredita que o programa de MPS foi “ultrapassado pelos eventos e crises do dia a dia”. Conradi e Fuggetta [2002], após a realização de diversos projectos em PME's situadas em Itália e na Escandinávia, concluem também que as prioridades de curto prazo e a turbulência dos mercados podem impedir ou mesmo abortar os esforços de MPS. O grupo Gartner estimou que 70% dos projectos de MPS falham num prazo de 3 anos [Evans 2003].

O relatório ESPI [1999] relata que 70% das avaliações CMM nos EUA não dão origem à elaboração do plano de melhoria do processo, ou seja, as organizações verificam o seu estado em relação ao referencial CMM, mas não iniciam nenhuma acção de melhoria.

O próprio SEI apresenta resultados que reforçam a ideia do alto risco associado a um projecto de MPS. O *survey* realizado por Goldenson e Herbsleb [1995] com uma amostra de 61 avaliações<sup>10</sup> obteve os seguintes resultados<sup>11</sup>:

- Na questão “Qual o sucesso com que as descobertas e recomendações da avaliação foram tratados?” 70% responderam “moderado”, “limitado” ou “pouco”
- 49% afirmaram que “houve muita desilusão com a falta de melhoria verificada
- 42% disseram que a iniciativa foi ultrapassada pelos eventos, crises e outros aspectos mais prioritários”
- 72% relataram que a escassez de recursos e tempo afectou o projecto de MPS

Estas conclusões foram indicadas embora  $\frac{2}{3}$  dos inquiridos afirmassem que os esforços de MPS foram largamente determinados pelas descobertas e recomendações da avaliação realizada.

Harter et al. [2000] realizaram um estudo empírico envolvendo a análise de 30 produtos de software numa grande empresa de Tecnologias de Informação (TIs). O objectivo foi investigar o relacionamento entre a maturidade do processo, qualidade, tempo de ciclo e esforço do desenvolvimento. As conclusões indicam que maiores níveis de maturidade estão associados a uma melhoria da qualidade do produto, mas também com um aumento no esforço de desenvolvimento. Contudo, a redução do tempo do ciclo e do esforço associado à melhoria de qualidade compensa o esforço associado à melhoria e assim o efeito líquido é a redução do tempo de ciclo e do esforço de desenvolvimento. Os autores desenvolveram um modelo para estudar as quatro variáveis de uma forma interrelacionada. As estimativas resultantes do estudo empírico são apresentadas na tabela 4.2.

---

<sup>10</sup> em diferentes empresas e incluindo 138 inquiridos.

<sup>11</sup> são apresentados apenas alguns valores.

Nível Maturidade CMM	Qualidade (LOC <sup>12</sup> / erro)	Tempo de ciclo (dias)	Esforço (pessoas-mês)
1	196	927	202
2	593	743	179
3	1134	653	167

Tabela 4.2 – Benefícios estimados da maturidade [Harter et al. 2000].

É importante notar que os autores referem as restrições deste estudo ligadas ao tipo de desenvolvimento, ou seja, software específico na área dos sistemas de informação de gestão, para *mainframes*.

O trabalho de King e Diaz [2002] foi realizado na GDDS<sup>13</sup> e analisa diversos aspectos do custo / benefício e do desempenho associado ao nível de maturidade. O estudo baseia-se nas métricas armazenadas pelas empresas durante os programas de melhoria para atingir o nível 2, 3, 4 e 5. A tabela 4.3 apresenta as tendências verificadas nos parâmetros: *rework*, detecção precoce de erros, qualidade prevista e produtividade.

Nível CMM	Percentagem de <i>Rework</i>	Detecção Precoce	Defeito / KSLOC <sup>14</sup>	Produtividade
2	23,2%	25,5%	3,20	1×
3	14,3%	41,5%	0,90	2×
4	9,5%	62,3%	0,22	1,9×
5	6,8%	87,3%	0,19	2,9×

Tabela 4.3 – Parâmetros dos projectos versus nível CMM [King e Diaz 2002].

A percentagem do *rework* é medição da percentagem do tempo de desenvolvimento do projecto dedicada a repetir o trabalho. A detecção precoce corresponde à medição da percentagem dos defeitos que são encontrados na fase onde foram inseridos. A qualidade prevista corresponde ao número de defeitos latentes ou o número de defeitos relatados pelo cliente por mil linhas de código fonte. Finalmente, a

<sup>12</sup> *Line Of Code*.

<sup>13</sup> General Dynamics Decision Systems com 360 profissionais envolvidos no desenvolvimento de software.

<sup>14</sup> *Kilo Source Line Of Code*.

qualidade é representada como um factor multiplicativo, calculado em relação à qualidade média do nível 2 CMM.

Em relação ao ROI (ver tabela 4.3) é calculado assumindo um investimento de 2,5% na melhoria do processo, para progredir um nível no modelo CMM.

Progressão CMM	Custo da MPS (horas)	Custo poupado em <i>rework</i> (horas)	ROI
Nível 4 para 5	884	1009	14%
Nível 3 para 4	1310	2744	109%
Nível 2 para 3	2544	6806	167%

Tabela 4.4 – ROI obtido com a progressão do nível CMM [King e Diaz 2002]

Refira-se que o ROI apresentado não toma em consideração o benefício adicional de realocação dos recursos disponibilizados.

Haley [1996] apresenta os dados da MPS da empresa Raytheon<sup>15</sup>, envolvendo 1200 profissionais de software em diversas áreas<sup>16</sup>. Este autor relata os resultados obtidos após um período de 7 anos, em que se verificou a progressão do nível de maturidade do CMM de 1 para 3. Os defeitos encontrados durante a fase de testes desceram 80%, o *rework* sofreu uma redução de mais de 50%, a produtividade aumentou 190%, a precisão das estimativas dos custos do software aumentou 93% e a qualidade do software aumentou 77%.

Nesta secção apresentaram-se apenas alguns casos demonstrativos dos retornos verificados como resultado das iniciativas de MPS. É importante referir que há um elevado número de estudos empíricos nesta área: Lipke e Butler [1992], Dion [1993], Curtis [1995], Diaz e Sligo [1997], Yamamura e Wigle [1997], Fowler [1997], Oldham et al. [1999], Cosgriff [1999] entre outros.

Na análise dos casos práticos foram ponderadas algumas questões importantes que envolvem as medições das iniciativas MPS: (1) Os casos apresentados são

<sup>15</sup> Raytheon Electronic Systems Equipment Division.

<sup>16</sup> controlo de tráfico, comunicações de satélite, radares, entre outras.



exemplos típicos ou apenas um grupo de histórias de sucesso? (2) Foi mesmo o programa de MPS que provocou os benefícios ou foram outros factores externos/internos? (3) As medições envolvem apenas alguns parâmetros, haverá alguns custos escondidos inerentes aos benefícios apresentados? (4) Será que não há casos de insucesso resultantes duma iniciativa de MPS?

Os estudos analisados confirmam a problemática estudada, ou seja, por um lado as iniciativas de MPS com sucesso trazem benefícios visíveis em termos do ROI, qualidade, tempo de ciclo, custo e produtividade, por outro lado, uma percentagem significativa dos projectos de MPS falha.

A secção seguinte procura determinar quais os factores críticos para o sucesso duma iniciativa de MPS.

## **4.2. Limitações, Problemas e Factores Críticos das Iniciativas de MPS**

Nesta secção procura-se identificar as limitações, problemas e factores críticos que permitem justificar o elevado número de iniciativas de MPS que são abandonadas. O trabalho de análise de casos práticos teve inicialmente um âmbito genérico, focando numa segunda fase no ambiente específico das pequenas empresas de software (PES).

Segundo Conradi e Fuggetta [2002], as abordagens à MPS são caracterizadas por duas dicotomias: disciplina versus trabalho criativo e riscos do cliente versus satisfação do utilizador. Com base nesta percepção, os autores apresentam seis teses para melhorar a aplicação dos referenciais de MPS:

1. As estratégias de melhoria devem estar orientadas aos objectivos do negócio e à inovação do produto
2. Iniciativas concretas e *bottom up*, devido a uma motivação maior dos profissionais de desenvolvimento
3. É dada demasiada ênfase à automatização dos processos de software
4. As análises custo - benefício necessitam de novos modelos de amortização
5. A MPS foca em mudanças culturais, assim é necessário *expertise* desta área

## 6. A MPS é sobre aprendizagem, não controlo como na garantia da qualidade

Dyba [2000] desenvolveu uma lista dos factores chave do sucesso da MPS, baseando-se inicialmente numa revisão da literatura e validando os resultados através de um estudo empírico realizado entre investigadores e profissionais de desenvolvimento. Os factores propostos são:

- Orientação ao negócio
- Envolvimento da liderança
- Participação dos funcionários
- Preocupação com medições
- Utilização do conhecimento existente (aprendizagem por experiência)
- Utilização de novo conhecimento (aprendizagem por experimentação)

No seguimento dos seus trabalhos Dyba [2000] mostrou empiricamente que “os aspectos organizacionais são tão importantes como a tecnologia para a MPS, ou talvez mais”.

Hall et al. [2002] realizaram um *survey*<sup>17</sup> para caracterizar a implementação da MPS em Inglaterra. A lista de factores considerada era composta por:

1. Factores Humanos
  - Líderes da MPS
  - Compromisso da gestão
  - Envolvimento do pessoal
2. Factores Organizacionais
  - Comunicação
  - Recursos
3. Factores de Implementação
  - Infra-estrutura
  - Definição de objectivos
  - “Colocação à medida”
  - Avaliação

---

<sup>17</sup> 85 organizações com menos de 100 funcionários.

Os inquiridos consideraram importante obter o compromisso da gestão sénior, “colocar à medida” a iniciativa de MPS para as necessidades específicas da organização e alinhar a iniciativa com os objectivos do negócio.

Fayad e Laitinen [1997] apresentam um conjunto de problemas ligados às avaliações do CMM, embora considerem que existem também nos outros modelos de melhoria:

- As avaliações são caras e para as organizações imaturas os resultados provavelmente têm pouco significado
- Os modelos de avaliação são artificialmente derivados (apresentam uma lista ideal de práticas)
- Os níveis de CMM são pesados de maneira muito desigual
- O tempo para progredir um nível é muito elevado considerando a turbulência e competitividade actuais
- Em organizações pequenas a sobrecarga para atingir e verificar os critérios pode ser proibitiva

Os autores consideram que as avaliações desviam a atenção do objectivo de melhorar o sistema e afirmam, “processos são ferramentas que ajudam a solucionar e não soluções eles mesmos” [Fayad e Laitinen 1998].

Segundo Evans [2002], as razões mais comuns para o insucesso das iniciativas de MPS são:

- Recursos inadequados
- Falha na criação de um clima de aceitação
- Falha na manutenção e acompanhamento das melhorias
- Falha nas ferramentas de apoio

Para Spence [2002] não é possível melhorar o processo de desenvolvimento de software sem tomar em consideração a cultura organizacional. Nesta perspectiva, o autor defende uma abordagem à melhoria do processo de software baseada na experiência e tomando em consideração os seguintes princípios:

- Executá-la como qualquer outro projecto do negócio
- Compreender o ambiente existente

- Construir sobre o melhor, não reinventar a roda
- Executar melhorias incrementais e mensuráveis
- Distribuir a responsabilidade do processo
- Manter as pessoas informadas e envolvidas

Wieggers [1999b] descreve a experiência de MPS numa pequena empresa a desenvolver para a Internet. Relata o sucesso obtido em cinco meses e aconselha que para o êxito de uma MPS numa empresa pequena em ambiente turbulento, é fundamental ser sensível à sua própria cultura e adaptar as iniciativas às suas necessidades essenciais bem como aos recursos disponíveis.

El Emam et al. [1999] realizaram um *survey* em organizações envolvidas na primeira fase dos *trials* SPICE e obtiveram os seguintes factores:

- Compromisso da gestão sénior
- Objectivos para a MPS claros e relevantes
- Atribuição de responsabilidades clara e compensatória na MPS
- Envolvimento do pessoal
- Pessoal respeitado na liderança da MPS
- Atribuição de recursos e tempo adequado
- Equipas de acção sobre o processo

Diversos autores referem que os modelos de melhoria do processo de desenvolvimento de software não tratam convenientemente os aspectos sociais. Johansen e Mathiassen [1998] argumentam que o CMM precisa de uma incidência maior na gestão. Aaen et al. [2001] referem que a complexidade da mudança organizacional sugerida pelo CMM necessita de uma abordagem mais centrada na gestão e não nos aspectos técnicos. No trabalho de Nielsen e Norbjerg [2001] é referido que o CMM necessita de ser complementado com teorias socialmente orientadas, para tratar de alterações e políticas organizacionais. Ngwenyama e Nielsen [2003] argumentam que a MPS é uma intervenção na cultura organizacional com o objectivo de alterá-la e como tal a teoria e prática da MPS não pode ignorar o corpo de conhecimento sobre cultura organizacional. Paulk et al. [1997], na sua antevisão sobre a

versão 2<sup>18</sup> do CMM referem a necessidade dos modelos de melhoria do processo de desenvolvimento de software tratarem explicitamente a gestão das mudanças organizacionais. O estudo de Siakas e Georgiadou [2002] avalia os efeitos dos factores culturais<sup>19</sup> nas iniciativas de MPS e conclui que os sistemas de gestão de qualidade devem depender da cultura onde a organização está situada.

Na literatura mais recente sobre a problemática da melhoria do processo de desenvolvimento de software sobressai o estudo do impacto da gestão do conhecimento. Ravichandran e Rai [2003] estudam o impacto da criação e utilização<sup>20</sup> do conhecimento na capacidade do processo de software. Os resultados mostram que a extensão com que o conhecimento é efectivamente utilizado no processo tem um efeito positivo significativo na sua capacidade. Por outro lado, o nível de utilização do conhecimento no processo é influenciado pela extensão com que o conhecimento é criado no contexto do processo. Mathiassen e Pourkomeylian [2003] referem que a “MPS é uma forma particular de criação, partilha e gestão do conhecimento sendo aconselhável tratar explicitamente a estratégia de gestão de conhecimento escondida”. Apostolou e Mentzas [2003] estudaram as diversas implementações de gestão do conhecimento em organizações de desenvolvimento de software. As recomendações foram: (1) focar na exploração das fontes do conhecimento; (2) facilitar a troca de conhecimento entre comunidades; (3) dar ênfase às necessidades e requisitos do cliente e (4) integrar a gestão do conhecimento dentro dos processos de negócio normais. Gasston e Halloran [1999] basearam-se num caso de estudo para sugerir que só é possível obter os maiores benefícios de uma iniciativa de MPS quando as organizações se tornam organizações “baseadas na aprendizagem”<sup>21</sup>. Schneider et al. [2002] descrevem a implementação da MPS baseada na gestão do conhecimento<sup>22</sup> e referem as dificuldades apresentadas.

A revisão da literatura efectuada mostra a existência de três grupos principais de problemas, relacionados com os modelos de melhoria do processo de desenvolvimento de software. Em primeiro lugar factores relacionados com os recursos humanos, ou seja,

---

<sup>18</sup> mais tarde abandonada em favor do CMMI.

<sup>19</sup> cultura organizacional e cultura nacional.

<sup>20</sup> *embedding*.

<sup>21</sup> *learning organizations*.

<sup>22</sup> especificamente a reutilização de experiências.

o compromisso da gestão e o envolvimento de todos os profissionais. Em segundo lugar, o problema da cultura organizacional. Reconhecendo que um esforço de MPS tem a ver directamente com mudança organizacional, parece óbvio que os modelos de MPS tratem explicitamente destes aspectos. Em terceiro lugar, a área de gestão do conhecimento. A necessidade de fazer face à crescente competitividade, turbulência e dinamismo dos mercados<sup>23</sup>, obriga a uma preocupação efectiva com a aprendizagem organizacional.

#### **4.2.1. Aplicação dos Modelos de MPS nas Pequenas Empresas de Software**

Na secção anterior apresentaram-se problemas das iniciativas de MPS, verificados independentemente do tamanho da organização de software. Nesta secção procura-se identificar problemas específicos verificados no contexto das pequenas empresas de software.

Segundo Orci e Laryd [2000], “apesar das dificuldades e problemas na implementação<sup>24</sup> é normalmente aceite que uma organização que desenvolve software, com sintomas de caos causado pela imaturidade, deve iniciar um programa de melhoria do processo de software”. Os autores referem que os modelos de MPS não podem ser aplicados directamente a pequenas empresas de software por diversas razões: (1) devido ao número de recursos, não é possível cumprir todos os papéis<sup>25</sup>, responsabilidades e tarefas; (2) os modelos têm que ser simplificados e reduzidos para não ultrapassar as possibilidades / necessidades das Pequenas Empresas de Software (PES); (3) devem incluir orientações para a avaliação e aplicação do modelo, com recursos internos<sup>26</sup> e (4) é fundamental que o modelo possua uma estratégia contínua para a MPS, caso contrário, as prioridades do dia-a-dia irão ultrapassar e abortar a iniciativa.

---

<sup>23</sup> particularmente na área de desenvolvimento de software.

<sup>24</sup> dos modelos de MPS.

<sup>25</sup> o CMM propõe mais de 25 papéis diferentes.

<sup>26</sup> devido à escassez de fundos as PES normalmente não possuem meios para suportar a avaliação e acompanhamento externo.

Rout et al. [2000] referem a necessidade de um método menos dispendioso e com uma aplicação mais rápida para executar a MPS nas pequenas empresas de software. Argumentam também a vantagem da conformidade com a ISO/IEC 15504, para que os resultados sejam vistos como fiáveis e objectivos. Nesta linha de pensamento, o SPIRE [1998] identifica a necessidade de uma abordagem específica<sup>27</sup> para a avaliação e melhoria do processo de software nas pequenas empresas de software. Segundo o SPIRE [1998], as PES enfrentam algumas barreiras especiais: estão menos conscientes do que as grandes empresas sobre os benefícios da MPS; falta-lhes experiência interna sobre a área de MPS; as crises do dia-a-dia e mudanças de prioridade provocam muitas vezes o abandono do plano de melhoria. Wiegers e Sturzenberger [2000] confirmam a mesma ideia, afirmando que “o aspecto mais comum para a falha da MPS é a falta de acompanhamento do plano de acção e da sua implementação”. Os custos<sup>28</sup> e a duração do ciclo de melhoria<sup>29</sup> são também elevados e muitas vezes proibitivos para as pequenas empresas de software [Villalon et al. 2002].

Johnson e Brodman [1997] realizaram um estudo em aproximadamente 200 pequenas organizações de software e concluíram que o CMM precisa de ser restringido<sup>30</sup> para se adaptar a pequenas organizações de software. Na mesma linha de pensamento, Kuvaja et al. [1999] e Horvat et al. [2000] referem a necessidade de adaptar os modelos de melhoria mais comuns às necessidades das PES.

Analisando a aplicação do CMM e da norma ISO/IEC 15504 às pequenas empresas de software, parece clara a necessidade de adaptar, simplificar e flexibilizar estes modelos.

---

<sup>27</sup> segundo o SPIRE, o CMM, ISO/IEC 15504 e ISO 9001 não são aplicáveis directamente às PES.

<sup>28</sup> segundo Herbsleb et al. [1994] cerca de \$245000 por ano.

<sup>29</sup> entre 18 e 24 meses [Zahran 1998].

<sup>30</sup> *tailored*.

### 4.3. Aplicação Prática em Pequenas Empresas de Software

Nesta secção apresenta-se uma breve descrição sobre três casos práticos de aplicação do CMM e da norma ISO/IEC 15504<sup>31</sup>, no contexto das PES portuguesas. Estes casos de estudo foram realizados entre 1999 e 2002 e serviram para experimentar, na prática, os modelos de melhoria referidos e também iniciar os testes de algumas componentes do 2MPspe. Os nomes das empresas participantes não são apresentados por questões relacionadas com a utilização / divulgação<sup>32</sup> de todos os dados recolhidos. As empresas serão denominadas<sup>33</sup> Alfa, Beta e Gama.

Os três projectos – descritos nesta secção – foram realizados seguindo um método de implementação<sup>34</sup> baseado no ciclo PDCA<sup>35</sup>. Para a execução da etapa de Planeamento<sup>36</sup> foram utilizadas quatro fases.

A primeira fase, denominada “Levantamento do Processo”, baseou-se na norma 12207 e teve como objectivos: (1) conhecimento inicial dos processos da empresa e recolha de documentação de “alto nível”; (2) levantamento informal da matriz recursos/processos e (3) selecção dos colaboradores a entrevistar na fase seguinte.

A fase 2, denominada “Diagnóstico dos Processos”, teve como base a norma ISO/IEC 15504 modificada e reduzida para as necessidades específicas da empresa. Esta fase foi executada utilizando um questionário<sup>37</sup>, desenvolvido através das práticas genéricas<sup>38</sup>, mas simplificado através dos *outputs* da fase de levantamento. O resultado da fase de diagnóstico foi um perfil individual da capacidade dos processos.

A terceira fase, denominada “Validação do Diagnóstico” utilizou o CMM e teve como objectivos principais: (1) confirmação dos resultados da fase de diagnóstico; (2) definição do nível de maturidade da organização e (3) selecção das áreas de melhoria

---

<sup>31</sup> na versão de 1998.

<sup>32</sup> duas empresas colocaram restrições à divulgação de dados sobre determinados processos.

<sup>33</sup> nomes fictícios.

<sup>34</sup> deu origem ao método de implementação do 2MPspe (M<sub>i</sub>2MP) – capítulo 8.

<sup>35</sup> descrito na secção 2.3.2.1.

<sup>36</sup> primeira etapa do método de implementação.

<sup>37</sup> ver anexo A

<sup>38</sup> propostas no ISO/IEC 15504-5.



possíveis. Entendeu-se que o CMM fornece uma definição mais clara das prioridades a ter em conta para progredir entre os níveis.

Finalmente, a fase 4, denominada “Definição das Áreas de Melhoria”, pretendeu analisar os resultados obtidos e desenvolver um plano de acção, apresentando as áreas-chaves a tratar e as estratégias adoptadas para atingir os objectivos. Inicialmente realizou-se a caracterização dos impactos previstos na organização, devido a melhorias independentes na capacidade dos processos. Esta avaliação de impactos possibilitou a definição das prioridades de melhoria e foi o ponto de partida para a definição das estratégias a utilizar em cada processo individualmente. Esta fase terminou com a definição do cronograma de acção e a caracterização quantitativa<sup>39</sup> dos objectivos do plano de melhoria.

#### **4.3.1. Descrição dos Casos Práticos**

Apresenta-se em seguida uma descrição de cada caso prático englobando a caracterização do contexto, o desenvolvimento do projecto e os resultados obtidos. A secção termina com a apresentação de algumas conclusões globais sobre as experiências práticas.

##### **Caso prático 1 – Alfa**

A empresa Alfa é uma pequena empresa de desenvolvimento de software. Possuía um produto na área de gestão comercial para PME's que representava 80% da sua facturação. Empregava 16 colaboradores<sup>40</sup>, dos quais 12 estavam dedicados ao desenvolvimento. A empresa tinha sofrido um crescimento rápido no último ano, tendo admitido recentemente 4 novos profissionais de desenvolvimento. A empresa encontrava-se num processo de mudança da plataforma de desenvolvimento e estava preocupada com os atrasos sucessivos do seu novo produto base.

---

<sup>39</sup> nível de capacidade a atingir para cada processo.

<sup>40</sup> 1º semestre de 1999.

A empresa apresentava claramente as características do nível 0 da norma ISO/IEC 15504, ou seja, poucos processos definidos e o sucesso muito dependente dos esforços individuais. O perfil de maturidade apresentava todos os processos avaliados no nível 0 (ver anexo A1). A validação do diagnóstico confirmou esses resultados.

As conclusões aconselharam a empresa a formalizar e difundir as técnicas de gestão de projectos. Verificou-se que a empresa abandonou o plano de melhoria devido à urgência de terminar o novo produto base. Os responsáveis da empresa esperavam uma intervenção nos processos técnicos, dando pouca importância aos processos de suporte, nomeadamente à gestão de projectos.

### **Caso prático 2 – Beta**

A empresa Beta tem oito colaboradores, dos quais seis estão dedicados a tarefas de desenvolvimento. A empresa desenvolve software específico, principalmente na área de gestão da produção. Possui uma carteira de cerca de trinta clientes da região e tem grandes preocupações no domínio da reutilização. É uma empresa recente e a iniciativa de melhoria surgiu da necessidade de melhorar a qualidade final dos produtos.

O projecto de melhoria teve início no segundo semestre do ano 2001. O resultado do diagnóstico dos processos (ver anexo A2) mostrou que a empresa se encontrava também no nível 0 da norma ISO/IEC 15504. Apenas os processos “Desenvolvimento dos requisitos”, “Implementação do software” e “Gestão de configurações” estavam no nível de capacidade 1, ou seja, eram executados. Em relação à validação do diagnóstico, verificou-se que as áreas chave do processo, gestão de requisitos e garantia da qualidade do software eram cumpridas em 50%. A gestão de projectos apresentava um nível de 75% e a gestão de configurações era totalmente executada<sup>41</sup>. O Plano de acção teve como base o diagnóstico e a avaliação do diagnóstico e teve como foco primário a definição de estratégias de melhoria para os processos de gestão de requisitos e garantia da qualidade. O foco secundário teve como objectivo as actividades de planeamento e acompanhamento do projecto de software. A implementação das melhorias nas áreas de requisitos e gestão de projectos pode

---

<sup>41</sup> refira-se que as áreas chave do processo eram avaliadas apenas pelas actividades.

considerar-se como com sucesso, trazendo resultados visíveis num curto prazo. A área de garantia da qualidade era apenas executada parcialmente e sem o nível de formalismo adequado, não se verificando melhorias significativas.

### **Caso prático 3 – Gama**

A empresa Gama possuía 20 funcionários e baseava o seu negócio num conjunto de aplicações para PMEs que incluía a Gestão Comercial, Gestão Pessoal, Contabilidade e Gestão de Imobilizado. Esta empresa não trabalha directamente com os clientes finais, mas possui uma rede de revendedores espalhados por todo o país. Além das aplicações referidas para PMEs, possui outras aplicações para áreas específicas<sup>42</sup>, mas com um peso pequeno no volume de facturação. A empresa tinha três equipas na área de desenvolvimento dedicadas à manutenção, novos produtos e Internet. A empresa Gama tem a percepção que dedica demasiado esforço à manutenção dos produtos e esta preocupação conduziu à iniciativa de MPS. O objectivo era então diagnosticar a situação e definir estratégias de melhoria.

Os resultados do diagnóstico dos processos e da validação encontram-se no anexo A3. Resumidamente, a empresa possui apenas um processo no nível 1, ou seja, o processo de manutenção. Em relação ao CMM, a empresa está no nível 1 e as áreas chave do processo do nível 2, encontram-se muito incompletas. Em relação às estratégias de melhoria, foram definidas três Áreas de Intervenção (AIs):

1. Infra-estrutura do processo, constituída pelos processos: definição do processo, recursos humanos e documentação.
2. Engenharia do produto, constituída pelos processos: desenvolvimento dos requisitos e implementação do software.
3. Sistema de qualidade, composta pelos processos: gestão da qualidade, garantia da qualidade, verificação e validação.

Embora existisse o apoio da gestão, verificou-se que apenas a AI da infra-estrutura do processo teve continuidade.

---

<sup>42</sup> por exemplo, gestão de seguros, gestão de clínicas, gestão de stands.

### 4.3.2. Conclusões Globais sobre os Projectos

Os projectos descritos na secção anterior não podem ser considerados casos de sucesso. Na realidade apenas na iniciativa da empresa Beta se verificaram melhorias significativas nas áreas apontadas. Na empresa Gama apenas a área da infra-estrutura do processo sofreu melhorias e na empresa Alfa nada mudou. Globalmente, podem ser retiradas algumas conclusões do trabalho de campo:

- É necessário gerir as expectativas. Verifica-se que alguns profissionais de software olham para a MPS como a “solução miraculosa” para todos os problemas. Quando as expectativas são demasiado elevadas, o resultado final é quase sempre o insucesso.
- Nas pequenas empresas de software, as crises do dia-a-dia definem as prioridades.
- É necessário um apoio visível por parte da gestão.
- A realização da avaliação é apenas o primeiro passo. O acompanhamento e estabilização das acções de melhoria é muito importante.
- É necessário perceber a cultura organizacional e ganhar o apoio dos executores dos processos. Não é possível melhorar um processo sem a colaboração dos intervenientes.
- A avaliação dos processos deve ser simples e pouco morosa.
- É importante que as primeiras acções de melhoria tenham resultados visíveis num curto prazo.
- É necessário tratar a abordagem à gestão do conhecimento predominante, definindo e gerindo uma infra-estrutura adequada.

Este capítulo apresentou a revisão da literatura realizada sobre os modelos de melhoria do processo de desenvolvimento de software bem como os casos práticos desenvolvidos para experimentar o CMM e a ISO/IEC 15504. O próximo capítulo faz a síntese dos aspectos principais sobre a aplicação prática dos modelos de MPS e propõe um novo modelo, denominado 2MPspe (Modelo de Melhoria do Processo de Software para Pequenas Empresas).

## Capítulo 5

---

### **5. Modelo de Melhoria do Processo de Software para Pequenas Empresas (2MPspe)**

A indústria do software está no centro de qualquer produto ou serviço moderno. Assim, melhorar a forma como desenvolvemos software ou os processos do software é um dos desafios chave das empresas, universidades e dos próprios governos. Neste contexto, as estratégias de Melhoria do Processo do Software são vistas como um dos esforços mais promissores para assegurar a qualidade do software, tratando de uma forma efectiva a problemática da “doença crónica” do software. Contudo, há ainda muitas questões em aberto.

#### **5.1. O Problema**

Como foi referido, a problemática inicial da investigação centrou-se nas questões sobre:

- (1) Os esforços de MPS trazem benefícios reais para as organizações?
- (2) Muitas iniciativas de MPS falham? Quais as razões do insucesso?

O capítulo anterior apresentou o estudo realizado sobre estas questões. Este estudo incluiu uma revisão exaustiva da literatura e o desenvolvimento de três casos práticos em PES da região norte de Portugal. As conclusões são evidentes:

1. Há uma grande taxa de insucesso na aplicação dos modelos de melhoria do processo de software
2. As organizações que implementam e institucionalizam uma MPS com sucesso obtêm ganhos significativos na produtividade, *rework* e precisão nas estimativas da duração e do custo e ROI

Das conclusões anteriores definiu-se o pressuposto da investigação:

### **Pressuposto**

Um projecto de melhoria do processo de desenvolvimento de software com sucesso traz benefícios substanciais para a qualidade do produto e para o valor do negócio, contudo há uma grande parte das iniciativas que falha.

A revisão da literatura e o trabalho de campo permitiram validar o pressuposto enunciado. A aplicação destas duas estratégias salientou dois factores importantes para o sucesso das MPS nas pequenas empresas. Estes factores são: (1) a abordagem à gestão do conhecimento [Gasston e Halloran 1999], [Mathiassen e Pourkomeylian 2003] e [Ravichandran e Rai 2003] e (2) a gestão da cultura organizacional [Paulk et al. 1997], [Johansen e Mathiassen 1998] e [Nielsen e Norbjerg 2001]. Tomando em consideração estes factores e a constatação enunciada definiu-se o propósito da investigação:

### **Propósito da investigação**

Pretende-se desenvolver um modelo de melhoria do processo de software adaptado às pequenas empresas de software na área de sistemas de informação de gestão e que trate explicitamente os factores contextuais da cultura organizacional da gestão do conhecimento.

Partindo do propósito definido e tendo como base a análise dos factores críticos de sucesso na MPS e o trabalho prático, definiram-se as seguintes características desejáveis para o novo modelo de MPS:

**Característica do novo modelo de MPS:**

- a) Adaptado às pequenas empresas de software a trabalhar na área dos sistemas de informação de gestão
- b) Tratar explicitamente os factores contextuais, cultura organizacional e gestão do conhecimento
- c) Conforme com a norma ISO/IEC 15504
- d) Ter correspondência definida com o CMM
- e) Cumprir os requisitos da norma ISO 9001:2000
- f) Tratar os problemas do envolvimento dos intervenientes nos processos
- g) Permitir ciclos<sup>1</sup> de melhoria rápidos
- h) Simples e robusto

O modelo proposto denomina-se 2MPspe e é apresentado globalmente na próxima secção.

**5.2. Descrição Geral do 2MPspe**

Nesta secção apresenta-se uma descrição de “alto nível” do modelo de melhoria do processo de software desenvolvido no âmbito deste trabalho. O modelo é designado 2MPspe (Modelo de Melhoria do Processo de Software para Pequenas Empresas) e pretende cumprir o propósito da investigação e as características enunciadas na secção anterior.

Podemos analisar a MPS na perspectiva técnica ou de gestão. Na perspectiva de gestão, temos que considerar a cultura organizacional como factor condicionante principal para implementar com sucesso as mudanças organizacionais necessárias à melhoria efectiva dos processos. Por outro lado, temos que considerar a Gestão do Conhecimento como abordagem epistemológica fundamental para analisar a captura, transferência e institucionalização do conhecimento sobre os processos.

---

<sup>1</sup> espaço de tempo entre o início do projecto e a apresentação do plano de acção

São diversos os autores que apontam a necessidade de tratar explicitamente a estratégia de gestão do conhecimento num modelo de melhoria do processo de software [Aaen 2003], [Mathiassen e Pourkomeylian 2003] e [Apostolou e Mentzas 2003]. Em relação à influência da cultura organizacional é o próprio Mark Paulk [Paulk et al. 1997] e outros especialistas [Zahran 1998], [Conradi e Fuggetta 2002] e [Ngwenyama e Nielsen 2003] que referem a sua importância num projecto de MPS.

Da secção anterior ressalta também a necessidade de balancear disciplina e criatividade [Boehm e Port 2001] e [Conradi e Fuggetta 2002]. Nesta área, as ideias de Ken Orr [2002] sobre “próximas práticas”, são um ponto de partida a considerar. Este autor classifica os esforços MPS num referencial conhecimento versus esforço (ver figura 5.1) e localiza os níveis do CMM e os modelos ágeis neste referencial. Na figura 5.1, no eixo vertical apresenta-se o crescimento do valor das práticas em termos do conhecimento. No eixo horizontal representa-se o crescimento do esforço em termos da documentação.

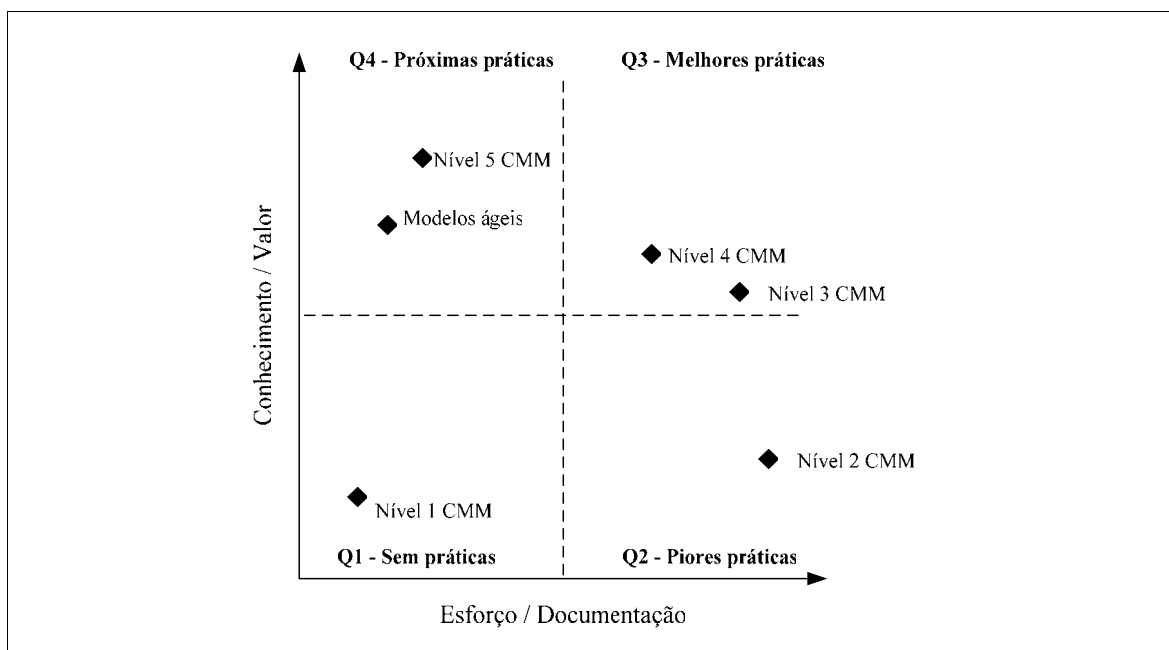


Figura 5.1 – Conhecimento versus esforço (adaptado de Orr [2002]).

São definidos quatro quadrantes, Q1 – sem práticas, Q2 – piores práticas, Q3 – melhores práticas e Q4 – próximas práticas, considerando o autor como as “próximas práticas” aquelas que se baseiam em tecnologias de ruptura. Da figura 5.1 torna-se claro uma excessiva documentação inerente principalmente ao nível 2, sem os ganhos esperados relativos ao conhecimento. A convergência de “ideias” em relação ao nível 5



do CMM e aos modelos ágeis é também apresentada, em concordância com Paulk [2002].

A figura 5.1 permite explicar um dos objectivos do 2MPspe, o tratamento explícito da abordagem à gestão do conhecimento dos processos, das equipas e dos indivíduos. Com a contextualização da gestão do conhecimento pretende-se atingir o quadrante das “próximas práticas”. Não dizendo quais são (como nos modelos ágeis), mas antes criando um ambiente propício a atingi-las (como no CMM, nível 5). Pensa-se então que as PES, seguindo o 2MPspe atingirão regiões cada vez mais valiosas atingindo no nível 3 as “próximas práticas” (ver figura 5.2).

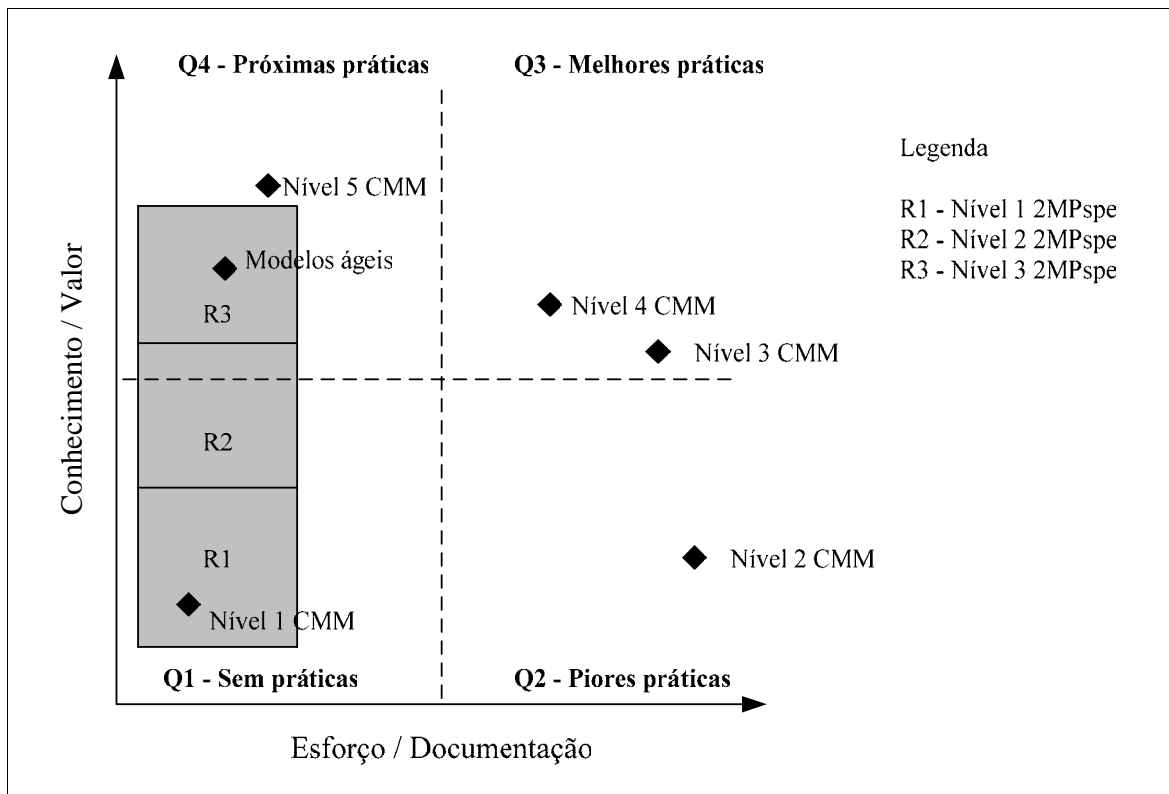


Figura 5.2 – Contexto do 2MPspe.

Outro objectivo do 2MPspe está ligado ao tratamento explícito da cultura organizacional predominante nas equipas e na organização. Este aspecto relaciona-se, como vimos, com o elevado número de insucessos nas iniciativas de MPS.

Considera-se então que a característica diferenciadora do 2MPspe em relação aos modelos de MPS analisados<sup>2</sup> na revisão da literatura é o tratamento explícito da gestão do conhecimento e da cultura organizacional.

Pretende-se também que o 2MPspe seja adaptado à realidade das PES, ou seja, apresente ciclos de melhoria rápidos e seja simples e robusto. Em relação à correspondência com as normas na área de MPS, pretende-se que seja conforme a ISO/IEC 15504, tenha uma correspondência definida com o CMM e cumpra os requisitos da norma ISO 9001:2000.

Como restrição principal do 2MPspe, refira-se a sua aplicação exclusiva ao desenvolvimento de software na área de sistemas de informação de gestão, também denominado software de negócio [Pressman 1997]. Esta área engloba os sistemas de processamento de dados tradicionais, COTS<sup>3</sup>, desenvolvimento para a Internet e sistemas Cliente/Servidor.

Os modelos de melhoria do processo baseiam-se no pressuposto<sup>4</sup> que o processo tem um impacto no produto. Verifica-se então que os modelos de MPS analisados<sup>5</sup> tratam a execução do processo e a sua capacidade para melhorar a qualidade dos produtos (ver figura 5.3 a.). Os factores do contexto são apenas tratados indirectamente através de orientações escritas<sup>6</sup>. Pretende-se que o 2MPspe trate explicitamente alguns dos factores do contexto do processo de desenvolvimento de software, nomeadamente: gestão do conhecimento e cultura organizacional (ver figura 5.3 b.).

---

<sup>2</sup> ver secção 3.3.

<sup>3</sup> *Component Off-The-Shelf*.

<sup>4</sup> ver secção 3.2.

<sup>5</sup> ver secção 3.3.1.

<sup>6</sup> ver, por exemplo, anexo C da parte 4 da norma ISO/IEC 15504.

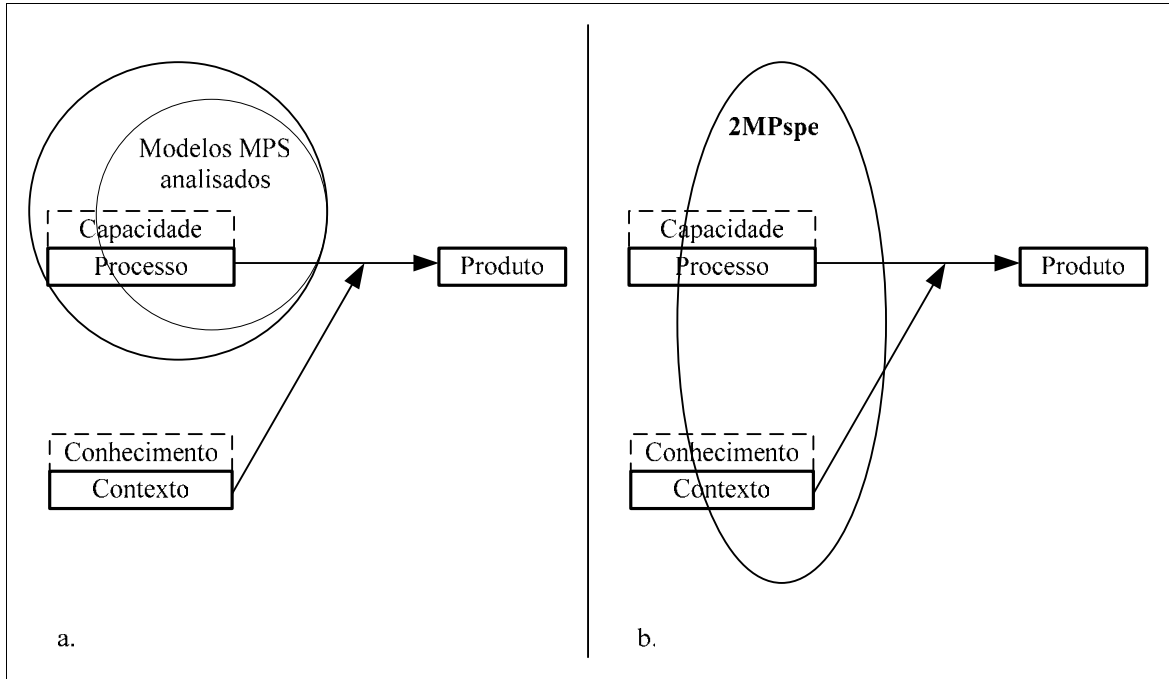


Figura 5.3 – 2MPspe versus modelos MPS analisados.

Em termos globais, o 2MPspe (ver figura 5.4) cobre três perspectivas da melhoria do processo de software:

- Conceitos, caracterização conceptual do referencial
- Ferramentas, instanciação dos conceitos em características mensuráveis
- Actividades, definição e organização das tarefas para executar a melhoria utilizando o modelo de avaliação

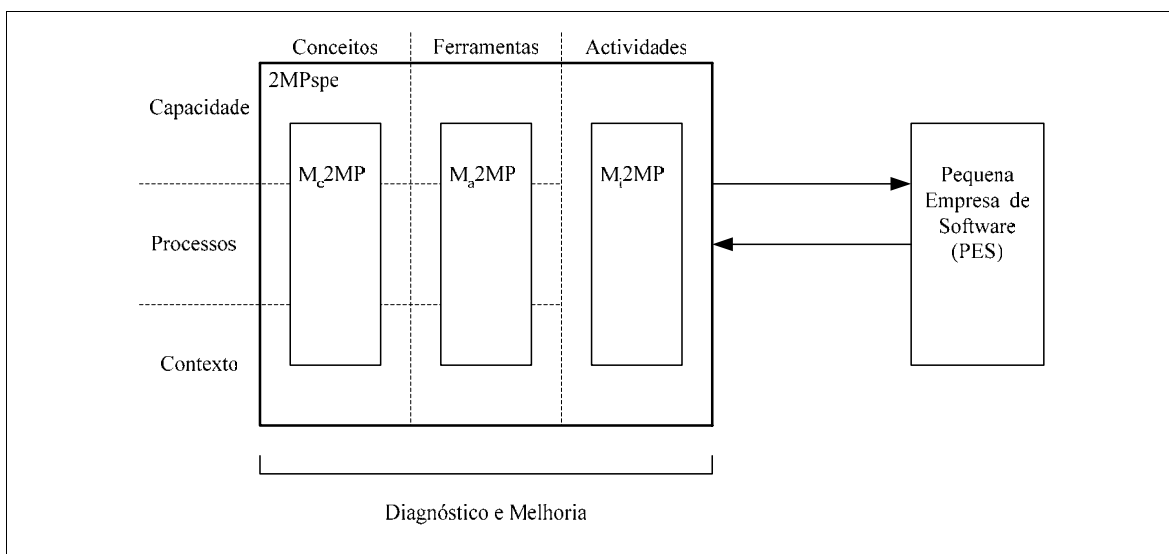


Figura 5.4 – Visão global do 2MPspe.

Esta arquitectura proposta no 2MPspe está alinhada com a norma ISO/IEC 15504, mas o 2MPspe pretende cobrir também a dimensão do contexto do processo.

O 2MPspe divide-se em três componentes: a componente  $M_c2MP$  (modelo conceptual) caracteriza a parte dos conceitos e define a abrangência do 2MPspe em termos dos processos considerados; o  $M_a2MP$  (modelo de avaliação) apresenta os indicadores característicos de cada atributo; a componente  $M_i2MP$  (método de implementação) abrange os aspectos relacionados com a execução da avaliação e o desenvolvimento da melhoria propriamente dita.

O modelo conceptual caracteriza o referencial utilizado pelo 2MPspe. O referencial possui três dimensões: a dimensão dos processos, a dimensão da capacidade e a dimensão do contexto. O  $M_c2MP$  define quais os processos incluídos no referencial e caracteriza o eixo da capacidade dos processos e do contexto dos processos. O modelo de avaliação do 2MPspe utiliza o referencial tridimensional para definir os indicadores que irão permitir caracterizar quantitativamente cada uma das dimensões. Estes indicadores do  $M_a2MP$  são utilizados na construção do questionário 2MPspe. Finalmente, o  $M_i2MP$  propõe a sequência de tarefas a cumprir para desenvolver o programa de melhoria do processo, com sucesso.

Em termos detalhados, o 2MPspe é apresentado na figura 5.5. Esta figura mostra os elementos de cada componente e os relacionamentos entre os elementos.

O  $M_c2MP$  define um referencial tridimensional. A dimensão dos processos caracteriza o modelo de referência dos processos (MRP) que descreve os processos do ciclo de vida do software incluídos no 2MPspe e detalha os relacionamentos entre os processos. A dimensão dos processos baseia-se na norma ISO/IEC 12207. A dimensão da capacidade dos processos tem três elementos: níveis de conhecimento, atributos de capacidade (ACs) e escala de classificação dos ACs. O eixo da capacidade tem como base a parte 2 da norma ISO/IEC 15504. A dimensão do conhecimento dos factores de contexto engloba também três elementos: níveis de conhecimento dos factores de contexto, atributos de conhecimento (AKs) e escala da classificação dos AKs. O  $M_c2MP$  cumpre os requisitos de conformidade apresentados na norma ISO/IEC 15504. Esta componente encontra-se descrita no capítulo 6.

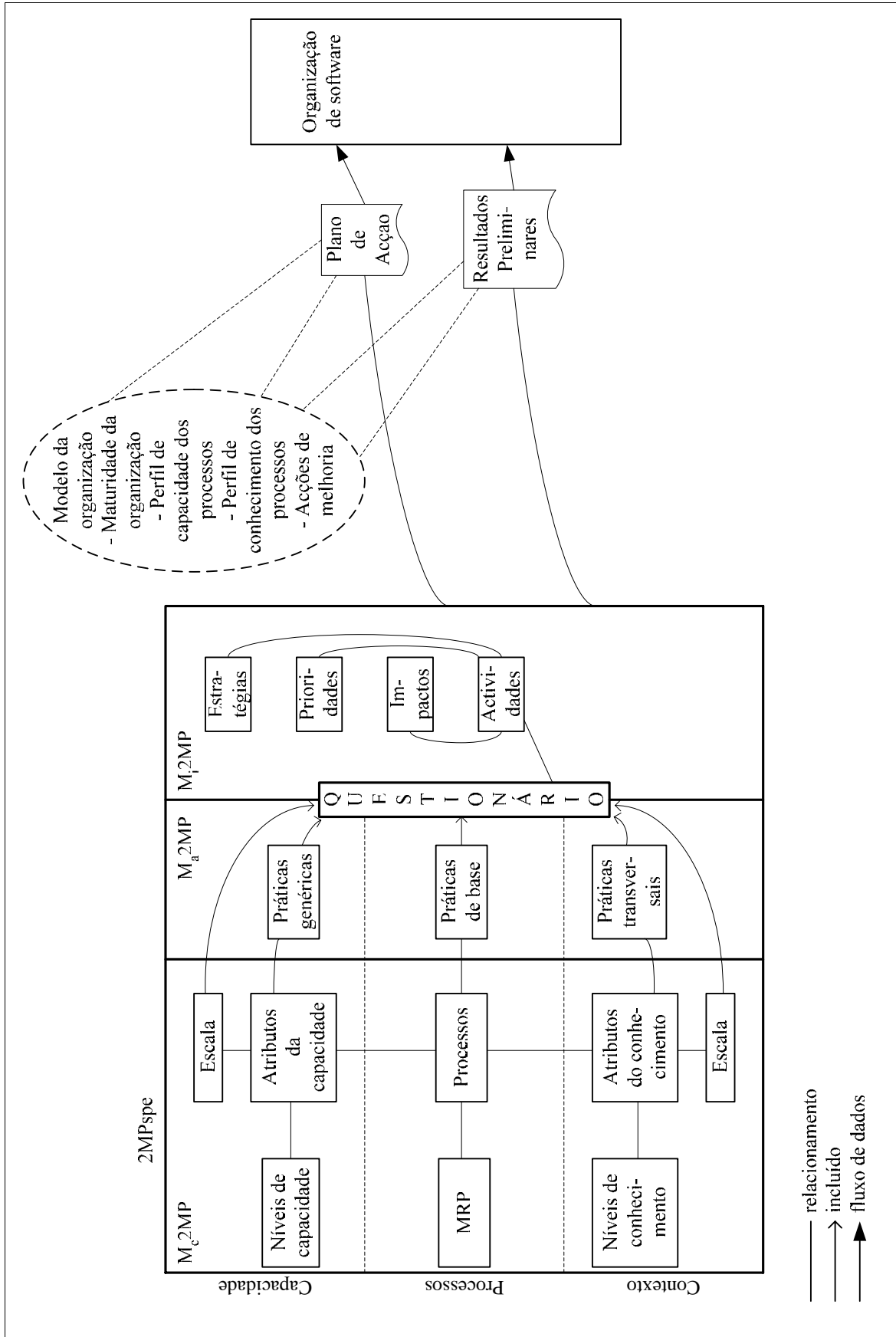


Figura 5.5 – Visão detalhada do 2MPspe.

O  $M_a2MP$  define o modelo de avaliação do processo, proposto no contexto do 2MPspe. Segundo a parte 2 da norma ISO/IEC 15504, um modelo de avaliação “forma a base para recolher evidência e classificar a capacidade do processo”. Nesta perspectiva, o  $M_a2MP$  define três tipos de indicadores: práticas de base, práticas genéricas e práticas transversais. As práticas de base permitem avaliar a execução do processo, as práticas genéricas permitem medir a capacidade do processo e as práticas transversais permitem avaliar o conhecimento dos factores contextuais utilizados no 2MPspe. Um indicador de uma prática suporta o julgamento da extensão com que um determinado atributo é atingido (ISO/IEC 15504-1). O  $M_a2MP$  define o questionário 2MPspe, onde são utilizadas as práticas e as escalas de classificação como base para as questões que permitirão realizar o diagnóstico do processo de desenvolvimento de software. O  $M_a2MP$  propõe também uma representação para os resultados obtidos com os dados recolhidos nos questionários. O  $M_a2MP$  tem como base a parte 5 da norma ISO/IEC 15504 e cumpre os requisitos de conformidades definidos na parte 2 da mesma norma. A componente  $M_a2MP$  é apresentada no capítulo 7 e o questionário encontra-se no anexo C.

Finalmente, a componente  $M_i2MP$  define o método de implementação que serve de base a uma iniciativa de melhoria do processo de desenvolvimento de software. O  $M_i2MP$  inclui a sequência de fases que um projecto de melhoria deve seguir, descrevendo as tarefas, as ferramentas e técnicas a utilizar e os papéis e responsabilidades. No  $M_i2MP$  há duas fases principais, isto é, o diagnóstico e a melhoria propriamente dita. Para a execução do diagnóstico são utilizados os questionários, entrevistas e a observação directa no local; os *output* desta fase são os resultados preliminares. A fase de melhoria utiliza as descobertas do diagnóstico e aplicando as estratégias, prioridades e impactos do 2MPspe deriva o plano de acção da melhoria do processo. O plano de acção servirá de base à implementação, verificação e institucionalização da melhoria. O  $M_i2MP$  define a duração e sequência das actividades estimando-se que será necessário apenas dois dias nas instalações da organização.

Nesta secção definiu-se o 2MPspe. O 2MPspe é uma proposta de método para aplicar em PES, com o objectivo de melhorar o seu processo de desenvolvimento de software. O 2MPspe não é uma teoria, é um método desenvolvido com uma abordagem de engenharia para resolver os problemas identificados nos modelos de MPS mais

divulgados. A descrição realizada neste capítulo e nos seguintes refere-se à primeira versão deste método.

Nos próximos capítulos é descrita em pormenor cada uma das componentes. No capítulo seis será detalhado o modelo conceptual ( $M_c2MP$ ) descrevendo-se cada um dos elementos. No capítulo 7 será apresentado o modelo de avaliação ( $M_a2MP$ ) e finalmente, no capítulo 8, descreve-se o método de implementação do 2MPspe, designado por  $M_i2MP$ .

## Capítulo 6

---

### 6. Modelo Conceptual do 2MPspe (M<sub>c</sub>2MP)

Neste capítulo apresenta-se o modelo conceptual que suporta o 2MPspe. O M<sub>c</sub>2MP é depois utilizado pelo modelo de avaliação para definir, em detalhe, as práticas a avaliar nas organizações de software. Relembrando os objectivos do 2MPspe, relacionados com: (1) a sua adaptação a pequenas empresas de desenvolvimento de software, a trabalhar na área dos sistemas de informação de gestão; (2) conformidade com a norma emergente ISO/IEC 15504; (3) correspondência definida para o CMM e (4) alinhada com a norma ISO 9001: 2000. Pretende-se também resolver explicitamente alguns problemas contextuais<sup>1</sup> associados a um esforço MPS, nomeadamente, a cultura organizacional, a gestão do conhecimento e o envolvimento dos recursos humanos, particularmente a gestão.

#### 6.1. Introdução

Na definição do M<sub>c</sub>2MP foram consideradas as propostas do CMM e da norma ISO/IEC 15504, enquadradas no contexto das pequenas empresas de software. O CMM apresenta um modelo com apenas uma dimensão [Wang 2001], ou seja, não há distinção entre a dimensão do processo e da capacidade (ver figura 6.1).

---

<sup>1</sup> ver capítulo 5.1.



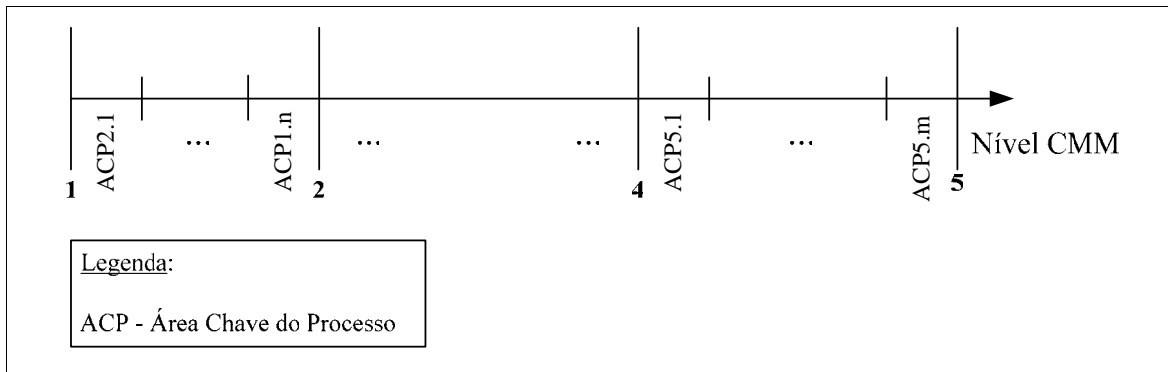


Figura 6.1 – Modelo unidimensional proposto pelo CMM.

Este facto provoca falta de visibilidade sobre a globalidade dos processos da organização, e ainda, dificuldade na realização de pequenas melhorias, centradas em determinados processos. Para subir um nível na escala de maturidade é necessário cumprir todas as áreas chave do processo para esse nível<sup>2</sup>.

A norma ISO/IEC 15504 apresenta um modelo de avaliação baseado num referencial com duas dimensões (ver figura 6.2): a dimensão dos processos e a dimensão da capacidade.

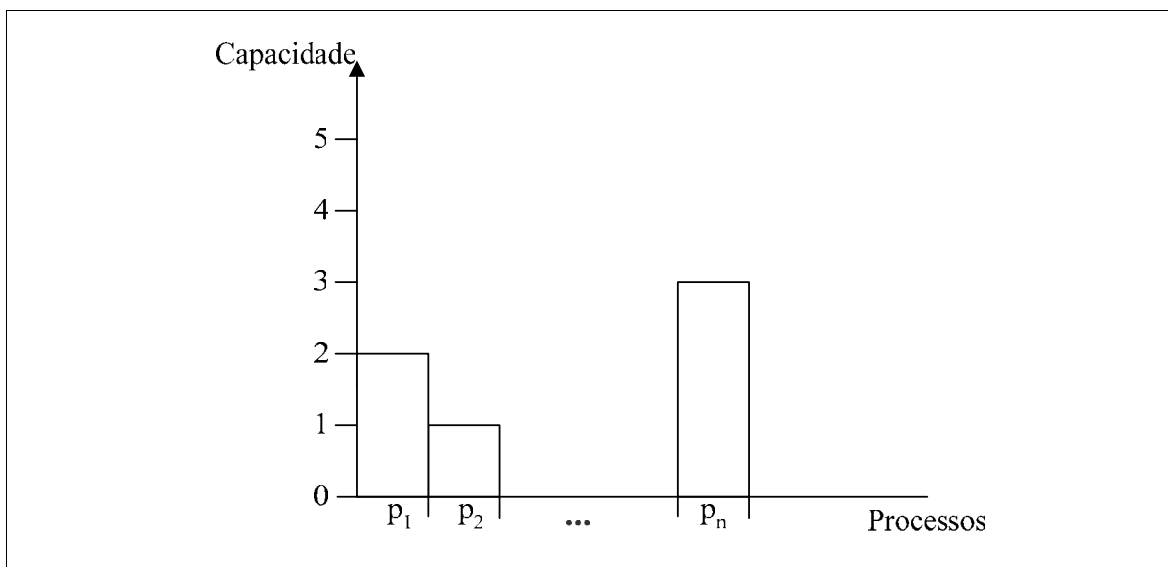


Figura 6.2 – Modelo bidimensional proposto pela norma ISO/IEC 15504.

<sup>2</sup> em média, passar do nível 1 para o nível 2 demora 22 meses e do nível 2 para o nível 3, 19 meses [SEI 2004].

A utilização destas duas dimensões permite obter uma imagem global da capacidade dos processos possibilitando também a melhoria em pequenas etapas, tratando apenas alguns processos específicos.

Como foi apresentado no capítulo quatro, verifica-se a existência de muitos casos de insucesso, quando estes modelos<sup>3</sup> são aplicados em projectos reais de avaliação e melhoria do processo de software. Pretende-se que o referencial 2MPspe trate explicitamente os problemas contextuais associados a um esforço MPS. Na revisão da literatura e nos casos práticos<sup>4</sup> foram detectados três grupos principais de problemas:

- (a) Mudança organizacional
- (b) Gestão do conhecimento
- (c) Envolvimento dos recursos humanos, particularmente a gestão.

No referencial 2MPspe proposto, o tipo (c) é tratado, acrescentando um nível intermédio<sup>5</sup> na dimensão da capacidade. Os tipos de problemas (a) e (b) são abrangidos através da dimensão do conhecimento<sup>6</sup>. A figura 6.3 apresenta o novo referencial, tridimensional, proposto pelo 2MPspe.

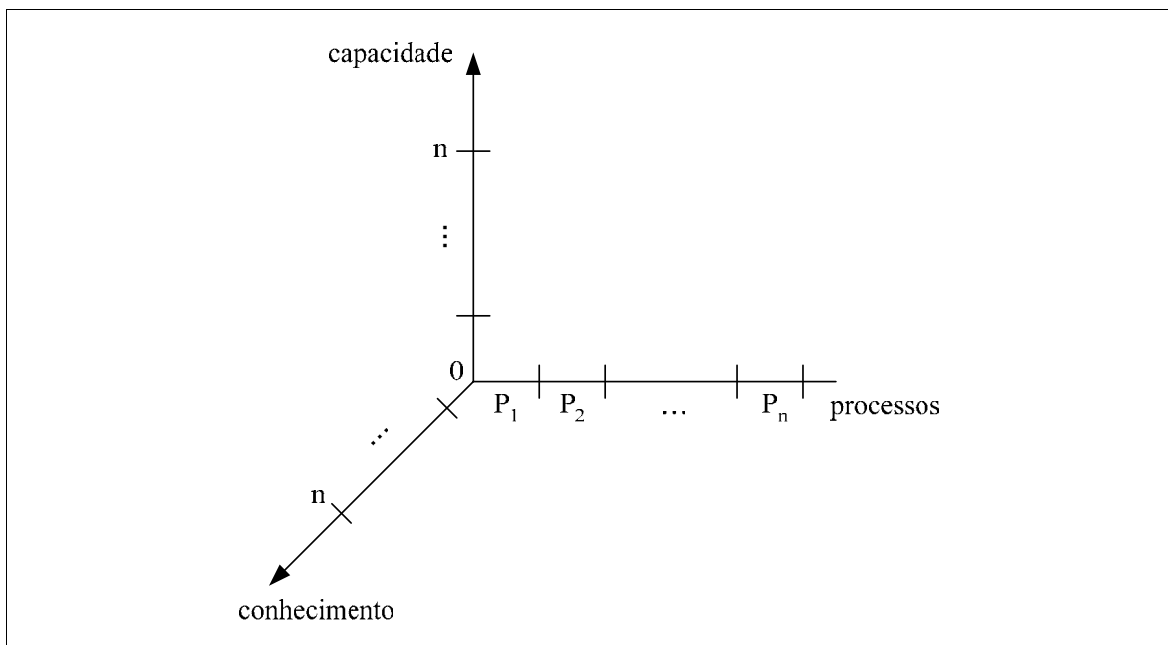


Figura 6.3 – Modelo tridimensional proposto pelo 2MPspe.

<sup>3</sup> o CMM e o SPICE, que deu origem à norma ISO/IEC 15504.

<sup>4</sup> ver capítulo 4.

<sup>5</sup> ver secção 6.3.2.

<sup>6</sup> ver secção 6.4.

O novo referencial inclui a dimensão dos processos, relacionados com o ambiente específico das pequenas empresas de software, a dimensão da capacidade, que permite avaliar as características de desempenho dos processos e finalmente a dimensão do conhecimento, permitindo avaliar o contexto da cultura organizacional e da gestão do conhecimento dos processos.

A figura 6.4 apresenta, simultaneamente, o referencial tridimensional no contexto da avaliação e melhoria do processo de software e o enquadramento genérico das componentes do 2MPspe.

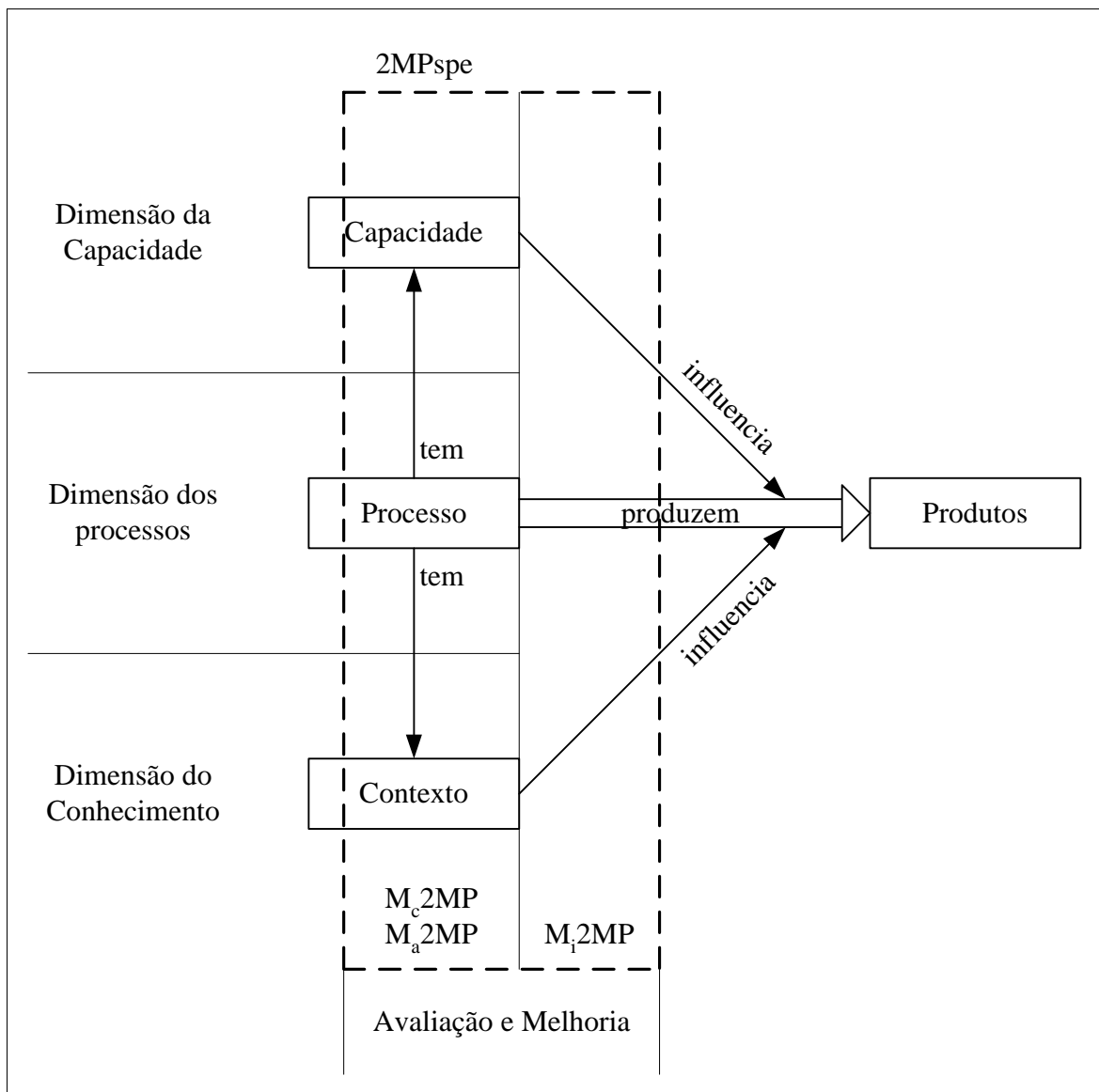


Figura 6.4 – Enquadramento do referencial tridimensional na área de avaliação e melhoria do processo de software.

É importante referir que, devido ao enquadramento das pequenas empresas de software foram efectuadas algumas restrições<sup>7</sup>: i) não são considerados todos os níveis de capacidade, ii) utiliza-se apenas um conjunto limitado de processos e iii) não são modeladas todas as variáveis contextuais<sup>8</sup>.

O modelo conceptual do 2MPspe ( $M_c2MP$ ) apresenta a infra-estrutura necessária para suportar o referencial tridimensional descrito. Na figura 6.5 apresenta-se o Modelo Conceptual do 2MPspe. A dimensão do processo é constituída pelo Modelo de Referência dos Processos (MRP) e pelos processos e baseia-se na norma ISO/IEC 12207, embora ajustada para as pequenas empresas de software. A dimensão da capacidade define os níveis de capacidade, os atributos da capacidade e a escala de Classificação e tem como base a norma ISO/IEC 15504. Nestas duas dimensões, existiu a preocupação de propor um modelo em conformidade com a norma emergente ISO/IEC 15504. A dimensão do conhecimento é constituída pelos níveis de conhecimento, pelos atributos de conhecimento do processo e utiliza a mesma escala de classificação apresentada na dimensão da capacidade.

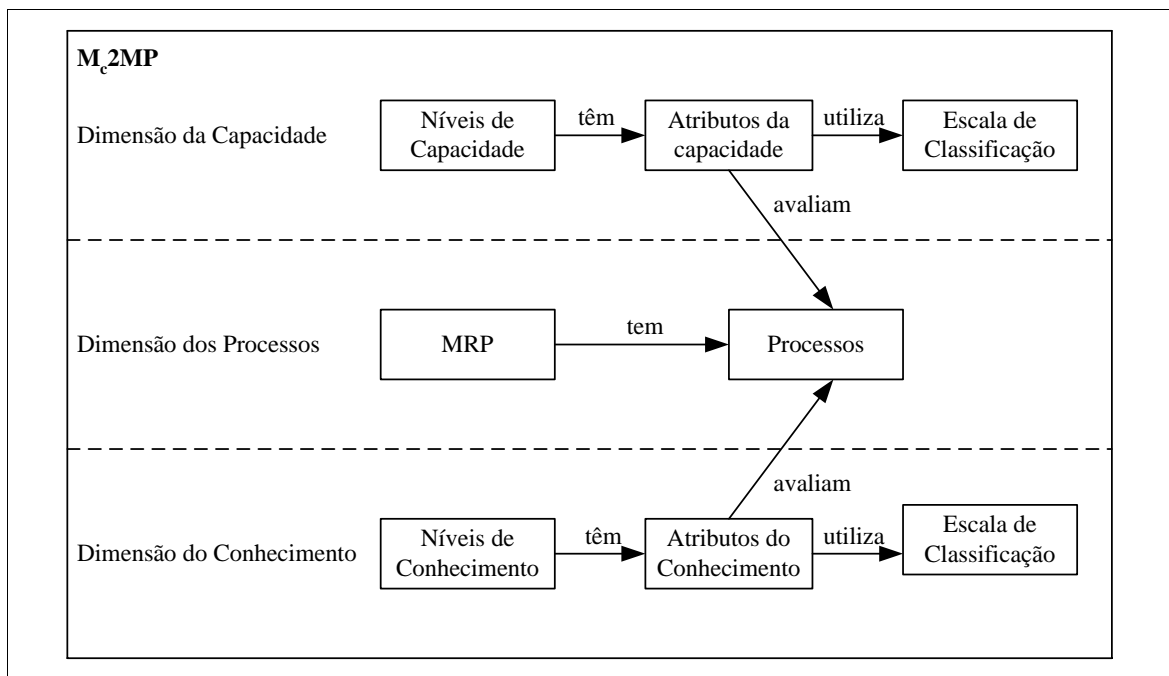


Figura 6.5 – Modelo Conceptual do 2MPspe.

Nas próximas secções desenvolve-se a descrição de cada uma das dimensões.

<sup>7</sup> são explicadas nas secções seguintes.

<sup>8</sup> modeladas com os atributos do conhecimento.

## 6.2. Dimensão dos Processos (Modelo de Referência dos Processos)

Um modelo de referência do processo deve descrever os processos do ciclo de vida do software em relação às metas e aos resultados a atingir, bem como a arquitectura que define os relacionamentos entre os processos [ISO/IEC 2004a]. Ainda seguindo a mesma norma, a meta do processo define os objectivos globais de alto nível<sup>9</sup> (preferencialmente mensuráveis), sendo complementada com a descrição dos resultados observáveis da execução do processo.

Através do modelo de referência dos processos, temos um mecanismo para relacionar o modelo de avaliação dos processos<sup>10</sup> com o referencial de avaliação 15504, proposto pela ISO. O modelo de avaliação dos processos, elemento da componente M<sub>a</sub>2MP, terá como base a descrição dos processos do modelo de referência. A parte 2 da norma ISO/IEC 15504 define os requisitos que devem ser cumpridos por qualquer modelo de referência do processo para estar em conformidade com a norma. Esta conformidade pretende assegurar que os resultados da avaliação<sup>11</sup> são traduzíveis de uma forma fidedigna e repetível para o perfil proposto pela ISO/IEC 15504.

Um modelo de referência do processo deve conter (ISO/IEC 15504 – 2):

- i) Uma declaração do seu âmbito
- ii) Descrições dos processos dentro do âmbito definido
- iii) Uma descrição do relacionamento entre o MRP e o contexto em que será utilizado
- iv) Uma descrição do relacionamento entre os processos definidos em ii)

O modelo de referência do processo deve ainda cumprir as seguintes restrições:

- v) Devem ser documentadas as acções desenvolvidas junto da comunidade alvo, para obter consenso:
  - a) Deve ser descrita a amplitude do esforço
  - b) Caso nenhuma acção tenha sido desenvolvida, deve ser explicitamente declarado o facto

---

<sup>9</sup> normalmente será apenas um parágrafo curto.

<sup>10</sup> ver capítulo 7.

<sup>11</sup> neste caso utilizando o referencial 2MPspe.

- vi) Um MRP no domínio de engenharia de software deve apresentar a correspondência sem ambiguidade para as metas e resultados do processo definidos na norma ISO/IEC 12207.
- vii) Os processos definidos no MRP devem ter uma definição e identificação únicos.
- viii) Qualquer elemento contido no MRP, além dos listados nestes pontos, deve ser considerado meramente informativo.

Nas próximas secções apresenta-se o Modelo de Referência dos Processos (MRP<sub>2MPspe</sub>) especificamente desenvolvido para o M<sub>c</sub>2MP

### 6.2.1. Âmbito do MRP<sub>2MPspe</sub>

O Modelo de Referência dos Processos proposto nesta secção pretende abranger as Pequenas Empresas de desenvolvimento de Software (PES) cuja área de trabalho são os Sistemas de Informação de Gestão (SIG)<sup>12</sup>. Como vimos os valores do ROI e outros parâmetros mostram os benefícios elevados<sup>13</sup> obtidos com iniciativas de MPS com sucesso.

As PES, para se manterem competitivas no mercado do software e para terem sucesso têm que apostar no desenvolvimento de software de alta qualidade, ou seja, devem apostar claramente na Melhoria do Processo de Software [SPIRE 1998], [Russ e McGregor 2000] e [Richardson 2002].

Como foi apresentado, a sensibilidade das PES aponta para a necessidade de realizar simplificações e ajustamentos ao CMM, SPICE e outros métodos, para ser possível a sua aplicação com êxito no ambiente específico das PES [Johnson e Brodman 1997] e [Kuvaja et al. 1999].

A definição de Pequena Empresa de desenvolvimento de Software não é consensual dentro da comunidade de software. Segundo a classificação da Comunidade

---

<sup>12</sup> *Management Information Systems* (MIS).

<sup>13</sup> ver capítulo 4.

Europeia, uma pequena empresa tem entre 1 e 50 funcionários [SPIRE 1998] e [Orci e Laryd 2000] mas Horvat et al. [2000] propõem uma tipologia em que as pequenas empresas possuem entre 1 e 15 funcionários, as médias empresas entre 15 e 50 e as grandes tem mais de 50 trabalhadores.

A realidade portuguesa e a especificidade do nosso mercado mostram um número significativo<sup>14</sup> de empresas de desenvolvimento de software com um número de funcionários inferior a 20.

Estas empresas, de uma forma global, podem ter dois perfis distintos, embora com problemas semelhantes em relação ao processo de software:

[Perfil 1] Empresas flexíveis, com um número restrito de clientes<sup>15</sup>, especializadas numa determinada área (p. ex. seguros, gestão da produção, Internet) e em que cada projecto apresenta características particulares

[Perfil 2] Empresas menos flexíveis que possuem um pequeno conjunto de aplicações. Desenvolvem software horizontal e uma grande parte do seu esforço é dedicado à manutenção

Para o âmbito deste trabalho consideram-se as empresas de desenvolvimento de software com menos de 20 funcionários, denominadas Pequenas Empresas de desenvolvimento de Software (PES). Como já foi referido, o 2MPspe aplica-se a empresas a desenvolver software do tipo Sistemas de Informação de Gestão (SIG).

As empresas a trabalhar na área de desenvolvimento de Sistemas de Informação para Gestão caracterizam-se por construir aplicações para controlar funções do negócio, por exemplo: salários, facturação ou marketing, incluindo tecnologias cliente - servidor e tecnologias Internet [Pressman 1997] e [Jones 2003].

Com base nos casos práticos<sup>16</sup>, apresenta-se em seguida um exemplo de uma empresa típica de cada um dos perfis descritos.

---

<sup>14</sup> aproximadamente 90% das empresas de software.

<sup>15</sup> normalmente com uma boa relação de confiança entre as partes.

<sup>16</sup> ver secção 4.3.1.

A empresa do perfil 1, denominada por XPTO1 está sediada no norte do país e trabalha unicamente para PMEs da região. É uma *Spin-off* da Universidade do Minho recente (2 anos de vida) e desenvolve utilizando preferencialmente Visual Basic e Access como SGBD. Adicionalmente faz desenvolvimento para Web, utilizando tecnologias Internet (html, PhP e outras). A empresa possui 7 colaboradores que acumulam diversas funções. O gerente acumula as tarefas de gestor do projecto, de analista e também faz desenvolvimento. Há uma equipa com três colaboradores que fazem apenas programação, verificando-se que um deles está incumbido também de realizar os testes e dar formação. Outro colaborador dedica-se em exclusivo ao desenvolvimento para a Internet, estando também encarregue do *site* da empresa e do marketing. Por último, há uma colaboradora dedicada à manutenção das aplicações já instaladas e uma funcionária administrativa (ver Figura 6.6).

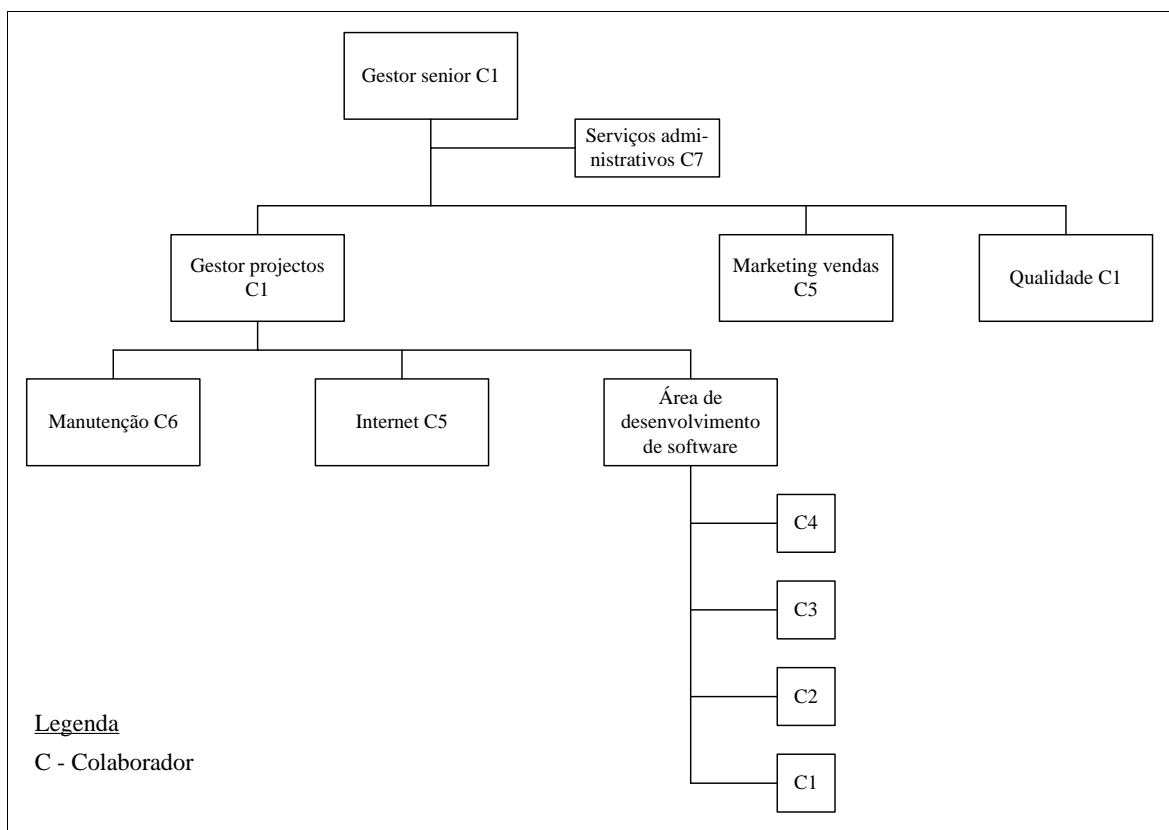


Figura 6.6 – Organigrama da empresa XPTO1.

A empresa possui um pequeno número de clientes (cerca de 25), verificando-se que os clientes estão globalmente satisfeitos com os projectos entregues. A XPTO1 tem muita dificuldade em estimar prazos e quase todos os projectos apresentam alguns



problemas, nomeadamente, horas extra, bastantes defeitos e prazos normalmente ultrapassados. O ponto forte é um bom relacionamento com os clientes.

Em relação ao desenvolvimento, verifica-se uma inexistência de documentação da concepção, falta de definição de procedimentos e normas, gestão de projecto não formalizada, gestão de configurações rudimentar e testes muitas vezes adulterados.

A empresa do perfil 2, denominada XPTO2 está sediada no norte do país e possui 18 colaboradores. É uma empresa com 10 anos de existência e trabalha fundamentalmente com três produtos que coloca no mercado através da sua rede de revendedores. Em relação ao organigrama (ver Figura 6.7), a empresa possui um director geral e cinco áreas chefiadas por um coordenador: Marketing, Assistência Pós Venda, Expedição, Qualidade e Desenvolvimento.

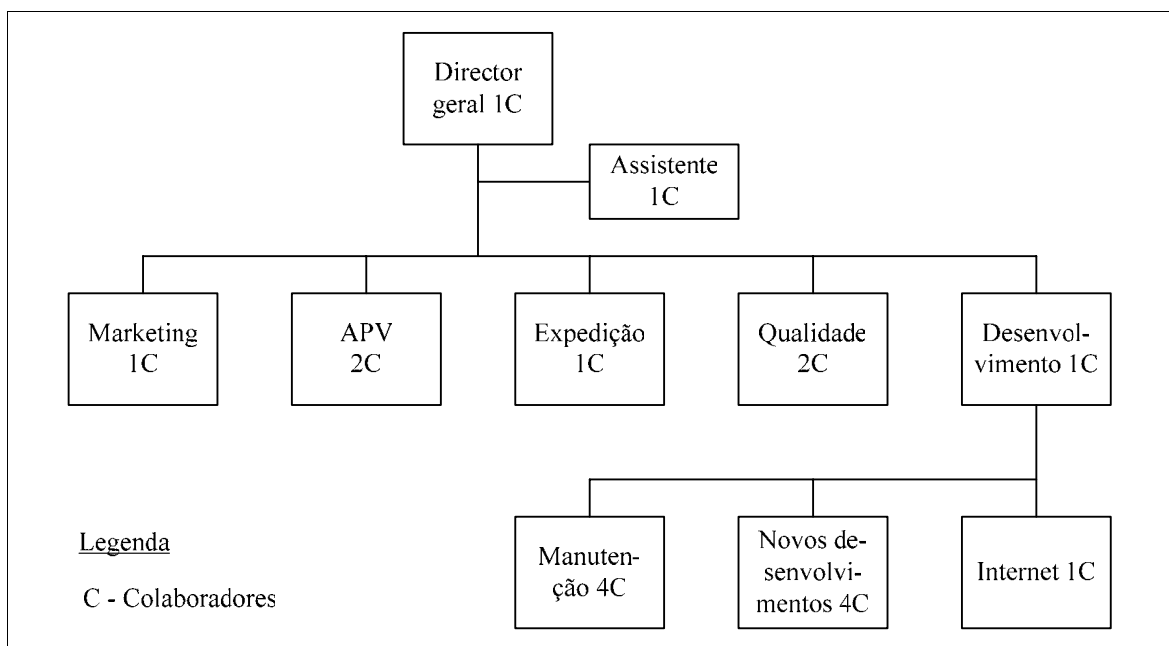


Figura 6.7 – Organigrama da empresa XPTO2.

A empresa possui ainda algum código em linguagens da 3ª geração, mas está continuamente a realizar a sua conversão. Todos os novos desenvolvimentos são realizados em Visual Basic, tendo como suporte o SQL SERVER 2000. A empresa possui cerca de 30 vendedores e estima que o seu número de clientes finais rondará os 600.

Em relação ao seu processo de desenvolvimento, a empresa sente preocupações recentes em relação à formalização e institucionalização dos processos, realiza o planeamento inicial dos novos projectos, embora note muitas falhas em relação ao seu acompanhamento e replaneamento. A realização de estimativas dos prazos é muito rudimentar, notando-se muitas falhas nesta área. Os profissionais da empresa sentem necessidades de melhoria em diversas áreas, nomeadamente, reutilização, gestão de configurações, documentação da análise e concepção e também na garantia da qualidade do software.

Estas PES, muito importantes no contexto do mercado português, apresentam um conjunto de características comuns que devem ser referidas:

- Escassez de recursos financeiros, o que provoca dificuldades em investir na Melhoria dos Processos de Software, aliado à dificuldade em disponibilizar Recursos Humanos em *full-time*.
- Dependência de um pequeno número de projectos, tornando cada projecto crucial para o bem estar financeiro da empresa
- Pequeno número de colaboradores, verificando-se a acumulação de tarefas
- Uma grande dependência nos indivíduos, fazendo com que os factores humanos e os aspectos psicológicos tenham um forte impacto em qualquer programa de melhoria
- Importância do relacionamento e satisfação do cliente
- Muitas horas extra solicitadas aos colaboradores e filosofia de “apagar fogos”
- Elevado nível de improvisação

Estas características particulares das PES serão tomadas em consideração na definição do Modelo de Referência do Processo, apresentado nas próximas secções.

### **6.2.2. Estrutura do $MRP_{2MP_{spe}}$**

Nesta secção seleccionam-se os processos que irão pertencer ao  $MRP_{2MP_{spe}}$  tendo em consideração o âmbito definido. Apresenta-se o relacionamento entre os processos e o relacionamento do  $MRP_{2MP_{spe}}$  com o contexto.

O MRP<sub>2MPspe</sub> foi desenvolvido para ultrapassar as dificuldades<sup>17</sup> apresentadas pelo SPICE e pelo CMM; tem como base a revisão da literatura efectuada e os casos práticos<sup>18</sup> desenvolvidos em PES da região norte. O ponto de partida do MRP foi o proposto pela norma ISO/IEC 12207 tendo em consideração o objectivo de atingir apenas o nível 3 da norma ISO/IEC 15504. Este requisito baseia-se na constatação do nível de Maturidade mais comum das PES ser o nível 0<sup>19</sup> e assim as preocupações emergentes nos níveis 4 e 5 estarem fora dos horizontes de curto e médio prazo das PES [SPIRE 1998].

Considerou-se benéfico realizar um estudo comparativo com outros MRPs. A selecção dos MRPs utilizados no estudo comparativo foi realizada seguindo três critérios: (1) cariz eminentemente prático; (2) divulgação e número elevado de casos práticos e (3) ter como população alvo as PES. Da aplicação destes critérios resultaram 8 modelos de melhoria do processo<sup>20</sup>.

Como já foi referido, a norma ISO/IEC 12207 documenta “o conjunto de processos de engenharia de software fundamentais para uma boa engenharia de software e cobrindo as melhores práticas” [ISO/IEC 2004b]. O MRP proposto divide-se em três categorias de processos (ver Figura 6.8), verificando-se a existência de 22 processos principais e 29 sub-processos (44 processos primários).

---

<sup>17</sup> ver secção 4.

<sup>18</sup> casos práticos descritos na secção 4.3.

<sup>19</sup> Rocha e Vasconcelos [2003] apresentam 5 empresas portuguesas classificadas no nível 1 do CMM.

<sup>20</sup> ver 6.2.3.1.

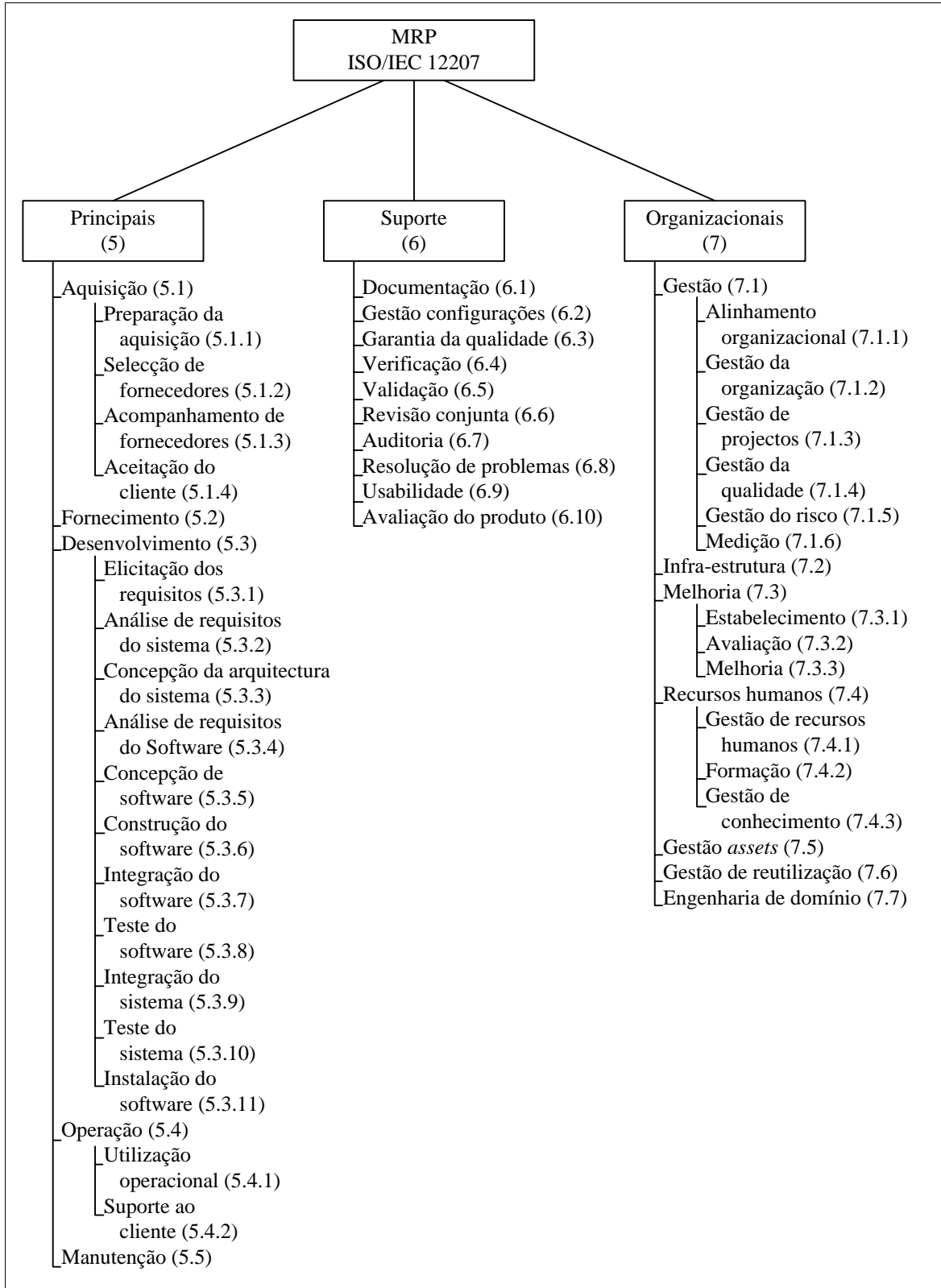


Figura 6.8 – Modelo de Referência dos processos proposto pela norma ISO/IEC 12207: 1995 / Amd1 [ISO/IEC 2002b].

Para a definição da estrutura do MRP<sub>2MPspe</sub> foram considerados os modelos apresentados pelo CMM e pelo BOOTSTRAP [Kuvaja 1999]. O CMM<sup>21</sup> propõe uma divisão em três categorias [Paulk et al. 1995]: processos de gestão, processos organizacionais e processos de engenharia. A figura 6.9 apresenta as áreas chave do processo dos três primeiros níveis.

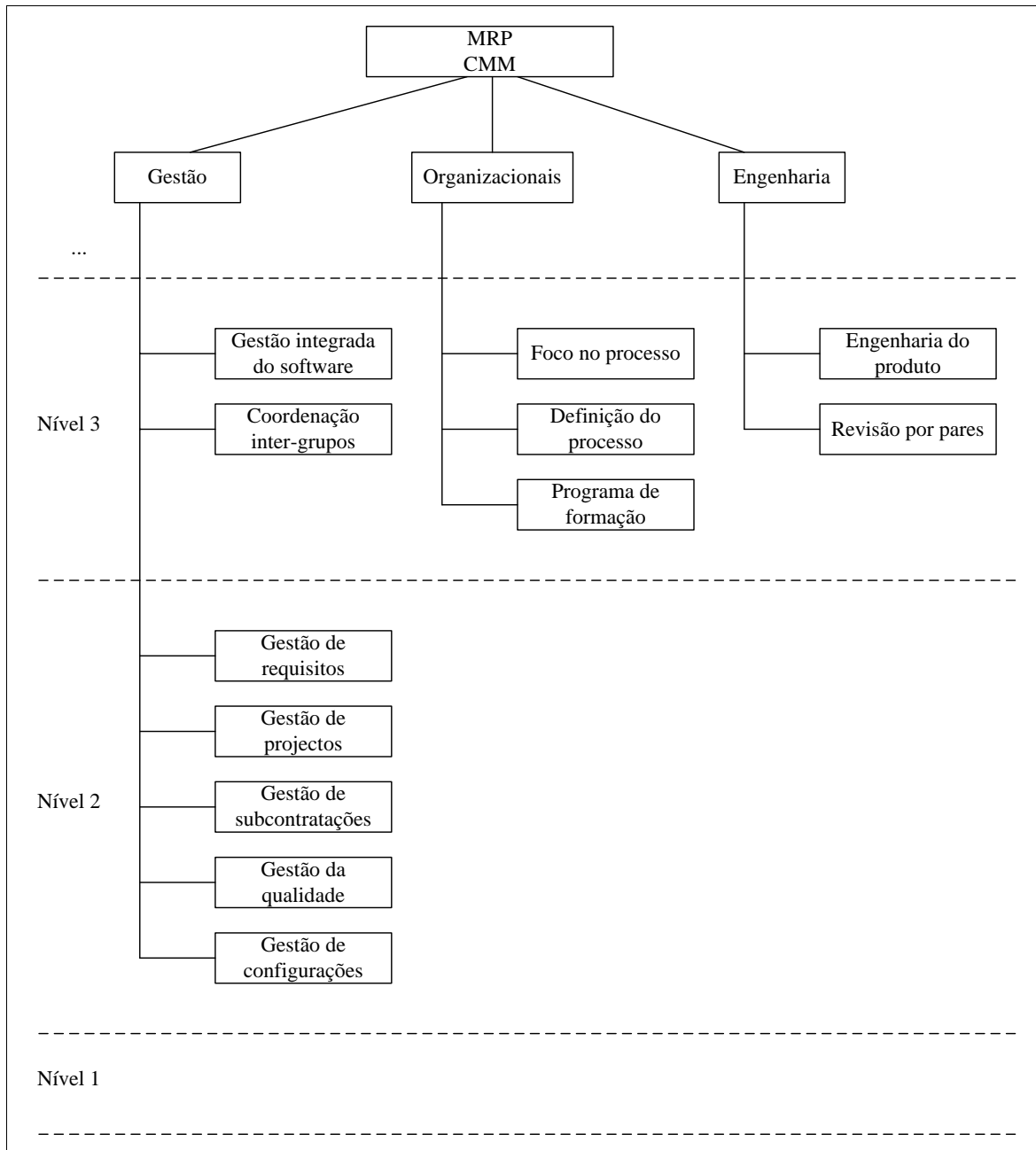


Figura 6.9 – Modelo de Referência dos Processos proposto pelo CMM.

<sup>21</sup> ver secção 3.3.1.1.

A metodologia BOOTSTRAP tinha, como vimos<sup>22</sup>, os objectivos iniciais de impulsionar a utilização de boas práticas de engenharia de software e conhecer as preocupações em relação à Melhoria do Processo de Software das empresas de desenvolvimento de software europeias. O BOOTSTRAP inclui as seguintes componentes [Kuvaja 1999]:

- Método de avaliação
- O modelo de referência dos processos
- Os níveis de capacidades
- Princípios para a clarificação e apresentação dos resultados
- Orientações para a melhoria do processo

Em relação ao MRP proposto pelo BOOTSTRAP, baseia-se em normas internacionalmente reconhecidas<sup>23</sup> e divide-se em três categorias de processos (ver Figura 6.10): processos organizacionais, processos metodológicos e processos tecnológicos.

Pretende-se que o  $MRP_{2MPspe}$  seja aplicável em PES que desenvolvam o produto software na área dos Sistemas de Informação de Gestão. O  $2MPspe$  deverá cumprir também os objectivos de estar em conformidade com a ISO/IEC 15504 e apresentar correspondência para o CMM e para a ISO 9001 : 2000.

---

<sup>22</sup> ver secção 3.3.1.4.3.

<sup>23</sup> ISO 9001 e 9000 – 3, ISO 12207, ISO 15504, ESA PSS – 05 – 0 e CMM.

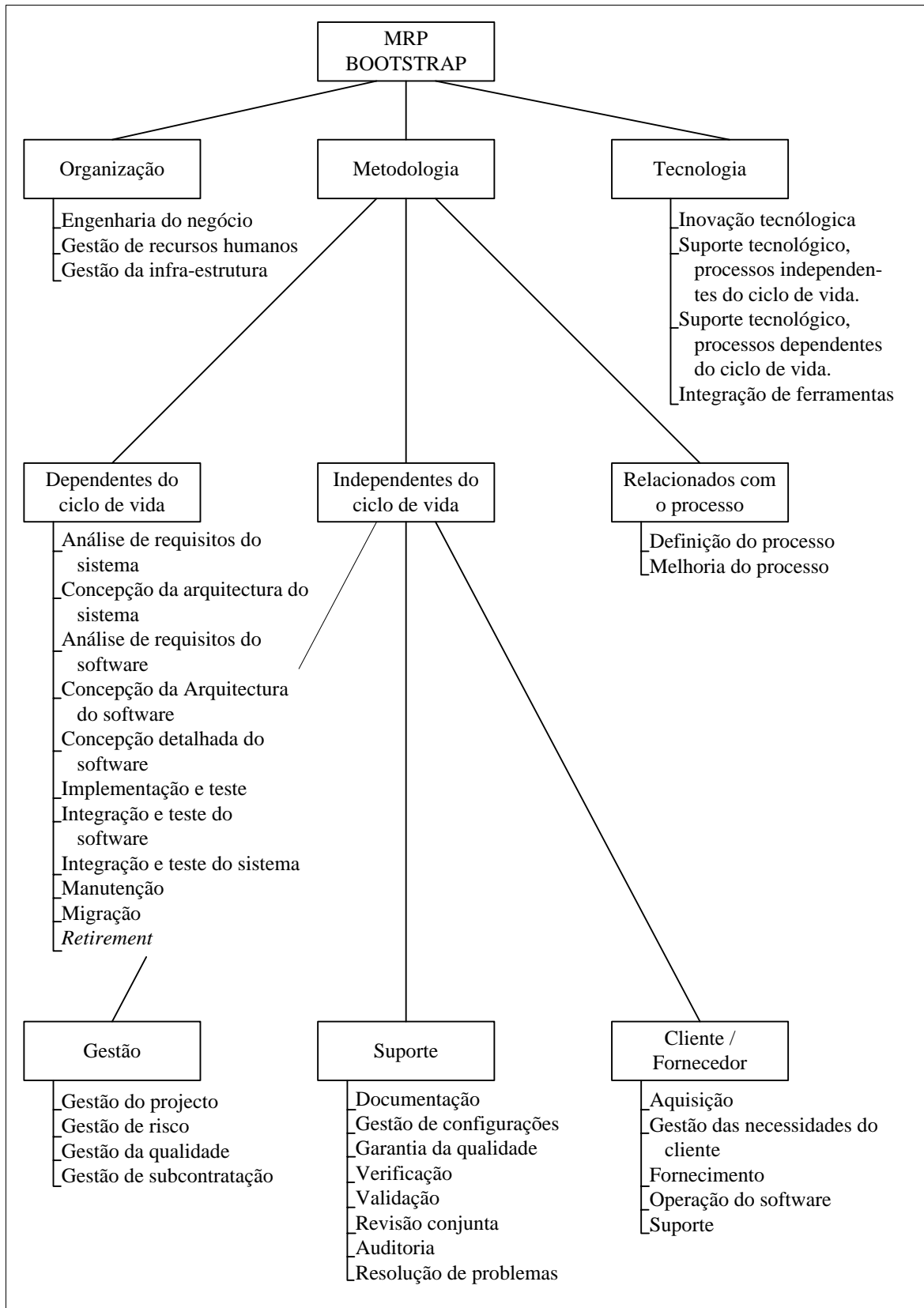


Figura 6.10 – Modelo de Referência dos Processo do BOOTSTRAP.

Nesta perspectiva dividiu-se o  $MRP_{2MPspe}$  em quatro áreas de processos (ver Figura 6.11):

- Engenharia, processos directamente relacionados com a realização do produto
- Suporte, processos utilizados por outros processos e com um contributo significativo para a qualidade final do produto
- Gestão, processos que contêm tarefas e actividades genéricas que podem ser utilizadas por quem faça a gestão de qualquer tipo de projecto ou processo
- Organizacionais, processos utilizados no nível da organização

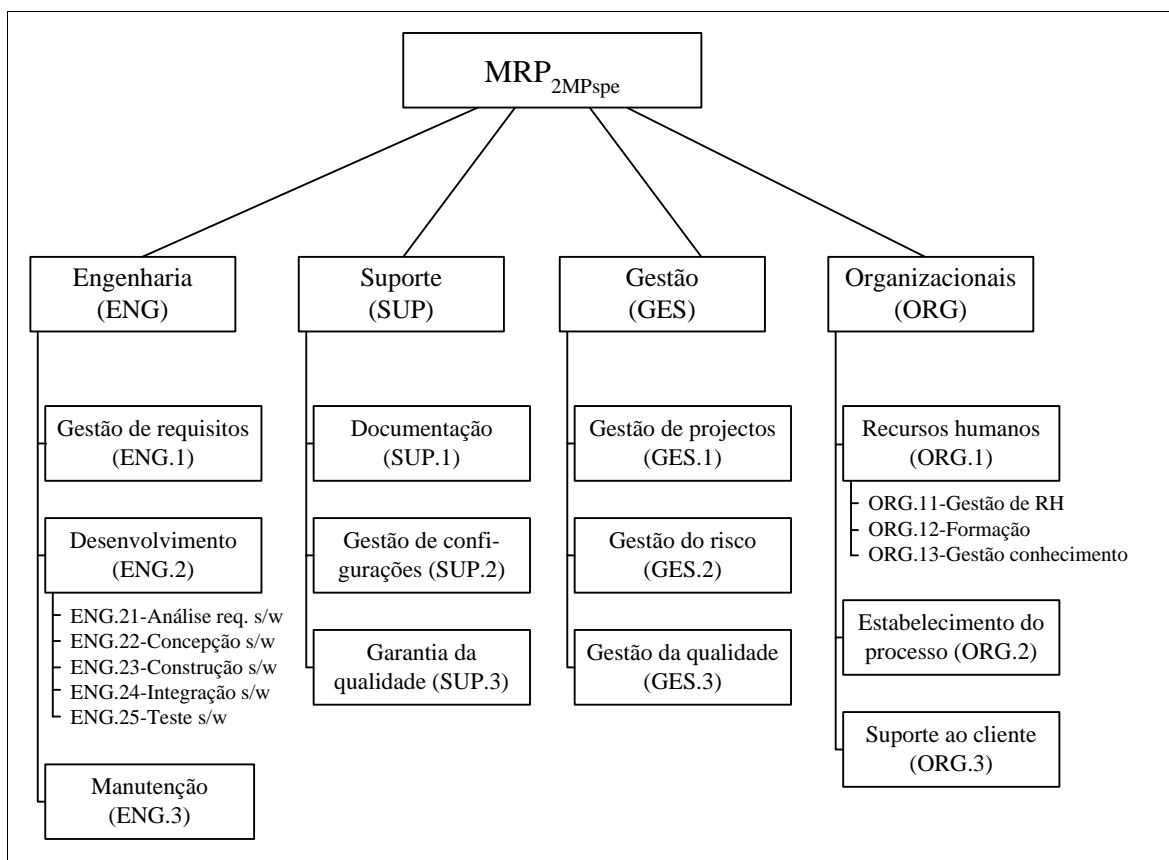


Figura 6.11 – Modelo de Referência dos Processos do 2MPspe.

O  $MRP_{2MPspe}$  divide-se em quatro áreas de processos constituídas por 12 processos principais. Uma vez que os processos de desenvolvimento e de recursos humanos estão agregados, o modelo proposto pelo 2MPspe possui 18 processos primários. Estes processos são considerados os processos prioritários nas PES e serão os



processos sujeitos à avaliação do desempenho, capacidade e conhecimento. A avaliação será executada utilizando o questionário 2MPspe<sup>24</sup>.

O processo de Desenvolvimento (ENG.2) é desagregado, considerando apenas um conjunto restrito dos processos propostos na norma ISO/IEC 12207. A selecção centrou-se nos processos directamente relacionados com a realização do software<sup>25</sup>, pensando-se assim cobrir uma perspectiva mais alinhada com as preocupações de curto prazo das PES. O processo Recursos Humanos (ORG.1) é desagregado nos processos: Gestão de Recursos Humanos (ORG.11), Formação (ORG.12) e Gestão de Conhecimento (ORG.13).

Os processos centrais serão os processos relacionados com a engenharia do produto, isto é, Gestão de Requisitos (ENG.1), Desenvolvimento do Software (ENG.2) e Manutenção (ENG.3). O processo de Gestão de Requisitos será tratado independentemente do processo global de desenvolvimento, por uma questão de alinhamento com o CMM e numa perspectiva de definir prioridades específicas para este processo. Como escreveu Brooks [1987], “A parte mais difícil na construção de um sistema de software é decidir precisamente o que construir... Nenhuma outra parte é mais difícil de rectificar mais tarde”.

Os processos Gestão de Configurações (SUP.2), Garantia da Qualidade (SUP.3), Gestão de Projectos (GES.1) e Recursos Humanos (ORG.1) foram seleccionados devido ao facto de permitirem atingir os níveis 2 e 3 da escala de maturidade do ISO/IEC 15504 e suportarem o conhecimento dos factores contextuais<sup>26</sup> tratados no 2MPspe. Os processos Documentação (SUP.1), Gestão do Risco (GES.2) e Estabelecimento do Processo (ORG.2), além dos objectivos anteriores, têm também uma perspectiva estratégica, ou seja, criar as fundações para atingir os níveis de capacidade mais elevados e possibilitar a melhoria contínua.

---

<sup>24</sup> ver secção 7.5.1.

<sup>25</sup> excluíram-se os processos relacionados com o “sistema”, sem querer significar uma menor importância destes processos.

<sup>26</sup> secção 6.4.

Especificamente com o objectivo de cumprir os requisitos exigidos pela norma ISO 9001: 2000 foi incluído o processo Gestão da Qualidade (GES.3). Este processo implica o estabelecimento de um sistema de gestão de qualidade para monitorizar a qualidade do produto e assim atingir a satisfação do cliente. Finalmente, para o ambiente operacional das PES foi também incluído o processo Suporte ao Cliente (ORG.3). Este processo tem como objectivo a definição e estabelecimento de um nível de serviço adequado para o cliente. Como já foi referido, nas PES, o relacionamento e satisfação do cliente é um factor crítico de sucesso para o negócio.

A descrição dos processos do  $MRP_{2MPspe}$  encontra-se na secção 6.2.4. Na próxima secção apresenta-se um estudo comparativo com outros MRPs seleccionados e algumas considerações finais sobre o Modelo de Referência do Processo do  $2MPspe$ .

### 6.2.2.1. Estudo Comparativo

Considerou-se importante, para validar o  $MRP_{2MPspe}$ , a realização de um estudo comparativo com outros MRPs. Dado o elevado número de modelos de melhoria do processo, foram aplicados três critérios para seleccionar um conjunto restrito. Os critérios utilizados foram: (1) cariz eminentemente prático; (2) divulgação e número elevado de casos práticos e (3) ter como população alvo as PES.

Foram escolhidos sete modelos cumprindo pelo menos dois dos critérios, a saber, ISO/IEC 15504 [ISO/IEC 2004a], QMIM<sup>27</sup> [Balla et al. 2001], RAPID<sup>28</sup> [Rout et al. 2000], FCS<sup>29</sup> - Motorola [Fitzgerald e O'Kane 1999], TAPISTRY<sup>30</sup> [Kuvaja et al. 1999], CMM [Paulk et al. 1995] e Trillium [Bell 1994]. Foi seleccionado também um modelo de maturidade empiricamente derivado [Drehmer e Dekleva 1993]. Embora este último modelo não cumpra nenhum dos critérios, apresenta a característica importante de ser derivado tanto de casos de sucesso como de casos de insucesso e ser derivado empiricamente do caminho evolutivo seguido pelas organizações.

---

<sup>27</sup> *Quality through Managed Improvement and Measurement.*

<sup>28</sup> *Rapid Assessment for Process Improvement for software Development.*

<sup>29</sup> Factores Críticos de Sucesso.

<sup>30</sup> *Tailored Application of Software Improvement Techniques for Small Enterprises.*

A análise apresentada em seguida foi realizada, procurando os processos chave propostos pelos modelos para atingir os níveis 2 e 3 da escala de classificação proposta pela norma ISO/IEC 15504, nomeadamente, Processo Gerido e Processo Estabelecido<sup>31</sup>. Apresenta-se em seguida uma breve descrição de cada modelo, mostrando o seu âmbito, restrições e processos propostos. Como base da comparação, foram utilizados os 22 processos principais propostos pela ISO/IEC 12207 (ver figura 6.8). Apenas os processos (7.1) Gestão e (7.4) Recursos Humanos foram desagregados nos seus componentes. Descrevem-se inicialmente os quatro modelos de âmbito genérico, ou seja, ISO/IEC 15504, FCS – Motorola, CMM e Trillium. Seguidamente apresentam-se os três modelos específicos para PES, QMIM, RAPID e TAPISTRY e finalmente o modelo empiricamente derivado.

#### **(ISO/IEC 15504: 2004)**

Esta norma encontra-se descrita na secção 3.3.1.2., contudo nesta análise comparativa, o objectivo é verificar quais os processos necessários para atingir os níveis de capacidade 2 e 3. Sabendo que cada nível de maturidade tem associado determinados Atributos do Processo (APs), interessa saber quais os processos relacionados com os atributos do processo dos níveis 2 e 3. A informação detalhada em seguida, baseia-se na parte 5 da norma ISO/IEC 15504 [ISO/IEC 2003].

Para atingir o nível 1<sup>32</sup> é necessário cumprir o atributo de execução do processo (AP 1.1). Este atributo do processo não tem nenhum processo relacionado.

O nível 2 (processo gerido) obriga ao cumprimento do atributo de gestão da execução (AP 2.1) e do atributo de gestão dos produtos de trabalho (AP 2.2). Os processos relacionados com o atributo do processo AP 2.1 são: gestão de projectos (7.1.3), gestão de risco (7.1.5), resolução de problemas (6.8) e garantia de qualidade (6.3). Em relação ao atributo do processo AP 2.2, os processos relacionados são: documentação (6.1), gestão de configurações (6.2), resolução de problemas (6.8), garantia da qualidade (6.3), verificação (6.4) e revisão conjunta (6.6) Verifica-se que o

---

<sup>31</sup> ver secção 3.3.1.1.

<sup>32</sup> processo executado.

nível 2 está centrado, principalmente, na área dos processos de suporte com a excepção da gestão de risco, classificada na categoria dos processos organizacionais.

O terceiro nível (processo estabelecido) possui dois atributos do processo: atributo de definição do processo (AP 3.1) e atributo de transferência<sup>33</sup> do processo (AP 3.2). Os processos relacionados com o AP 3.1 são: gestão da organização (7.1.2), estabelecimento do processo (7.3.1), gestão de recursos humanos (7.4.1), gestão do conhecimento (7.4.3), infra-estrutura (7.2), gestão de *assets* (7.5), engenharia do domínio (7.7) e auditoria (6.7). Finalmente, em relação a AP 3.2, temos os seguintes processos: gestão da organização (7.1.2), gestão do projecto (7.1.3), gestão da qualidade (7.1.4), formação (7.4.2) infra-estrutura (7.2) e auditoria (6.7).

Refira-se ainda, que independentemente do nível a atingir num plano de melhoria, a norma aconselha que os processos alvo pertençam ao grupo de processos primários (ISO/IEC 2003b). Na parte 4 da norma, sobre o estabelecimento da capacidade alvo é referido como passo 1, “Seleccionar os processos primários do ciclo de vida, excluindo os processos não relevantes para os requisitos especificados” (ISO/IEC 2003b). Apresenta-se na coluna 2 da tabela 6.2 o resumo da norma 15504.

### **(FCS<sup>34</sup> - Motorola: 1999)**

A abordagem utilizada na Motorola, para a melhoria do processo de software, baseou-se no método dos Factores Críticos de Sucesso (FCS) [Butler e Fitzgerald 1999], aplicado às áreas chave do processo características dos níveis propostos pelo CMM. Apresenta-se apenas a estratégia seguida para cumprir os níveis 2 e 3 do CMM. O objectivo de aplicar o conceito dos FCS prende-se com o carácter dinâmico e de melhoria contínua associado aos FCS ao contrário das KPAs que possuem um valor binário.

Para atingir o nível 2 do CMM, foram definidos quatro FCS: papéis dedicados ao planeamento e acompanhamento do projecto (FCS 2.1); procedimentos formalizados para gerir as subcontratações (FCS 2.2); institucionalização da garantia da qualidade do

---

<sup>33</sup> *Deployment.*

<sup>34</sup> Factores Críticos de Sucesso.

software (FCS 2.3) e automatização da gestão de configurações do software, como parte intrínseca do ciclo de vida do software (FCS 2.4).

No nível 3 são utilizados cinco FCS, a saber: definição simples, mas formal dos processos e caracterização da estratégia para “colocar à medida” (FCS 3.1); cultura de melhoria do processo de software, forte e difundida por todos os níveis da organização (FCS 3.2); bom programa de formação com ênfase nas tecnologias líderes (FCS 3.3); forte cooperação tanto interna como com outras organizações externas (FCS 3.4) e realização da revisão por pares em todos os produtos do software (FCS 3.5) [Fitzgerald e O’Kane 1999].

A correspondência desta estratégia de melhoria com a ISO 12207 encontra-se na tabela 6.2 (coluna 3). Nota-se um forte alinhamento com as KPAs do nível 2 do CMM, embora os FCS do nível 3 apresentem características mais abrangentes e ligadas a factores humanos.

### ***(Capability Maturity Model – CMM: 1995)***

O CMM encontra-se descrito na secção 3.3.1.1. Para realizar este estudo comparativo foi necessário analisar as actividades<sup>35</sup> das áreas chave do processo<sup>36</sup> e realizar a correspondência para os processos da ISO/IEC 12207.

Nas áreas chave do processo (KPA) do nível 2<sup>37</sup> do CMM, verifica-se que a KPA Gestão de Requisitos tem correspondência directa para (5.3.1) Gestão de Requisitos. As KPAs Planeamento do Projecto de Software (PPS) e Acompanhamento do projecto de Software (APS) cobrem os processos (5.2) Fornecimento, (7.1.2) Gestão da Organização e Gestão de projectos (7.1.3). A actividade 13 do APS cobre a Revisão Conjunta (6.6) e a actividade 13 de PPS inclui a gestão de risco (7.1.5). A KPA Gestão de Subcontratações<sup>38</sup> corresponde ao processo (5.1) Aquisição. A KPA Garantia da Qualidade do Software cobre os processos (6.3) Garantia da Qualidade e (6.7)

---

<sup>35</sup> utiliza-se a numeração definida no CMM para as actividades.

<sup>36</sup> *Key Process area* – KPA.

<sup>37</sup> denominado nível repetível.

<sup>38</sup> quando a organização subcontrata trabalho.

Auditoria. Finalmente, a Gestão de Configurações cobre o processo com o mesmo nome (6.2) e a actividade 5 inclui o processo de resolução de problemas (6.8).

Em relação ao nível Gerido<sup>39</sup>, a KPA, Foco no Processo Organizacional, cobre a Avaliação (7.3.2) e Melhoria (7.3.3) do Processo. A KPA, Definição do Processo Organizacional corresponde aos processos Infra-estrutura (7.2) e Estabelecimento do Processo (7.3.1). A KPA, Programa de Formação cobre os processos de gestão de recursos humanos (7.4.1) e formação (7.4.2). Os processos gestão da organização (7.1.2) e gestão de projectos (7.1.3) da ISO 12207 estão também englobados na KPA Gestão Integrada do Software. No que se refere à área chave do processo, Engenharia do Produto de Software, verifica-se que engloba os processos de desenvolvimento (5.3) e suporte ao cliente (5.4.2). A actividade 8 inclui a documentação, as actividades 5 e 6 cobrem a verificação (6.4), a actividade 5 corresponde também à validação (6.5.) e finalmente, as actividades 7 e 9 cobrem a avaliação do produto (6.10). Sobre a engenharia do produto, refira-se que o CMM não inclui os sub-processos relacionados com o sistema, nomeadamente: análise, concepção da arquitectura, integração e teste. Por fim, as KPAs, Coordenação Intergrupos e Revisões por pares não possuem processos semelhantes na norma ISO 12207.

Refira-se que o processo Medição (7.1.6) proposto na ISO 12207 é coberto pela característica comum denominada “medições e análise” e não directamente por uma actividade ou KPA. O posicionamento do CMM no contexto da norma ISO/IEC 12207 pode ser analisado na tabela 6.2 (coluna 4), ressaltando a falta de cobertura do processo (5.5) Manutenção. É importante comentar que os processos da ISO 12207 com cobertura através de actividades dentro das KPAs, apresentam um nível de detalhe menor no CMM. Este facto foi tomado em conta na concepção do MRP<sub>2MPspe</sub>.

### **(Trillium – Bell: 1994)**

O modelo Trillium<sup>40</sup> foi desenvolvido pela Bell Canada em 1994, tendo como base principal o CMM. O objectivo do modelo é fornecer orientações para um programa

---

<sup>39</sup> nível 3 do CMM.

<sup>40</sup> ver secção 3.3.1.4.1.

de melhoria contínua do processo. Embora seja orientado para as telecomunicações, pode ser utilizado em qualquer outro tipo de organização [Zahran 1998].

Para o estudo comparativo interessa analisar os *roadmaps* e as práticas propostas para os níveis 2 e 3. A tabela 6.1 apresenta a correspondência com a ISO/IEC 12207.

Área de capacidade	Roadmap	Nível 2	Nível 3
Qualidade do processo organizacional	Gestão da qualidade	--	(7.1.4)
	Engenharia do processo de negócio	--	--
Gestão e desenvolvimento dos recursos humanos	Gestão e desenvolvimento dos recursos humanos	--	(7.4.1) (7.4.2)
Processo	Definição do processo	--	(7.3.1) (7.2)
	Gestão da tecnologia	--	--
	Engenharia e melhoria do processo	--	(7.3.2) (7.3.3)
	Medições	(7.1.6)	(7.1.4)
Gestão	Gestão do processo	(7.1.3) (7.1.5)	--
	Gestão de subcontratações	(5.1)	--
	Relacionamento cliente – fornecedor	(5.1)	--
	Gestão de requisitos	(5.3.1)	--
	Estimação	(7.1.3)	--
Sistema de qualidade	Sistema de qualidade	(6.3) (6.6) (6.7)	--
Práticas de desenvolvimento	Processo de desenvolvimento	(5.3)	--
	Técnicas de desenvolvimento	(5.3)	--
	Documentação interna	(6.1)	--
	Verificação validação	(5.3)	(6.4) (6.5)
	Gestão de configurações	(6.2)	--
	Reutilização	--	(7.6)
Ambiente de desenvolvimento	Ambiente de desenvolvimento	(7.2)	--
Suporte ao cliente	Sistema de resolução de problemas	(6.8)	--
	Engenharia de usabilidade	(6.9)	--
	Modelo do custo do ciclo de vida	--	--
	Documentação do utilizador	(5.4.2)	--
	Engenharia do cliente	(5.2)	--
	Formação do utilizador	(5.2)	--

Nota: A designação dos processos ISO/IEC 12207 encontra-se na tabela 6.2.

Tabela 6.1 – Correspondência dos roadmaps com a ISO 12207.

O *roadmap* gestão da qualidade corresponde ao processo (7.1.4), mas vai mais além englobando também um programa TQM, formação TQM e comunicação com os funcionários. Verifica-se que os seguintes *roadmaps* não têm nenhuma cobertura directa na norma ISO/IEC 12207: engenharia do processo de negócio, gestão da tecnologia, gestão da confiança e modelos de custeio do ciclo de vida.

**(QMIM – IQSOFT: 2001)**

Este modelo resultou de uma experiência<sup>41</sup> de melhoria do processo de software que decorreu na empresa IQSOFT<sup>42</sup>, sediada na Hungria. A empresa abrange três áreas de actividade: desenvolvimento de software, integração de software e implementação de software. Os projectos têm características muito diversificadas embora o seu tamanho seja normalmente pequeno ou médio [Balla et al. 2001].

O QMIM baseia-se num referencial de qualidade (ver figura 6.12), na dimensão horizontal situam-se os objectos centrais da melhoria, ou seja, gestão de projectos, processo técnico e produto. A dimensão vertical representa a especificação da qualidade com as áreas de definição, atributos de qualidade e métricas.

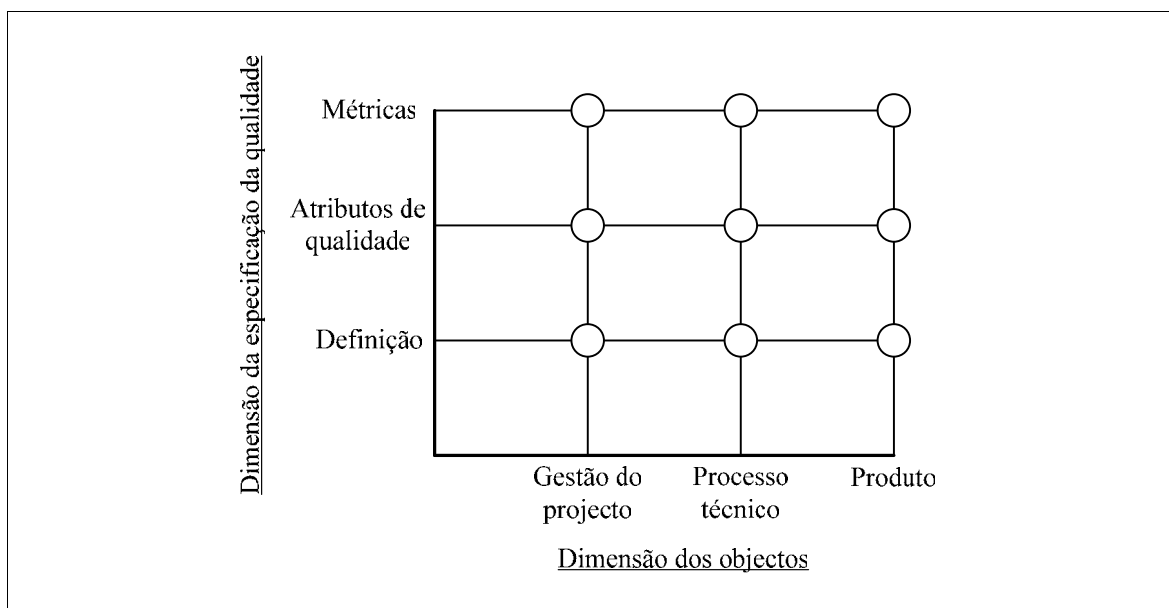


Figura 6.12 – Referencial de Qualidade do QMIM [Balla et al. 2001].

O QMIM sugere uma abordagem assente nas seguintes actividades básicas, para tratar dos aspectos da qualidade e melhoria:

1. Definir gestão de projectos (englobando os Recursos Humanos)
2. Definir processos técnicos
3. Definir produtos

<sup>41</sup> o esforço de melhoria durou 8 anos.

<sup>42</sup> com 74 colaboradores.



#### 4. Definir métricas de suporte

A correspondência com os processos da norma ISO 12207 encontra-se na coluna 6 da tabela 6.2.

#### **(RAPID – *Rapid Assessment for Process Improvement for software Development: 2000*)**

O modelo RAPID [Rout et al. 2000] é um modelo de melhoria do processo de software, específico para PMEs na área de software e conforme a norma ISO/IEC 15504. Encontra-se descrito na secção 3.3.1.4.2.

O modelo RAPID adopta um MRP restringido apenas a oito processos. Os processos centrais da avaliação e melhoria são o desenvolvimento (5.3) e a elicitação de requisitos (5.3.1). Refira-se que o processo de desenvolvimento é avaliado de uma forma agregada. Em relação aos processos necessários para atingir o nível de gestão do processo, o RAPID selecciona as áreas de gestão de projectos (7.1.3), gestão de configurações (6.2), garantia da qualidade (6.3) e resolução de problemas (6.8). Finalmente os processos, gestão de risco (7.1.5) e estabelecimento do processo (7.3.1) foram escolhidos para fornecerem os “alicerces” para atingir os níveis mais elevados da escala de maturidade. A correspondência deste modelo com a ISO/IEC 12207 encontra-se na coluna 7 da tabela 6.2.

#### **(TAPISTRY 1999)**

O Tapisstry é um modelo de melhoria do processo de software desenvolvido especificamente para pequenas e médias empresas de desenvolvimento de software. Utiliza o Bootcheck, que é uma metodologia de avaliação do processo, baseada no BOOTSTRAP<sup>43</sup> [Kuvaja 1999] e [Messnarz 1997] mas calibrada<sup>44</sup> para o enquadramento específico das PME de desenvolvimento de software. A selecção dos processos foi realizada em duas etapas: (1) recomendação de um painel de especialistas

---

<sup>43</sup> ver secção 3.3.1.4.3.

<sup>44</sup> o número de processos foi reduzido de 37 para 19.

do ESI<sup>45</sup> e do instituto Bootstrap e (2) validação e finalização baseada no *feedback* de 61 organizações<sup>46</sup> que testaram o modelo.

O Tapesty utiliza os seis níveis de capacidade propostos pela norma ISO 15504<sup>47</sup> e propõe uma estratégia de melhoria constituída por cinco passos [Kuvaja et al. 1999]:

- Passo 1 – Definição das necessidades de negócio da organização
- Passo 2 – Utilização do Bootcheck para avaliação do processo (auto-avaliação)
- Passo 3 – Definição de objectivos e prioridades de melhoria
- Passo 4 – Definição das acções de melhoria do processo
- Passo 5 – Definição do orçamento e calendarização da melhoria do processo

Este modelo de avaliação e melhoria é aplicado através de um *Workshop* com duração de dois dias e apoio de especialistas na área de melhoria do processo do software. Os processos avaliados pelo Tapesty são baseados na norma ISO/IEC 15504 e, assim, têm também correspondência directa com a norma 12207. Os processos propostos pelo Tapesty são apresentados na tabela 6.2 (coluna 8).

O Tapesty foi aplicado em diversos países europeus<sup>48</sup> com um *feedback* positivo dos participantes, resultando apenas como aspectos menos positivos a duração do *Workshop*, referindo os participantes que um *workshop* mais longo resultaria numa auto-avaliação mais aprofundada. Foi também referida a necessidade dos colaboradores envolvidos terem um conhecimento profundo dos processos para as conclusões serem realistas.

### **(Empiricamente derivado – 1993)**

O modelo empiricamente derivado foi apresentado por Drehmer e Dekleva [1993] e pretende responder à pergunta “será que o caminho de melhoria sugerido pelo CMM representa a progressão evolutiva natural?” [El Emam e Goldenson 1999]. Os

---

<sup>45</sup> European Software Institute.

<sup>46</sup> Europa e Índia.

<sup>47</sup> ver secção 3.3.1.2.

<sup>48</sup> Finlândia, Inglaterra, Bélgica, Itália e Alemanha.

resultados não comprovam<sup>49</sup> o caminho do CMM e os autores sugerem um caminho alternativo empiricamente derivado. Os cinco primeiros níveis são:

- Nível 1 – Revisões e controlo das mudanças
- Nível 2 – Processo standard e gestão do projecto
- Nível 3 – Revisão da gestão e controlo de configurações
- Nível 4 – Melhoria do processo de software
- Nível 5 – Gestão da cobertura dos testes e revisões

Deve referir-se que este caminho foi empiricamente derivado de casos de sucesso e insucesso, pelo que não sabemos se realmente conduz a melhorias ou provoca o insucesso. A correspondência deste modelo com a ISO 12207 encontra-se na tabela 6.2 (coluna 9).

Na tabela 6.2 apresenta-se o resultado do estudo comparativo. Na primeira coluna encontram-se os processos propostos para o MRP do 2MPspe. As quatro colunas seguintes apresentam os modelos com um âmbito genérico. Nas colunas 6, 7 e 8 encontram-se os modelos específicos para PES e finalmente, na coluna 9, apresenta-se o modelo empiricamente derivado.

Da análise da tabela 6.2 verifica-se uma correspondência quase completa com os três métodos específicos das PES. As excepções referem-se ao processo de Resolução de Problemas proposto no Rapid e no Tapisstry e os processos de Usabilidade, Avaliação do Produto e Medição propostos no QMIM. Em relação ao processo Resolução de Problemas, considera-se que no contexto das PES, embora importante não é prioritário. Verifica-se que devido ao pequeno número de funcionários, este processo tem um cariz informal, mas normalmente funciona de modo adequado. Em relação à proposta do QMIM, o MRP<sub>2MPspe</sub> tem as actividades de medição distribuídas pelos processos seleccionados, não sendo por isso um processo estratégico para o âmbito das PES.

---

<sup>49</sup> os autores afirmam mesmo que o caminho parece “arbitrário”.

Cat. 12207	COD 12207	Designação 12207	1		2		3		4		5		6	7	8	9			
			MRP 2MP	15504		Motorola		CMM		Trillium		QMIM					Ra pid	Tapis try	Em p.
				N2	N3	N2	N3	N2	N3	N2	N3								
Primários	5.1	Aquisição				f2.2		gs		X									
	5.2	Fornecimento						gp		X									
	5.3.1	Gestão Req.	●					gr		X		X	X						
	5.3	Desenvol.	●						ep	X		X	X	X	X	X			
	5.4.2	Sup. Cliente	●						ep						X				
	5.5	Manutenção	●												X				
Suporte	6.1	Document.	●	X				ep	X			X		X	X	X			
	6.2	Gestão Conf.	●	X		f2.4		gc		X		X	X	X	X	X			
	6.3	G. Qualidade	●	X		f2.3		gq		X		X	X	X	X	X			
	6.4	Verificação		X					ep		X								
	6.5	Validação							ep		X								
	6.6	Revisão Conj		X				gp		X						X			
	6.7	Auditoria			X			gq		X									
	6.8	R. Problemas		X				gc		X			X	X					
	6.9	Usabilidade								X		X							
	6.10	Av. Produto							ep			X							
Organizacionais	7.1.1	Alinh. Org.									X								
	7.1.2	Gestão Org.			X			gp	gi		X								
	7.1.3	G. Projectos	●	X	X	f2.1		gp		X		X	X	X	X	X			
	7.1.4	G. Qualidade	●		X						X	X			X	X			
	7.1.5	Gestão Risco	●	X				gp		X			X	X					
	7.1.6	Medição			X				cc	X		X							
	7.2	Infraestrutura			X				dp	X	X								
	7.3.1	Est. Processo	●		X		f3.1		dp		X	X	X	X	X	X			
	7.3.2	Avaliação							fp		X					X			
	7.3.3	Melhoria					f3.2		fp		X					X			
	7.4.1	GRH	●		X				pf		X	X			X				
	7.4.2	Formação	●		X		f3.3		pf		X								
	7.4.3	Gestão Conh.	●		X														
	7.5	Gestão Assets			X				dp										
	7.6	Gestão Reut.									X								
7.7	Eng. Domínio			X															
Outros (Não 12207)	CMM	C. Intergrupo							ci										
	CMM	R. por pares				f3.5		rp											
	Motorola	Cultura SPI				f3.2													
	Motorola	Cooperação				f3.4													
	Trillium	Gestão Tecn.									X								
	Trillium	Eng. Negócio									X								
	Trillium	G. Confiança									X								
	Trillium	Custeio C.V.								X									

**Legenda:**  
 cc – Características comuns  
 ci – Coordenação Intergrupos  
 dp – Definição do Processo Organizacional  
 ep – Engenharia do Produto de Software  
 f – Factor Crítico de Sucesso  
 fp – Foco no Processo Organizacional  
 gc – Gestão de Configurações  
 gi – Gestão integrada do Software  
 gp – Gestão de Projectos inclui Planeamento do Projecto de Software e Acompanhamento do Projecto de Software  
 gq – Garantia da Qualidade do Software  
 gr – Gestão de Requisitos  
 gs – Gestão de Subcontratações (não obrigatório)  
 pf – Programa de Formação  
 rp – Revisão por pares

Tabela 6.2 – Tabela comparativa entre os MRPs seleccionados.

Em relação aos métodos com um contexto de aplicação mais alargado<sup>50</sup>, verifica-se – como seria de esperar – que o 2MPspe engloba um conjunto menor de processos. Este facto deve-se a duas razões principais:

1. O contexto específico das PES. Alguns processos utilizados pelos métodos abrangentes são menos prioritários<sup>51</sup> no ambiente das pequenas empresas de software, por exemplo, Gestão de Domínio, Auditoria e Revisão Conjunta.
2. Algumas práticas dos processos excluídos são cumpridas indirectamente nos processos seleccionados, por exemplo, a Medição encontra-se dispersa por outros processos, a Infra-estrutura do processo está parcialmente no Estabelecimento do processo e a Verificação e Validação encontram-se no processo de Garantia da Qualidade do Software.

#### **6.2.2.2. Restrições do MRP Proposto**

A restrição principal do MRP<sub>2MPspe</sub> refere-se à sua aplicação a PES. Para cumprir este objectivo restringiu-se de modo significativo<sup>52</sup> o número de processos avaliados. O modelo proposto encontra-se dividido em quatro áreas de processos, nomeadamente: Engenharia, Suporte, Gestão e Organização, englobando no total 18 processos primários.

Analisando os processos propostos pela norma ISO/IEC 12207 verifica-se que, na categoria dos processos Primários, foram excluídos o processo de aquisição e fornecimento e a selecção dos processos de desenvolvimento centrou-se na área da realização e acompanhamento do produto específico, no caso, o software. Em relação aos processos Suporte, foram seleccionados os processos de documentação, gestão de configurações e garantia da qualidade. Os processos de Suporte, verificação, validação, revisão conjunta e resolução de problemas, também relacionados com a qualidade, são tratados como possíveis estratégias a adoptar na garantia da qualidade. Finalmente, a usabilidade e avaliação do produto foram considerados menos prioritários nas PES. No

---

<sup>50</sup> colunas 2 a 5 na tabela 6.2.

<sup>51</sup> embora algumas práticas possam estar alinhadas com os objectivos de curto prazo das PES.

<sup>52</sup> 45 para 18 processos primários.

que se refere à categoria de processos Organizacionais, na sub-área de gestão seleccionam-se os processos de gestão de projectos, gestão do risco e gestão da qualidade. Consideraram-se não prioritários os processos de alinhamento organizacional, gestão da organização e medição. Finalmente, ainda na categoria de processos Organizacionais, incluíram-se os processos de recursos humanos e estabelecimento do processo. Esta escolha deve-se à necessidade de definir formalmente os processos<sup>53</sup> e tratar os problemas relacionados com recursos humanos, sempre importantes nas pequenas empresas de software. Refira-se a inclusão do processo de suporte ao cliente com o objectivo específico de melhorar e consolidar o relacionamento com os clientes.

O MRP<sub>2MPspe</sub> apresenta também a restrição de pretender atingir apenas o nível 3 do ISO/IEC 15504, isto é, excluíram-se o nível 4, denominado previsível, caracterizado pela gestão quantitativa e o nível 5, designado optimizado. Os processos relacionados com os níveis 4 e 5 são: alinhamento organizacional, medição, avaliação do processo, melhoria do processo, infra-estrutura, gestão de *assets* e gestão do domínio. Estes processos não foram incluídos no 2MPspe, verificando-se a mesma opção no Rapid e Tapistry.

Na próxima secção apresenta-se uma descrição detalhada de cada um dos processos seleccionados.

### 6.2.3. Descrição dos Processos do MRP<sub>2MPspe</sub>

A parte central da definição de um modelo de referência do processo consiste em descrever os processos dentro do âmbito do modelo. A parte 2 da norma ISO 15504 [ISO/IEC 2004b], define os seguintes requisitos para as descrições dos processos:

- a) O processo deve ser descrito em termos do seu propósito e resultados. O propósito deverá consistir numa declaração definindo os objectivos de alto nível a atingir e o conjunto dos resultados deverá mostrar que o propósito foi atingido com sucesso

---

<sup>53</sup> nas suas várias dimensões.

- b) O conjunto dos resultados do processo deverá ser o necessário e suficiente para atingir o propósito definido. Considera-se um resultado do processo, a produção de um *output*, uma alteração significativa no estado do processo ou o cumprimento de uma restrição<sup>54</sup> específica do processo
- c) A descrição dos processos não deve incluir nenhum aspecto do referencial de classificação, excepto para o nível 1

Nesta secção faz-se a descrição dos processos propostos no MRP<sub>2MPspe</sub>. Esta descrição englobará um enquadramento, o propósito e finalmente os resultados. A base desta descrição é a norma ISO/IEC 12207 [ISO/IEC 2002b].

### **6.2.3.1. Categoria de Processos ENG - Engenharia**

A categoria dos processos de engenharia engloba os processos directamente relacionados com a realização do produto. Podem também ser denominados como processos técnicos sendo normalmente a área onde as PES investem fundos, quando pretendem iniciar um programa de melhoria<sup>55</sup>. [Blanco et al. 2001].

#### **6.2.3.1.1. Processo ENG.1 – Gestão dos Requisitos**

Um requisito é uma propriedade que tem que ser exibida pelo sistema desenvolvido, para resolver um determinado problema [Thayer e Dorfman 1997] e [Pfleeger 1998]. Os requisitos podem ser classificados em requisitos do produto ou do processo [Sommerville e Sawyer 1997]. Os requisitos do produto referem-se às propriedades do software e são normalmente divididos em requisitos funcionais, que caracterizam as capacidades do produto e requisitos não funcionais também denominados como restrições ou requisitos de qualidade. Os requisitos do processo referem-se a restrições sobre o processo de desenvolvimento. Os requisitos devem ser

---

<sup>54</sup>por exemplo requisitos ou objectivos,

<sup>55</sup>quase sempre condenado ao abandono ou insucesso.

claros e não ambíguos e sempre que possível expressos de uma forma quantitativa, para facilitar a sua validação durante o processo de validação do produto.

A Gestão dos requisitos é a primeira fase na compreensão do problema a ser resolvido. É fundamentalmente uma actividade humana de descoberta, envolvendo a identificação dos “interessados”<sup>56</sup> e o estabelecimento de canais de comunicação<sup>57</sup> entre o cliente e a equipa de desenvolvimento.

O propósito do processo de gestão dos requisitos é recolher, processar e acompanhar as necessidades e requisitos do cliente durante o ciclo de vida do produto.

Resultados<sup>58</sup>:

- Definição da versão de base a partir dos requisitos aprovados
- Mecanismo de tratamento e gestão das alterações aos requisitos
- Mecanismo para acompanhamento das necessidades do cliente
- Mecanismo para permitir ao cliente saber o estado dos pedidos de alterações aos requisitos
- Estabelecimento de canais de comunicação com o cliente
- Identificação de melhorias resultantes dos pedidos do cliente e mudanças tecnológicas e avaliação do impacto

#### **6.2.3.1.2. Processo ENG.2 – Desenvolvimento**

O propósito do processo de desenvolvimento do software é transformar um conjunto de requisitos no software final. A preocupação está centrada no ambiente de desenvolvimento de software das PES e neste contexto incluem-se os seguintes sub-processos: Análise dos Requisitos do Software, Concepção do Software, Construção do Software, Integração do Software e Teste do Software.

---

<sup>56</sup> *stakeholders*.

<sup>57</sup> preferencialmente baseados numa sólida relação de confiança.

<sup>58</sup> resultados da implementação com sucesso do processo.



### **Processo ENG.21 – Análise dos Requisitos do Software**

A perspectiva tradicional da análise de requisitos do software centra-se na modelação conceptual utilizando um método de análise. Embora o desenvolvimento de modelos seja uma actividade fundamental na análise de requisitos<sup>59</sup> é importante também a classificação dos requisitos [SWEBOK 2001]. Pressman [1997] afirma que a análise de requisitos permite a especificação das funcionalidades e desempenho do software, indicando as suas fronteiras e estabelecendo as restrições que o software deve cumprir.

O propósito do processo de análise de requisitos do software é estabelecer os requisitos do software.

Resultados:

- São definidos os requisitos atribuídos ao software e os seus interfaces
- Os requisitos são analisados em relação à correcção e facilidade de teste
- São definidas prioridades na implementação dos requisitos
- Os requisitos do software são aprovados e actualizados sempre que necessário
- São avaliadas as alterações aos requisitos do software
- É desenvolvida uma versão base dos requisitos do software

### **Processo ENG.22 – Concepção do Software**

Segundo a definição do IEEE [1991], a concepção é “o processo de definir a arquitectura, componentes, interfaces e outras características do software ou do componente”, é também “o resultado deste processo”. O resultado da concepção tem que descrever o modo como o sistema é decomposto e organizado em componentes e as ligações entre as componentes. A tarefa corresponde à procura de soluções para o problema, assegurando que é possível obter uma solução informática completa e

---

<sup>59</sup> na perspectiva de compreender melhor o problema e não de iniciar a concepção da solução.

exequível. No ciclo de vida clássico, a concepção é dividida em concepção de alto nível<sup>60</sup> e concepção detalhada do software.

O propósito do processo de concepção do software é definir uma solução para o software que acomode os requisitos e possa ser testada em relação a eles.

Resultados:

- É concebida a arquitectura do software
- São definidos os interfaces internos e externos para cada componente do software
- É desenvolvida concepção detalhada
- É estabelecida a ligação e consistência com os requisitos do software

### **Processo ENG.23 – Construção do Software**

O SWEBOK [2001] define a construção do software como “um acto fundamental da engenharia de software” que consiste “na construção de software que funcione e com significado, através da combinação da codificação, validação e teste (teste das unidades) pelo programador”. Consiste então na criação e verificação dos componentes codificados, a partir dos quais o produto final será obtido. Nesta actividade podem ser utilizadas diversas ferramentas, nomeadamente: compiladoras, *debuggers*, geradores de código, ferramentas de documentação e outras.

O propósito do processo de construção do software é produzir unidades de software executáveis que implementem correctamente a concepção do software.

Resultados:

- As unidades de software definidas na concepção são produzidas
- São definidos critérios de verificação para todas as unidades de software
- É estabelecida a ligação e consistência com a concepção do software
- É realizada a verificação das unidades de software

---

<sup>60</sup> ou da arquitectura.

**Processo ENG.24 – Integração do Software**

O nível de integração necessário e como consequência a importância do processo depende do tamanho e complexidade do software. No caso de um sistema informático muito simples, constituído apenas por uma unidade, não é necessário este processo. Uma das actividades importantes da integração é o teste de cada item desenvolvido na perspectiva de descobrir os erros o mais cedo possível.

O propósito do processo de integração do software é combinar as unidades de software produzindo itens integrados consistentes com a especificação do software<sup>61</sup>.

Resultados:

- São produzidos os itens integrados
- São verificados os itens de software
- São registados os resultados dos testes de integração
- É estabelecida a ligação e consistência com as unidades de software

**Processo ENG.25 – Teste do Software**

Os testes das unidades e os testes de integração concentram-se em verificar a correcção dos elementos e das suas interacções, conforme são construídos. O teste do software preocupa-se em detectar as falhas no produto e em corrigir os erros que provocaram essas falhas. Este processo preocupa-se em verificar o comportamento do software integrado, esperando-se que a maioria das falhas tenha sido detectada durante os testes anteriores.

O propósito do processo de testes do software é confirmar que o produto de software integrado cumpre os requisitos definidos.

---

<sup>61</sup> documento resultante do processo de concepção de software.

Resultados:

- São desenvolvidos critérios para demonstrar que os requisitos do software foram cumpridos
- O software integrado é verificado usando os critérios definidos
- São registados os resultados dos testes
- É desenvolvida e aplicada uma estratégia de testes de regressão

#### **6.2.3.1.3. Processo ENG.3 – Manutenção do Software**

A norma do IEEE para a manutenção do software [IEEE 1998] define a manutenção do software como a modificação do produto software, depois da entrega, para corrigir falhas, melhorar o desempenho ou outros atributos, ou para adaptar o produto a um ambiente modificado. Embora a percepção comum sobre manutenção indique que é uma actividade principalmente centrada na correcção de erros, estudos recentes apontam que a maioria<sup>62</sup> dos esforços de manutenção são acções não correctivas [Pressman 1997] e [Pigoski 1997]. Nas PES verifica-se uma forte preocupação com esta área, principalmente devido ao elevado número de recursos humanos envolvidos directamente com a manutenção. A norma ISO/IEC 14764 [ISO/IEC 2000a] define quatro categorias de manutenção:

- Correctiva. Modificação reactiva executada depois da entrega, com o objectivo de corrigir problemas descobertos.
- Adaptativa. Modificação realizada depois da entrega para manter o produto utilizável, quando o ambiente se altera.
- Perfectiva. Modificação realizada para melhorar o desempenho.
- Preventiva. Modificação realizada para detectar e corrigir problemas latentes, antes de se tornarem falhas efectivas.

O propósito do processo de manutenção do software é modificar o produto depois da entrega para corrigir falhas, melhorar o desempenho ou adaptá-lo a um ambiente modificado.

---

<sup>62</sup> mais de 80%.

Resultados:

- É desenvolvida a estratégia de manutenção
- É identificado o impacto das modificações
- É actualizada a documentação do software
- São desenvolvidas as modificações e testadas
- É efectuada a migração dos *upgrades*
- Os produtos são retirados de utilização
- As modificações do software são comunicadas a todos os intervenientes

### **6.2.3.2. Categoria de Processos SUP – Suporte.**

A categoria de processos de suporte engloba processos que podem ser utilizados por qualquer outro processo em vários pontos do ciclo de vida do software. Uma organização de software pretende não só desenvolver um bom produto, na perspectiva técnica, mas também um produto que satisfaça os requisitos explícitos do cliente. Os processos de suporte são a fundação para o fornecimento de um produto viável.

#### **6.2.3.2.1. Processo SUP.1 – Documentação**

O processo de documentação é concebido para registar toda a informação produzida por um processo ou actividade do ciclo de vida. Este processo inclui um conjunto de actividades para planeamento, concepção, produção, edição, distribuição e manutenção dos documentos. Documentação refere-se ao processo de criação do documento e ao próprio documento em si. Para os documentos serem úteis e cumprirem o seu propósito devem assegurar que os utilizadores podem [BS 1986]:

- Obter uma visão global do produto
- Encontrar a informação correcta
- Compreendê-lo
- Aplicá-lo, conseguindo executar as tarefas correctamente

As classes de documentos mais importantes são: a documentação do utilizador, documentação do desenvolvimento e documentação técnica [SPIRE 1998]. A melhoria desta área nas PES necessita de um acompanhamento rigoroso.

O propósito do processo de documentação é desenvolver e manter a informação armazenada sobre o software, produzida por um processo.

Resultados:

- Está desenvolvida uma estratégia que identifica a documentação a ser produzida
- Estão identificadas as normas aplicáveis ao desenvolvimento da documentação
- Está identificada a documentação a ser produzida pelos processos
- O propósito e conteúdo de toda a documentação é especificado, revisto e aprovado
- A documentação é desenvolvida e disponibilizada de acordo com as normas identificadas
- A documentação é mantida de acordo com os critérios definidos

#### **6.2.3.2.2. Processo SUP.2 – Gestão de Configurações**

O CMM [Paulk et al. 1995] refere que a gestão de configurações do software “envolve a identificação da configuração do software (produtos de trabalho seleccionados e as suas descrições) em determinados pontos no tempo, controlando de forma sistemática as alterações à configuração e mantendo a integridade e consistência da configuração através do ciclo de vida do software”. Define os produtos de trabalho a colocar sob a gestão de configurações como os produtos entregues ao cliente (por exemplo, a especificação dos requisitos do software e o código) e elementos necessários para criar esses produtos do software (por exemplo, o computador).

Na perspectiva da gestão, a gestão de configurações (GC) controla a evolução e integridade de um produto através da identificação dos seus elementos, gerindo e controlando as alterações e verificando, registando e relatando informação sobre a configuração. Na perspectiva dos profissionais de desenvolvimento a GC facilita o

desenvolvimento e a implementação de modificações [SWEBOK 2001]. Nas PES, a GC é normalmente executada, mas carece de uma definição e estabelecimento.

O propósito do processo de gestão de configurações é estabelecer e manter a integridade de todos os produtos de trabalho de um processo ou projecto e disponibilizá-los a todos os intervenientes.

Resultados:

- É desenvolvida uma estratégia para a gestão de configurações
- Todos os itens gerados pelo processo ou projectos são identificados, definidos e é criada uma versão base
- São controladas as modificações e versões dos itens
- São disponibilizadas as modificações e versões
- O estado dos itens e pedidos de modificações são registados e relatados
- É assegurado que os itens estão completos e consistentes
- É controlado o armazenamento, manuseamento e entrega dos itens

#### **6.2.3.2.3. Processo SUP.3 – Garantia da Qualidade**

Para uma organização entregar produtos com sucesso é necessário, em primeiro lugar, que os produtos de trabalho estejam conforme os requisitos. É fundamental também, que além de cumprirem os requisitos, eles possuam uma determinada qualidade [SPIRE 1998]. A garantia da qualidade fornece precisamente a visibilidade adequada à gestão, sobre o processo utilizado e sobre os produtos construídos. O papel da garantia da qualidade é então “assegurar que os processos planeados são apropriados e são depois implementados de acordo com o plano e que as medições apropriadas sobre os processos são fornecidas aos níveis apropriados da organização” [SWEBOK 2001]. Refira-se também que a garantia da qualidade é apenas uma parte do sistema de qualidade requerido pela norma ISO 9001: 2000, mas é uma parte muito importante. Embora normalmente a garantia da qualidade seja implementada com uma equipa independente do desenvolvimento, nas PES, estas actividades são executadas a tempo parcial pelos profissionais de desenvolvimento ou nos melhores cenários, por um

profissional em exclusividade. A introdução / melhoria desta área nas PES necessita de uma gestão de expectativas cuidadosa.

O propósito do processo de garantia da qualidade é fornecer garantia que os produtos de trabalho e processos cumprem os requisitos especificados e implícitos e aderem aos planos estabelecidos.

Resultados:

- É desenvolvida uma estratégia para conduzir a garantia da qualidade
- É produzida e mantida evidência sobre a garantia da qualidade
- São identificados e registados problemas e/ou não conformidades com os requisitos aprovados
- É verificada a aderência dos produtos, processos e actividades às normas, procedimentos e requisitos aplicáveis

### **6.2.3.3. Categoria de Processos GES - Gestão**

A categoria de processos de gestão engloba os processos que contêm práticas que podem ser utilizadas por profissionais que façam a gestão de qualquer tipo de projecto ou processo no ciclo de vida do software [ISO/IEC 2003]. Os processos desta categoria preocupam-se com o sucesso dos projectos, na perspectiva de produzir um produto com as funcionalidades e qualidade pretendidas, nos prazos previstos, dentro do orçamento concordado e evitando riscos desnecessários [SPIRE 1998].

#### **6.2.3.3.1. Processo GES.1 – Gestão do Projecto**

Segundo o PMI [2000], a gestão de projecto “é a aplicação de conhecimento, aptidões, ferramentas e técnicas às actividades do projecto, de modo a cumprir os requisitos. A gestão de projectos é executada através do uso de processos tais como:



início, planeamento, execução, controlo e fecho. As actividades da gestão de projectos envolvem:

- Necessidades concorrentes: âmbito, tempo, custo, risco e qualidade
- Partes interessadas com diferentes expectativas e necessidades
- Requisitos identificados”

É uma área em que as PES apresentam fortes preocupações, embora as suas práticas sejam normalmente insatisfatórias.

O propósito do processo da gestão do projecto é identificar, estabelecer, coordenar e monitorizar as actividades, tarefas e recursos necessários para um projecto produzir o produto e/ou serviço, no contexto das restrições e requisitos do projecto.

Resultados:

- É definido o âmbito do projecto
- É avaliada a exequibilidade do projecto
- São desenvolvidas estimativas sobre as tarefas e os recursos necessários
- São definidas as dependências entre os elementos do projecto
- São desenvolvidos e implementados os planos para a execução do projecto
- O progresso do projecto é monitorizado e relatado
- São executadas acções para corrigir os desvios verificados no plano

#### **6.2.3.3.2. Processo GES.2 – Gestão do Risco**

Charette [1989] apresenta a seguinte definição conceptual de risco: “Primeiro, o risco preocupa-se com acontecimentos futuros... Segundo, o risco envolve mudança... e terceiro, o risco envolve escolha”. Em relação ao software, o risco inerente envolve duas características [Higuera 1995], incerteza e perda. Quando é desenvolvida a análise dos riscos é importante quantificar o nível de incerteza e o grau de perda associado a cada risco. Assim, devem ser consideradas várias categorias de riscos [Pressman 1997]:

- Riscos do projecto, que ameaçam o plano do projecto

- Riscos técnicos, que ameaçam a qualidade do produto
- Riscos do negócio, ameaçam a viabilidade do software a ser construído

A gestão de riscos é normalmente uma área fora das preocupações das PES. Verifica-se que pode ser uma área efectiva de melhorias uma vez que não exige um grande investimento<sup>63</sup> e resulta em benefícios visíveis. Apresenta a dificuldade de necessitar de *expertise* na área. A gestão do risco cobre a avaliação do risco (identificação, análise e definição de prioridades) e o controlo do risco (planeamento, resolução e monitorização).

O propósito do processo de gestão de risco é identificar, gerir e mitigar de modo contínuo os riscos.

Resultados:

- É determinado o âmbito da gestão de risco
- São definidas e implementadas estratégias apropriadas de gestão do risco
- São identificados os riscos do projecto
- Os riscos são analisados
- São seleccionadas técnicas de monitorização dos riscos
- São executadas as acções necessárias para evitar o impacto dos riscos

#### **6.2.3.3.3. Processo GES.3 – Gestão da Qualidade**

O sistema de gestão da qualidade é o tema central na norma ISO 9001:2000<sup>64</sup>, como foi apresentado um dos objectivos do 2MPspe é possuir uma correspondência para esta norma. O processo GES.3 é o processo central para cumprir a certificação ISO 9001:2000. A gestão da qualidade envolve o estabelecimento de um foco na monitorização da qualidade do produto e do processo tanto no nível do projecto como no nível organizacional. A implementação deste processo numa PES obriga a um

---

<sup>63</sup> tanto em fundos como em recursos humanos.

<sup>64</sup> ver secção 3.3.1.3.

acompanhamento cuidadoso das acções pretendidas, uma vez que os benefícios directos são difíceis de medir e os resultados serão mais visíveis a médio/longo prazo.

O propósito do processo de gestão da qualidade é monitorizar a qualidade dos produtos e serviços, no nível da organização e do projecto de modo a assegurar a satisfação do cliente.

Resultados:

- São estabelecidos objectivos de qualidade com base nos requisitos implícitos e explícitos do cliente
- É desenvolvida a estratégia global para atingir os objectivos definidos
- É estabelecido um sistema de gestão da qualidade para implementar a estratégia
- São executadas actividades de garantia e controlo da qualidade e o seu desempenho é confirmado
- É monitorizado o desempenho actual em relação aos objectivos de qualidade
- São realizadas acções apropriadas quando os objectos de qualidade não são atingidos

#### **6.2.3.4. Processos ORG - Organizacionais**

A categoria de processos organizacionais preocupa-se com o ambiente organizacional onde é desenvolvido o software. Estes processos ajudam no estabelecimento, implementação e melhoria para tornar a organização mais efectiva. Consideram então os objectivos do negócio, a organização global, a gestão das pessoas e a infra-estrutura necessária independente do projecto [SPIRE 1998].

##### **6.2.3.4.1. Processo ORG.1 – Recursos Humanos**

Embora o desenvolvimento de software seja cada vez mais suportado por tecnologias melhoradas, os recursos humanos continuam a ser o factor mais importante

neste processo. As capacidades dos profissionais variam muito mas a formação, motivação e ambiente podem dar um grande contributo na melhoria global do desempenho. O processo de recursos humanos tem como propósito, fornecer a organização com os recursos humanos adequados e manter as suas competências consistentes com as necessidades do negócio.

As PES, para sobreviverem no mercado, têm obrigatoriamente que apostar nos seus Recursos Humanos, nesta perspectiva a gestão de recursos humanos (ORG.11), a formação (ORG.12) e a gestão do conhecimento (ORG.13) são processos prioritários.

### **Processo ORG.11 – Gestão de Recursos Humanos**

Os profissionais de software devem possuir *Know-how* tecnológico, conhecimento sobre métodos e competência social. A competência social está a tornar-se cada vez mais importante devido às necessidades de trabalho em equipa características do desenvolvimento de software actual [Katzenbach e Smith 1993]. O SPIRE [1998] baseado na experiência industrial, apresenta um possível perfil típico de um Engenheiro de Software:

- Conhecimento técnico especializado (25%)
- Experiência, conhecimento do domínio, capacidade para trabalhar em equipa (20%)
- Capacidade de cooperação e comunicação, qualidades humanas (18%)
- Independência, flexibilidade, capacidade de planeamento, utilização de ferramentas (14%)
- Pensamento abstracto, criatividade, bons resultados no trabalho (12%)
- Orientação ao utilizador (5%)
- Motivação (5%)

O propósito do processo de Gestão de Recursos Humanos é fornecer a organização e os projectos com indivíduos que possuam aptidões e conhecimento para executar os seus papéis efectivamente e para trabalharem em conjunto como uma equipa coesa.

**Resultados:**

- São identificados e recrutados indivíduos com aptidões e competências necessárias
- É suportada uma interacção efectiva entre indivíduos e equipas
- A força de trabalho tem aptidões para partilhar informação e coordenar as suas actividades de modo eficiente
- São definidos critérios objectivos para avaliar o desempenho das equipas e dos indivíduos, fornecendo *feedback* sobre o desempenho e assim possibilitando melhorias.

**Processo ORG.12 – Formação**

O processo de formação envolve em primeiro lugar a identificação das necessidades de formação na organização, nos projectos e nos indivíduos e em seguida desenvolver ou contratar formação para atingir as necessidades identificadas [Paulk et al. 1995]. A formação tem que ser planeada e implementada com antecedência no ciclo de desenvolvimento. Devido às características peculiares do desenvolvimento do software<sup>65</sup>, as PES apresentam preocupações reais nesta área.

O propósito do processo de formação é desenvolver as aptidões e conhecimento dos indivíduos de modo a que possam executar os seus papéis efectivamente.

**Resultados:**

- É desenvolvida ou contratada a formação adequada para cobrir as necessidades identificadas
- A formação é conduzida para assegurar que todos os indivíduos têm as aptidões necessárias para cumprir os seus papéis

---

<sup>65</sup> em que o conhecimento e aptidões do pessoal tem um papel fundamental.

### **Processo ORG.13 – Gestão do Conhecimento**

A área de gestão do conhecimento encontra-se descrita em detalhe na secção 6.4.2.1. Sucintamente, pode definir-se a gestão do conhecimento como englobando actividades relacionadas com a criação, aquisição, captura, partilha e utilização de conhecimento [Quintas et al. 1996].

O propósito do processo de gestão do conhecimento é assegurar que o conhecimento, informação e aptidões dos profissionais são coligidas, partilhadas, reutilizadas e melhoradas através da organização.

Resultados:

- É estabelecida e mantida uma infra-estrutura para partilhar informação
- O conhecimento é partilhado e disponibilizado através da organização
- Está seleccionada uma estratégia de gestão de conhecimento apropriada

#### **6.2.3.4.2. Processo ORG.2 – Estabelecimento do Processo**

O processo engloba a definição do modelo do processo que incluirá as actividades necessárias para o desenvolvimento de software. Há vários modelos do processo<sup>66</sup>, com diferentes níveis de detalhe, mas o primeiro passo é analisar a situação actual. Verificar como os processos são actualmente executados e decidir como devem ser executados. A definição do processo deve incluir orientações sobre como “colocar à medida”.

O propósito do processo de estabelecimento do processo é estabelecer um conjunto de processos organizacionais para todos os processos do ciclo de vida do software de forma a serem aplicados às suas actividades do negócio.

---

<sup>66</sup> ver secção 2.2.1.

**Resultados:**

- Um conjunto de processos normalizado é estabelecido.
- São identificadas as tarefas detalhadas, actividades e produtos de trabalho associados com o processo normalizado, junto com as características de desempenho esperadas
- É desenvolvida uma estratégia para “colocar à medida” o processo normalizado, de acordo com as necessidades do projecto
- É recolhida e mantida informação relativa ao uso do processo normalizado em projectos específicos

**6.2.3.4.3. Processo ORG.3 – Suporte ao Cliente**

A organização após a instalação do software precisa de fornecer um nível de serviço e suporte adequado para os produtos de software. É necessário assegurar a formação apropriada a todo o pessoal envolvido, verificar se o software é utilizado de modo adequado e assegurar que os utilizadores conhecem e utilizam todo o potencial do software instalado. Como foi referido, um bom relacionamento com os clientes é fundamental nas PES e nesta perspectiva, este processo está incluído no MRP<sub>2MPspe</sub>.

O propósito do processo de Suporte ao Cliente é estabelecer e manter um nível de serviço aceitável, realizando assistência e consultoria ao cliente para suportar a utilização efectiva do produto.

**Práticas de base:**

- São identificadas as necessidades de serviços para o suporte ao cliente
- É avaliada a satisfação do cliente em relação ao produto e aos serviços de suporte
- Os pedidos e questões do cliente são resolvidos através do suporte operacional
- As necessidades de suporte do cliente são atingidas através do fornecimento de serviços adequados

### 6.3. Dimensão da Capacidade (Referencial de Medição).

O referencial de medição é constituído pelas componentes: níveis de capacidade, atributos da capacidade e escala de classificação [ISO/IEC 2004b].

Um nível de capacidade representa uma aptidão de um processo para atingir o objectivo requerido, isto é, mede “quão bem” o processo é gerido, permitindo assim atingir a sua meta e contribuir para os objectivos globais da organização [SPIRE 1998]. Um atributo da capacidade é “uma característica mensurável da capacidade do processo, aplicável a qualquer processo” [ISO/IEC 2004a]. Os atributos da capacidade permitem avaliar se um determinado nível de capacidade foi atingido. Cada atributo permite medir se um determinado aspecto<sup>67</sup> da capacidade foi atingido<sup>68</sup>. Finalmente, a extensão com que o atributo é alcançado é medida numa escala ordinal de medição, denominada escala de classificação (ver Figura 6.13)

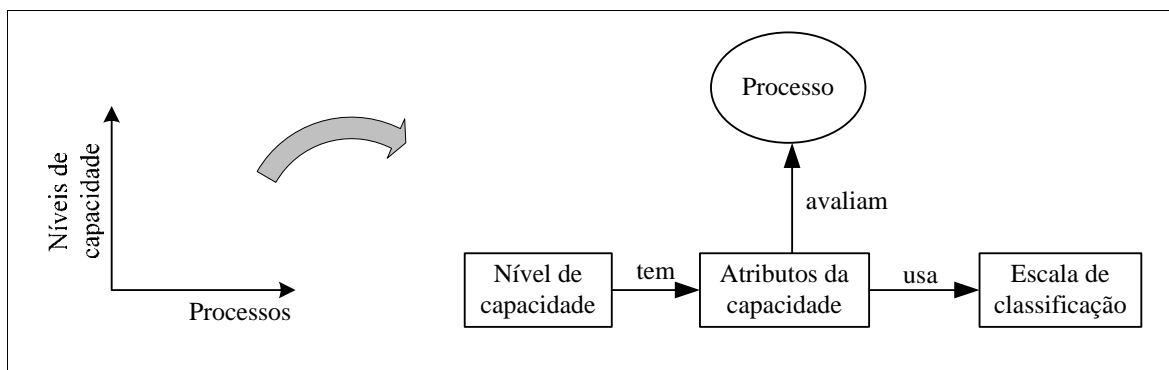


Figura 6.13 – Referencial de medição [ISO/IEC 2004b].

Nas secções seguintes apresentam-se cada uma das componentes do referencial de medição.

<sup>67</sup> por exemplo, gestão dos produtos de trabalho ou gestão dos recursos.

<sup>68</sup> para todos os processos do MRP seleccionados.



### 6.3.1. Níveis de Capacidade

A parte 2 da norma ISO/IEC 15504 propõe uma escala com 6 níveis para modelar os níveis de capacidade do processo. A escala começa no nível 0 (o mais baixo) em que o processo, ou não é executado ou é executado de forma incompleta e termina no nível 5 (o mais elevado), onde o processo é continuamente melhorado, através da sua monitorização. Esta escala define a rota de melhoria para os processos.

O referencial de medição do 2MPspe utiliza apenas os quatro primeiros níveis: 0 – Incompleto, 1 – Executado, 2 – Gerido e 3 – Definido. Esta simplificação<sup>69</sup>, deve-se à maturidade característica das PES, ou seja, normalmente a sua capacidade está no nível 0 ou 1. Verificando-se que as suas preocupações<sup>70</sup> prioritárias não englobam a compreensão quantitativa dos processos nem a apetência para a optimização contínua, características dos níveis 4 e 5.

Na dimensão da capacidade do 2MPspe utiliza-se ainda o sub-nível 1.5, designado como valorizado. Este sub-nível situa-se entre os níveis 1 e 2 e pretende caracterizar o objectivo de um determinado processo além de ser executado, ser também reconhecido (ou valorizado) pelos intervenientes, incluindo gestão.

Há dois grupos de razões que justificam a inclusão deste sub-nível, um relacionado com o ambiente operacional das PES e o outro com enfoque no estudo dos casos de insucesso dos projectos de MPS.

Verifica-se que nas pequenas empresas de desenvolvimento de software há normalmente um número bastante reduzido de recursos humanos, o que provoca acumulação de tarefas<sup>71</sup> nos colaboradores. Associado a este facto constata-se que naturalmente, cada colaborador acaba por dar maior prioridade a determinados processos, em detrimento de outros. Por outro lado, os gestores quase sempre dão maior importância<sup>72</sup> às questões e tarefas com impacto directo no produto.

---

<sup>69</sup> também utilizada no SPIRE [1998].

<sup>70</sup> mesmo de médio / longo prazo.

<sup>71</sup> ver casos práticos (secção 4.3.1) e descrição de PES típicos (secção 6.2.1).

<sup>72</sup> muitas vezes inconscientemente.

Diversos autores<sup>73</sup> [Dyba 2000], [Sassenburg 2001], [Evans 2002] e [Rainer e Hall 2003] apresentam como factores importantes para o sucesso de uma iniciativa MPS, o apoio da gestão e o envolvimento do pessoal. Estes dois aspectos, traduzidos para o enquadramento das PES, devem ser analisados no nível dos processos e não apenas no âmbito da MPS. Nesta perspectiva, o nível 1.5 pretende reforçar a importância dos processos em todos os intervenientes.

O 2MPspe utiliza então 4 níveis de capacidade (ver Figura 6.14) e um sub-nível entre os níveis 1 e 2.

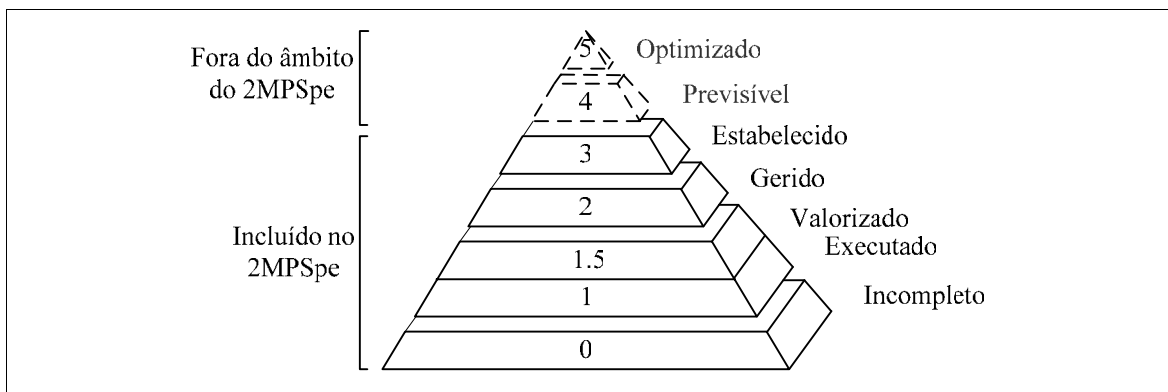


Figura 6.14 – Níveis de capacidade do 2MPspe (adaptado de SPIRE [1998]).

Apresenta-se em seguida uma caracterização de cada nível [El Emam 1998] e [ISO/IEC 2004b].

Um processo está no nível 0, quando não está implementado ou há uma falha geral em atingir os resultados esperados. Considera-se que um processo está parcialmente no nível 1, caso sejam observados alguns produtos de trabalho ou práticas relacionadas com o processo.

No nível 1, o processo atinge o seu objectivo global e este facto é testemunhado através da identificação dos produtos de trabalho.

Um processo estará no nível 1.5, quando além de ser executado é possível identificar “uma consciência” da importância do processo, ou seja, os intervenientes e a

<sup>73</sup> no capítulo 4 encontram-se mais detalhes sobre causas do insucesso.

própria gestão reconhecem o valor<sup>74</sup> do conjunto de actividades. Pretende-se avaliar até que ponto os “alicerces” humanos e organizacionais do processo estão devidamente estruturados.

No nível 2, o processo anteriormente descrito como valorizado, é agora executado de uma forma gerida e tendo como base objectivos bem definidos. Os produtos de trabalho têm uma qualidade aceitável e estão em conformidade com normas e requisitos especificados.

Finalmente, no nível 3, o processo é executado utilizando uma versão do processo normalizado. Verifica-se então que o processo tem a capacidade de atingir os resultados esperados, uma vez que a definição do processo utiliza os princípios<sup>75</sup> de engenharia e são atribuídos os recursos com as competências necessárias para atingir o desempenho esperado.

### **6.3.2. Atributos da Capacidade (ACs)**

A medição da capacidade dos processos é feita através dos atributos da capacidade. Cada atributo da capacidade mede uma determinada característica do processo. Os atributos da capacidade são aplicáveis a todos os processos do MRP e ao conjunto das classificações dos ACs para um processo é denominado perfil do processo.

Apresentam-se em seguida os atributos da capacidade utilizados para avaliar até que ponto o nível foi atingido. Nesta secção apresentam-se as definições propostas pela parte 2 da norma ISO/IEC 15504, desenvolvendo-se uma explicação<sup>76</sup> para cada AC.

#### **Nível 0: Processo Incompleto**

AC: Não tem

---

<sup>74</sup> e a necessidade de melhoria.

<sup>75</sup> princípios para a utilização prática do conhecimento científico.

<sup>76</sup> com enfoque no ambiente das PES.

**Nível 1: Processo Executado**

“AC1.1 Atributo da execução do processo

O atributo da execução do processo mede a extensão com que o processo atinge os seus resultados. Como resultado deste AC ser totalmente atingido:

- a) o processo atinge os resultados definidos

(ISO/IEC 15504-2, 5.2.1)

O AC1.1, ao contrário dos atributos da capacidade dos níveis seguintes, tem indicadores específicos para cada processo.

Estes indicadores globalmente são:

- Actividades que contribuem para o propósito do processo e transformam artefactos de entrada em artefactos de saída
- Produtos de trabalho resultantes

O AC1.1 é então avaliado pelos produtos de trabalho identificáveis e pelas práticas observáveis na organização.

**Nível 1.5: Processo valorizado**

AC1.51 Atributo de valorização do processo

O atributo de valorização do processo mede a extensão com que os intervenientes (directos e indirectos<sup>77</sup>) reconhecem o valor acrescentado para o negócio, resultante de uma efectiva realização do processo. Como resultado deste AC ser totalmente atingido:

- a) São identificados todos os intervenientes do processo
- b) É compreendida a importância das práticas e resultados do processo, no contexto do negócio

O AC1.51, tal como outros atributos da capacidade dos níveis seguintes requer a execução de outro processo organizacional ou de suporte. Para identificar os intervenientes e difundir a contribuição do processo é necessário que o processo

---

<sup>77</sup> gestores e outros profissionais não envolvidos directamente na execução do processo.

Estabelecimento do Processo (ORG.2) do MRP<sub>2</sub>MPspe, seja, pelo menos parcialmente executado.

## **Nível 2: Processo gerido**

### “AC2.1 Atributo de gestão da execução

O atributo de gestão da execução mede, até que ponto o processo é gerido para produzir produtos de trabalho que vão ao encontro dos objectivos definidos. Como resultado deste AC ser totalmente atingido:

- a) Os objectivos para a execução do processo estão identificados
- b) A execução do processo é planeada e monitorizada
- c) A execução do processo é ajustada para cumprir os planos
- d) Responsabilidades e autoridades para executar o processo, são definidas, atribuídas e comunicadas
- e) Os recursos e a informação necessária para a execução do processo são identificados, disponibilizados, atribuídos e utilizados
- f) Os interfaces entre as partes envolvidas são geridos para assegurar a comunicação efectiva e uma clara atribuição de responsabilidade

(ISO/IEC 15504-2, 5.3.1)

O atributo da capacidade 2.1 refere-se à aplicação de técnicas de gestão<sup>78</sup> para assegurar que os objectivos da execução e de qualidade do processo são atingidos. Este atributo da capacidade requer a actuação do processo de gestão de projectos (GES.1). Uma das preocupações principais consiste na identificação dos objectivos da execução do processo, incluindo um ou mais dos seguintes aspectos: qualidade dos produtos de trabalho, duração do ciclo de execução do processo e utilização de recursos. Este atributo só pode ser atingido com uma disponibilização efectiva dos recursos necessários e uma definição clara de responsabilidades e autoridade.

### “AC2.2. Atributo de gestão do produto de trabalho

---

<sup>78</sup> planeamento, acompanhamento e verificação.

O atributo de gestão do produto de trabalho mede a extensão com que os produtos de trabalho produzidos são geridos de forma apropriada. Como resultado deste AC ser totalmente atingido:

- a) Os requisitos para os produtos de trabalho são definidos
- b) Os requisitos para a documentação e controlo dos produtos de trabalho são definidos
- c) Os produtos de trabalho são identificados, documentados e controlados de forma apropriada
- d) Os produtos de trabalho são revistos e, se necessário, ajustados para cumprir os requisitos”

(ISO/IEC 15504-2, 5.3.2)

O atributo AC2.2 refere-se à definição dos requisitos, documentação, garantia de qualidade e monitorização da qualidade, dos produtos de trabalho. Para atingir este atributo da capacidade é necessário o contributo de alguns processos de suporte, definidos no MRP<sub>2MPspe</sub>: (SUP.1) documentação, para definição dos documentos produzidos pelo processo; (SUP.2) gestão de configurações, permitindo manter a integridade dos produtos de trabalho e (SUP.3) garantia de qualidade, assegurando a qualidade dos produtos de trabalho.

### **Nível 3 – Processo estabelecido**

#### “AC 3.1 Atributo de definição do processo

O atributo de definição do processo mede a extensão com que o processo normalizado é mantido para suportar a transferência<sup>79</sup> do processo definido. Como resultado deste atributo ser totalmente atingido:

- a) É definido um processo normalizado, incluindo orientações para “colocar à medida” e descrevendo os elementos fundamentais que têm que ser incorporados no processo definido
- b) É determinada a sequência e interacção do processo normalizado, com outros processos

---

<sup>79</sup> *deployment.*

- c) Como parte do processo normalizado são identificados papéis e competências necessárias para executar o processo
- d) Como parte do processo normalizado são identificados a infra-estrutura e o ambiente de trabalho necessários para executar o processo
- e) São determinados, métodos apropriados para monitorizar a efectividade e conveniência do processo

Nota: O processo normalizado pode ser utilizado “sem alterações” quando é feita a transferência do processo definido, neste caso as orientações para colocar à medida não são necessárias.”

(ISO/IEC 15504-2, 5.4.1)

O AC3.1 preocupa-se com a existência de um processo normalizado. Este atributo define o conteúdo do processo normalizado e identifica a necessidade de existir um conjunto de orientações para colocar o processo à medida. Finalmente, é necessário definir estratégias para monitorizar quão conveniente e efectivo é o processo normalizado. Este atributo da capacidade apoia-se noutros processos do MRP<sub>2</sub>MP<sub>spe</sub>, nomeadamente, o estabelecimento do processo (ORG.1), a gestão do conhecimento do processo (ORG.13) e a gestão de recursos humanos (ORG.11).

“AC3.2 Atributo de transferência do processo

O atributo de transferência do processo mede a extensão com que o processo normalizado é efectivamente transferido como um processo definido, para atingir os resultados do processo. Como resultado deste atributo ser totalmente atingido:

- a) É utilizado um processo definido que se baseia num processo normalizado, seleccionado e/ou colocado à medida de forma apropriada
- b) São atribuídos e comunicados os papéis, responsabilidades e autoridade necessários para executar o processo definido
- c) Os profissionais que executam o processo definido são competentes, com base numa educação, formação e experiência apropriados
- d) Os recursos requeridos e a informação necessária para executar o processo definido são disponibilizados, atribuídos e utilizados
- e) A infra-estrutura e o ambiente de trabalho requeridos para executar o processo definido são disponibilizados, geridos e mantidos

- f) Dados apropriados são coligidos e analisados, como base para a compreensão do comportamento, para demonstrar a adequação e efectividade do processo e para avaliar onde a melhoria contínua do processo pode ser realizada”

(ISO/IEC 15504-2, 5.4.2)

O AC3.2 define as características necessárias para a transferência efectiva do processo normalizado. Refere-se à atribuição de recursos, definição de competências e adequação da infra-estrutura do processo. Este atributo da capacidade requer o apoio de outros processos do MRP<sub>2MPspe</sub>: (SUP.3) gestão da qualidade e (ORG.2) gestão de recursos humanos.

A tabela 6.3 apresenta os níveis de capacidade e respectivos atributos da capacidade, utilizados no modelo 2MPspe.

Nível de Capacidade	Atributo da Capacidade
Nível 3 – Estabelecido	AC3.2 – Transferência do processo AC3.1 – Definição do processo
Nível 2 – Gerido	AC2.2 – Gestão dos produtos de trabalho AC2.1 – Gestão da execução
Nível 1.51 – Valorizado Nível 1 – Executado	AC1.51 – Valorização do processo AC1.1 – Execução do processo
Nível 0 – Incompleto	

*Tabela 6.3 – Níveis de capacidade e respectivos atributos da capacidade.*

### 6.3.3. Escala de Classificação dos ACs.

Como vimos, um atributo da capacidade representa uma característica mensurável de qualquer processo. É necessário definir uma escala de classificação que



permita descrever a extensão com que é atingido determinado atributo. A parte 2<sup>80</sup> da norma ISO/IEC 15504 propõe uma escala ordinal de quatro pontos, adoptou-se esta escala no 2MPspe. Apresenta-se em seguida a definição de cada um dos valores da escala e a tradução para uma escala percentual contínua [ISO/IEC 2004b].

- **N** Não atingido. Há pouca ou nenhuma evidência que o atributo é atingido (0 a 15%).
- **P** Parcialmente atingido. Há evidência que mostra que o atributo é algumas vezes atingido, embora alguns aspectos tenham um carácter imprevisível (>15% até 50%).
- **L** Largamente atingido. O atributo definido é atingido de uma forma significativa. O processo pode apresentar variações no desempenho em algumas áreas (>50 até 85%).
- **C** Completamente atingido. Há evidência que mostra que o atributo definido é atingido de uma forma completa no processo avaliado. Não foram detectadas fraquezas significativas (>85% até 100%).

Refira-se que os valores numéricos<sup>81</sup> associados a cada ponto da escala, servem apenas como indicadores, não sendo necessário armazenar essa informação.

Sobre a classificação dos atributos da capacidade, a parte 2<sup>82</sup> da norma 15504, prescreve que cada atributo da capacidade deve ser classificado até ao nível mais alto definido no âmbito da avaliação e utilizando a escala ordinal definida anteriormente. Em relação à designação das classificações, a mesma norma<sup>83</sup> considera que “deve ser atribuído um identificador a cada classificação do atributo da capacidade, que armazene o nome do processo e o atributo da capacidade avaliado”.

Considerou-se importante investigar a origem e justificação para a escala ordinal de quatro valores proposta pela norma, ISO/IEC 15504<sup>84</sup> para classificar os atributos da

---

<sup>80</sup> secções 5.7.1 e 5.7.2.

<sup>81</sup> em percentagem.

<sup>82</sup> secção 5.7.2.

<sup>83</sup> secção 5.7.3, da parte 2 da norma 15504.

<sup>84</sup> mas com origem no SPICE [1995].

capacidade. É reconhecido [Guilford 1954] que uma escala com poucos valores<sup>85</sup> não fornece discriminação suficiente, verificando-se a perda de informação, uma vez que os indivíduos conseguem normalmente detalhar mais a classificação. No outro extremo, uma escala com demasiados pontos<sup>86</sup>, poderá fornecer uma discriminação tão “fina” que ultrapassa a capacidade de julgamento dos inquiridos. Alguns estudos mostram que o número óptimo<sup>87</sup> de valores é cinco [Lissitz e Green 1975] e [Van de Ven e Ferry 1998], embora sugiram que em determinadas circunstâncias<sup>88</sup> é aconselhável uma escala com menos pontos. Craigmyle [1998] justifica a utilização de uma escala de quatro pontos no SPICE devido à constatação de que uma escala com um ponto central desequilibra os resultados. Verifica-se que este ponto é a opção mais fácil<sup>89</sup>, em vez do inquirido aprofundar a análise e seleccionar outra opção. Verifica-se então que a escala de classificação proposta se adequa ao objectivo de medir os atributos da capacidade.

A tabela 6.4 apresenta o nível de capacidade atingido por um determinado processo, este nível de capacidade é obtido através das classificações dos atributos da capacidade.

Refira-se que o modelo de classificação apresentado na tabela anterior permite alguma flexibilidade nos atributos específicos de cada nível<sup>90</sup>, embora os atributos dos níveis anteriores tenham que ser classificados como completamente atingidos. Este facto torna o referencial de classificação menos sensível a avaliações subjectivas.

A tabela 6.4 resume o referencial de classificação, apresentando as componentes principais e seus relacionamentos. A próxima secção apresenta a terceira dimensão proposta no 2MPspe, isto é, a dimensão do conhecimento do processo.

---

<sup>85</sup> no limite uma escala binária.

<sup>86</sup> no limite uma escala contínua.

<sup>87</sup> escala mais fidedigna.

<sup>88</sup> inquiridos sem um conhecimento profundo da área.

<sup>89</sup> naturalmente a opção seleccionada.

<sup>90</sup> classificação L ou C.

Nível	Atributos da capacidade	Classificação
1	AC1 – Execução do processo	L ou C
1.5	AC1 – Execução do processo AC1.51 – Valorização do processo	C L ou C
2	AC1 – Execução do processo AC1.51 – Valorização do processo AC2.1 – Gestão da execução AC2.2 – Gestão do produto de trabalho	C C L ou C L ou C
3	AC1 – Execução do processo AC1.51 – Valorização do processo AC2.1 – Gestão da execução AC2.2 – Gestão do produto de trabalho AC3.1 – Definição do processo AC3.2 – Transferência do processo	C C C C L ou C L ou C

Legenda:

C – Completamente atingido

L – Largamente atingido

Tabela 6.4 – Níveis derivados das classificações dos ACs (adaptado de ISO/IEC 2004b).

## 6.4. Dimensão do Conhecimento

A dimensão do conhecimento do M<sub>c</sub>2MP modela alguns factores contextuais que afectam os processos de uma organização de software. O objectivo é que este conjunto de problemas seja explicitamente conhecido e conscientemente tratado. Em relação ao referencial tridimensional referido, esta secção apresenta o sistema de eixos Conhecimento × Processo (ver figura 6.15)

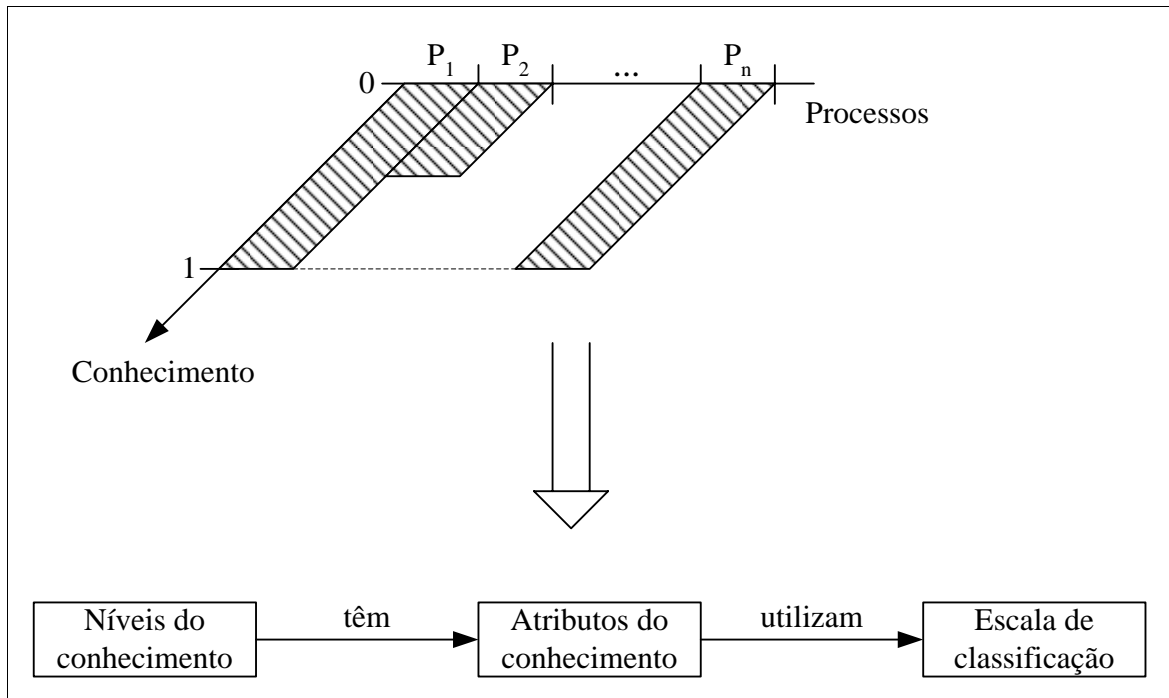


Figura 6.15 – Detalhe da dimensão do conhecimento.

Refira-se que o referencial bidimensional Conhecimento  $\times$  Capacidade (ver figura 6.16 poderá ser utilizado<sup>91</sup> no contexto do método de implementação do 2MPspe (M<sub>i</sub>2MP), em duas vertentes:

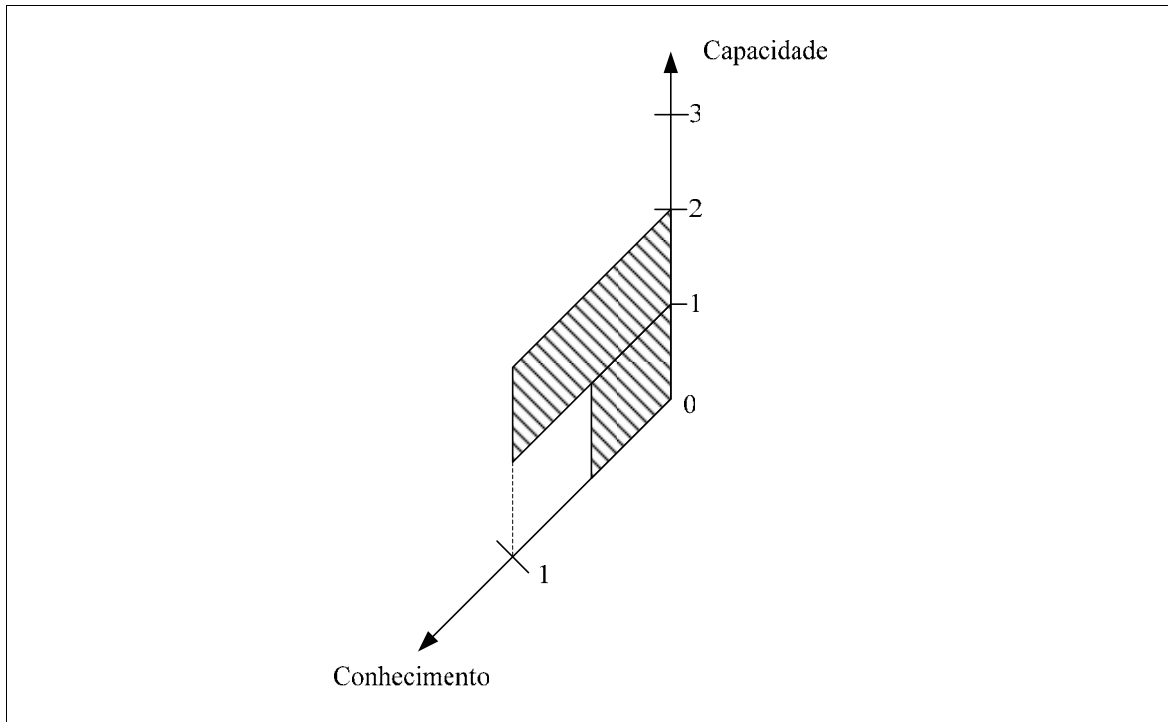
- Detalhar o impacto do conhecimento do contexto na transição entre níveis de capacidade
- Definir orientações sobre a capacidade dos processos directamente relacionados com os atributos do conhecimento

O referencial Conhecimento  $\times$  Capacidade será utilizado apenas no âmbito do M<sub>i</sub>2PM, uma vez que apresenta o “como” realizar as tarefas, ultrapassando os objectivos do modelo de avaliação centrados no “quê”.

Ainda sobre a figura 6.15, central nesta secção, a dimensão do conhecimento é constituída pelas componentes: Níveis de Conhecimento, Atributos do Conhecimento (AKs) e Escala de Classificação. O nível de conhecimento mede até que ponto os aspectos contextuais são explicitamente conhecidos, permitindo assim o seu tratamento consciente, na perspectiva do sucesso do programa de melhoria do processo de

<sup>91</sup> perspectiva-se incluir esta característica numa próxima versão.

software. Um atributo do conhecimento permite a avaliação de um factor específico do contexto do processo e é aplicável a todos os processos do MRP. A extensão com que o atributo do conhecimento é alcançado é medida através da escala de classificação, tal como se verifica com os atributos da capacidade (na dimensão da capacidade).



*Figura 6.16 – Referencial Conhecimento×Capacidade.*

As secções seguintes detalham e justificam as opções tomadas no desenvolvimento do referencial de conhecimento dos factores de contexto.

#### **6.4.1. Níveis de Conhecimento**

O eixo do conhecimento mede o nível de conhecimento explícito<sup>92</sup> sobre os factores contextuais que afectam os processos. No 2MPspe utiliza-se uma escala com apenas dois níveis: o nível 0, designado como nível implícito, em que os problemas contextuais ou são desconhecidos, ou são tratados apenas de uma forma implícita e o

---

<sup>92</sup> que a organização de software possui.

nível 1, denominado nível explícito, em que todos<sup>93</sup> os aspectos contextuais são tratados de uma forma consciente.

O número de aspectos contextuais considerados dependerá do âmbito de aplicação<sup>94</sup> do modelo e coincide com o número de atributos de conhecimento utilizados. No caso desta versão do 2MPspe são considerados dois factores contextuais: a cultura organizacional e a gestão do conhecimento. Considera-se um factor contextual como um aspecto que influencia o desempenho do processo independentemente do seu nível de capacidade e tem impacto sobre o sucesso no aperfeiçoamento do processo. Os factores contextuais caracterizam-se também pelo seu carácter dinâmico ao longo do tempo. É importante referir que não se afirma ser impossível melhorar a capacidade de um processo sem tratar explicitamente estes factores, mas antes, que se irá diminuir significativamente os casos de insucesso quando os aspectos contextuais são tratados explicitamente.

Os níveis de conhecimento são aplicados a cada processo do modelo de referência do processo e embora a escala contemple apenas dois níveis, utilizam-se subníveis<sup>95</sup> para modelar o conhecimento explícito completo de alguns factores contextuais. Por exemplo, caso fossem considerados quatro factores de contexto, com os correspondentes atributos de conhecimento, designados  $AK_1$ ,  $AK_2$ ,  $AK_3$  e  $AK_4$ , então, através da figura 6.17, poderíamos afirmar que, apenas o processo  $P_3$  estava no nível 1 de conhecimento, ou seja, tratava explicitamente todos os factores contextuais. O processo  $P_1$  trata 50% dos factores contextuais e os processos  $P_2$  e  $P_n$  tratam apenas 25% dos factores contextuais considerados.

Verifica-se então que os subníveis intermédios representam o facto de serem tratados alguns aspectos contextuais de forma explícita.

---

<sup>93</sup> referentes aos atributos de conhecimento utilizados.

<sup>94</sup> pensando em versões futuras.

<sup>95</sup> dependentes do número de atributos do conhecimento.

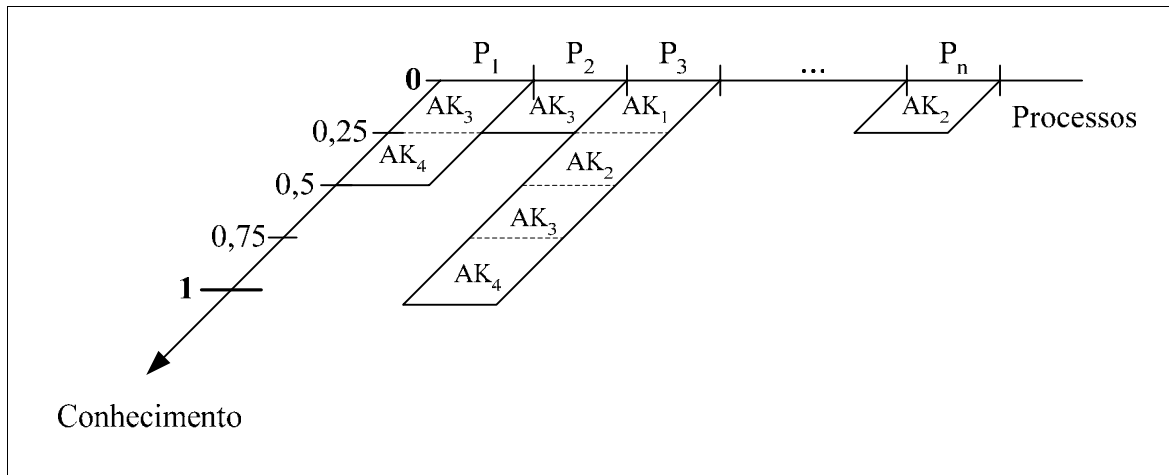


Figura 6.17 – Exemplo do nível de conhecimento de um conjunto de processos.

#### 6.4.2. Atributos do Conhecimento

A medição do nível de conhecimento do contexto dos processos é feita através dos atributos do conhecimento. Cada atributo do conhecimento mede um factor específico do contexto do processo, sendo aplicável a todos os processos do modelo de referência do processo.

Como foi apresentado no capítulo 4, há dois factores contextuais que influenciam de forma determinante o sucesso de um programa de melhoria do processo de software, isto é, a gestão do conhecimento e a cultura organizacional. Pretende-se, nesta versão do 2MPspe, tratar estes dois factores. Nesta perspectiva, para uma organização estar no nível de conhecimento explícito terá que atingir os seguintes atributos de conhecimento: (AK1.1) Atributo de gestão do conhecimento do processo e (AK1.2) Atributo de gestão da cultura organizacional.

Nas próximas secções desenvolve-se o enquadramento de cada atributo do conhecimento e apresenta-se a sua definição formal.

#### 6.4.2.1. Atributo de Gestão do Conhecimento do Processo

É reconhecido que a engenharia de software é uma actividade baseada no conhecimento [Mathiassen e Pourkomeylian 2003]. Simultaneamente, as organizações a trabalhar no negócio do desenvolvimento de software devem ter uma preocupação constante na adopção de novas tecnologias e na melhoria das suas práticas organizacionais. Este foco está ligado à sobrevivência da organização num mercado em constante mudança e ao ganho de vantagens competitivas. Há diversos trabalhos [Baskerville e Pries-Heje 1999], [Kautz e Thaysen 2001] e [Apostolou e Mentzas 2003] que demonstram o interesse em aplicar a gestão do conhecimento às organizações de software.

A gestão do conhecimento<sup>96</sup> é uma área emergente e ligada ao domínio da gestão<sup>97</sup> [Quintas et al. 1997]. Este facto e os antecedentes variados<sup>98</sup> dos cientistas e práticos a trabalhar na gestão do conhecimento, tornam as definições díspares e pouco consensuais. Apresenta-se, em seguida, uma pequena amostra recolhida na literatura:

“A gestão do conhecimento engloba práticas e processos que se preocupam com a criação, aquisição, captura, partilha e utilização de conhecimento, aptidões e especializações” [Quintas et al. 1996]

“A gestão do conhecimento é a actividade que se preocupa com a estratégia e tácticas para gerir os *assets* centrados nos recursos humanos” [Brooking 1997]

“Forças ambientais poderosas estão a mudar o mundo da gestão no século XXI. Estas forças chamam uma mudança fundamental nos processos organizacionais e na estratégia de recursos humanos. Isto é a gestão do conhecimento” [Taylor e al. 1997].

“Gestão do conhecimento é uma frase colectiva para um grupo de práticas e processos utilizados pelas organizações para aumentar o seu valor, melhorando de forma efectiva a geração e aplicação do seu capital intelectual” [Marr et al. 2003]

---

<sup>96</sup> denominada normalmente KM (*Knowledge Management*).

<sup>97</sup> eclético e com características subjectivas.

<sup>98</sup> por exemplo, economia, engenharia, sociologia, psicologia, etc.



Destas definições ressaltam dois aspectos importantes [McAdam e McCreedy 1999]: (1) a gestão do conhecimento relaciona-se tanto com a teoria como com a prática e (2) a gestão do conhecimento ultrapassa claramente a área das tecnologias de informação (TIs). Embora as TIs sejam o suporte, não são o ponto de interesse central. O foco central está nas pessoas e na aprendizagem.

Para Boisot [1995] os processos de gestão de conhecimento são metaprocessos<sup>99</sup> que não podem ser uniformemente observados como os processos físicos e diferem de acordo com os seus meios de criação, natureza, armazenamento, transmissão e modo de utilização. Verifica-se então que não há duas implementações iguais da gestão de conhecimento, uma vez que os recursos humanos são centrais em todas as aplicações de gestão de conhecimentos e cada indivíduo<sup>100</sup> tem percepções e filosofias diferentes sobre o conhecimento<sup>101</sup> [Marr et al. 2003]

A gestão do conhecimento refere-se então à “condução” do capital social e intelectual para melhorar as capacidades de aprendizagem organizacional, reconhecendo que o conhecimento, e não simplesmente a informação, é a fonte primária do potencial inovador de uma organização [Castells 1996] e [Marshall 1997].

A área de gestão de conhecimento é rica em modelos, apresentando diversas perspectivas. Por exemplo, o modelo proposto por Nonaka e Takeuchi [1995], apresenta uma representação de alto nível da gestão de conhecimento, considerada essencialmente como um processo de criação de conhecimento (ver figura 6.18).

		<u>Para</u>	
		Tácito	Explícito
<u>De</u>	Tácito	Socialização	Exteriorização
	Explícito	Interiorização	Combinação

Figura 6.18 – Modelo de gestão de conhecimento de Nonaka e Takeuchi [1995].

<sup>99</sup> no 2MPspe denominados factores contextuais.

<sup>100</sup> ou conjunto de indivíduos.

<sup>101</sup> perspectivas epistemológicas dos indivíduos [Pike e Roos 2002].

Neste modelo, o conhecimento é visto como constituído por dois elementos, a saber: o conhecimento tácito, definido como não verbalizado, intuitivo e não articulado [Polanyi 1962] e o conhecimento explícito, especificado em textos, modelos, desenhos, etc. O modelo assume que o conhecimento tácito pode ser transferido para conhecimento tácito através da camaradagem do dia a dia (socialização) e para conhecimento explícito através da formalização de um corpo de conhecimento (exteriorização). Através da tradução da teoria em prática (interiorização) o conhecimento explícito pode ser transformado em conhecimento tácito e finalmente a combinação de teorias existentes permite a passagem de conhecimento explícito para conhecimento explícito.

Este modelo é importante na compreensão da criação e transferência do conhecimento, mas no âmbito deste trabalho, interessa considerar modelos práticos de implementação da estratégia de gestão de conhecimento. Da revisão da literatura sobre gestão de conhecimento surgiram três estratégias principais denominadas, abordagem por codificação, abordagem por personalização e abordagem conexionista [Venzin et al. 1998], [Swan et al. 1999] e [Hansen et al. 1999].

A abordagem por codificação assenta na perspectiva da organização cognitiva, onde o conhecimento válido, localizado no cérebro, é identificado, capturado e processado através das TIs e assim pode ser utilizado noutros contextos [Swan et al. 1999]. Esta abordagem tem como papel central a utilização de ferramentas sofisticadas com base nas TIs. Pode ser denominada também como “centrada no produto” ou “cognitivismo”.

A abordagem por personalização vê a gestão do conhecimento como um processo de comunicação social onde a observação objectiva é impossível. Verifica-se então que o conhecimento está intimamente ligado à pessoa que o desenvolveu e a sua partilha é realizada através de contactos pessoais [Apostolou e Mentzas 2003]. O objectivo principal das TIs é apenas facilitar a comunicação entre as pessoas. Esta abordagem é designada também como “centrada no processo” ou “baseada em comunidades”.

A abordagem conexionista vê as organizações como redes auto organizadas compostas por relacionamentos. As regras sobre como a informação é processada não são universais, sofrendo variações locais [Venzin et al. 1998]. O conhecimento é possuído pelos indivíduos mas também é expresso em regulamentos através dos quais os profissionais cooperam em comunidades sociais [Kogut e Zander 1995].

A tabela 6.5 faz a correspondência entre a abordagem à gestão do conhecimento e os processos de conhecimento.

<b>Abordagem Processo</b>	<b>Codificação</b>	<b>Conexionismo</b>	<b>Personalização</b>
Codificação do conhecimento	O conhecimento pode ser codificado	Algum conhecimento precisa de informação sobre o contexto	<i>Não aplicável</i>
Armazenamento do Conhecimento	O conhecimento pode ser armazenado em Bases de Dados.	Armazenamento de informação e regras locais	<i>Não aplicável</i>
Mapeamento do conhecimento	Mapeamento directo do conhecimento	Mapeamento de informação e regras	Mapeamento dos indivíduos com conhecimento
Partilha do Conhecimento	Sistemas inteligentes e BD de conhecimento	Sistemas inteligentes, reuniões fóruns, <i>groupware</i>	Reuniões não reguladas, conversas casuais, discussões abertas
Aquisição do Conhecimento	Capturado em sistemas TIs, revisões da literatura	Relacionamentos externos, convertem o conhecimento para o contexto interno	Redes abertas e relacionamentos casuais
Criação do Conhecimento	Pode ser gerida directamente	Não regulado dentro das regras locais	Abertura, liberdade, caos aparente.

*Tabela 6.5 – Abordagem ao conhecimento e os processos de conhecimento (adaptado de Marr et al. [2003]).*

Foram apresentadas três abordagens à gestão do conhecimento. Definir o perfil de gestão de conhecimento de um determinado objecto consiste então em identificar a abordagem à gestão do conhecimento preferencial e avaliar o impacto da abordagem no

processo. Neste contexto, o atributo de conhecimento AK1.1 apresenta a definição seguinte,

### **AK1.1 – Atributo de gestão do conhecimento do processo**

O AK1.1 mede a extensão com que a abordagem à gestão do conhecimento no processo está explicitamente identificada e aplicada. Como resultado deste atributo ser totalmente atingido:

- a) O perfil de gestão de conhecimento do(s) indivíduo(s) e da(s) equipa(s) está identificado
- b) O perfil de gestão de conhecimento específico do processo está identificado
- c) O impacto dos perfis no processo está documentado
- d) Os perfis de Gestão do Conhecimento estão alinhados
- e) Há uma infra-estrutura adequada para as necessidades do processo

Refira-se que nas PES, devido ao número reduzido de recursos humanos, o perfil de conhecimento dos indivíduos e das equipas pode ser identificado em termos globais<sup>102</sup>.

#### **6.4.2.2. Atributo de Gestão da Cultura Organizacional**

É reconhecido pela comunidade de software que a área de melhoria do processo de software, está directamente relacionada com mudanças organizacionais [Paulk et al. 1997], [Spence 2002] e [Dyba 2002]. Deste facto surge naturalmente a questão: como é possível implementar, com sucesso, um programa de melhoria do processo de software, sem tratar explicitamente as questões relacionadas com a cultura organizacional? [Nielsen e Norbjerg 2001] e [Siakas e Georgiadou 2002]. Como afirma Lundberg [1989], “A cultura organizacional determina muito sobre o que podemos fazer enquanto tentamos gerir a mudança”.

---

<sup>102</sup> mais detalhe na secção 7.5.1.

Não há uma definição consensual sobre cultura organizacional, contudo, Hofstede [1984] formaliza cultura como “a programação colectiva da mente que distingue os membros de um grupo humano de outro”. Outros vêm a cultura organizacional como estruturas sociais (símbolos, normas e significados partilhados) que dão sentido às realidades organizacionais [Pettigrew 1979] e [Morgan 1997].

Embora as definições sejam variadas e cubram diferentes perspectivas, a maioria dos autores concorda que a cultura é a base sobre a qual as acções organizacionais são construídas e estabelecidas [Hofstede 1991], [Spence 2002] e [Dyba 2002]. Neste trabalho analisa-se a cultura organizacional como uma realidade dinâmica e segue-se a proposta de Smircich [1983], que apresenta a cultura organizacional como uma estrutura cognitiva e social. A parte cognitiva está embebida no cérebro dos profissionais, mas a parte social é exteriorizada através dos designados artefactos organizacionais<sup>103</sup>.

Para o âmbito deste trabalho foi seleccionado o referencial<sup>104</sup> de Quinn e McGrath [1985] para classificar o tipo de cultura organizacional preponderante associada à organização como um “todo” e também associada aos grupos de profissionais<sup>105</sup> intervenientes nos processos. Este referencial foi escolhido devido ao seu foco nos problemas de mudança organizacional e ao seu alinhamento com as quatro escolas da teoria organizacional<sup>106</sup>. Este referencial caracteriza quatro tipos de cultura organizacional: hierárquica, racional, consensual e em desenvolvimento (tabela 6.6).

Em termos gerais, considera-se que o protótipo da cultura hierárquica são as organizações militares, embora se possa encontrar em muitas empresas, em diferentes áreas. O protótipo do tipo racional é a célula de produção, orientada à produtividade e eficiência. A cultura consensual é orientada à coesão, manutenção e moral da equipa. Finalmente, a cultura do desenvolvimento é orientada ao crescimento e o enfoque principal é o desenvolvimento humano [Ngwenyama e Nielsen 2003]. Como seria de esperar, uma organização pode apresentar um “*mix*” de culturas [Trice e Beyer 1984],

---

<sup>103</sup> há várias categorizações para estes artefactos, por exemplo, Schein [1985] ou Lundberg [1989].

<sup>104</sup> denominado “valores em competição”.

<sup>105</sup> equipas.

<sup>106</sup> processo interno, objectivo racional, relações humanas e sistemas abertos [Morgan 1997].

embora normalmente exista um tipo preponderante. Em relação à caracterização da cultura organizacional de uma equipa, verifica-se um maior alinhamento a um determinado tipo de cultura ideal.

<b>Tipo C.O.</b> <b>Aspecto</b>	<b>Hierárquica</b>	<b>Racional</b>	<b>Consensual</b>	<b>Em desenvolvimento</b>
Orientação organizacional	Estabilidade e controlo	Produtividade e eficiência	Coesão e moral	Flexibilidade, adaptabilidade, prontidão
Objectivos organizacionais	Execução de regulamentos	Cumprir objectivos	Manutenção do grupo	Crescimento e desenvolvimento
Estrutura organizacional	Tarefas repetidas; regras e políticas formais	Tarefas complexas; responsabilidades baseada na especialização	Tarefas complexas; grupos de trabalho colaborantes	Tarefas complexas; grupos de trabalho colaborantes
Base do poder	Conhecimento de regras e procedimentos organizacionais	Competência	Capacidade de cultivar relacionamentos	Valores
Tomada de decisão	<i>Top-down</i>	Centrada em objectivos sistemática e analítica	Participativa, deliberativa	Orgânica, intuitiva
Liderança	Conservador, cauteloso	Orientado a objectivos	Construtor da equipa, apoiante	Idealista, orientado ao risco
Cumprimento	Monitorização e controlo	Acordo contratual	Compromisso com o processo	Compromisso com valores
Avaliação	Cumprimento das regras	Nível de produtividade	Qualidade dos relacionamentos	Intensidade do esforço
Reacção à mudança	Resistente	Abertura à mudança, orientadas a objectivos	Abertura à mudança	A mudança faz parte do crescimento

Tabela 6.6 – Referencial dos valores em competição [Quinn e McGrath 1985].

Foi apresentado um referencial que permite categorizar os tipos de conhecimento organizacional. A aplicação deste referencial em termos de impactos e acções específicas num projecto de melhoria de software, será feito através de orientações no Método de implementação<sup>107</sup> do 2MPspe. Na perspectiva da avaliação do conhecimento dos factores contextuais pretende-se verificar se é aplicado um modelo de classificação das culturas organizacionais<sup>108</sup>. Neste contexto, o atributo de conhecimento AK1.2, é definido do seguinte modo:

<sup>107</sup> a desenvolver numa próxima versão do 2MPspe.

<sup>108</sup> o modelo dos valores em competição ou outro.

### **AK1.2 – Atributo de gestão da cultura organizacional**

O AK1.2 mede até que ponto o tipo de cultura organizacional no processo está explicitamente identificada e aplicada. Como resultado deste atributo ser totalmente atingido:

- a) O tipo de cultura organizacional predominante na equipa interveniente no processo está identificado
- b) O impacto da cultura organizacional no processo está documentado
- c) Há um sistema de acompanhamento da dinâmica da cultura organizacional da equipa

Nas pequenas empresas de software, a cultura organizacional identificada na organização é normalmente partilhada por todos os processos. Este facto deve-se ao reduzido número de funcionários, provocando uma cultura organizacional única em todos os grupos<sup>109</sup>.

#### **6.4.3. Escala de Classificação dos AKs**

Como foi definido, cada atributo do conhecimento mede um factor específico do contexto do processo. É necessário definir uma escala de classificação que permita descrever a extensão com que um determinado atributo do conhecimento é considerado. Para uniformizar o referencial da capacidade e o referencial do conhecimento do contexto, utiliza-se a mesma escala ordinal de quatro pontos<sup>110</sup>. A definição de cada um dos valores é diferente da apresentada anteriormente. Cada atributo do conhecimento é classificado como:

- **N** Não atingido. O atributo do conhecimento não é considerado.
- **P** Parcialmente atingido. O atributo do conhecimento está identificado.
- **L** Largamente atingido. O atributo do conhecimento está identificado e os seus impactos estão documentados.

---

<sup>109</sup> ou até um único grupo.

<sup>110</sup> ver secção 6.3.3.

- **C** Completamente atingido. O atributo está identificado, os seus impactos são conhecidos e está definida uma estratégia de acompanhamento da evolução do factor contextual.

Utilizando esta escala de classificação, um processo estará no nível do conhecimento explícito se todos os atributos do conhecimento estiverem classificados como L ou C.

O nível 1 mostra que todos os factores contextuais, considerados no modelo, são tratados. Caso apenas um factor obtenha a classificação L ou C, o processo estará no nível intermédio 0,5. Este valor indica que 50% dos factores contextuais são tratados de forma explícita.

A tabela 6.7 apresenta o nível de conhecimento atingido por um determinado processo, este nível de conhecimento é obtido através das classificações dos atributos do conhecimento.

Nível do Conhecimento	Atributos do Conhecimento	Classificação
0,5	AK1 – Gestão do conhecimento	L ou C
	AK2 – Gestão da cultura organizacional	<u>ou</u> L ou C
1	AK1 – Gestão do conhecimento	L ou C
	AK2 – Gestão da cultura organizacional	<u>e</u> L ou C

<p><u>Legenda:</u>  C – Completamente atingido  L – Largamente atingido</p>
---

*Tabela 6.7 – Níveis de conhecimento versus Atributos de conhecimento.*

A secção seguinte mostra a representação da capacidade e do conhecimento no referencial tridimensional proposto no 2MPspe.



## 6.5 Representação no Sistema de Eixos Tridimensional

Como foi apresentado o 2MPspe propõe um referencial tridimensional para a caracterização da maturidade organizacional. A figura 6.19 mostra o referencial composto por três dimensões: processos, representa o conjunto de processos seleccionados no  $MRP_{2MPspe}$ ; capacidade, representa o nível de capacidade dos processos e conhecimento, representa o nível de conhecimento dos factores de contexto considerados: gestão de conhecimento e gestão da cultura organizacional. O referencial designa-se por:

$$\text{processos (P)} \times \text{capacidade (C)} \times \text{conhecimento (K)} = (P \times C \times K)$$

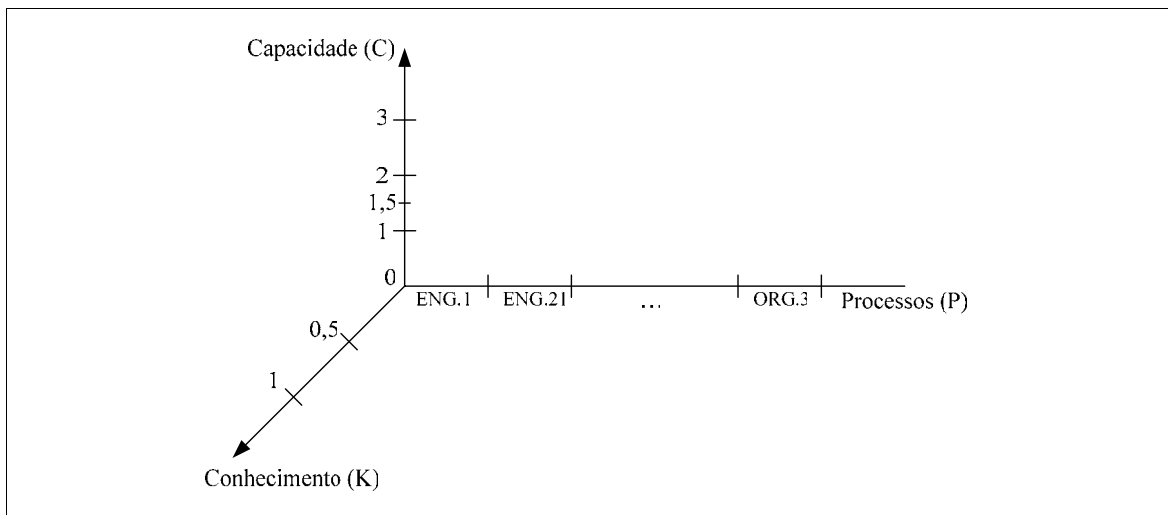


Figura 6.19 – Referencial proposto pelo 2MPspe.

Na figura 6.19, a variável qualitativa  $P_1, P_2, \dots, P_n$  corresponde aos processos do 2MPspe, ou seja,  $P_1 = \text{ENG.1}$  (Gestão de Requisitos) ...  $P_{18} = \text{ORG.3}$  (Suporte ao Cliente).

Refira-se que, nesta secção apresenta-se uma análise “teórica” da representação no referencial tridimensional. A representação dos resultados em termos práticos, no contexto do modelo de avaliação ( $M_a2MP$ ) é apresentada na secção 7.5.2.

No contexto de utilização do 2MPspe apresentam-se três cenários hipotéticos de utilização:

- (1) Utilização do 2MPspe apenas para avaliar a capacidade dos processos. Embora com consciência das probabilidades de insucesso da iniciativa é uma hipótese a considerar.
- (2) Utilização do 2MPspe para avaliar o conhecimento dos factores contextuais de uma organização. No caso de uma empresa que tivesse realizado uma avaliação<sup>111</sup> da capacidade dos processos há pouco tempo e pretendesse conhecer a percepção dos factores contextuais.
- (3) Utilização do 2MPspe para avaliar a maturidade da organização, avaliando a capacidade e o conhecimento dos seus processos.

No cenário 1, a iniciativa a desenvolver envolvia apenas o referencial bidimensional, processos (P) × capacidade (C). A figura 6.20 representa as iniciativas de melhoria tradicionais e também o cenário 1 no 2MPspe.

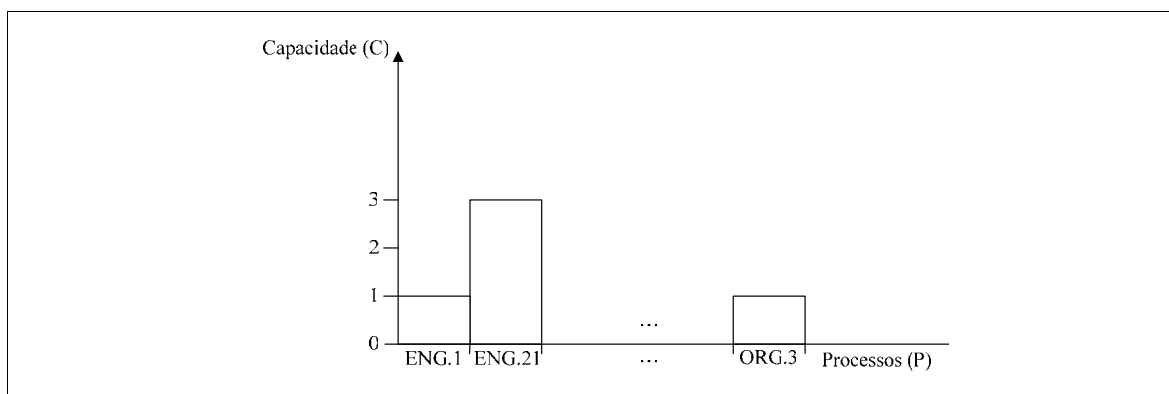


Figura 6.20 – Cenário 1: Processos (P) × Capacidade (C).

O plano de melhoria teria como base a figura 6.20, verificando-se quais os processos com maior capacidade e aplicando-se as prioridades definidas.

No cenário 2, o projecto de melhoria referia-se apenas à avaliação do conhecimento dos factores contextuais, ou seja, envolvia o referencial, processos(P)×conhecimento(K).

<sup>111</sup> ISO 15504, SPIRE, RAPID,...

A figura 6.21 representa um cenário hipotético de avaliação da gestão do conhecimento e da cultura organizacional.

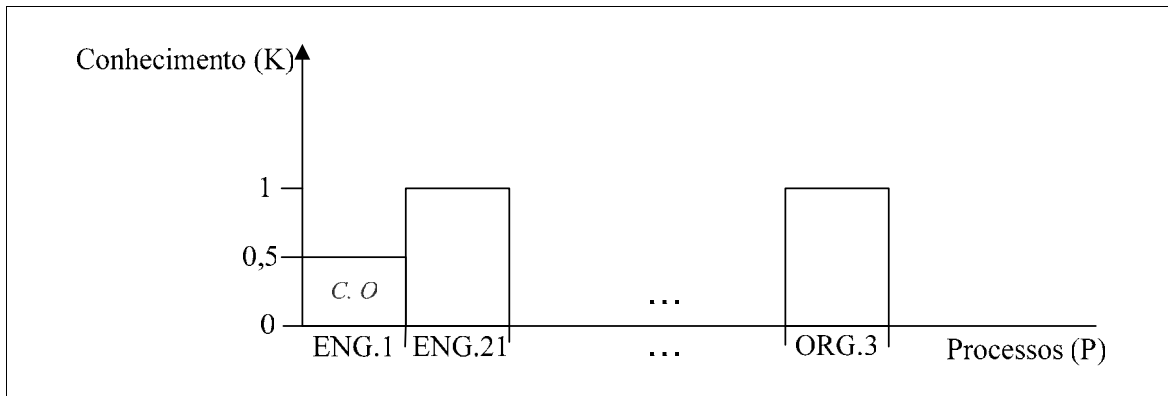


Figura 6.21 – Cenário2: Processos (P) × Conhecimento (K).

No caso do cenário 2, a avaliação realizada teria impacto no plano de acção já desenvolvido. Certamente novas prioridades seriam definidas e surgiria uma compreensão melhorada da capacidade dos processos e dos caminhos a adoptar na iniciativa.

Finalmente, no cenário 3, apresenta-se a utilização completa do 2MPspe, realizando a avaliação da capacidade e do conhecimento dos processos. Cada processo pode ser representado pelo terno (P, C, K), por exemplo (ORG.2; 0; 0,5) e o conjunto de ternos representa o perfil de capacidade e conhecimentos dos processos (ver figura 6.22).

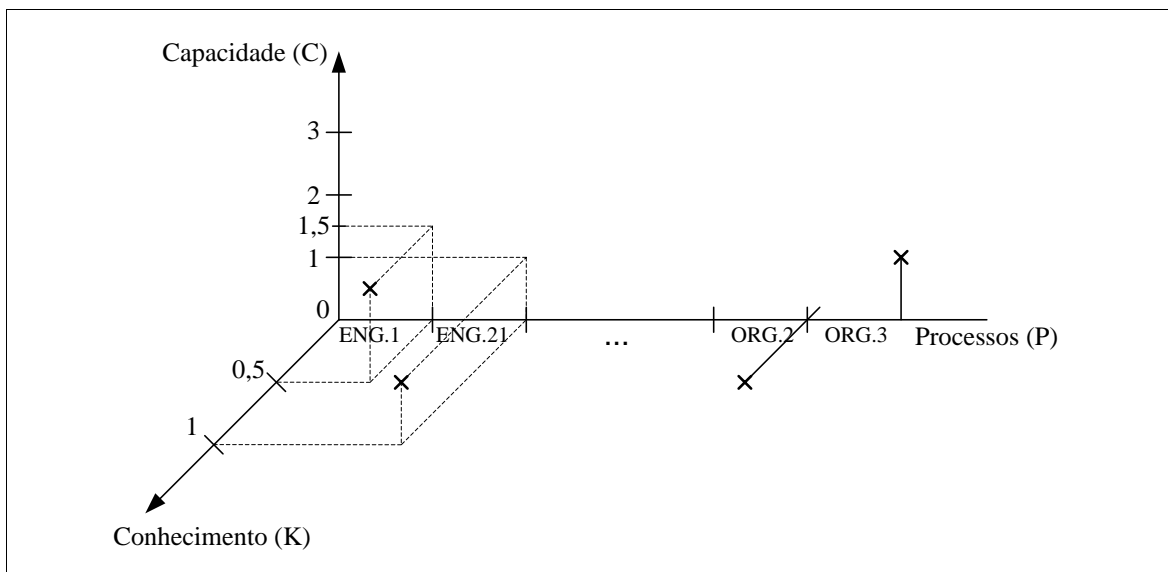


Figura 6.22 – Cenário 3: Processo × Capacidade × Conhecimento.

A figura 6.22 dá-nos uma perspectiva completa sobre a capacidade e conhecimento. Para uma análise mais detalhada deve-se representar nos eixos  $P \times C$  (figura 6.23 a.),  $P \times K$  (figura 6.23 b.),  $K \times C$  (figura 6.23 c.).

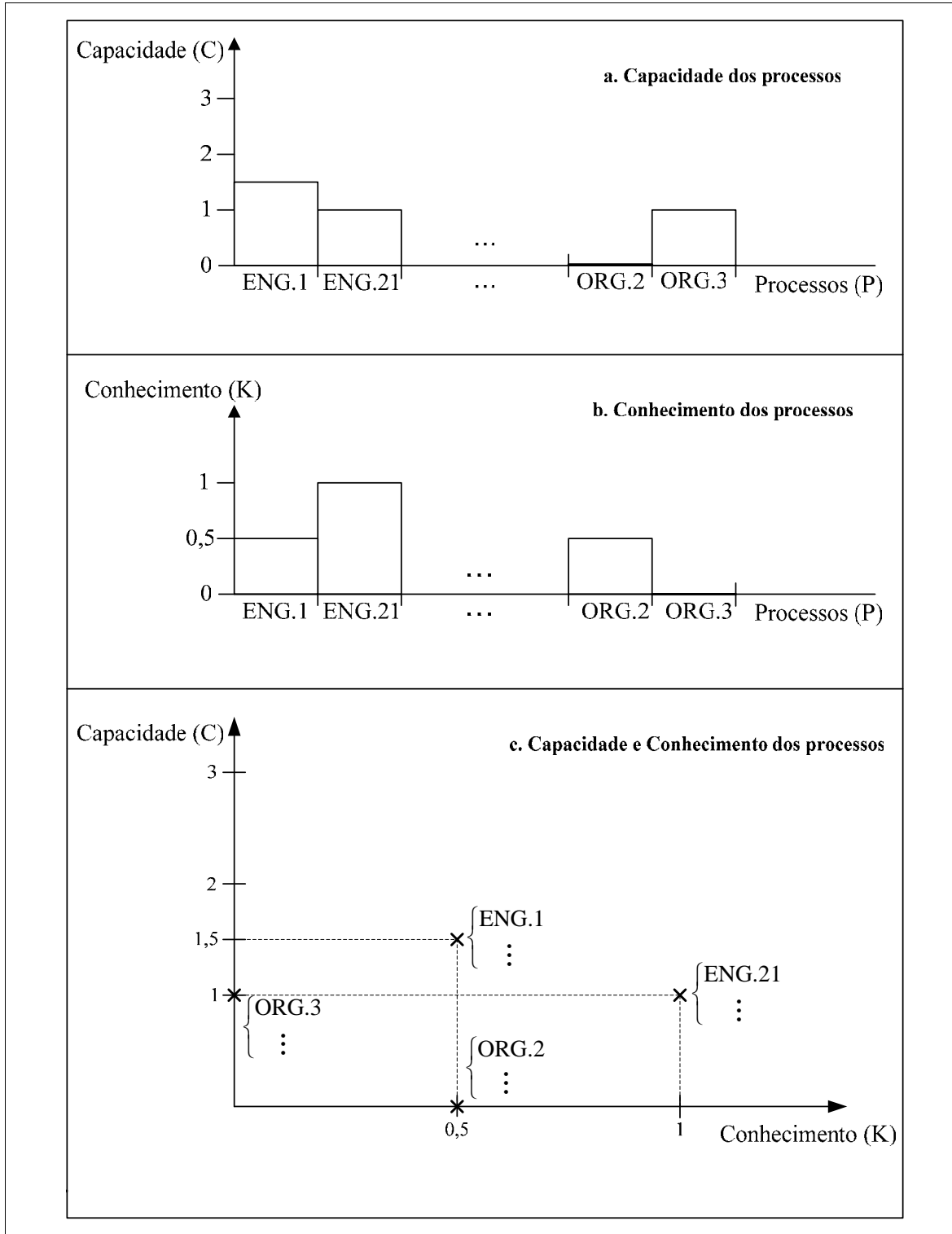


Figura 6.23 – Cenário 3: Representação em duas dimensões.

As representações a. e b. da figura 6.23 já foram explicadas no início da secção, refira-se a “riqueza” da representação c., pois consegue dar uma perspectiva global elucidativa sobre a maturidade da organização e simultaneamente detalhar os níveis de capacidade e conhecimento atingidos por cada processo.

A representação no referencial tridimensional permite também obter diversos níveis de abstracção, bastando para isso substituir o eixo dos processos (variável qualitativa) pelo eixo da categoria dos processos (ver figura 6.24) ou pelo eixo da organização (ver figura 6.25).

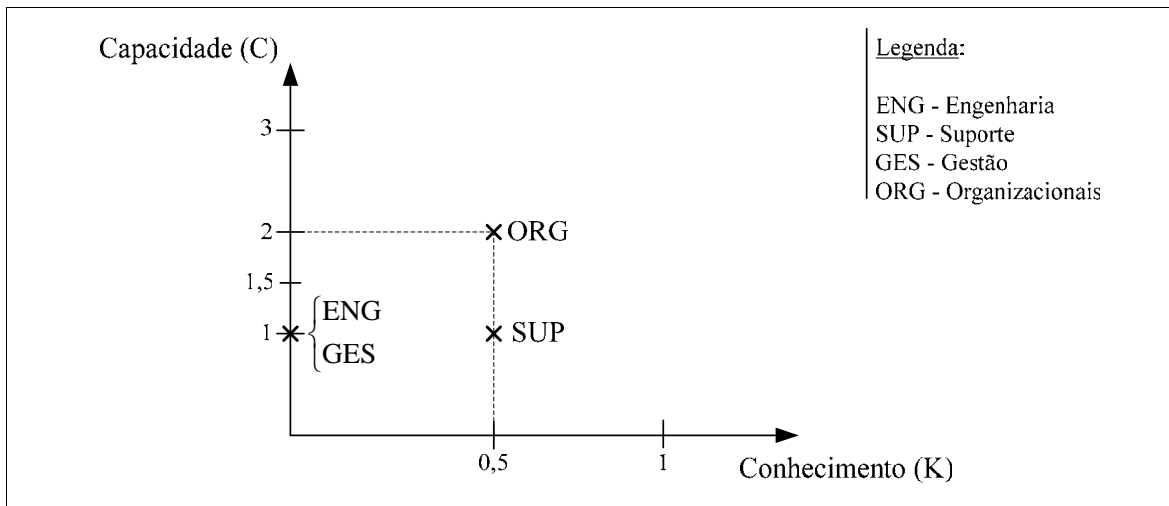


Figura 6.24 – Capacidade e conhecimento das categorias dos processos.

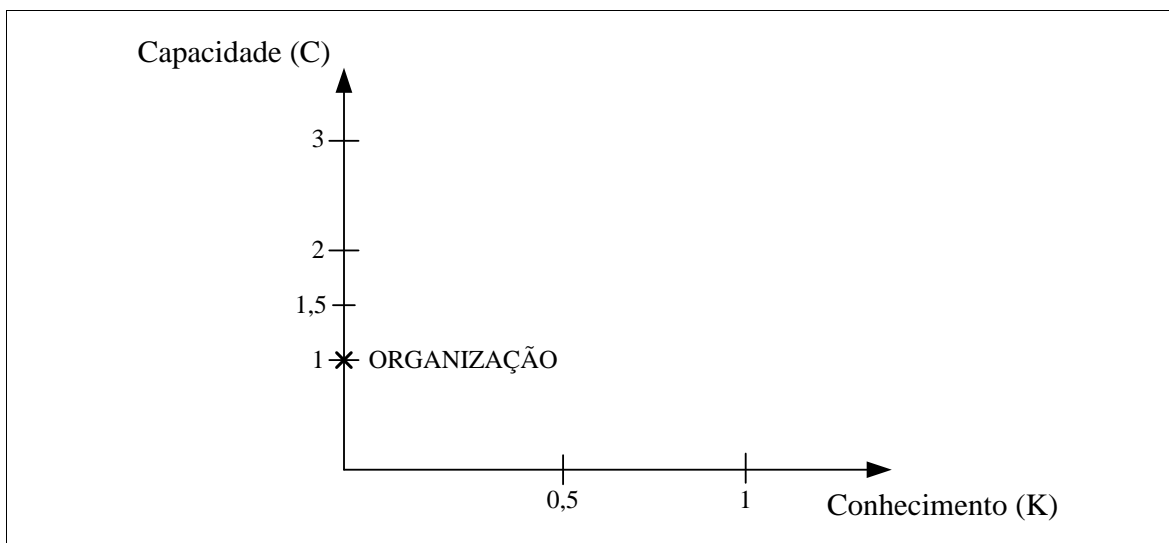


Figura 6.25 – Maturidade da Organização.

No caso da representação da maturidade da organização, uma vez que o eixo da variável qualitativa tem apenas um elemento pode fazer-se a representação no referencial bidimensional constituído pelas duas variáveis quantitativas.

Neste capítulo apresentou-se o modelo conceptual do 2MPspe e descreveu-se cada uma das suas componentes. O próximo capítulo apresenta o modelo de avaliação do processo do 2MPspe ( $M_a2MP$ ).

## Capítulo 7

---

### 7. Modelo de Avaliação (M<sub>a</sub>2MP)

Este capítulo apresenta o modelo de avaliação do processo proposto para o 2MPspe. O M<sub>a</sub>2MP fornece o conjunto de indicadores a partir dos quais são comparados os processos de uma organização e tem como *output* o Questionário da Maturidade<sup>1</sup>. Para o desenvolvimento do Modelo de avaliação utilizou-se o M<sub>c</sub>2MP, seguindo as orientações da parte 2 da norma ISO/IEC 15504. Refira-se que o questionário 2MPspe e o Método de implementação (M<sub>i</sub>2MP) serão a base para a execução de uma iniciativa de melhoria.

#### 7.1. Introdução

O modelo de avaliação suporta as três dimensões utilizadas no M<sub>i</sub>2MP e descreve as melhores práticas que irão permitir caracterizar os processos (ver figura 7.1).

O M<sub>a</sub>2MP cobre as três dimensões através das práticas de base, das práticas genéricas e das práticas transversais. As práticas de base descrevem as actividades desejáveis dos processos permitindo avaliar a sua execução. As práticas genéricas detalham os atributos de capacidade suportando a avaliação da capacidade dos

---

<sup>1</sup> denominado Questionário 2MPspe.

processos. Finalmente, as práticas transversais instanciam os atributos do conhecimento possibilitando a avaliação do conhecimento do contexto do processo.

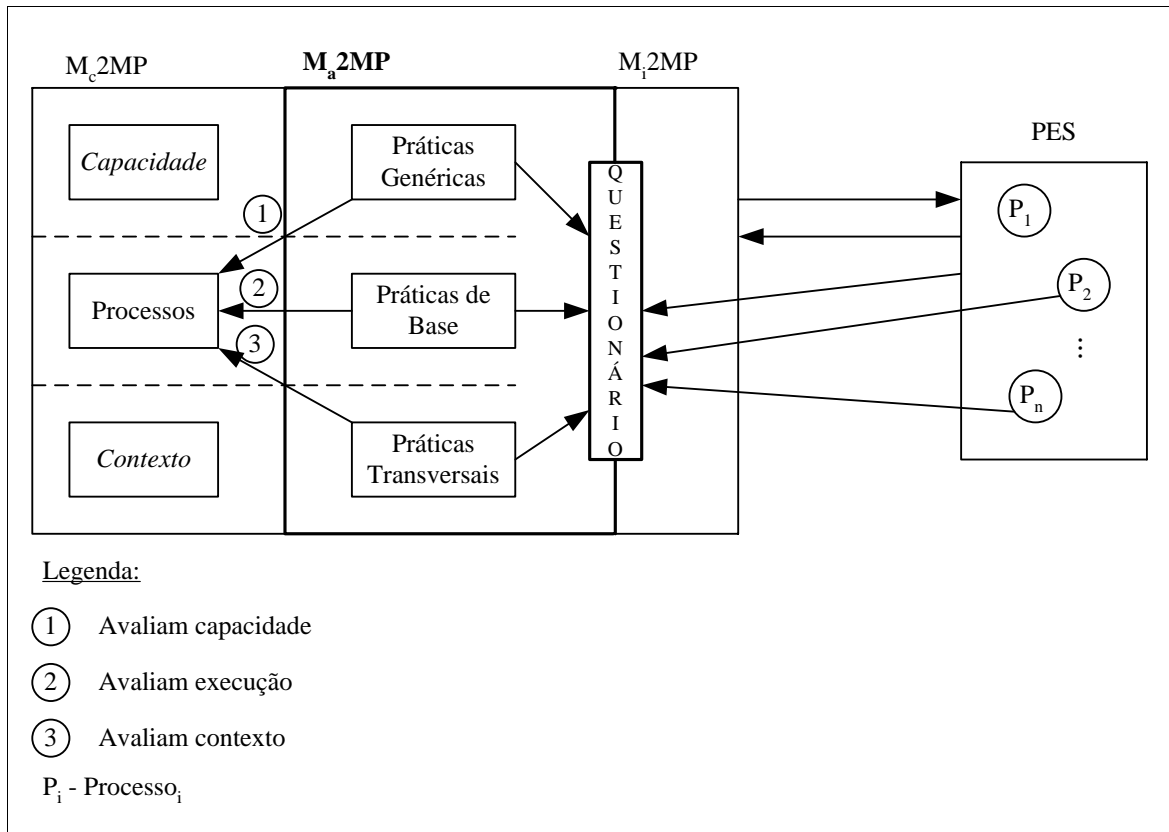


Figura 7.1 – Enquadramento do  $M_a2MP$ .

A secção 6.3 da parte 2 da norma ISO/IEC 15504 define as características de conformidade dos modelos de avaliação do processo. Começa por referir que um modelo de avaliação “forma a base para coligir evidência e classificar a capacidade do processo” [ISO/IEC 2004b]. Em seguida, define os requisitos de um modelo de avaliação do processo em termos de: (1) definição do seu propósito, âmbito e elementos; (2) correspondência para o MRP e para o referencial de medição e (3) mecanismo para a apresentação consistente dos resultados. Em relação à conformidade, é necessário cumprir as seguintes indicações:

→ “Um modelo de avaliação do processo tem que declarar o seu âmbito de cobertura em termos de:

- a) O modelo de referência do processo seleccionado
- b) Os processos seleccionados do modelo de referência do processo



c) Os níveis de capacidade seleccionados do referencial de medição”

(ISO/IEC 15504-2, 6.3.2)

No caso do  $M_a2MP$ , o modelo de referência dos processos é o  $MRP_{2MPspe}$  e são seleccionados todos os processos do MRP. Em relação aos níveis de capacidade, são tratados os três primeiros níveis, acrescentando-se o subnível 1.5, denominado “Processo Valorizado”.

→ “O modelo de avaliação do processo tem que basear-se num conjunto de indicadores que tratem explicitamente as metas e resultados, tal como definido no MRP seleccionado e que demonstrem que os atributos do processo foram atingidos”

(ISO/IEC 15504-2, 6.3.3)

Os indicadores propostos no  $M_a2MP$  são apresentados nas secções 7.2, 7.3 e 7.4.

→ “O modelo de avaliação do processo tem que fornecer uma correspondência explícita dos elementos relevantes do modelo para os processos do MRP seleccionado e para os atributos do processo relevantes do referencial de medição”

(ISO/IEC 15504-2, 6.3.4)

As secções seguintes descrevem as práticas de base, práticas genéricas e práticas transversais e apresentam a correspondência para os elementos do  $M_c2MP$ .

→ “O modelo de avaliação do processo tem que fornecer um mecanismo formal e verificável para representar os resultados de uma avaliação, como um conjunto de classificações dos atributos do processo”

(ISO/IEC 15504-2, 6.3.5)

A secção 7.5.2 apresenta um mecanismo para representar os resultados.

Nas próximas secções apresentam-se os indicadores utilizados para avaliação das características dos processos. A última secção descreve a representação dos resultados proposta.

## 7.2. Práticas de Base

As práticas de base e os produtos de trabalho são os indicadores da execução do processo do nível 1 [ISO/IEC 2003]. Uma prática de base é então uma actividade cuja execução permite atingir o propósito do processo. As práticas de base são descritas num nível abstracto, especificando “o que” deve ser feito, mas não “como” deve ser feito.

O anexo C da parte 5 da norma ISO/IEC 15504 apresenta orientações para a definição das práticas de base. As práticas de base são utilizadas como actividades de engenharia de software ou gestão que tratam o propósito e os resultados de um processo particular. Utiliza-se a identificação para as práticas de base proposta no anexo C. Uma prática de base será identificada pela sequência: cp.pr.PB.np, em que os códigos são:

- cp, identifica a categoria do processo
- pr, identifica o número do processo dentro da categoria
- PB, texto (Prática de Base)
- np, número da prática no processo

Por exemplo, SUP.1.PB.3, refere-se à prática de base número 3 do processo de “Documentação” da categoria do processo de “Suporte”. Para a definição das práticas de base utiliza-se a descrição dos processos<sup>2</sup> do MRP<sub>2MPspe</sub> e a parte 5 da norma ISO/IEC 15504<sup>3</sup>. São também referidos os produtos de trabalho principais para cada processo. A justificação das exclusões efectuadas devido ao contexto das PES encontra-se no anexo B.

### 7.2.1. Processos de Engenharia (ENG)

São os processos directamente relacionados com a construção do produto. No caso do 2MPspe consideram-se os processos de Gestão de Requisitos, Desenvolvimento do Software e Manutenção.

---

<sup>2</sup> ver secção 6.2.3.

<sup>3</sup> apresenta um exemplo de um modelo de avaliação do processo.

### **7.2.1.1. Gestão de Requisitos (ENG.1)**

Como vimos, o propósito da Gestão de requisitos é recolher, processar e acompanhar as necessidades e requisitos do cliente durante o ciclo da vida do produto.

Práticas de base (ENG.1.PB.n)

1. Obter os requisitos e pedidos do cliente
2. Compreender as expectativas do cliente
3. Obter a aprovação dos requisitos
4. Estabelecer a versão de base dos requisitos do cliente
5. Gerir os pedidos de alterações dos requisitos realizados pelo cliente
6. Estabelecer os mecanismos para manter o cliente informado sobre o estado dos requisitos

Os produtos de trabalho que devem ser identificados são: relatório de controlo de alterações, e requisitos de cliente.

### **7.2.1.2. Desenvolvimento do Software (ENG.2)**

#### **7.2.1.2.1. Análise dos Requisitos do Software (ENG.21)**

O propósito da análise de requisitos é estabelecer os requisitos do software

Práticas de base (ENG.21.PB.n)

1. Especificar os requisitos do software
2. Desenvolver critérios para o teste de aceitação do software
3. Avaliar e actualizar os requisitos do software
4. Desenvolver versão base dos requisitos do software

Os produtos de trabalho a identificar são: relatório de controlo de alterações, requisitos de interface e requisitos do software.

#### **7.2.1.2.2. Concepção do Software (ENG.22)**

O propósito da concepção do software é definir uma solução para o software que acomode os requisitos do software e possa ser testada em relação a eles.

Práticas de base (ENG.22.PBn)

1. Descrever a arquitectura do software e as interfaces entre os elementos
2. Desenvolver a concepção detalhada
3. Avaliar e verificar a especificação do software
4. Assegurar a consistência com os requisitos do software

Os produtos de trabalho resultantes são: especificação da Base de Dados, especificação de alto nível do software, especificação detalhada do software e registo de consistência com os requisitos de software.

#### **7.2.1.2.3. Construção do Software (ENG.23)**

O propósito da construção do software é produzir unidades de software executáveis que implementem correctamente a concepção do software.

Práticas de base (ENG.23.PB.n)

1. Desenvolver procedimentos para a verificação das unidades
2. Produção das unidades de software
3. Assegurar a consistência com a especificação do software
4. Verificar as unidades de software

Os produtos de trabalho resultantes são: procedimentos de teste das unidades de software, unidades de software, resultados dos testes e registo de consistência com a especificação do software.

#### **7.2.1.2.4. Integração do Software (ENG.24)**

O propósito da integração do software é combinar as unidades de software produzindo itens integrados consistentes com a especificação do software.

Práticas de base (ENG.24.PB.n)

1. Desenvolver a estratégia de integração do software
2. Desenvolver os testes para os itens integrados
3. Integrar os itens de software
4. Testar os itens integrados
5. Assegurar a consistência com a especificação do software
6. Realizar os testes de regressão dos itens de software
7. Integrar o produto de software

Os produtos de trabalho da integração do software são: itens de software, procedimentos de teste de integração, produto integrado, resultados dos testes e registo de consistência com a especificação do software.

#### **7.2.1.2.5. Teste do Software (ENG.25)**

O propósito do teste do software é confirmar que o produto final cumpre os requisitos definidos.

Práticas de base (ENG.25.PB.n)

1. Desenvolver os testes para o produto integrado
2. Testar o produto integrado

3. Realizar os testes de regressão do produto integrado
4. Preparar o produto para entrega

Os produtos de trabalho resultantes deste processo são: item de configuração, produto testado, relatório de incidente, procedimentos de teste, relatórios do teste e manual do utilizador.

### **7.2.1.3. Manutenção do Software (ENG.3)**

O propósito da manutenção do software é modificar o produto depois da entrega para corrigir falhas, melhorar o desempenho ou adaptá-lo a um ambiente modificado.

Práticas de base (ENG.3.PB.n)

1. Desenvolver a estratégia de manutenção
2. Estabelecer procedimentos de resposta
3. Analisar problemas e alterações do utilizador
4. Implementar e testar modificações
5. Realizar o *upgrade* do software do utilizador
6. Retirar o software obsoleto

Os produtos de trabalho resultantes são: item de configuração, relatório de controlo/acompanhamento da mudança, estratégia de manutenção, resultados dos testes, versão do produto e registo de consistência com os requisitos do software.

### **7.2.2. Processos de Suporte (SUP)**

São os processos utilizados por outros processos. No 2MPspe, esta categoria engloba os processos de documentação, gestão de configurações e garantia da qualidade.

### **7.2.2.1. Documentação (SUP.1)**

O propósito da documentação é desenvolver e manter a informação armazenada sobre o software, produzida por um processo.

#### Práticas de base (SUP.1.PB.n)

1. Estabelecer normas para os documentos
2. Especificar os requisitos dos documentos
3. Identificar os documentos a produzir
4. Desenvolver os documentos
5. Distribuir documentos
6. Manter documentos

Os produtos de trabalho são: relatório de controlo / acompanhamento de mudanças, registo de aceitação, registo de entrega e produto de trabalho.

### **7.2.2.2. Gestão de Configurações (SUP.2)**

O propósito da gestão de configurações é estabelecer e manter a integridade de todos os produtos de trabalho de um processo ou projecto e disponibilizá-los a todos os intervenientes.

#### Práticas de base (SUP.2.PB.n)

1. Identificar itens de configuração
2. Estabelecer versão interna e versão para entrega
3. Manter descrição dos itens de configuração
4. Controlar modificações e versões
5. Manter histórico dos itens de configuração
6. Relatar o estado da configuração
7. Gerir cópias de segurança, armazenamento, arquivo, manuseamento e entrega dos itens

Os produtos de trabalho de saída são: item de configuração, guia de manuseamento e armazenamento, produto integrado e registo do estado do progresso.

### **7.2.2.3. Garantia da Qualidade (SUP.3)**

O propósito da garantia da qualidade é fornecer garantia que os produtos de trabalho e processos cumprem os requisitos especificados e implícitos e aderem aos planos estabelecidos.

Práticas de base (SUP.3.PBn)

1. Desenvolver a estratégia de garantia da qualidade
2. Definir registos de qualidade do projecto
3. Assegurar a qualidade das actividades do processo e dos produtos de trabalho
4. Produzir e armazenar registos das actividades de garantia
5. Resolver problemas

Os produtos de trabalho identificados são: procedimentos de garantia da qualidade, descrição dos processos, normas e critérios de qualidade e relatório de problemas.

### **7.2.3. Processos de Gestão (GES)**

Esta categoria refere-se aos processos que englobam tarefas e actividades genéricas utilizadas por qualquer profissional na gestão dos seus processos. No 2MPspe incluem-se os processos: gestão do projecto, gestão do risco e gestão da qualidade.



### **7.2.3.1. Gestão do Projecto (GES.1)**

O propósito da gestão do projecto é identificar, estabelecer, coordenar e monitorizar as actividades, tarefas e recursos necessários para um projecto produzir o produto, considerando as restrições e requisitos do projecto.

Práticas de base (GES.1.PB.n)

1. Definir o âmbito do trabalho
2. Definir o ciclo de vida do projecto
3. Desenvolver estimativas dos atributos do projecto
4. Definir as actividades do projecto
5. Definir o cronograma do projecto
6. Estabelecer o plano do projecto
7. Implementar o projecto
8. Monitorizar parâmetros do projecto
9. Rever o progresso do projecto
10. Agir para corrigir desvios

Os produtos de trabalho são: estimativas, plano do projecto, modelo do ciclo de vida, medições do projecto e relatório do estado do projecto.

### **7.2.3.2. Gestão do Risco (GES.2)**

O propósito do processo de gestão do risco é identificar, gerir e mitigar de modo contínuo os riscos.

Práticas de base (GES.2.PB.n)

1. Identificar os riscos
2. Analisar os riscos e determinar prioridades
3. Definir e executar acções para reduzir o risco
4. Monitorizar riscos
5. Executar acções correctivas

Os produtos de trabalho a identificar são: registo de controlo do risco, plano de gestão do risco e relatório de análise do risco.

### **7.2.3.3. Gestão da Qualidade (GES.3)**

O propósito da gestão da qualidade é monitorizar a qualidade dos produtos e serviços ao nível do projecto e da organização de modo a assegurar a satisfação do cliente.

Práticas de base (GES.3.PB.n)

1. Estabelecer objectivos de qualidade
2. Estabelecer estratégia global
3. Definir critérios de qualidade
4. Estabelecer um sistema de gestão da qualidade
5. Avaliar até que ponto os objectivos da qualidade são atingidos
6. Executar acções correctivas

Os produtos de trabalho são: objectivos de qualidade, medição de qualidade, plano de qualidade, política de qualidade, registo de revisão, registo de acção correctiva, análise de resultados e critérios de aceitação.

### **7.2.4. Processos Organizacionais (ORG)**

Os processos organizacionais preocupam-se com o ambiente e actividades no nível da organização. No caso do 2MPspe incluem-se: recursos humanos, estabelecimento do processo e suporte ao cliente.

### **7.2.4.1. Processo de Recursos Humanos (ORG.1)**

#### **7.2.4.1.1. Gestão de Recursos Humanos (ORG.11)**

O propósito da Gestão de Recursos Humanos é prover a organização e os projectos com indivíduos que possuam aptidões e conhecimentos para executar os seus papéis efectivamente.

Práticas de base (ORG.11.PB.n)

1. Identificar aptidões e competências necessárias
2. Definir critérios de avaliação
3. Recrutar pessoal qualificado
4. Desenvolver aptidões e competências
5. Avaliar desempenho do pessoal
6. Fornecer *feedback* sobre desempenho
7. Manter registo do pessoal

Os produtos de trabalho são: plano de recursos humanos, avaliação do desempenho do pessoal, plano de formação, registo do pessoal, registo de formação e critérios de desempenho.

#### **7.2.4.1.2. Formação (ORG.12)**

O propósito da formação é desenvolver as aptidões e conhecimento dos indivíduos de modo a que possam executar os seus papéis de modo efectivo.

Práticas de base (ORG.12.PB.n)

1. Identificar necessidades de formação
2. Desenvolver ou adquirir formação
3. Preparar execução da formação

4. Dar formação ao pessoal
5. Manter registo da formação
6. Avaliar efectividade da formação

Os produtos de trabalho identificados são: plano de formação, material de formação e registo de formação

#### **7.2.4.1.3. Gestão do Conhecimento (ORG.13)**

O propósito da gestão do conhecimento é assegurar que o conhecimento, informação e aptidões dos profissionais são coligidas, partilhadas, reutilizadas e melhoradas através da organização.

Práticas de base (ORG.13.PB.n)

1. Estabelecer o sistema de gestão do conhecimento
2. Desenvolver uma estratégia de Gestão do Conhecimento
3. Gerir os *assets* de conhecimento

Os produtos de trabalho são: repositório do conhecimento, estratégia de gestão do conhecimento.

#### **7.2.4.2. Estabelecimento do Processo (ORG.2)**

O propósito do estabelecimento do processo é estabelecer um conjunto de processos organizacionais para todos os processos do ciclo de vida do software.

Práticas de base (ORG.2.PB.n)

1. Definir objectivos dos processos
2. Definir e documentar os processos
3. Identificar expectativas de desempenho

4. Estabelecer orientações para colocar à medida
5. Manter dados sobre o processo
6. Manter descrição dos processos normalizados

Os produtos de trabalho são: descrição do processo, registo de qualidade, registo de revisão, análise de resultados e repositório das descrições dos processos.

#### **7.2.4.3. Suporte ao Cliente (ORG.3)**

O propósito do suporte ao cliente é estabelecer e manter um nível de serviço aceitável, realizando assistência e consultoria ao cliente para suportar a utilização efectiva do produto.

Práticas de base (ORG.3.PB.n)

1. Estabelecer o suporte ao cliente
2. Fornecer formação e documentação aos utilizadores
3. Determinar o nível de satisfação do cliente
4. Comunicar a satisfação do cliente aos profissionais

Os produtos de trabalho resultantes são: dados sobre a satisfação do cliente, relatório de problemas, medição do nível de serviço e registo de formação.

### **7.3. Práticas Genéricas**

Como vimos, os atributos da capacidade (ACs) avaliam a capacidade dos processos e são medidos através da escala de classificação proposta<sup>4</sup>. Para realizar a medição dos ACs utilizam-se os indicadores de práticas genéricas. Definindo-se um indicador como “um atributo objectivo ou característica de uma prática ou produto de

---

<sup>4</sup> ver secção 6.3.3.

trabalho que suporta o julgamento da execução ou capacidade de um processo implementado” [ISO/IEC 2003].

A avaliação da execução do processo (nível 1) é realizada através do AC1.1 e utilizando as indicações das práticas de base e produtos de trabalho. A avaliação da capacidade do processo (nível 1.5 a nível 3) é realizada medindo os ACs de cada nível, com apoio dos indicadores de práticas genéricas.

Para a identificação das práticas genéricas utiliza-se a seguinte sequência: ac.PG.np, com:

- ac, número do atributo da capacidade
- PG, texto (Prática Genérica)
- np, número da prática genérica (dentro do mesmo AC)

Por exemplo, 3.2PG.3, refere-se à prática genérica número 3 referente ao atributo da capacidade AC3.2<sup>5</sup>.

### **7.3.1. Nível 1: Processo Executado**

O nível 1 possui um atributo da capacidade, AC1.1, denominado atributo da execução do processo. Este AC é avaliado através das práticas de base e produtos do trabalho específicos para cada processo. Pretende-se verificar que os resultados do processo são atingidos, através da execução das práticas de base e produção dos produtos de trabalho identificados.

---

<sup>5</sup> segundo atributo da capacidade do nível de capacidade 3 (atributo de transferência do processo).

### 7.3.2. Nível 1.5: Processo Valorizado

O processo é executado e verifica-se também o reconhecimento da importância da sua execução. Este nível é avaliado com o atributo da capacidade AC1.51, denominado atributo de valorização do processo. O AC possui duas práticas genéricas:

1.51.PG.1 – *Identificar os intervenientes do processo*, não só os intervenientes directos (executores), mas também responsáveis, consultores e outros intervenientes indirectos

1.51.PG.2 – *Reconhecer a importância da execução*, todos os intervenientes estão conscientes da contribuição do processo para os objectivos do negócio

### 7.3.3. Nível 2: Processo Gerido

Neste nível, o processo valorizado está agora implementado de uma forma gerida e os produtos de trabalho estão estabelecidos, controlados e mantidos de forma adequada. Este nível possui dois ACs:

- AC2.1 – *Atributo de gestão da execução*. Mede a extensão com que o processo é gerido de modo a produzir produtos de trabalho que cumprem os objectivos definidos. Para medir este AC utilizam-se cinco indicadores de práticas genéricas:

2.1.PG.1 – *Identificar os objectivos da execução*. Devem incluir: qualidade dos artefactos; tempo de ciclo e utilização de recursos

2.1.PG.2 – *Planear a execução do processo*, incluindo atribuição de responsabilidades, desenvolvimento do plano, definição de prazos e planeamento de revisões

- 2.1.PG.3 – *Monitorizar e controlar a execução do processo*, incluindo execução de acções quando os resultados não são atingidos, ajustamentos no plano e no cronograma, caso sejam necessários
- 2.1.PG.4 – *Atribuir e utilizar recursos*, incluindo identificação e disponibilização de recursos e disponibilização da informação necessária para executar o processo
- 2.1.PG.5 – *Gestão dos interfaces entre os intervenientes*. Os intervenientes foram identificados no nível 1.5 e é necessário atribuir responsabilidades e gerir as ligações
- AC2.2 – *Atributo de gestão do produto de trabalho*. Mede até que ponto o processo é gerido para produzir produtos de trabalho documentados, controlados e verificados de forma adequada. Este AC é caracterizado com apoio de quatro indicadores de práticas genéricas:
    - 2.2.PG.1 – *Definir os requisitos dos produtos de trabalho*, incluindo critérios de qualidade e critérios para aprovação e revisão
    - 2.2.PG.2 – *Definir requisitos de documentação e controlo*, incluindo a identificação e definição das dependências entre os produtos de trabalho
    - 2.2.PG.3 – *Identificar, documentar e controlar os produtos de trabalho*. Realizar estas actividades de acordo com os requisitos, incluindo a disponibilização do produto de trabalho através de mecanismos de acesso adequados
    - 2.2.PG.4 – *Rever e ajustar os produtos de trabalho*. Quando necessário, realizar estas actividades de modo a que os produtos de trabalho cumpram os requisitos definidos



### 7.3.4. Nível 3: Processo Estabelecido

O processo neste nível é executado a partir de uma versão do processo normalizado. Como vimos, o processo tem a capacidade de atingir os resultados esperados uma vez que o processo normalizado utiliza os princípios de engenharia e são atribuídos os recursos com as competências necessárias para atingir o desempenho esperado. O nível estabelecido possui dois atributos da capacidade:

- AC3.1 – *Atributo de definição do processo*. Mede até que ponto o processo normalizado é mantido de modo a suportar a transferência efectiva do processo definido<sup>6</sup>. Este AC é caracterizado através de cinco indicadores de práticas genéricas:
  - 3.1.PG.1 – *Definir o processo normalizado*, incluindo as necessidades e contexto para a transferência, orientações para colocar à medida o processo normalizado e procedimentos para suportar a implementação do processo
  - 3.1.PG.2 – *Determinar a sequência e interação entre os processos*
  - 3.1.PG.3 – *Identificar autoridade, papéis, responsabilidades e competências*, necessárias para a execução do processo
  - 3.1.PG.4 – *Identificar os requisitos para a infra-estrutura e para o ambiente de trabalho*, incluindo métodos, ferramentas, hardware, redes e outros
  - 3.1.PG.5 – *Determinar métodos adaptados para monitorizar o processo*, incluindo os critérios para aferir a efectividade do processo e os dados necessários para monitorizar os critérios
- AC3.2 – *Atributo de transferência do processo*. Mede a extensão com que o processo normalizado é efectivamente transferido como um processo

---

<sup>6</sup> o processo definido é uma instância do processo normalizado para um determinado projecto.

definido, permitindo atingir os resultados do processo. Para avaliar o AC3.2 utilizam-se cinco indicadores de práticas genéricas:

- 3.2.PG.1 – *Transferir um processo definido*, através da colocação à medida do processo normalizado
- 3.2.PG.2 – *Atribuir recursos e disponibilizar informação*, de modo a possibilitar a transferência do processo definido. Devem ser atribuídos e comunicados os papéis, responsabilidades e autoridade
- 3.2.PG.3 - *Assegurar as competências necessárias*, incluindo a identificação das competências apropriadas e a disponibilização da formação adequada
- 3.2.PG.4 – *Fornecer a infra-estrutura adequada*, incluindo a disponibilização, utilização e manutenção da infra-estrutura
- 3.2.PG.5 – *Coligir e analisar dados sobre o desempenho do processo*, permitindo compreender o comportamento e efectividade do processo definido

#### **7.4. Práticas Transversais**

Os atributos do conhecimento representam o nível de conhecimento dos factores contextuais dos processos. Para avaliar de uma forma objectiva os AKs utilizam-se os indicadores de práticas transversais.

Como vimos os AKs medem o nível de conhecimento explícito sobre os factores contextuais que afectam os processos. O 2MPspe propõe uma escala com dois níveis: o nível 0, designado como nível implícito e o nível 1, denominado nível explícito.

Para a identificação das práticas transversais utiliza-se a seguinte sequência: ak.PT.np, com:

- ak, número do atributo do conhecimento

- PT, texto (Prática Transversal)
- np, número da prática transversal (dentro do mesmo AK)

Por exemplo, 1.2.PT.2, indica a prática transversal nº 2 do atributo do conhecimento 1.2 (atributo de gestão da cultura organizacional).

#### **7.4.1. AK1.1 – Atributo de Gestão do Conhecimento**

O AK1.1 mede a extensão com que a abordagem à gestão do conhecimento no processo está explicitamente identificada e aplicada. Para caracterizar objectivamente o AK1.1 utilizam-se quatro práticas transversais:

1.1.PT.1 – *Identificar o perfil de gestão do conhecimento*, no nível dos indivíduos, das equipas e específico do processo.

1.1.PT.2 – *Documentar o impacto dos perfis no processo*, incluindo os interfaces entre os diversos perfis.

1.1.PT.3 – *Alinhar os perfis de gestão do conhecimento*.

1.1.PT.4 – *Fornecer a infra-estrutura adequada*, incluindo a disponibilização, implementação e manutenção da infra-estrutura adequada para o tipo de conhecimento identificado.

#### **7.4.2. AK1.2 – Atributo de Gestão da Cultura Organizacional**

O AK1.2 mede até que ponto o tipo de cultura organizacional no processo está explicitamente identificada e aplicada. Este atributo do conhecimento é caracterizado através de três indicadores de práticas transversais.

1.2.PT.1 – *Identificar o tipo de cultura organizacional predominante.*

1.2.PT.2 – *Documentar o impacto da cultura organizacional no processo.*

1.2.PT.3 – *Definir e manter um sistema de acompanhamento da dinâmica da cultura organizacional.*

Refira-se que, no contexto das PES, os atributos de conhecimento dos factores de contexto serão normalmente avaliados no nível da organização global. O avaliador externo, ao aplicar os questionários, deverá usar a sua experiência para medir os AKs em determinados processos específicos. Note-se que, em organizações com poucos recursos humanos, a cultura organizacional predominante é a mesma em todos os processos. Refira-se também que o foco do AK1.2 é o balanceamento sempre complexo entre criatividade e disciplina ou flexibilidade e rigidez nos processos.

## **7.5. Aplicação do $M_a2MP$**

Como vimos, o  $M_c2MP$ <sup>7</sup> apresenta um referencial tridimensional que é a base do modelo de melhoria  $2MP_{spe}$ . As três dimensões referem-se a:

- (1) Processos. Conjunto de processos considerados, seguindo o  $MRP_{2MP_{spe}}$
- (2) Capacidade. Níveis de capacidade dos processos e respectivos atributos da capacidade
- (3) Conhecimento. Níveis de conhecimento dos factores contextuais e respectivos atributos do conhecimento

O modelo de avaliação dos processos  $M_a2MP$ , apresentado neste capítulo, define os indicadores a utilizar de modo a caracterizar os processos em relação à sua capacidade e em relação ao conhecimento dos factores de contexto. Definiram-se três tipos de indicadores de práticas: práticas de base, utilizadas para avaliar se o processo é executado e identificadas como “nº capacidade.PB.nº prática”; práticas genéricas, utilizadas para caracterizar a capacidade do processo e designadas como, “nº atributo

---

<sup>7</sup> ver capítulo 6.

processo.PG.nº prática” e práticas transversais, utilizadas para avaliar o conhecimento dos factores do contexto do processo e identificadas como, “nº atributo conhecimento.PT.nº prática”.

Um modelo de avaliação do processo, segundo a norma ISO/IEC 15504, é a base para coligir evidências e classificar a capacidade do processo. No caso do 2MPspe, é também a base para classificar o conhecimento dos factores do contexto do processo. Para cumprir os requisitos de conformidade especificados na parte 2 da norma ISO/IEC 15504, definiu-se o propósito, âmbito e elementos do  $M_a2MP$ , desenvolveu-se a correspondência para o  $MRP_{2MPspe}$  e para o referencial de medição. É necessário ainda definir um mecanismo para representar os resultados. A representação dos resultados no 2MPspe é proposta na secção 7.5.2.

A ferramenta principal do  $M_a2MP$  é o questionário do 2MPspe<sup>8</sup>, que utiliza os indicadores das práticas para medir a capacidade e o conhecimento dos processos. O questionário é descrito na próxima secção e os detalhes da sua aplicação prática são apresentados no contexto do método de implementação  $M_i2MP$ . Refira-se que o 2MPspe é um modelo de melhoria do processo de software, concebido com a preocupação central de servir as PES, mas o questionário desenvolvido pode ser utilizado também no contexto duma iniciativa de determinação da capacidade do processo. A figura 7.2 mostra os relacionamentos entre a avaliação dos processos e as iniciativas de melhoria do processo e determinação da capacidade do processo.

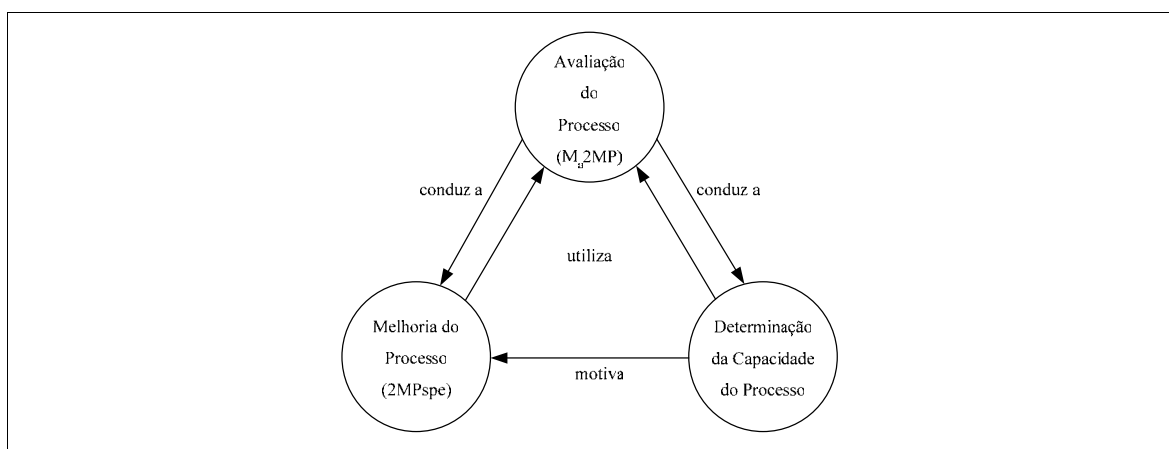


Figura 7.2 – Contexto da determinação da capacidade do processo (adaptado de [ISO/IEC 2004a]).

<sup>8</sup> ver anexo C.

A aplicação do questionário no contexto duma iniciativa de melhoria baseada no 2MPspe, permite caracterizar as práticas actuais da organização em termos da capacidade e conhecimento dos processos. A análise dos resultados permitirá identificar pontos fortes, pontos fracos e riscos inerentes aos processos e assim definir prioridades de melhoria.

A determinação da capacidade do processo preocupa-se com a análise comparativa da capacidade dos processos, em relação a um perfil de capacidade alvo. Esta análise pretende identificar os riscos ligados à execução de um projecto utilizando os processos seleccionados [ISO/IEC 2004a]. Embora não seja o propósito específico, é viável utilizar o M<sub>a</sub>2MP para determinar a capacidade e/ou conhecimento do processo.

### 7.5.1. Questionário do 2MPspe

O questionário do 2MPspe foi concebido para ser respondido sem apoio externo<sup>9</sup>. Nesta perspectiva, é aplicado parcialmente a especialistas internos das áreas inquiridas. O objectivo da aplicação do questionário é recolher dados sobre a execução, capacidade e conhecimento dos processos. Estes dados serão analisados e verificados por um “profissional externo” no âmbito do M<sub>i</sub>2MP.

O questionário 2MPspe<sup>10</sup> divide-se em quatro secções e um apêndice:

- i) - Introdução
  - ii) - Orientações para o preenchimento
  - iii) - Descrição dos processos
  - iv) - Questões sobre os processos
- Apêndice: Indicadores dos atributos

A parte inicial recolhe informação sobre o entrevistado, incluindo: nome, idade, cargo na empresa, anos na empresa, habilitações, projecto actual, responsabilidades no projecto e data da entrevista. A secção i) esclarece os objectivos do questionário e

---

<sup>9</sup> após a reunião de apresentação da iniciativa (ver secção 8.2.2)

<sup>10</sup> ver anexo C.

enquadra-o no contexto da MPS. A secção ii) apresenta o modo como as questões devem ser respondidas e define as escalas de classificações utilizadas para as práticas. Na secção iii) apresenta-se uma breve explicação de cada processo. Na secção iv) encontram-se as questões sobre cada um dos processos divididas em práticas de base (actividades executadas), documentos a identificar e atributos do processo. Finalmente, no apêndice, apresenta-se uma descrição dos indicadores das práticas genéricas e das práticas transversais. Estes indicadores servem de apoio para as questões relacionadas com a capacidade e conhecimento dos processos.

O questionário completo segue a estrutura do  $MRP_{2MPspe}$ , englobando 4 categorias de processos, 18 processos primários e 244 questões. A tabela 7.1 mostra a distribuição das questões pelas categorias de processos e pelas práticas.

<b>Categoria de processos</b>	<b>ENG</b>	<b>SUP</b>	<b>GES</b>	<b>ORG</b>	<b>4</b>
<b>Número de processos</b>	7	3	3	5	18
<b>Questões</b>					<b>TOTAL</b>
<b>PB</b>	35	18	21	26	100
<b>PG</b>	42	18	18	30	108
<b>PT</b>	14	6	6	10	36
<b>TOTAL</b>	91	42	45	66	244

*Tabela 7.1 – Detalhe do questionário 2MPspe.*

Embora sendo um número elevado de questões, os inquiridos apenas irão responder a um conjunto restrito de perguntas relacionadas com a sua área de trabalho. Assim, por exemplo, um engenheiro de software a trabalhar nas áreas de concepção, construção e integração irá responder a 39 questões<sup>11</sup>.

As perguntas sobre as práticas de base são diferentes para todos os processos, uma vez que representam actividades específicas de cada processo. Em relação às questões sobre as práticas genéricas e transversais, refira-se que são iguais para todos os processos, mudando apenas o nome do processo. Temos então,

---

<sup>11</sup> 15 questões sobre as PBs da concepção, construção e integração e 24 questões sobre as PGs e PTs dos três processos.

**Práticas genéricas:**

AC1.1 – Atributo de execução do processo.

Questão: O (A) «nome do processo» utiliza as práticas de base adequadas e resulta nos produtos de trabalho identificados?

AC1.5 – Atributo de valorização do processo

Questão: Os intervenientes do processo «nome do processo» estão identificados e reconhecem a importância da sua execução?

AC2.1 – Atributo de gestão da execução.

Questão: O (A) «nome do processo» é gerido de forma a produzir os recursos de trabalho com os recursos previstos e dentro do prazo?

AC2.2 – Atributo de gestão do produto de trabalho.

Questão: O (A) «nome do processo» é gerido para produzir produtos de trabalho documentados, controlados e verificados, em linha com os objectivos de qualidade definidos?

AC3.1 – Atributo de definição do processo.

Questão: O (A) «nome do processo» é baseado num processo normalizado?

AC3.2 – Atributo de transferência do processo.

Questão: O (A) «nome do processo» utiliza Recursos Humanos adaptados e uma infraestrutura do processo adequada, de modo a permitir a transferência efectiva como um processo definido?

A resposta à questão sobre AC1.1 baseia-se na avaliação das práticas de base e na identificação dos produtos de trabalho. As respostas às restantes questões baseiam-se nos indicadores de práticas genéricas<sup>12</sup>.

**Práticas transversais:**

AK1.1 – Atributo de gestão do conhecimento.

Questão: O (A) «nome do processo» utiliza uma abordagem à gestão do conhecimento explicitamente identificada e aplicada?

AK1.2 – Atributo de gestão da cultura organizacional.

Questão: O tipo de cultura organizacional predominante no processo «nome do processo» está explicitamente identificada e aplicada?

---

<sup>12</sup> ver anexo C (questionário 2MPspe).



As respostas às questões sobre os atributos do conhecimento dos factores do contexto têm como base os indicadores de práticas transversais.

A aplicação do questionário nas PES implica três níveis de configurações:

- (1) Relacionadas com o reduzido número de profissionais. Poderá permitir apenas uma questão para definir a cultura organizacional predominante (AK1.2) dos processos. Poderá permitir também que o indicador 1.1.PT.1, referente ao perfil de gestão de conhecimento das equipas possa ser analisado em termos globais, com apoio de uma única questão.
- (2) Relacionadas com a área específica de cada entrevistado. Permite a redução do número de processos<sup>13</sup> em cada questionário individual.
- (3) Relacionadas com a validação da objectividade das questões sobre as práticas de base em relação ao contexto específico da organização alvo. Para as questões potencialmente subjectivas serão definidos critérios objectivos de validação de modo a adaptar o questionário às práticas da empresa. Por exemplo, o teste de software (ENG.25.PB.2) não necessitaria de critérios auxiliares devido à sua objectividade inerente, ou seja, basta verificar o caderno de testes envolvido. No caso oposto, a prática de base GES.1.PB.1, "Âmbito definido", no caso de uma empresa que siga a norma PMI, o documento "*Scope Statement*" e a WBS provam a definição do âmbito. Noutra empresa um documento em prosa validado e assinado pelo gestor do projecto e cliente pode ser suficiente.

O questionário é a ferramenta central do 2MPspe, uma vez que permite a recolha de evidências sobre a capacidade e conhecimento dos processos. A definição do plano de melhoria terá como base os dados recolhidos através desta ferramenta. Nesta perspectiva procedeu-se à validação do questionário do 2MPspe. Esta validação foi realizada com o apoio de um painel de experts<sup>14</sup> e utilizando o método experimental designado "experiência em ambiente sintético". O processo de validação teve duas iterações e resultou na 3ª versão do questionário 2MPspe.

---

<sup>13</sup> com a conseqüente redução do número de questões.

<sup>14</sup> estratégia também utilizada nos trabalhos de El Emam e Madhavji [1996] e Dyba [2000] na área da MPS.

A primeira iteração foi realizada por dois especialistas do meio académico numa perspectiva de validar o conteúdo do questionário. A 2ª iteração incluiu três gestores de projectos de pequenas empresas de desenvolvimento de software. Foi solicitado a este painel o preenchimento do questionário e foi avaliado um conjunto de variáveis. As variáveis controladas foram o tempo, o nº de questões levantadas e a comparação entre os resultados obtidos e a percepção do investigador. A avaliação destas variáveis originou a reformulação de algumas questões.

### 7.5.2. Representação dos Resultados

Para cumprir a conformidade com a norma ISO/IEC 15504-2, é necessário definir um mecanismo para representar os resultados da avaliação. Os requisitos de conformidade referem que o mecanismo deve ser formal e verificável.

Para a representação dos resultados propõem-se dois níveis de detalhe e três níveis de abstracção organizacional. Os níveis de detalhe são:

- 1) Perspectiva global. Onde apenas são apresentados os níveis de capacidade e os níveis de conhecimento para cada elemento<sup>15</sup> (ver exemplo da figura 7.3);

Conhecimento		Processo	Capacidade			
1	0,5		1	1,5	2	3
		ENG.1				
		ENG.21				
		⋮				
		ORG.3				

Figura 7.3 – Representação dos resultados na perspectiva global.

<sup>15</sup> poderá ser processos, categoria de processos ou organização.

- 2) Perspectiva detalhada. Onde são apresentadas as classificações atingidas nos atributos de capacidade e nos atributos de conhecimento e a percentagem de práticas de base executadas (ver exemplo na figura 7.4)

AK1.2	AK1.1	Processo	PB	AC1.1	AC1.51	AC2.1	AC2.2	AC3.1	AC3.2
L	C	ENG.1	90%	C	C	C	L	P	P
P	L	ENG.21	60%	C	L	L	P	N	N
		⋮							
P	N	ORG.3	40%	L	L	P	N	N	N

Legenda:

- C** Completamente atingido
- L** Largamente atingido
- P** Parcialmente atingido
- N** Não atingido

Figura 7.4 – Representação dos resultados na perspectiva detalhada.

Os níveis de abstracção organizacional são obtidos substituindo a coluna processos – ver figuras 7.3 e 7.4 – por categoria de processos ou organização. Os níveis de abstracção propostos são:

- a) Perspectiva da organização. Representação da maturidade da organização, utilizando a perspectiva global ou detalhada. Refira-se que a maturidade de uma organização é dada pelo nível de capacidade e pelo nível de conhecimento dos factores de contexto. A maturidade da organização pode ser representada pelo par (nível capacidade, nível conhecimento) ou (C, K).
- b) Perspectiva da categoria de processos. Representação do perfil de maturidade das categorias de processos.

- c) Perspectiva do processo. Representação do perfil de maturidade dos processos. Esta representação pode ser realizada utilizando a perspectiva detalhada ou a perspectiva global.

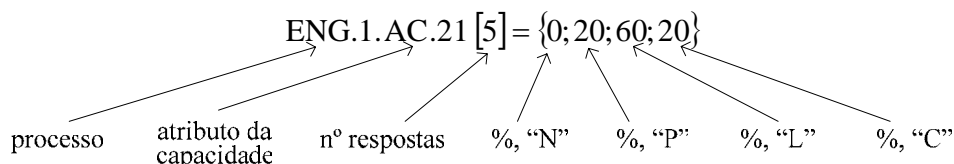
Na representação dos resultados, é possível utilizar qualquer combinação entre os níveis de detalhe e os níveis organizacionais. Verifica-se que as representações mais comuns são:

- A perspectiva global associada à perspectiva da organização ou das categorias dos processos
- A perspectiva detalhada associada à perspectiva dos processos

Em relação ao tratamento dos dados recolhidos a partir dos questionários, são utilizadas três regras para calcular as classificações:

(Regra 1) O valor PB de um processo, representa a percentagem de práticas de base que são executadas. Este valor é calculado através da média aritmética dos valores recolhidos nos inquéritos.

(Regra 2) A classificação dos ACs e AKs é atribuída pelo avaliador externo, tendo como base os dados dos inquiridos e as suas percepções sobre os indicadores das práticas. Para a representação das classificações calculadas a partir dos questionários, utiliza-se a notação proposta pelo SPICE [1998]. Por exemplo, se tivéssemos 5 respostas para o AC2.1 do processo ENG.1, com uma resposta “P”, três respostas “L” e uma resposta “C”, a representação seria:



A classificação final deste AC dependeria da avaliação no local realizada pelo avaliador externo, tomando em consideração as opiniões expressas nos questionários.

(Regra 3) As classificações dos ACs e dos AKs para as categorias de processos e para a organização são determinadas atribuindo o menor valor do conjunto origem. Por exemplo, a partir do perfil de maturidade dos processos: (SUP.1; 2; 0), (SUP.2; 1; 0,5) e (SUP.3; 2; 0), calcula-se o valor da maturidade da categoria de processos de suporte, (SUP; 1; 0).

Neste capítulo apresentou-se o modelo de avaliação do 2MPspe, denominado  $M_a2MP$ . Definiram-se as componentes do modelo, ou seja, práticas de base, práticas genéricas e práticas transversais e partindo destes indicadores dos processos elaborou-se o questionário 2MPspe. Como foi referido, o  $M_a2MP$  está conforme a norma ISO/IEC 15504. O próximo capítulo apresenta o método de implementação do 2MPspe, designado  $M_i2MP$ .

## Capítulo 8

---

### 8. Método de Implementação do 2MPspe (M<sub>i</sub>2MP)

Neste capítulo apresenta-se o método de implementação do 2MPspe, designado M<sub>i</sub>2MP. Este método apresenta as actividades que irão conduzir uma iniciativa de melhoria do processo de software. Na concepção do M<sub>i</sub>2MP foram considerados alguns aspectos prioritários<sup>1</sup>: adaptação às PES; ciclos de melhoria rápidos; simples e alinhado com os objectivos do negócio.

#### 8.1. Introdução

Como foi referido (ver figura 7.2 do capítulo anterior), uma das componentes principais de um modelo de melhoria é o processo de avaliação. Nesta perspectiva, é necessário cumprir os requisitos apresentados na parte normativa da ISO/IEC 15504, sobre as actividades obrigatórias num processo de avaliação. A figura 8.1 apresenta as actividades, entradas, saídas e papéis e responsabilidades propostos pela parte 2 da norma ISO/IEC 15504.

---

<sup>1</sup> derivados dos objectivos do 2MPspe.

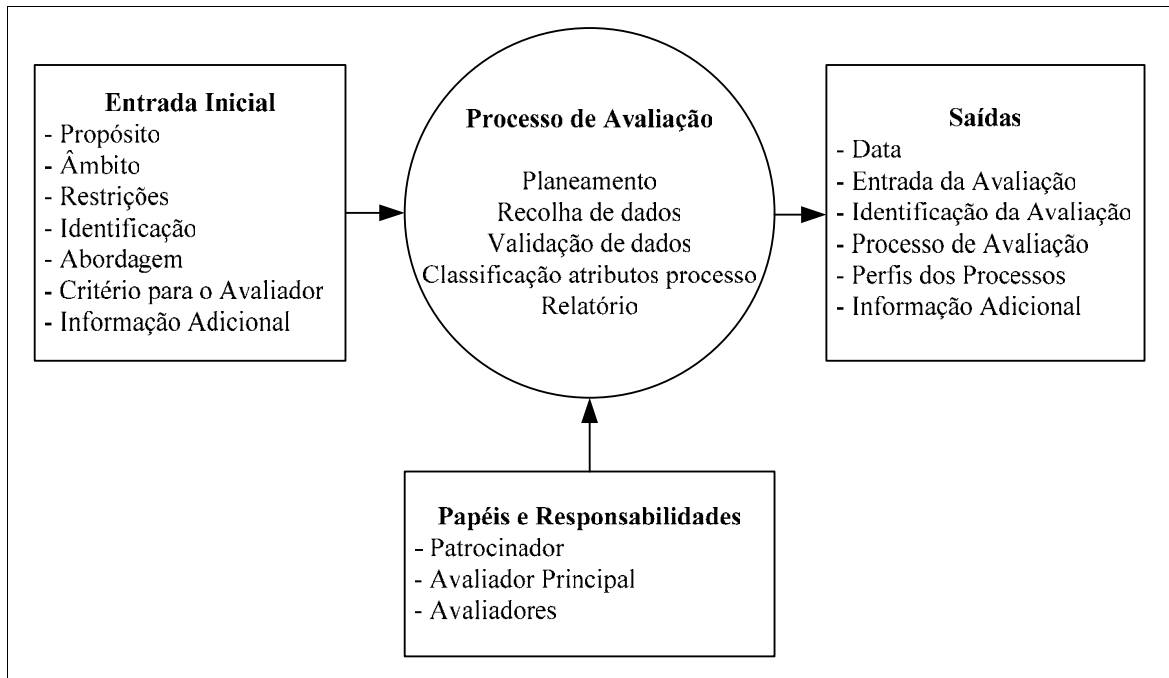


Figura 8.1 – Actividades no processo de avaliação (adaptado de ISO/IEC [2004b]).

A parte 2 refere que um processo de avaliação documentado deve englobar, no mínimo, as seguintes actividades:

- a) Planeamento. Incluindo as entradas necessárias, as actividades a executar, os recursos e o cronograma, a identificação e responsabilidades dos participantes, critérios de conformidade e as saídas planeadas.
- b) Recolha de dados. Os dados necessários para avaliar os processos devem ser recolhidos de uma forma sistemática.
- c) Validação dos dados. Os dados recolhidos devem ser validados para confirmar a objectividade da evidência, assegurar que os dados são suficientes e representativos e assegurar a consistência dos dados.
- d) Classificação dos atributos do processo. Com base nos dados validados deve ser atribuída uma classificação a cada atributo do processo. No caso do 2MPspe os atributos do processo serão os atributos da capacidade e os atributos do conhecimento.

- e) Elaboração do relatório. Os resultados da avaliação devem ser documentados e comunicados ao patrocinador.

No anexo D encontra-se o texto completo com os requisitos de conformidade tal como definidos pela norma ISO/IEC 15504, para o processo de avaliação.

No contexto da melhoria do processo, a avaliação do processo fornece um meio para caracterizar o estado actual da organização em termos da maturidade dos processos avaliados. O  $M_i2MP$  surge da orientação expressa na parte 4 da ISO/IEC 15504 que afirma que a organização deve utilizar “um método adaptado – definindo papéis, técnicas e actividades específicas apropriadas – para implementar” a melhoria do processo.

Em termos globais, o método de implementação do 2MPspe define as actividades, técnicas, ferramentas e papéis a utilizar numa iniciativa de melhoria do processo de software (ver figura 8.2).

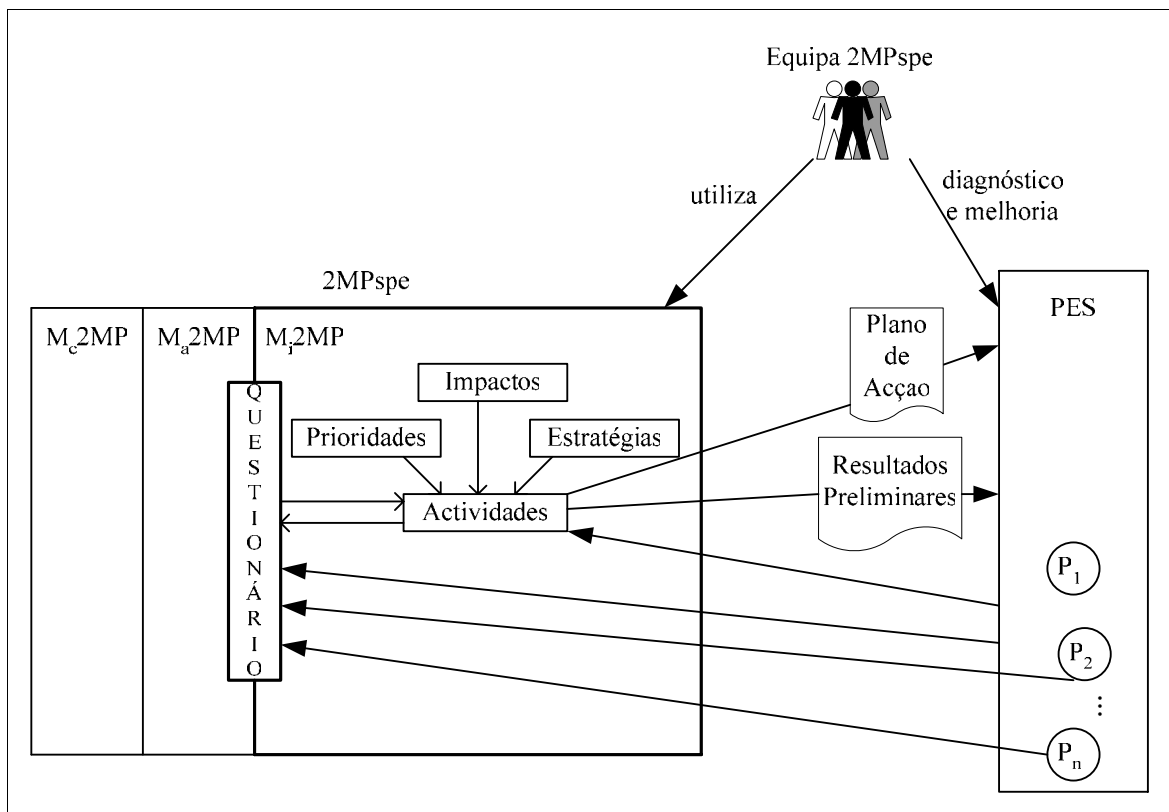


Figura 8.2 – Enquadramento do  $M_i2MP$ .



Zahran [1998] propõe as seguintes fases genéricas para o processo de melhoria (ver figura 8.3): Pré-planeamento, ciclo de avaliação, criação do plano de acção, atribuição de responsabilidades, implementação e institucionalização.

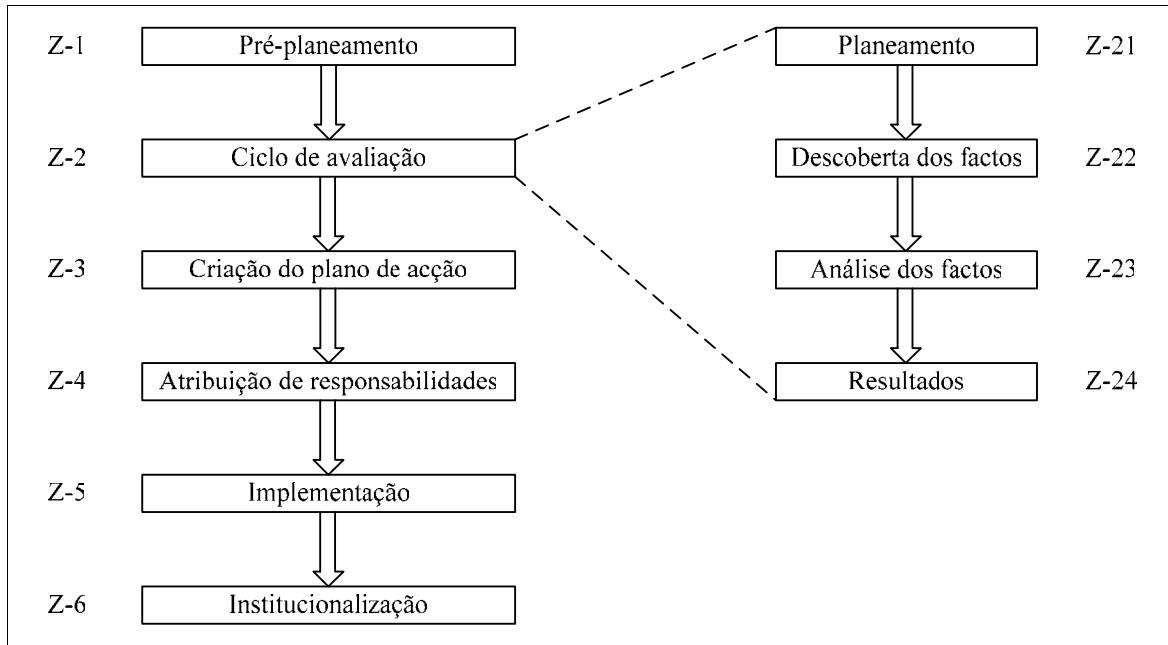


Figura 8.3 – Fases genéricas do processo de avaliação (adaptado de Zahran [1998]).

A fase de pré-planeamento (Z-1) inclui a investigação das necessidades de negócio da organização, o estudo de viabilidade da melhoria, definição do âmbito e fronteiras da melhoria e a definição do patrocinador. A fase de avaliação (Z-2) consiste na execução do diagnóstico e corresponde às fases propostas na parte 2 da ISO/IEC 15504<sup>2</sup>. A fase de criação do plano de acção (Z-3) corresponde à utilização das recomendações resultantes da avaliação para desenvolver o plano de acção da MPS. A fase Z-4 baseia-se no plano de acção para atribuir novos papéis e responsabilidade dentro da organização, necessários para implementar as melhorias. A implementação (Z-5) consiste na execução das tarefas de melhoria, que podem envolver a re-concepção do processo, a concepção de novos processos ou a melhoria do ambiente do processo. Finalmente, a fase Z-6 (Institucionalização) definida por Paulk et al. [1995] como “a construção da infra-estrutura e cultura que suporta métodos, práticas e procedimentos de modo que seja a forma natural de fazer negócio, mesmo depois daqueles que o definiram originalmente, se retirarem”.

<sup>2</sup> ver início desta secção.

Verifica-se que os modelos de melhoria do processo de software cobrem de forma detalhada as fases Z-1, Z-2 e Z-3 onde a equipa externa tem o papel principal. As fases Z-4, Z-5 e Z-6 são executadas pela organização alvo e o modelo de melhoria fornece orientações para a condução.

Os modelos de melhoria do processo de software<sup>3</sup> incluem sempre um método de implementação, englobando todas ou apenas algumas das fases genéricas propostas por Zahran [1998]. Apresenta-se seguidamente uma breve descrição dos métodos de implementação mais divulgados, nomeadamente: IDEAL [McFeely 1995], Tapistry [Kuvaja et al. 1999], Rapid [Rout et al. 2000], CBA-IP<sup>4</sup> [Dunaway e Masters 2001] e ISO/IEC 15504 [ISO/IEC 2004d].

O IDEAL é o acrónimo para (I) Início (D) Diagnóstico (E) Estabelecimento (A) Agir (L) *Levering* e mostra as actividades envolvidas numa iniciativa de melhoria do processo de software. Na figura 8.4 apresenta-se o modelo IDEAL e faz-se a correspondência com as fases genéricas propostas por Zahran.

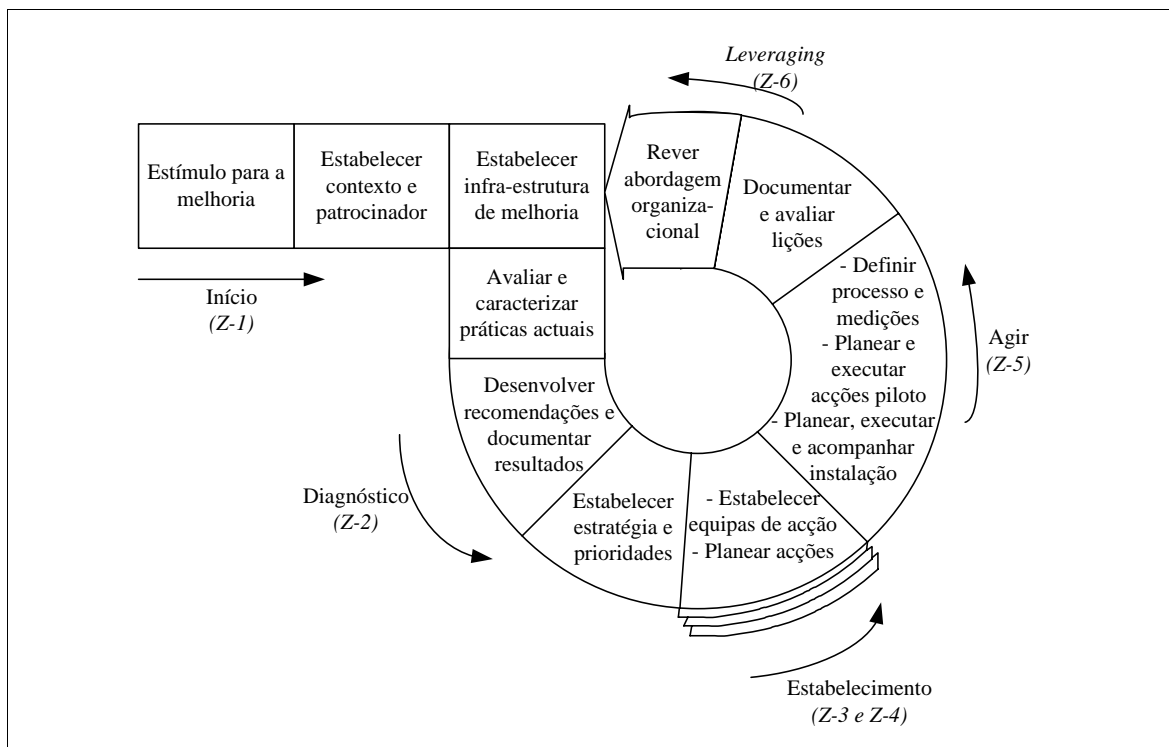


Figura 8.4 – Abordagem IDEAL.

<sup>3</sup> ver secção 3.3.

<sup>4</sup> CMM – Based Appraisal for Internal Process Improvement.

O Tapistry baseia-se num *workshop* de 2 dias realizado na organização, para suportar a auto-avaliação com o BootCheck<sup>5</sup>. Durante o *workshop* os participantes internos são acompanhados na avaliação dos processos, na análise dos resultados e no desenvolvimento do plano de melhoria (ver figura 8.6).

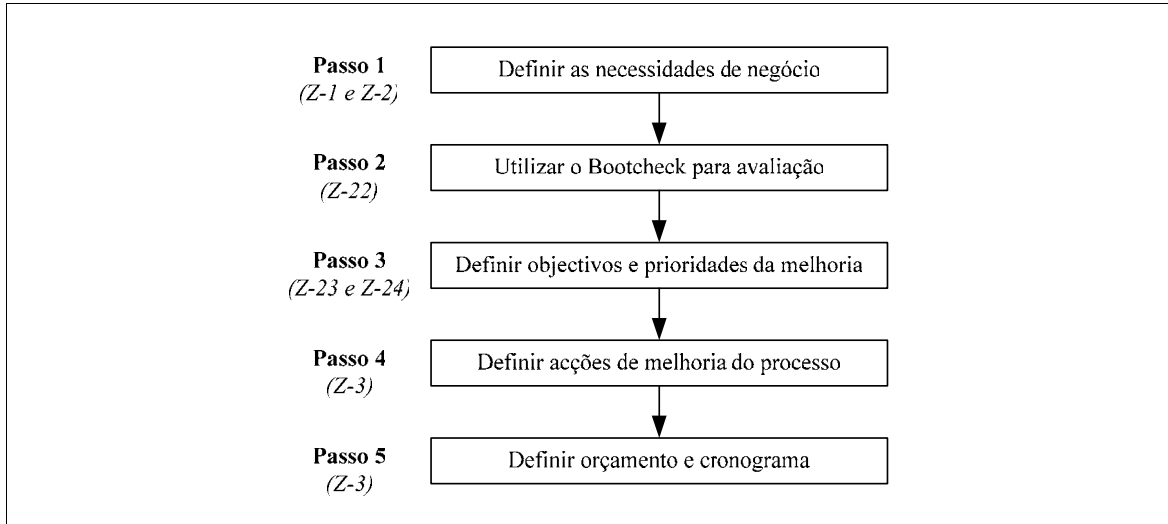


Figura 8.5 – Método de implementação do Tapistry.

Segundo Rout et al. [2000], o Rapid necessita apenas de um dia nas instalações da organização alvo, baseia-se numa discussão conduzida por um avaliador externo e tem como suporte um questionário. As tarefas propostas pelo Rapid para executar o diagnóstico encontram-se na tabela 8.1.

Tarefa	Pessoas	Tempo
Preparar e enviar questionário demográfico (Z-1)	Líder da equipa	15 min.
Preencher questionário demográfico (Z-1)	Patrocinador	15 min.
Preparar plano de avaliação (Z-21)	Líder da equipa	30 min.
Preparar instrumento para avaliação (Z-21)	Líder da equipa	30 min.
Conduzir avaliação Rapid (Z-22)	Líder da equipa Avaliador de suporte Participantes	8 h. 8 h. 8 h.
Preparar relatório (Z-23 e Z-24)	Líder da equipa Avaliador de suporte	6 h. 4 h.

Tabela 8.1 – Esforço de uma avaliação Rapid.

<sup>5</sup> ferramenta informática baseada no Bootstrap.

O CBA-IPI propõe três fases principais: (1) preparar e planejar avaliação; (2) conduzir a avaliação e (3) relatar resultados. Cada fase possui diversas actividades (ver figura 8.5). Os relatos sobre 260 avaliações mostram que 80% das iniciativas precisaram entre 5 e 10 dias no local e a moda foi 5 dias. Em relação ao intervalo temporal típico para executar as tarefas, o método apresenta: Planeamento e formação (mês 1-2); Avaliação no local (mês 3); Entrega do relatório final (mês 4); Desenvolvimento de plano de acção (mês 5) e Implementação (mês 6 - 24).

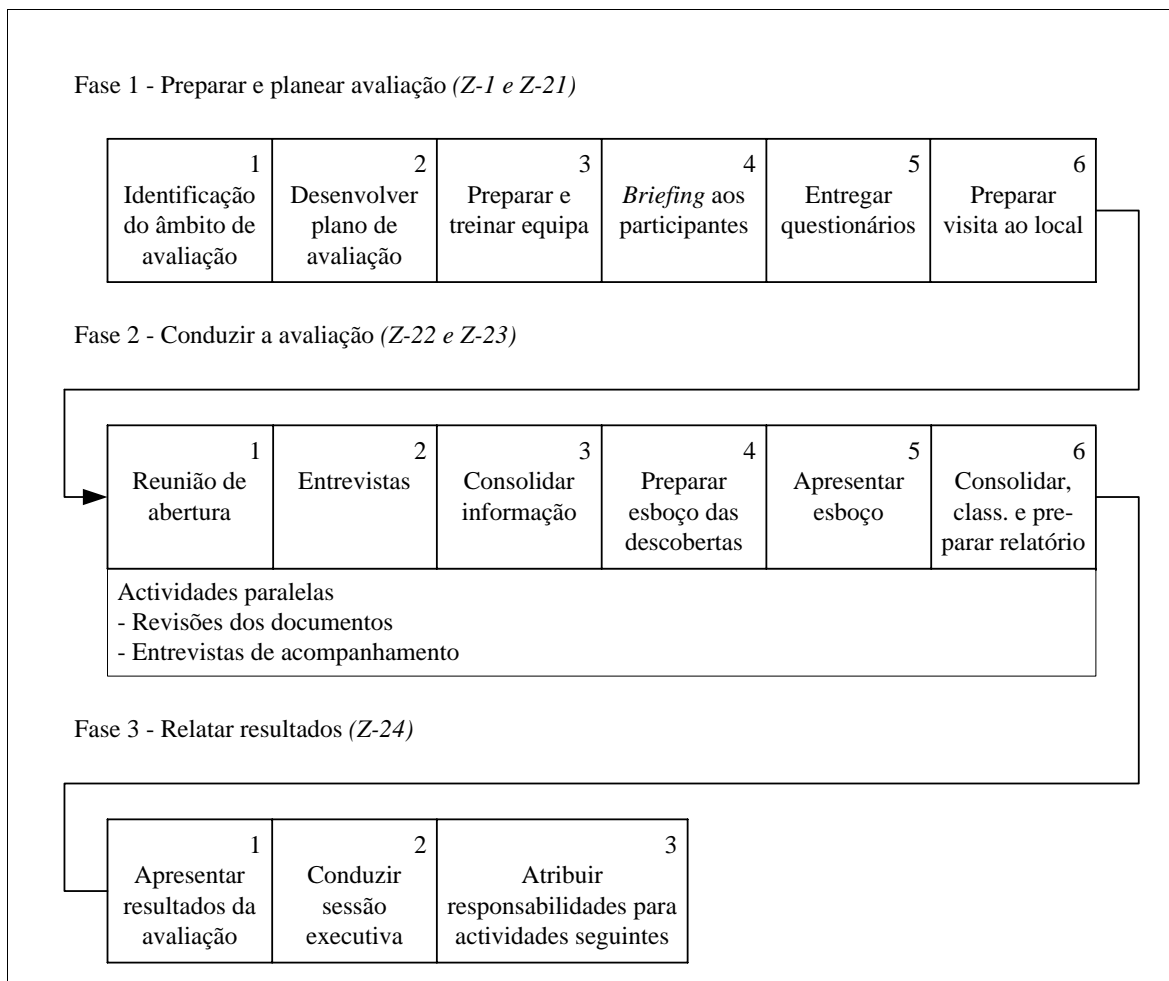


Figura 8.6 – Actividades do método CBA-IPI.

Sobre a implementação da melhoria do processo, a norma seguida neste trabalho, propõe os seguintes resultados [ISO/IEC 2004d]:

- Revisões<sup>6</sup> aos processos normalizados da organização
- Execução de alterações no processo normalizado e definido de forma controlada

<sup>6</sup> com periodicidade apropriada.

- Monitorização das actividades de melhoria do processo
- Análise e utilização de dados históricos, técnicos e dos diagnósticos para melhorar os processos
- Recolha, manutenção e utilização de dados sobre o custo da qualidade, para melhorar os processos da organização

A figura 8.7 mostra o método de melhoria proposto na parte 4 da norma ISO/IEC 15504. Os círculos representam os passos do processo de melhoria e as setas representam o fluxo de informação.

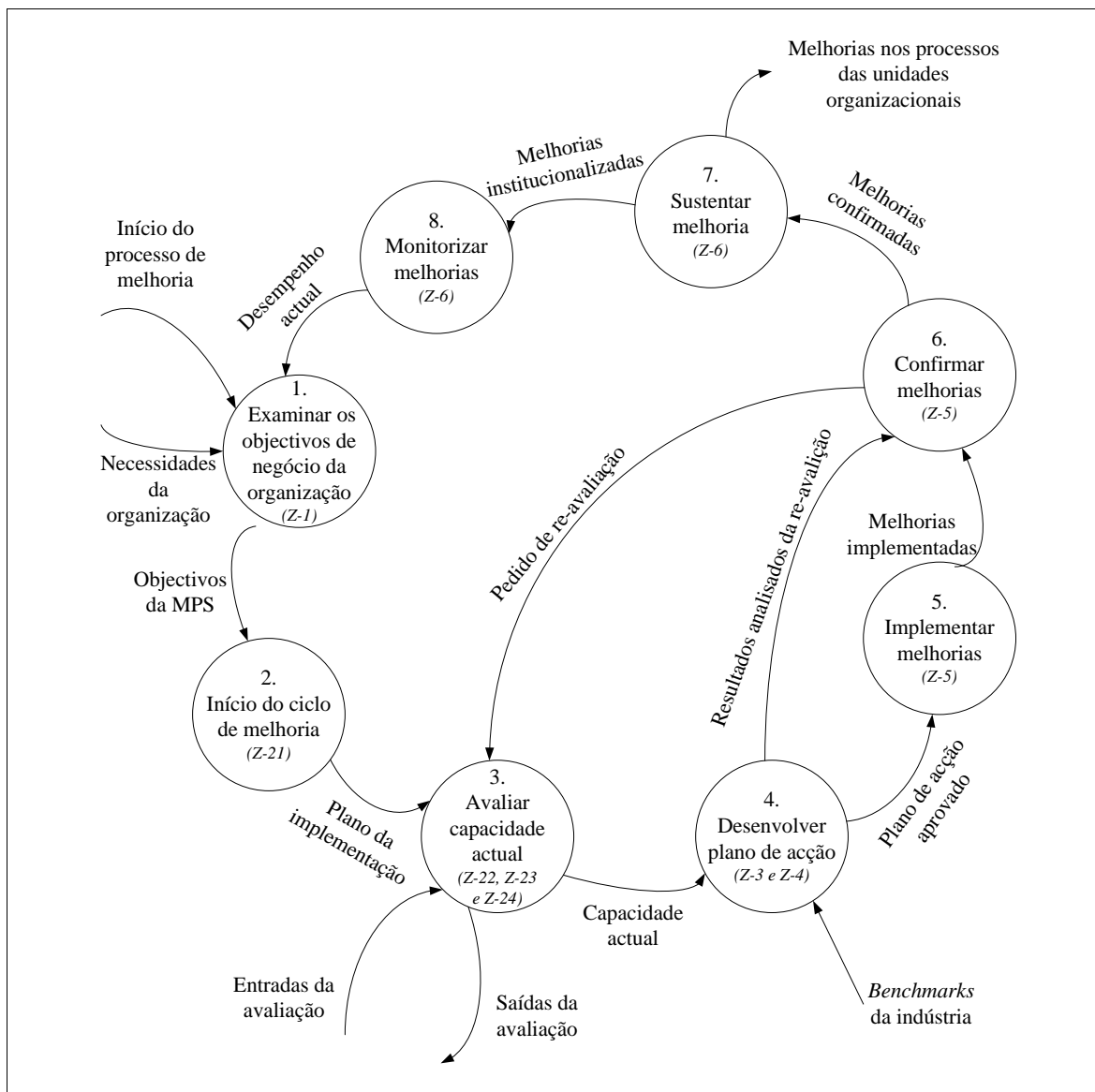


Figura 8.7 – Método de melhoria da ISO/IEC 15504.

Os métodos de implementação apresentados foram tomados em consideração na concepção do M<sub>i</sub>2MP. Devido aos objectivos do 2MPspe, nomeadamente, a sua aplicação às PES, a necessidade de ciclos de melhoria curtos e a simplicidade, pretendeu-se um maior alinhamento com os métodos de implementação do Rapid, Tapistry e o CBA-IPI.

## 8.2. Descrição do M<sub>i</sub>2MP

O M<sub>i</sub>2MP segue as fases genéricas propostas por Zahran [1998]. Este modelo (ver figura 8.8) foi concebido com base nos modelos analisados e nos casos práticos descritos<sup>7</sup>. As etapas seguidas nos casos práticos foram: planeamento, diagnóstico<sup>8</sup>, definição do plano de melhoria e implementação do plano. O M<sub>i</sub>2MP é um aperfeiçoamento deste método utilizado inicialmente para executar a iniciativa de melhoria.

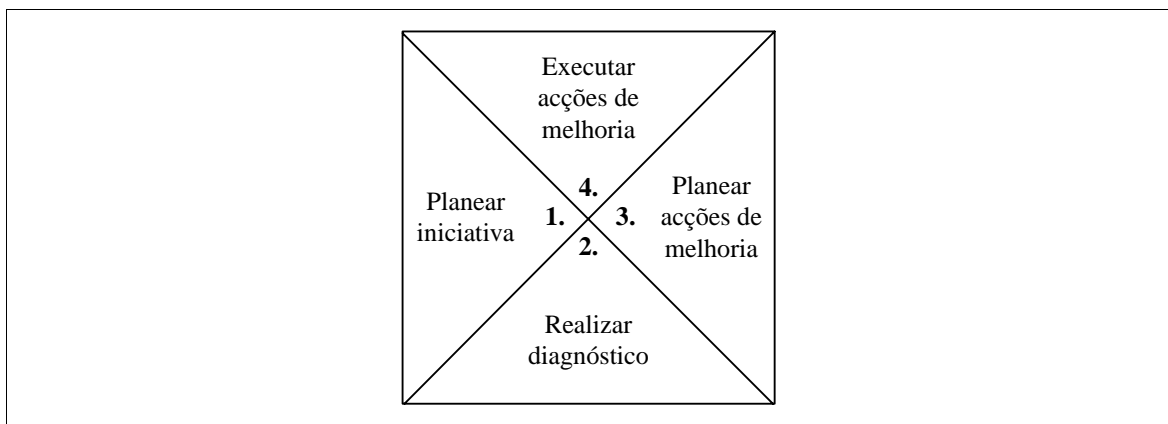


Figura 8.8 – Etapas do M<sub>i</sub>2MP.

Em relação às fases genéricas temos:

1. Planear iniciativa → Z-1 e Z-21
2. Realizar diagnóstico → Z-22, Z-23 e Z-24
3. Planear acções de melhoria → Z-3 e Z-4

<sup>7</sup> ver secção 4.3.

<sup>8</sup> verificou-se que na aplicação prática de uma iniciativa de melhoria se deve evitar a palavra “avaliação”, devido à carga negativa associada. Assim, neste capítulo e no terreno utiliza-se diagnóstico em vez de avaliação.

4. Executar acções de melhoria → Z-5 e Z-6

No método de implementação M<sub>i</sub>2MP cada etapa é constituída por diversas tarefas realizadas de forma sequencial (ver figura 8.9)

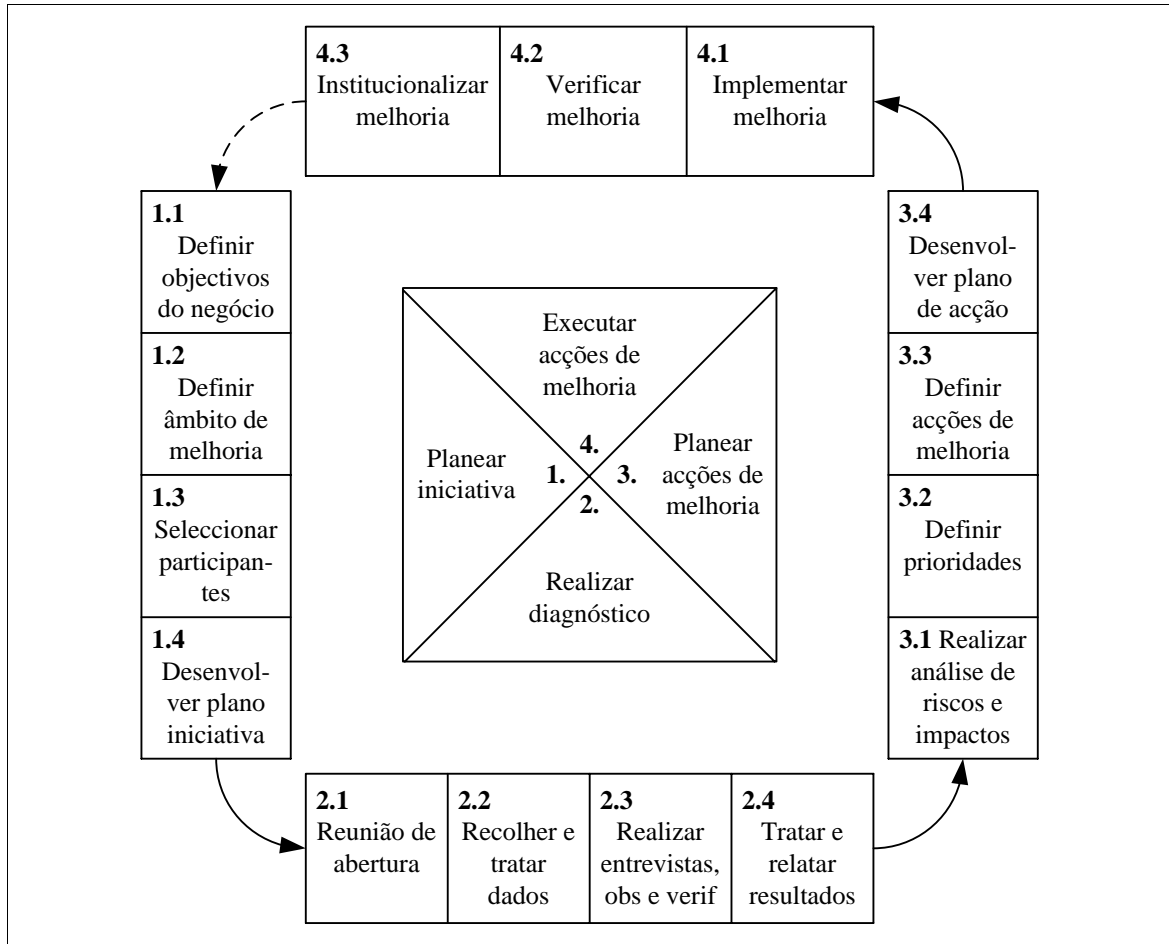


Figura 8.9 – Tarefas do M<sub>i</sub>2MP.

A execução de uma iniciativa de melhoria seguindo o M<sub>i</sub>2MP será realizada por uma equipa externa<sup>9</sup>. A figura 8.10 apresenta a intervenção da equipa externa e dos participantes internos, no programa de MPS.

<sup>9</sup> também designada equipa 2MPspe.

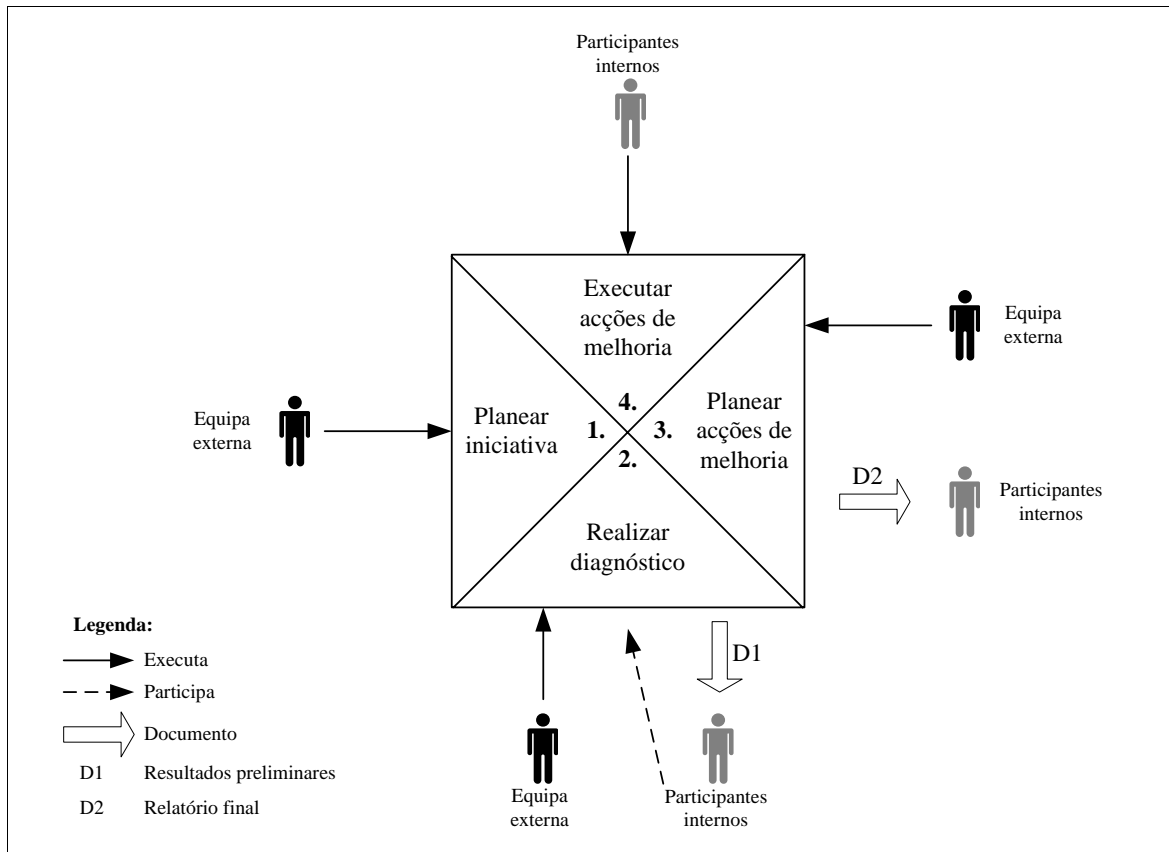


Figura 8.10 – Intervenção dos participantes no programa de melhoria

Verifica-se que os profissionais da organização alvo da melhoria participam na etapa de diagnóstico e executam as acções de melhoria seleccionadas na etapa 3. O patrocinador da melhoria recebe o documento, “Resultados preliminares” (D1) resultante da fase de diagnóstico e recebe o “Relatório final” (D2) definindo a descrição e planeamento das acções de melhoria. A equipa externa<sup>10</sup> executa as três primeiras fases do M<sub>i</sub>2MP.

O método de implementação do 2MPspe fornece orientações sobre a execução das tarefas e propõe a utilização de diversas técnicas e ferramentas. A etapa de diagnóstico usa como ferramenta de suporte o questionário 2MPspe e utiliza a técnica de representação dos resultados definida no M<sub>a</sub>2MP, para desenvolver o documento “Resultados preliminares”. Na realização do diagnóstico são também efectuadas entrevistas e observações directas aos processos. No planeamento das acções de melhoria (etapa 3) são aplicados três níveis de filtragem para seleccionar os processos

<sup>10</sup> com 1 ou 2 elementos.



alvo: impactos 2MPspe, prioridades 2MPspe e estratégias 2MPspe. Os processos pré-seleccionados serão ainda discutidos com os participantes seniores da organização de modo a obter a convergência com os objectivos do negócio.

Nas próximas secções descrevem-se as etapas do M<sub>i</sub>2MP.

### 8.2.1. Etapa 1 – Planear Iniciativa

O planeamento da melhoria engloba as actividades de preparação da iniciativa e desenvolvimento do plano, incluindo recursos, tempos e logística (ver figura 8.11).

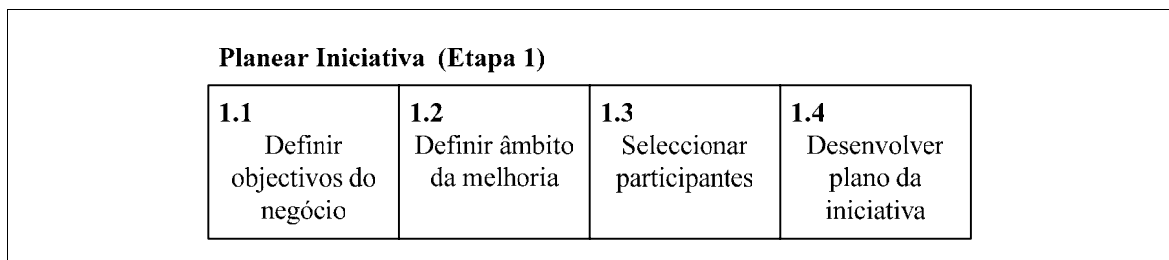


Figura 8.11 – Etapa 1: “Planear Iniciativa”

A tarefa de definição de objectivos do negócio inclui inicialmente identificar o patrocinador da iniciativa de melhoria e assegurar o seu compromisso. Seguidamente, é necessário definir o(s) Impulsionador(es) internos do Processo de Melhoria (IPM) cuja tarefa é facilitar a “ponte” com a organização. Finalmente, é necessário definir o estímulo que levou à percepção da necessidade de melhoria e simultaneamente identificar as necessidades da organização e objectivos do negócio. Segundo o SPIRE [1998] estes parâmetros são um “mix” da lista seguinte:

- Custos do desenvolvimento
- Custo de manutenção
- Tempo para o mercado
- Falta de tempo
- Qualidade do produto final
- Capacidade para controlar o projecto e os riscos do produto

A equipa 2MPspe incluirá<sup>11</sup> um ou dois elementos dependendo do tamanho da organização e esta primeira tarefa terá como suporte uma reunião com o patrocinador<sup>12</sup> e o(s) IPM(s).

A segunda tarefa consiste na definição do âmbito da melhoria e inclui a caracterização do contexto organizacional, no mínimo [ISO/IEC 2004d]:

- Tamanho da organização
- Domínio de aplicação dos produtos ou serviços
- Complexidade dos produtos
- Características de qualidade dos produtos

É necessário também definir restrições específicas, incluindo disponibilidade de recursos, confidencialidade dos dados, prazos e fundos. Finalmente, é definido o âmbito da melhoria, englobando, projectos diagnosticados, fronteiras da melhoria, objectivos do diagnóstico e alinhamento com as necessidades da organização. A definição do âmbito será executada pela equipa externa e incluirá uma reunião com o(s) IPM(s). Esta reunião servirá para preencher os documentos “Matriz de Impactos” e “Capacidade Alvo”.

A matriz de impactos (ver tabela 8.2) serve para alinhar os processos com as necessidades da organização e classificar os impactos percebidos.

Necessidades do negócio e objectivos da organização	Processos para suportar a necessidade	Impactos na necessidade (Grande/Médio/Pequeno)
1. Aumentar qualidade do produto final	SUP.3 GES.3 ENG.25 ORG.12	Médio Pequeno Grande Médio
2. Diminuir os custos da manutenção	ENG.3 ENG.1 SUP.3	Grande Médio Médio
...	...	...

Tabela 8.2 – Exemplo da “Matriz de impactos” (adaptado de SPIRE [1998]).

<sup>11</sup> a constituição da equipa externa é definida no final desta 1ª tarefa.

<sup>12</sup> caso o patrocinador seja um gestor de topo não terá mais envolvimento directo na iniciativa, recebendo apenas o relatório final.

O documento da “Capacidade Alvo” (ver tabela 8.3) permitirá saber as expectativas dos IPMs em relação à capacidade dos processos. Note-se que o conhecimento dos factores de contexto (atributos AK) é pré-definido para o nível explícito devido ao seu impacto global na melhoria.

	Nível 1 (Exec.)	Nível 2 (Ger.)	Nível 3 (Estab.)		Estratégico (S/N)	Impacto (G/M/P)
ENG.1 – Gestão de requisitos		X			S	G
ENG.21 – Análise de requisitos s/w		X			S	G
ENG.22 – Concepção do s/w		X			S	M
...						
ORG.3 – Suporte ao cliente	X				N	G

Tabela 8.3 – Exemplo do documento “Capacidade alvo”.

A tarefa de selecção dos participantes tem início com o desenvolvimento da matriz recursos / processos. Esta matriz apresenta as diversas responsabilidades dos intervenientes nos processos. Com base na matriz, a equipa 2MPspe procederá à selecção dos profissionais da organização que irão responder ao questionário. A selecção pretende cobrir todos os processos do MRP<sub>2MPspe</sub>, utilizando os profissionais com o conhecimento adequado.

No desenvolvimento do plano da iniciativa é necessário inicialmente definir os recursos, detalhar o cronograma e desenvolver a análise dos riscos. Seguidamente, é executada a configuração dos questionários 2MPspe e é preparada a visita às instalações da organização para diagnosticar as práticas actuais. A preparação inclui a solicitação dos documentos necessários e da logística de apoio.

A configuração do questionário do 2MPspe envolve três tarefas<sup>13</sup> relacionadas com: (1) a área específica de cada entrevistado; (2) a validação da objectividade das questões sobre as práticas de base e (3) o número de profissionais da organização alvo.

<sup>13</sup> ver secção 7.5.1.

O *output* da primeira etapa do M<sub>i</sub>2MP é o plano da MPS, esta etapa é executada pela equipa externa, incluindo duas reuniões com elementos da organização (patrocinador e IPMs).

### 8.2.2. Etapa 2 – Realizar Diagnóstico.

A etapa de realização do diagnóstico trata da análise das práticas actuais da organização (ver figura 8.12). A ferramenta principal é o questionário 2MPspe mas são realizadas entrevistas e observações directas para confirmar os dados recolhidos.

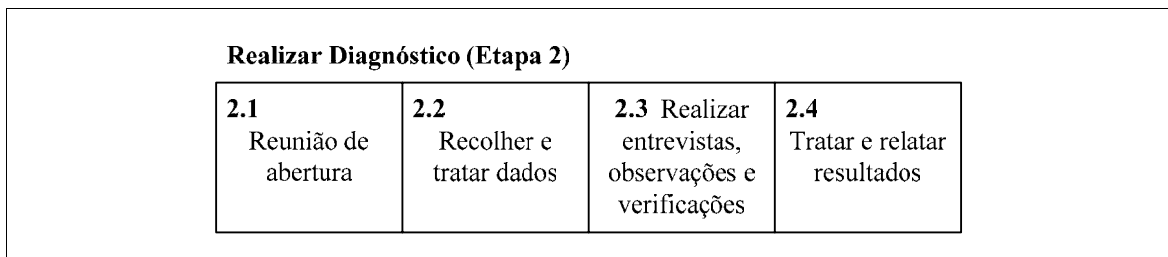


Figura 8.12 – Etapa 2: “Realizar Diagnóstico”

A tarefa inicial da etapa 2 do M<sub>i</sub>2MP consiste na condução da reunião de abertura, englobando todos os participantes na iniciativa de melhoria. Esta reunião serve para expor os objectivos da iniciativa de melhoria, gerir as expectativas dos intervenientes e enquadrar o diagnóstico no contexto da melhoria. O patrocinador da iniciativa deverá abrir a sessão para demonstrar de forma visível o seu apoio e compromisso. Seguidamente, os IPMs deverão também demonstrar o seu apoio e contextualizar a iniciativa. Finalmente, a equipa externa apresentará os detalhes da melhoria, do diagnóstico e dos questionários. A reunião terminará com um espaço de “questões livres” e com a entrega dos questionários aos participantes.

A tarefa de recolha e tratamento dos questionários tem início com a recepção dos questionários, realizando-se, em seguida, o cálculo preliminar dos níveis de capacidade e de conhecimento dos processos. O passo seguinte consistirá na análise dos dados e na procura de valores com grandes discrepâncias. As entrevistas devem ser

preparadas utilizando os dados dos questionários e as observações directas e acções de verificação devem também ser calibradas com as descobertas dos questionários.

A realização de entrevistas, observações directas e verificações permitem recolher factos para caracterizar de forma fundamentada as práticas actuais da organização. As entrevistas devem centrar a atenção nos seguintes aspectos:

- Compreender o modo como o trabalho é executado
- Compreender quais os processos utilizados
- Identificar o tipo de cultura organizacional preponderante
- Compreender os relacionamentos entre os processos
- Identificar a abordagem à gestão do conhecimento nos processos, nos indivíduos e nas equipas
- Identificar áreas que os entrevistados pensam que devem ser melhoradas
- Verificar indirectamente dados específicos dos questionários

As observações directas, tal como as entrevistas são realizadas pela equipa 2MPspe. Estas acções têm como objectivo principal identificar e analisar os produtos de trabalho. Paralelamente, as observações devem permitir verificar questões relacionadas com:

- O modo de executar os processos
- Os procedimentos e normas escritas
- A cultura organizacional preponderante (através dos símbolos visíveis externamente<sup>14</sup>)
- A partilha do conhecimento entre os intervenientes
- Os relacionamentos entre processos

Esta tarefa deve terminar com acções<sup>15</sup> específicas para verificar a consistência, objectividade e representatividade dos dados.

A tarefa de tratamento e relato dos resultados faz a consolidação de toda a informação recolhida, derivando a maturidade da organização e o perfil de maturidade dos processos. A maturidade no 2MPspe é representada por dois valores que avaliam a

---

<sup>14</sup> logótipos, slogans e padrões de comportamento.

<sup>15</sup> discussões abertas, entrevistas ou observações directas.

capacidade e o conhecimento dos factores de contexto. O *output* desta tarefa é o relatório designado “Resultados Preliminares” que será formalmente apresentado ao patrocinador e aos IMPs. No desenvolvimento deste relatório utiliza-se a representação de resultados<sup>16</sup> proposta pelo M<sub>a</sub>2MP, mostrando de forma detalhada o estado actual da organização em relação às práticas utilizadas.

### 8.2.3. Etapa 3 – Planear Acções de Melhoria.

A etapa de planeamento das acções de melhoria baseia-se nas descobertas do diagnóstico para daí derivar o plano das acções de melhoria (ver figura 8.13). Os resultados do diagnóstico são analisados pela equipa 2MPspe, de acordo com as estratégias, riscos e impactos 2MPspe. Este esboço do plano de melhoria é depois calibrado com a visão e sensibilidade dos IPMs. É importante incluir nas acções de melhoria as percepções dos profissionais internos e preferencialmente chegar a uma posição consensual.

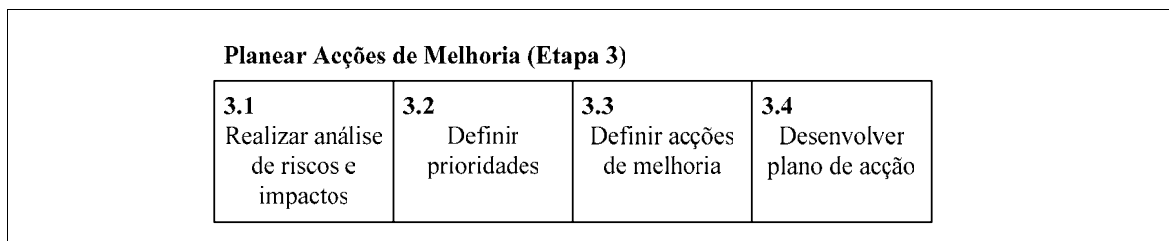


Figura 8.13 – Etapa 3: “Planear Acções de Melhoria”

A análise de riscos e impactos baseia-se nos documentos “Matriz de impactos” e “Capacidade alvo” para desenvolver prioridades para as acções de melhoria. A equipa 2MPspe deverá preencher a primeira versão do documento “Prioridades dos processos” (ver tabela 8.4). A explicação completa deste documento encontra-se na próxima secção.

<sup>16</sup> ver secção 7.5.2

Processo	Capacidade alvo	Capacidade avaliada	Factores contexto		Estratégico			Impacto na organização	Risco do processo	Risco do contexto	Prioridades		
			G.C	C.O	s/n	Imp	N.I				T	2MP	C
<b>ENG.1</b>	2	1	X	X	S	G	2	Grande	Médio	Baixo	1	1	1
<b>ENG.21</b>	2	0		X	S	G	1	Médio	Alto	Médio	2	1	2
...													
<b>ORG.3</b>	2	2	X	X	S	G	2	Grande	Baixo	Baixo	2	3	3

Legenda:	
Prioridades T	– Teórica
Prioridades 2MP	– 2MPspe
Prioridades C	– Consensual
Escala de prioridades:	1 – Prioridade Máxima
	2 – Prioridade Alta
	3 – Prioridade Média
	4 – Prioridade Baixa
N.I.	– Número de indicações
Imp	– Impacto directo

Tabela 8.4 – Exemplo do documento “Prioridades dos processos”.

Durante esta tarefa, a equipa 2MPspe utiliza os dados do diagnóstico e do plano da iniciativa para derivar as prioridades 2MPspe (penúltima coluna da tabela 8.4). Este procedimento é realizado para cada processo do  $MRP_{2MPspe}$  e consiste em classificar a prioridade entre 1 – Máxima Prioridade e 4 – Baixa Prioridade. Esta classificação é realizada ponderando 3 parâmetros, isto é, a prioridade é máxima (nível 1) quando:

- O Impacto na organização é “(G)rande” e a capacidade avaliada “0” ou “1”. O impacto deve ser verificado através da coluna, “Processo Estratégico” classificada com “(S)im”, o número de indicações das necessidades da organização “>0” e o impacto directo, “(G)rande”
- O Risco do processo classificado como “Alto”
- A Prioridade (T)eórica classificada como 1 (Máxima prioridade)

A tarefa definição das prioridades é realizada com os IPMs e servirá para chegar a uma posição consensual sobre a prioridade atribuída a cada um dos processos. Deve ser preenchida a última coluna do documento “Capacidade Alvo” com a classificação da prioridade (C)onsensual. Esta tarefa é desenvolvida numa reunião entre a equipa 2MPspe e os IPMs.

A definição das acções de melhoria é realizada pela equipa 2MPspe e consiste em analisar os processos com prioridade consensual = 1 (Máxima)<sup>17</sup> e definir quais as acções a serem executadas para melhorar os processos alvo. As acções devem ser realistas e sempre que possível especificar de forma quantitativa os objectivos a atingir. As acções de melhoria são desenvolvidas com base no perfil de maturidade do processo, isto é, considerando as práticas de base, as práticas genéricas e as práticas transversais. Em seguida, as acções de melhoria devem ser agrupadas em “grupos de acções”, utilizando alguns critérios:

- Fronteiras das acções de melhoria (organização, equipa, indivíduo)
- Atributos do processo específicos (por exemplo, cultura organizacional, estabelecimento do processo, formação)
- Prazos de implementação iguais

Finalmente, estes “pacotes de acções de melhoria” terão a filtragem final em relação a: custo, disponibilidade de recursos e alinhamento com a estratégia organizacional. Esta reunião incluirá os IPMs e validará cada grupo de acções de melhoria, analisando a estimativa de custos, necessidades de recursos, alinhamento com a estratégia e pesando com os benefícios esperados. Estas três vertentes seguem a proposta do OPM3 [PMI 2003] e são preponderantes no ambiente das PES.

O desenvolvimento do plano de acção e atribuição de responsabilidade consiste em detalhar cada grupo de acções de melhoria, especificando: objectivos, actividades, recursos, responsabilidades, riscos, cronograma e monitorização. A figura 8.14 mostra o desenvolvimento do Plano de acção.

---

<sup>17</sup> poderão ser incluídos os processos com prioridade 2 (alta).



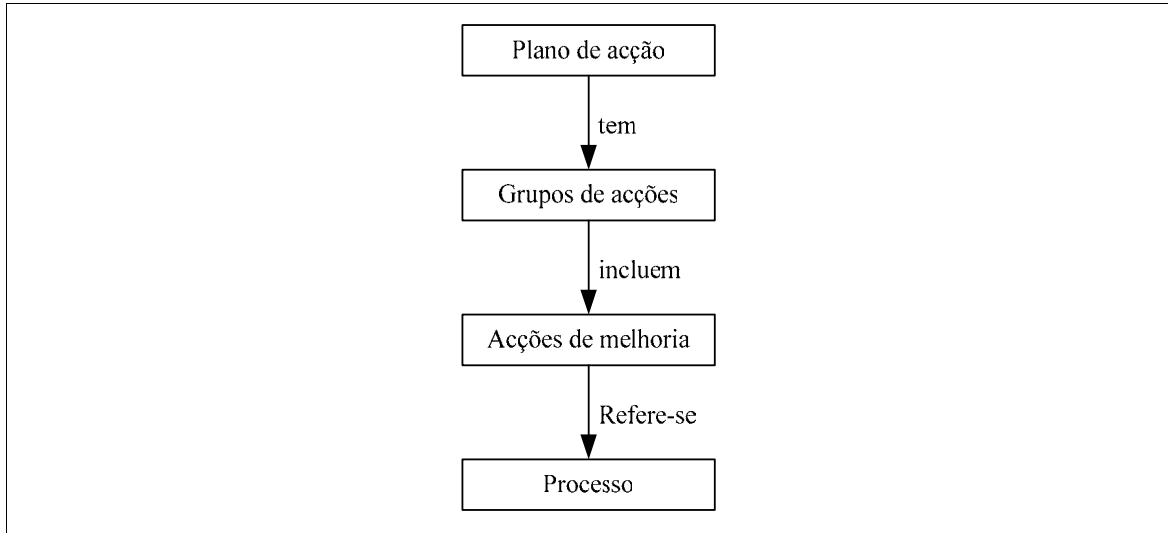


Figura 8.14 – Desenvolvimento do Plano de Acção.

O plano de acção será formalmente apresentado ao patrocinador e aos IPMs e servirá como suporte para a etapa seguinte.

**8.2.3.1. Desenvolvimento do Documento “Prioridades dos Processos”**

Como foi referido, o documento “Prioridades dos Processos” é o documento central da etapa de planeamento das acções de melhoria. Nesta secção apresenta-se uma explicação sucinta dos elementos deste documento. A figura 8.15 apresenta os elementos apenas para um processo do MRP<sub>2MPspe</sub><sup>18</sup>.

	Recolhidos				Calculados			Saída					
Coluna	a)	b)	c)		d)			e)	f)	g)	h)		
Processo	Capacidade alvo	Capacidade avaliada	Factores contexto		Estratégico			Impacto na organização	Risco do processo	Risco do contexto	Prioridades		
			G.C	C.O	s/n	Imp	N.I				T	2MP	C
ENG.1	2	1	0,5	0,5	S	G	2	Grande	Médio	Baixo	1	2	2

Figura 8.15 – Exemplo do documento “Prioridades dos processos”.

<sup>18</sup> o procedimento é repetido para todos os processos.

Em relação aos parâmetros recolhidos, a coluna a) é derivada directamente do documento da “Capacidade Alvo” preenchido na tarefa de definição do âmbito da melhoria. A capacidade avaliada (coluna b) e os factores de contexto (coluna c) são recolhidos directamente dos “Resultados preliminares”.

Na coluna d), que pretende verificar quão estratégico é o processo, a primeira coluna (Sim/Não) é recolhida no documento “Capacidade Alvo”. As colunas d2 e d3 são derivadas da “Matriz de Impactos”, o número de indicações é calculado, contabilizando o número de vezes que o processo é referido e a coluna d3. (impacto nas necessidades do negócio) é calculada, atribuindo o valor máximo de cada processo<sup>19</sup>. Finalmente, o impacto na organização (coluna e) é recolhido directamente no documento “Capacidade Alvo”.

As colunas calculadas (f e g) têm como base os perfis do processo alvo e avaliado. A ferramenta base para esta análise do risco é a representação dos resultados na perspectiva detalhada<sup>20</sup> (ver figura 8.16).

AK1.2	AK1.1	Processo		AC1.1	AC1.51	AC2.1	AC2.2	AC3.1	AC3.2
C	C	ENG.1	Alvo	C	C	C	L	N	N
L	C		Avaliado	C	L	L	P	N	N
C	C	ENG.21	Alvo	C	L	N	N	N	N
P	L		Avaliado	L	N	N	N	N	N
		⋮							

**Legenda:**

- C** Completamente atingido
- L** Largamente atingido
- P** Parcialmente atingido
- N** Não atingido

Figura 8.16 – Perfis alvo e avaliado dos processos.

<sup>19</sup> no caso de um processo não referido em nenhuma necessidade, a coluna d2 ficará com “N”, Não referido e d3 com zero.

<sup>20</sup> ver secção 7.5.2.

O cálculo do risco do processo segue a abordagem proposta pela parte 4 da norma ISO/IEC 15504. A norma propõe a utilização do conceito de lacunas<sup>21</sup> no atributo da capacidade sempre que a classificação diagnosticada para um AC é menor que a classificação desejada para o atributo da capacidade (ver tabela 8.5).

Classificação desejada para o atributo	Classificação avaliada para atributo	Lacuna no atributo
Completamente atingido	Completamente atingido	Nenhuma
	Largamente atingido	Pequena
	Parcialmente atingido	Grande
	Não atingido	Grande
Largamente atingido	Completamente atingido	Nenhuma
	Largamente atingido	Nenhuma
	Parcialmente atingido	Pequena
	Não atingido	Grande

Tabela 8.5 – Lacunas nos atributos (adaptado de ISO/IEC [2004d]).

A probabilidade de ocorrerem problemas num processo é derivada da extensão das lacunas entre o perfil do processo alvo e avaliado.

A tabela 8.6 apresenta o risco associado com o processo, derivado das lacunas nos ACs e no nível onde ocorrem.

Número de lacunas nos ACs e nível de capacidade	Risco no processo
Nenhuma lacuna pequena nem grande	Risco reduzido
Nenhuma lacuna no 1, 1.5 e apenas pequenas lacunas no 2, 3	Risco baixo
Apenas lacunas pequenas ou apenas 1 lacuna grande no 2, 3	Risco médio
Uma lacuna grande no 1, 1.5 ou mais que uma grande no 2, 3	Risco alto

Tabela 8.6 – Risco associado às lacunas nos ACs.

Para calcular o risco do contexto utiliza-se também a tabela das lacunas dos atributos, mas neste caso, o risco derivado é apresentado na tabela 8.7.

---

<sup>21</sup> *gaps*.

Número de lacunas nos AKs	Risco do contexto
Nenhuma lacuna pequena nem grande	Risco reduzido
Apenas lacunas pequenas	Risco baixo
Uma lacuna grande	Risco médio
Apenas lacunas grandes	Risco alto

Tabela 8.7 – Risco associado ao contexto.

Aplicando os valores tabelados para o risco do processo e do contexto, calcula-se os valores necessários para preencher as colunas f) e g).

A figura 8.17 apresenta um exemplo da derivação dos riscos.

ENG.1 - Gestão dos Requisitos								
AC1.2	AC1.1		AP1.1	AP1.51	AP2.1	AP2.2	AP3.1	AP3.2
C	C	Perfil Alvo	C	C	C	L	N	N
L	C	Perfil Avaliado	C	L	L	P	N	N
Pequena	Nenhuma	Lacuna do atributo	Nenhuma	Pequena	Pequena	Pequena	-	-
<b>Risco baixo</b>		<b>Risco</b>	<b>Risco médio</b>					
Contexto			Processo					

Figura 8.17 – Derivação dos riscos.

Da figura 8.17 conclui-se que o risco da gestão de requisitos é médio, as acções de melhoria não são prioritárias e o risco do contexto do processo é baixo (isto é, as acções de melhoria têm um ambiente propício para o sucesso).

Finalmente, na coluna h, são apresentadas as prioridades associadas ao processo. A coluna h1 mostra a prioridade teórica (recolhida directamente da figura 8.18), na coluna h2 apresentam-se as prioridades 2MP (desenvolvidas pela equipa 2MPspe com

base nos dados do documento “Prioridades dos Processos”) e a coluna h3, apresenta a prioridade consensual obtida através de uma reunião com os IPMs.

As prioridades teóricas foram desenvolvidas a partir das áreas chave do processo do CMM<sup>22</sup> e dos atributos da capacidade necessários para atingir os níveis 2 e 3 da norma ISO/IEC 15504<sup>23</sup>. Resumidamente, o CMM propõe as prioridades do nível 2 centradas nos processos de suporte (Gestão de Configurações e Garantia da Qualidade) e nos processos de gestão (Gestão de projectos e Gestão dos riscos). O nível 3 centra-se nos processos organizacionais (Recursos humanos e Estabelecimento do processo) e nos processos de engenharia.

Em relação à norma ISO/IEC 15504, é referido que os processos de engenharia são tratados desde o início da iniciativa. Os processos relacionados com o nível 2 são os de suporte (Documentação, Gestão de Configurações e Garantia da Qualidade) e gestão (Gestão de projectos e Gestão do risco). No nível 3, os processos relacionam-se com a organização (Recursos humanos e Estabelecimento do processo) e gestão da qualidade.

Estas orientações do CMM e da norma ISO/IEC 15504 foram calibradas com os resultados apresentados por Blanco et al. [2001]. Este trabalho estuda os dados de mais de 400 experiências de melhoria do processo (PIEs<sup>24</sup>) e conclui que as categorias de processo tratadas foram:

- Engenharia (65%)
- Suporte (14%)
- Gestão (8%)
- Organização (8%)
- Cliente – Fornecedor (4%)

A figura 8.18 apresenta as prioridades teóricas associadas a cada processo do MRP<sub>2MPspe</sub>. Refira-se que em termos gerais é dada maior ênfase à categoria de suporte (prioridade máxima), os processos da categoria engenharia possuem prioridade alta e dois processos organizacionais têm prioridade média. Uma vez que o MRP<sub>2MPspe</sub> é um

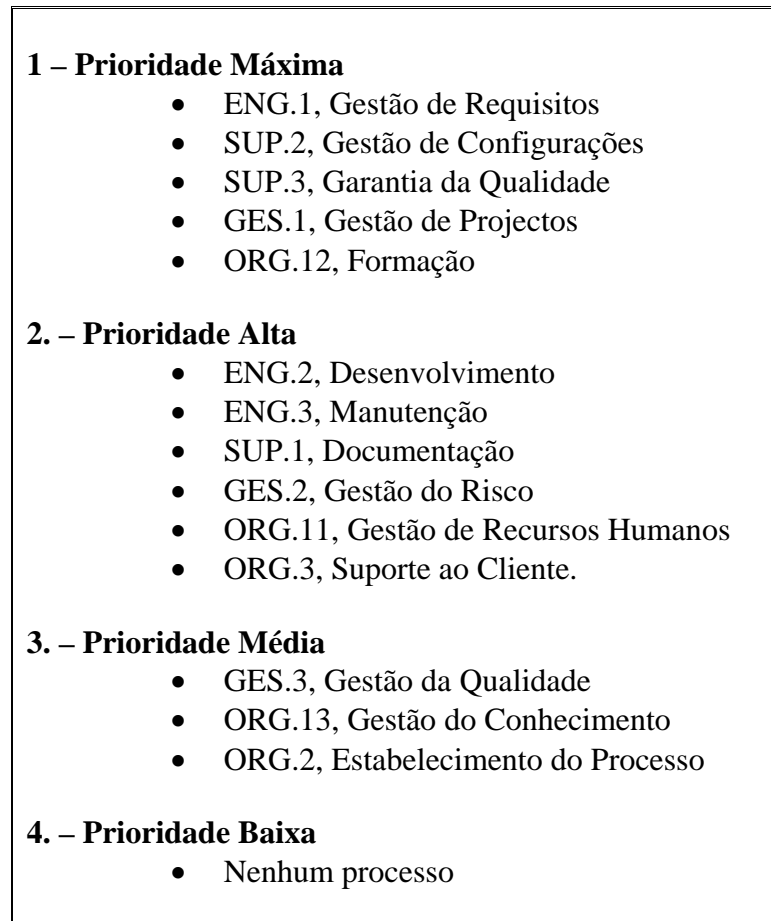
---

<sup>22</sup> ver capítulo 6, figura 6.9.

<sup>23</sup> ver capítulo 6, tabela 6.2.

<sup>24</sup> PIE – *Process Improvement Experience*.

modelo calibrado<sup>25</sup> para PES, não há nenhum processo com prioridade teórica baixa, os processos considerados menos prioritários foram retirados.



*Figura 8.18 – Prioridades teóricas do 2MPspe.*

As prioridades teóricas associadas aos processos serão ajustadas com os dados das iniciativas 2MPspe, ou seja, espera-se introduzir uma componente para reflectir as necessidades reais das PES.

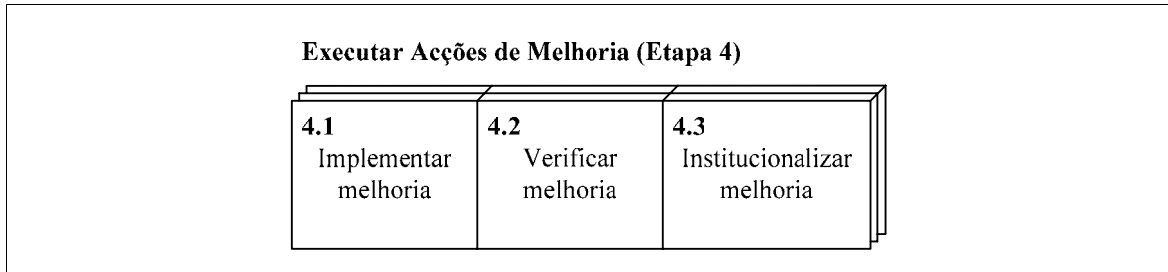
#### **8.2.4. Etapa 4 – Executar Acções de Melhoria.**

A etapa de acompanhamento das acções de melhoria refere-se à execução do plano de acção desenvolvido na etapa anterior (ver figura 8.19). Esta etapa é

---

<sup>25</sup> através da redução dos processos.

responsabilidade dos profissionais da organização sujeita à iniciativa de melhoria. O papel da equipa 2MPspe é apenas de acompanhamento esporádico.



*Figura 8.19 – Etapa 4: “Executar acções de melhoria”*

A etapa consiste no acompanhamento de várias acções de melhoria, em paralelo ou sequenciais<sup>26</sup> e cada grupo de acções de melhoria é implementado, verificado e institucionalizado.

A tarefa de implementação das acções de melhoria deverá seguir uma determinada estratégia de implementação. A estratégia mais comum consiste em experimentar num projecto-piloto, ou no outro extremo implementar em toda a organização em simultâneo. Independentemente da estratégia deve ser preparado um plano detalhado da implementação tendo como base o plano de acção. Este plano detalhado deve conter: objectivos, estratégia, responsabilidades, cronograma, descrição das tarefas e gestão do risco. Durante a implementação, o plano deve ser acompanhado pela gestão para assegurar o seu cumprimento.

A verificação das melhorias servirá para confirmar que os objectivos esperados foram cumpridos e os benefícios desejados foram atingidos. É necessário verificar que os processos e práticas alvo foram realmente adoptados. A organização deve também reavaliar os riscos associados à iniciativa reavaliar os custos e benefícios [ISO/IEC 2003].

A institucionalização das melhorias consiste em manter as melhorias obtidas e segundo o SPIRE [1998] é um dos grandes desafios das iniciativas de melhoria. Simultaneamente, caso a implementação tenha sido realizada num projecto-piloto, é

<sup>26</sup> de acordo com o definido no plano de acção.

necessário difundir as práticas e processos por toda a organização. Esta difusão deve ser planeada e é necessário dar a formação adequada aos recursos humanos.

### 8.3. Utilização Prática do M<sub>i</sub>2MP

O M<sub>i</sub>2MP apresenta tarefas que devem ser seguidas para executar uma iniciativa de melhoria. Como vimos, no contexto das PES, os recursos humanos são escassos e os “sobressaltos” são constantes. Com esta preocupação, o M<sub>i</sub>2MP propõe um ciclo inicial de melhoria<sup>27</sup> de apenas duas semanas (10 dias úteis) com 16 horas estimadas para a realização das actividades do diagnóstico nas instalações da organização alvo (ver figura 8.20).

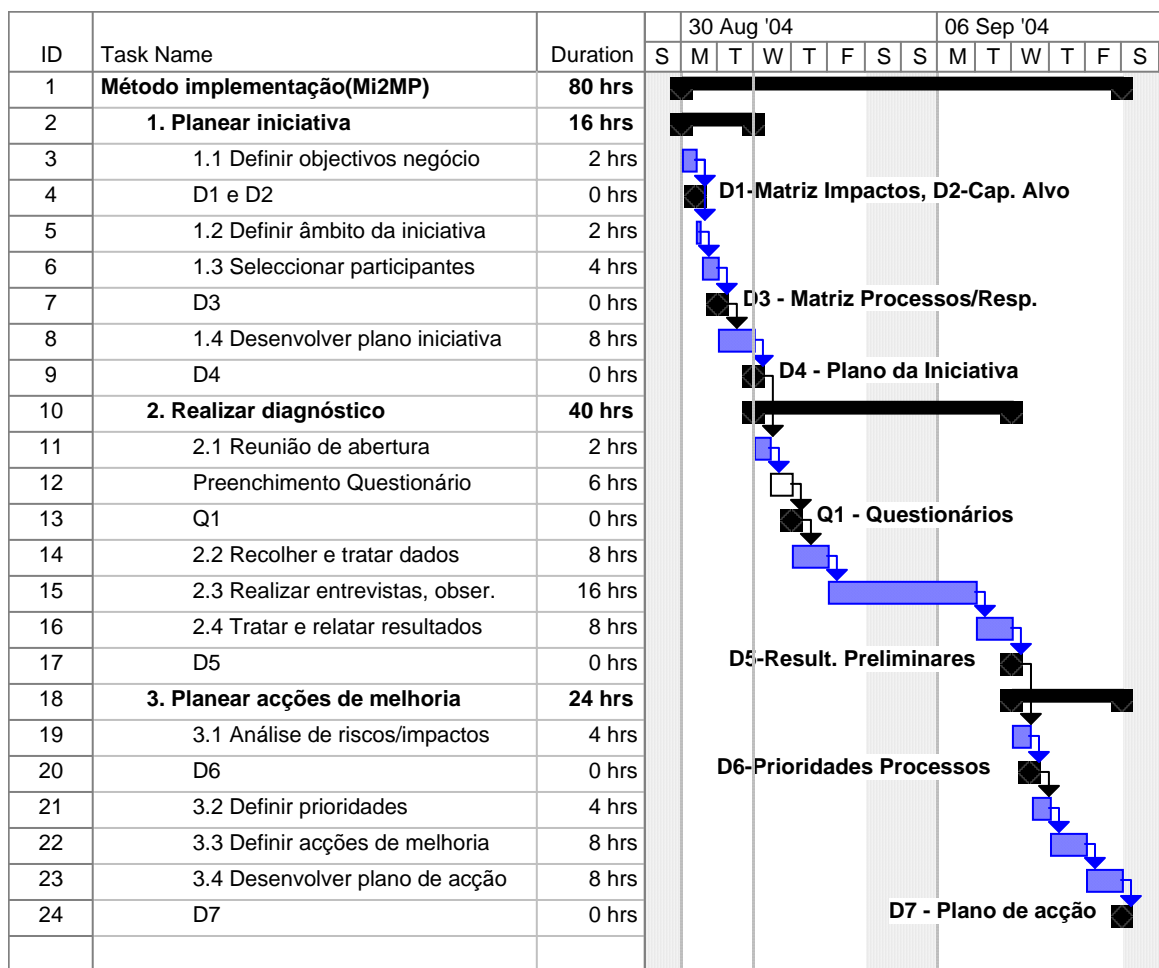


Figura 8.20 – Diagrama de Gantt da iniciativa de Melhoria.

<sup>27</sup> tempo desde o início da iniciativa até à entrega do plano de acção.



Em relação aos recursos humanos, o maior esforço, em termos de intervenientes internos, centra-se nos IPMs, com participação directa ou indirecta em 7 reuniões de trabalho. O patrocinador estará presente na reunião de arranque da iniciativa e na reunião para apresentação do “plano de acção”. Finalmente, os participantes seleccionados deverão responder ao questionário e serão entrevistados durante a etapa de diagnóstico. Em termos de esforço, temos:

Equipa 2MP	→	160 pessoa – hora
IPMs	→	30 pessoa – hora
Patrocinador	→	5 pessoa – hora
Participantes internos	→	60 pessoa – hora
Estimando-se um esforço total de 255 pessoa – hora		

Os documentos apresentados na organização são:

- Resultados preliminares, mostra o estado das práticas da organização, desenvolvido através da etapa de diagnóstico.
- Plano de acção, especifica em detalhe as acções de melhoria propostas.

Os documentos utilizados internamente pela equipa 2MP são: Matriz de impactos, Capacidade alvo, Matriz processos / responsabilidades, Plano iniciativa e Prioridades dos processos.

Neste capítulo realizou-se a descrição do M<sub>i</sub>2MP. No próximo capítulo desenvolvem-se as conclusões sobre este projecto de investigação. Apresenta-se uma síntese do trabalho realizado. Em seguida sistematiza-se as contribuições principais. Na 3ª secção apresentam-se propostas de trabalho futuro e na 4ª secção são tecidas algumas considerações finais.

## Capítulo 9

---

### 9. Conclusões

Neste último capítulo desenvolvem-se as conclusões. Nesta perspectiva apresenta-se inicialmente uma síntese do trabalho realizado. Na secção seguinte expõem-se as contribuições e resultados principais. Na terceira secção desenvolvem-se propostas de projectos de trabalho futuro por forma a dar continuidade e a promover a utilização do 2MPspe. Na última secção são apresentadas algumas considerações finais.

#### 9.1. Síntese do Trabalho Realizado

Como vimos, o objectivo da MPS é melhorar continuamente a efectividade e eficiência da organização através da utilização dos processos e do seu alinhamento com as necessidades do negócio [ISO/IEC 2004d]. Nesta perspectiva, os modelos de MPS sugerem caminhos de melhoria através da avaliação dos processos em relação a um conjunto de melhores práticas. Embora haja um elevado número de modelos de melhoria do processo de desenvolvimento de software, verificou-se que há ainda algumas questões em aberto. Definiu-se então como finalidade da investigação o desenvolvimento de um modelo de melhoria do processo de software adaptado às pequenas empresas de software a desenvolver na área dos sistemas de informação de gestão que trate explicitamente os factores contextuais da cultura organizacional e da gestão do conhecimento. Este objectivo global da investigação tem origem na

constatação que uma iniciativa de MPS com sucesso resulta em benefícios comprovados na qualidade do produto e no valor do negócio mas grande parte das iniciativas falha. Para atingir a finalidade da investigação foi concebido o 2MPspe.

Para cumprir a finalidade enunciada para a investigação foi definido um conjunto de objectivos a atingir cuja satisfação permitiu alcançar um conjunto importante de resultados e contribuições. Os objectivos definidos foram:

- (1) Enquadrar a área de melhoria do processo de desenvolvimento de software na engenharia de software
- (2) Rever e analisar os modelos de MPS mais divulgados
- (3) Avaliar os benefícios da aplicação prática da MPS e caracterizar os FCS
- (4) Conceber o 2MPspe
- (5) Validar o 2MPspe
- (6) Propor projectos de trabalho futuro

Para cumprir o **primeiro objectivo** realizou-se uma revisão bibliográfica para enquadrar a área da melhoria do processo de desenvolvimento de software na disciplina de engenharia de software e simultaneamente clarificar conceitos e termos utilizados. Nesta perspectiva foi utilizado o SWEBOK [2001] que define os conteúdos da disciplina de engenharia de software. Em relação à categorização proposta pelo referido guia, este projecto de investigação centra-se principalmente nas áreas de conhecimento do processo de engenharia do software e da qualidade do software. A área do processo de engenharia de software engloba por um lado as actividades técnicas e de gestão dentro do processo de engenharia de software realizadas durante a aquisição, desenvolvimento e manutenção. Por outro lado preocupa-se com a definição, implementação, gestão, mudança e melhoria do próprio processo [SWEBOK 2001]. Esta tarefa teve como resultado a revisão dos conceitos e perspectivas sobre as duas áreas do conhecimento referidas.

O **segundo objectivo** visou a revisão dos modelos de melhoria do processo de desenvolvimento de software mais divulgados. Refira-se que na realização desta tarefa houve duas preocupações principais: analisar a arquitectura e fundamentos dos modelos e focar a revisão nos modelos específicos para PES. A revisão incidiu, de modo particular, nos modelos que utilizam o paradigma *benchmarking*, mas incluiu também

modelos do paradigma analítico e a norma ISO/IEC 12207 (processos do ciclo de vida do software). Considerou-se importante descrever e analisar as metodologias ágeis devido às suas características inovadoras, nomeadamente, minimização dos processos, centrada nos indivíduos, constante mudança dos requisitos e colaboração com o cliente. Como resultado deste segundo objectivo foi analisado o espectro de modelos de MPS existentes na literatura. Conclui-se que a norma recentemente publicada, ISO/IEC 15504 define requisitos de conformidade para os modelos de MPS e notou-se uma forte preocupação da maioria dos modelos recentes em cumprir esses requisitos.

O **terceiro objectivo** prendeu-se com a avaliação dos benefícios da aplicação prática dos modelos de MPS. Pretendeu-se também analisar os factores críticos de sucesso das iniciativas da MPS. Para a consecução deste objectivo foi realizada uma pesquisa bibliográfica exaustiva e foram executados três casos de estudo em pequenas empresas da região norte. O trabalho desenvolvido permitiu concluir que: (1) há uma grande taxa de insucesso na aplicação dos modelos de MPS e (2) as organizações que implementam e institucionalizam uma MPS com sucesso obtêm ganhos significativos na produtividade, *rework*, precisão nas estimativas da duração e custos e no ROI. Como resultados deste terceiro objectivo, foi definido um conjunto de factores críticos de sucesso para uma iniciativa MPS. A revisão bibliográfica e os casos práticos permitiram definir um conjunto de 8 características desejáveis para o 2MPspe, isto é:

- a) Adaptado às pequenas empresas de software
- b) Tratamento explícito dos factores contextuais, cultura organizacional e gestão do conhecimento
- c) Cumprir os requisitos da norma ISO/IEC 15504
- d) Ter correspondência definida com o CMM
- e) Cumprir os requisitos da norma ISO 9001: 2000
- f) Tratar os problemas do envolvimento dos intervenientes no processo
- g) Permitir ciclos de melhoria rápidos
- h) Simples e robusto

A concepção e validação do 2MPspe representam o **quarto e quinto objectivos** definidos para este trabalho de investigação. No desenvolvimento do 2MPspe utilizou-se uma arquitectura com três componentes, a saber: modelo conceptual (M<sub>c</sub>2MP),

modelo de avaliação ( $M_a2MP$ ) e método de implementação ( $M_i2MP$ ). Esta arquitectura segue as orientações da parte 2 da norma ISO/IEC 15504.

No  $M_c2MP$  propõe-se um referencial tridimensional, englobando as dimensões do processo, da capacidade e do conhecimento dos factores de contexto. Para a definição do eixo dos processos, concebeu-se um modelo de referência dos processos ( $MRP_{2MPspe}$ ), baseado na norma ISO/IEC 12207, mas restringido para o contexto das pequenas empresas de software. A descrição de cada processo do  $MRP_{2MPspe}$  foi realizada definindo o seu propósito e os resultados esperados. Para a dimensão da capacidade dos processos definiram-se os níveis de capacidade do processo, tendo como base os níveis propostos na norma ISO/IEC 15504. Tendo em consideração o ambiente das PES, não foram incluídos os níveis 4 (previsível) e 5 (otimizado) propostos na norma, mas foi adicionado um sub-nível, denominado valorizado. Este sub-nível pretende avaliar se os intervenientes do processo compreendem a importância das práticas e resultados do processo. Ainda no eixo da capacidade, foram propostos os atributos da capacidade (ACs) específicos de cada nível e a escala de classificação utilizada para avaliar os ACs. Em relação à dimensão do conhecimento dos factores de contexto, definiram-se dois níveis de conhecimento, designados nível 0 – nível do conhecimento implícito e nível 1 – nível do conhecimento explícito. No nível 0 não é atingido nenhum atributo do conhecimento (AK), enquanto no nível 1 são explicitamente tratados os atributos do conhecimento AK1 e AK2, respectivamente, atributo da gestão do conhecimento e atributo da gestão da cultura organizacional. Para medir os atributos do conhecimento é proposta uma escala de classificação com quatro pontos. Ainda englobado no  $M_c2MP$  é apresentada a representação da maturidade dos processos no eixo tridimensional, definindo o seu “significado”.

O desenvolvimento do  $M_a2MP$  centrou-se na definição do conjunto de indicadores a partir dos quais são avaliados os processos de uma organização. Um indicador é visto como uma característica de uma prática que suporta o julgamento da execução, da capacidade ou do conhecimento de um processo. Nesta perspectiva, um modelo de avaliação do processo é a base para a recolha de dados e classificação da capacidade e do conhecimento dos processos. Foram então definidos indicadores para as três dimensões propostas no  $M_c2MP$ , ou seja: práticas de base para avaliar a execução do processo; práticas genéricas para a dimensão da capacidade do processo e

finalmente, práticas transversais para a dimensão do conhecimento. Com base nos indicadores definidos, foi concebido o questionário 2MPspe que permite realizar o diagnóstico das práticas utilizadas pela organização de software. Para cumprir os requisitos de conformidade da norma ISO/IEC 15504 foi proposto um mecanismo para representar os resultados obtidos no diagnóstico.

Finalmente, foi desenvolvido o método de implementação do 2MPspe, designado  $M_i2MP$ . Este método define as actividades, técnicas e ferramentas que irão ser utilizadas numa iniciativa de melhoria do processo de desenvolvimento de software. O  $M_i2MP$  propõe quatro fases principais: 1. planear iniciativa; 2. realizar diagnóstico; 3. planear acções de melhoria e 4. executar as acções de melhoria. Os documentos resultantes do  $M_i2MP$  são os “Resultados preliminares” e o “Plano de acção”. No  $M_i2MP$  estima-se um ciclo inicial de melhoria<sup>1</sup> de 10 dias úteis, incluindo 16 horas para as actividades do diagnóstico, realizadas nas instalações da organização de software.

O resultado do quarto e quinto objectivo foi a primeira versão do 2MPspe. Refira-se que as componentes do modelo foram validadas indirectamente através do estudo de casos. O questionário 2MPspe foi validado num ambiente controlado, através de um painel de *experts* da área académica e da área empresarial.

O **sexto objectivo** desenvolve-se na secção 9.3.

## 9.2. Contribuições Principais

As contribuições principais deste trabalho estão ligadas ao cumprimento da finalidade enunciada e dos objectivos para atingir essa finalidade da investigação. Assim, considera-se que há três contribuições principais:

1. Como resultado da revisão bibliográfica realizada inicialmente, enquadrou-se a área de melhoria do processo de desenvolvimento de software na disciplina de

---

<sup>1</sup> constituído pelas 3 fases iniciais

engenharia de software. Simultaneamente foi possível descrever e sistematizar conceitos e aspectos relevantes na área de melhoria do processo de software.

2. A segunda contribuição está ligada ao enfoque nos modelos de melhoria do processo de desenvolvimento de software. As estratégias seguidas foram a pesquisa bibliográfica e o estudo de casos. A contribuição refere-se, por um lado, a uma sistematização das arquitecturas e dos fundamentos inerentes aos diversos modelos de MPS. Por outro lado, identificaram-se os benefícios obtidos a partir de uma iniciativa de melhoria do processo com sucesso e definiram-se os factores críticos de sucesso para um projecto de MPS. Este estudo teve como preocupação central, analisar os modelos de MPS específicos para o contexto das PES. Como resultado do estudo sobre os modelos MPS identificou-se um conjunto de características desejáveis para o 2MPspe.

3. Finalmente, a contribuição principal deste trabalho de investigação está ligada à concepção do 2MPspe. Esta primeira versão do modelo pretendeu tratar um conjunto de aspectos que foram identificados como importantes para o sucesso de uma iniciativa de MPS. Assim, pretendeu-se tratar o conjunto das oito características desejáveis, definidas no 3º objectivo da investigação. A inovação do 2MPspe está ligada ao tratamento simultâneo deste conjunto de características. Apresenta-se em seguida o trabalho desenvolvido para atingir as características desejáveis.

A primeira característica (a) definida foi adaptação às pequenas empresas de software. Para cumprir este aspecto, o  $MRP_{2MPspe}$ , definido no contexto do  $M_c2MP$  foi restringido, incluindo apenas 18 processos primários. O esforço da iniciativa foi estimado em 255 pessoas-hora e englobando apenas dois dias<sup>2</sup> nas instalações da empresa. Estes valores parecem “confortáveis” para o contexto das PES.

A segunda característica (b) identificada liga-se ao tratamento explícito dos factores contextuais, cultura organizacional e gestão do conhecimento. Esta característica foi tratada no nível conceptual, propondo um referencial tridimensional que engloba as seguintes dimensões: processos, capacidade e conhecimentos dos factores do contexto. Este referencial tem repercussões no modelo de avaliação,  $M_a2MP$

---

<sup>2</sup> 16 horas

e conseqüentemente no questionário 2MPspe. A base das acções de melhoria é o referido questionário, que inclui perguntas sobre a cultura organizacional predominante e sobre a abordagem à gestão do conhecimento.

Como terceira característica (c) definiu-se o cumprimento dos requisitos da norma ISO/IEC 15504. Como foi apresentado ao longo dos capítulos 6, 7 e 8, a concepção do 2MPspe teve como base as cinco partes da norma ISO/IEC 15504. Os requisitos e orientações da norma são apresentados na descrição das componentes do 2MPspe e a própria arquitectura do modelo segue a proposta da parte 2 da norma ISO/IEC 15504<sup>3</sup>. O referencial tridimensional proposto no M<sub>c</sub>2MP permite um alinhamento completo com a norma, caso a iniciativa de melhoria se desenvolva utilizando apenas as dimensões dos processos e da capacidade.

A quarta característica (d) está ligada à formalização de uma correspondência com o CMM. Como foi apresentado no capítulo 6, a concepção do modelo de referência dos processos e a selecção dos processos incluídos no MRP<sub>2MPspe</sub> teve em consideração as propostas do CMM. Verifica-se então que, em termos globais, o 2MPspe não inclui as áreas chave do processo, coordenação intergrupos e revisão por pares, características do nível 3 do CMM e a gestão de subcontratações do nível 2. Em relação à gestão de subcontratações, é a única KPA não obrigatória do CMM. A revisão por pares foi considerada como uma possível estratégia a utilizar no processo de garantia da qualidade. Finalmente, a coordenação intergrupos foi considerada menos prioritária no contexto das PES.

A quinta característica (e) refere-se ao cumprimento dos requisitos da norma ISO/IEC 9001:2000. Como foi apresentado no capítulo 6, o processo de gestão da qualidade (GES.3) foi incluído especificamente para realizar o alinhamento com a norma ISO 9001:2000. O processo GES.3 implica o estabelecimento de um sistema de gestão da qualidade para monitorizar a qualidade do produto e do processo e assim assegurar a satisfação do cliente.

---

<sup>3</sup> parte normativa.



O sexto aspecto (f) engloba o envolvimento dos intervenientes no processo. Esta característica foi derivada do estudo sobre os factores críticos de sucesso nas iniciativas de MPS, em que se verificou a importância do compromisso e envolvimento dos recursos humanos. Este aspecto foi tratado incluindo o sub-nível 1.5 nos níveis de capacidade do processo. Este sub-nível designa-se “processo valorizado” e pretende assegurar que os intervenientes do processo reconhecem o valor acrescentado para o negócio resultante de uma efectiva realização do processo. Considera-se que para o processo ser gerido (nível 2) tem que ser inicialmente executado (nível 1) e em seguida valorizado (sub-nível 1.5).

A sétima característica (g) refere-se à duração reduzida da iniciativa de MPS. Como foi referido o M<sub>i</sub>2MP estima que são necessárias duas semanas para o ciclo inicial de melhoria, ou seja, o tempo entre o arranque do projecto e a entrega do plano de acção. Considera-se que é uma duração aceitável, mesmo para o contexto das PES.

Finalmente, a última característica (h) estava ligada à simplicidade e robustez do 2MPspe. Embora estes atributos apresentem alguma subjectividade, considera-se que o método de implementação proposto no 2MPspe é simples e tem as tarefas bem formalizadas. Por outro lado, verifica-se que a arquitectura proposta na componente M<sub>c</sub>2MP é robusta, em grande parte devido ao alinhamento com a norma recém publicada ISO/IEC 15504.

Considera-se que a primeira versão do 2MPspe possui as características desejadas, mas antes de continuar a sua evolução é necessário testar a arquitectura de base e o modelo de uma forma global.

### **9.3. Trabalho Futuro**

O trabalho aqui apresentado culminou na primeira versão do 2MPspe. Apresenta-se nesta secção um conjunto de projectos e propostas que visam a evolução do modelo e a promoção da sua utilização nas pequenas empresas de software. Como

foi referido, é necessário, numa primeira fase testar a arquitectura de base e as três componentes numa forma global.

Numa segunda fase considera-se importante a sua evolução e transferência para o meio empresarial através dos seguintes projectos:

- Desenvolver uma ferramenta baseada em PC para suportar as fases propostas no M<sub>1</sub>2MP. Os objectivos são simplificar a recolha e tratamento dos dados e automatizar algumas tarefas do método.
- Utilizar a simulação dinâmica para construir cenários permitindo analisar de forma detalhada os benefícios da MPS nos factores de qualidade. O objectivo é utilizar o paradigma analítico para conseguir estimar quantitativamente os efeitos das diversas acções de melhoria.
- Integrar no 2MPspe uma componente relacionada com a área de melhoria do processo pessoal. O objectivo é facilitar a melhoria do processo organizacional através da difusão de uma cultura de melhoria pessoal.
- Desenvolver um método simples de auto-avaliação, sem recurso a profissionais externos. Preferencialmente utilizando a Internet e fornecendo dados fidedignos de *benchmark*.
- Formalizar a influência dos factores de contexto nos atributos da capacidade<sup>4</sup>. Embora esta formalização ultrapasse os objectivos de um modelo de MPS, uma vez que definirá o “como” executar as tarefas, parece interessante orientar a abordagem à gestão do conhecimento e o tipo de cultura organizacional específicos de cada nível de capacidade. Será um pouco definir o balanço “ideal” entre flexibilidade e rigidez nos processos.
- Promover iniciativas de aplicação do 2MPspe em larga escala realçando o seu mapeamento para o CMM e para a ISO 9001:2000. O objectivo é melhorar o

---

<sup>4</sup> no referencial tridimensional proposto consistirá em estudar os eixos C × K (capacidade × conhecimentos) sem a dimensão dos processos.

processo de desenvolvimento de software das PES de uma forma efectiva. E disseminar a cultura da melhoria contínua.

#### **9.4. Considerações Finais**

A qualidade do software tem recebido muita atenção recentemente tanto no meio empresarial, como nas universidades. Este projecto de investigação centrou-se numa das abordagens mais comuns aos problemas da qualidade do software, ou seja, a melhoria do processo de desenvolvimento de software.

Após a identificação de algumas restrições e problemas relacionados com os modelos de MPS mais divulgados definiu-se o seguinte propósito para este trabalho: desenvolver um modelo de melhoria do processo de software adaptado às pequenas empresas de software na área de sistemas de informação de gestão e que trate explicitamente os factores contextuais da cultura organizacional e da gestão do conhecimento. Foi identificado também um conjunto de características desejáveis para o novo modelo.

O 2MPspe é o contributo principal deste trabalho de doutoramento, considerando-se que foi atingido o propósito enunciado. Numa perspectiva detalhada, considera-se que o 2MPspe cumpre as oito características desejáveis para um novo modelo de MPS.

Para a consecução deste trabalho foram desenvolvidos seis objectivos parciais com tarefas e resultados esperados. Como foi apresentado na secção 9.1 considera-se que estes objectivos foram também atingidos.

Como em qualquer trabalho de doutoramento foram sentidas algumas dificuldades principalmente em três vertentes. Por um lado a emergência da área de MPS e em termos práticos o longo período de evolução da norma ISO nesta área. Inicialmente foi utilizada a versão de 1995 do SPICE, numa 2ª fase foram utilizadas as

9 partes da ISO/IEC TR 15504 de 1998 e finalmente a norma recentemente editada ISO/IEC 15504 de 2004.

Noutra vertente a abrangência da área de MPS. Embora seja possível identificar as áreas de conhecimento mais directamente relacionadas, ou seja, o processo de engenharia de software e a qualidade do software. A MPS é horizontal à maioria das áreas do conhecimento definidas no SWEBOK [2001] englobando – numa perspectiva de diagnóstico e melhoria – todos os processos do ciclo de vida do software.

Finalmente, o cariz prático que se pretendeu associar ao 2MPspe. A turbulência do ambiente e a maturidade das pequenas empresas de software não facilita o desenvolvimento de casos de estudo.

Detalhados os contributos deste projecto, considera-se que foi satisfeita a finalidade que justificou a sua realização e espera-se ter contribuído de forma efectiva para o enriquecimento da disciplina de engenharia de software.

## Bibliografia

---

- Aaen, I. (2003). "Software Process Improvement: Blueprints versus Recipes." *IEEE Software*, 20(5), 86-93.
- Aaen, I., Arendt, J., Mathiassen, L. e Ngwenyama, O. (2001). "A Conceptual MAP of Process Improvement." *Scandinavian Journal of Information Systems*, 13.
- Abdel-Hamid, T. e Madnick, E. (1982). "A Model of Software Project Management Dynamics." *6ª COMPSAC*.
- Abdel-Hamid, T. e Madnick, E. (1991). *Software Project Dynamics: An Integrated Approach*, Prentice-Hall.
- Ambler, S. (2000). "Completing the Unified Process With Process Patterns." Ambysoft.
- Apostolou, D. e Mentzas, G. (2003). "Experiences from Knowledge Management Implementations in Companies of the Software Sector." *Business Process Management*, 9(3), 354-381.
- Arthur, J. e Nance, R. (1990). "A Framework for Assessing the Adequacy and Effectiveness of Software Development Methodologies." *Proc. 15ª S.E. Workshop*.
- Balla, K., Bemelmans, T., Kusters, R. e Trienekens, J. (2001). "Quality through Managed Improvement and Measurement (QMIM): Towards a Phased Development and Implementation of a Quality System for a Software Company." *Software Quality Journal*, 9, 177-193.
- Basili, V. (2004). "New Year's Resolutions for Software Quality." *IEEE Software*, 21(1), 12-13.
- Basili, V. e Rombach, H. (1988). "The TAME Project: Towards Improvement-Oriented Software Environments." *IEEE Transactions on Software Engineering*, 14(6), 759-773.
- Baskerville, R. e Pries-Heje, J. (1999). "Managing Knowledge Capability and Maturity." *Information Systems: Current Issues and Future Changes*, J. Larsen T; Levine L. e Degron, ed., IFIP/Kluwer, 175-196.
- Beck, K. (2000). *Extreme Programming Explained*, Addison-Wesley.
- Bell, C. (1994). "Trillium - Model for Telecom Product Development & Support Process Capability." *MPI6*, Bell Canada.
- Blanco, M., Gutierrez, P. e Satriani, G. (2001). "SPI Patterns: Learning from Experience." *IEEE Software*, 18(3), 28 - 35.
- Boehm, B. (1976). "Software Engineering." *IEEE Transactions on Computers*, C-25(12).
- Boehm, B. (1988). "A Spiral Model for Software Development and Enhancement." *Computer*, 21(5), 61-72.
- Boehm, B. (2004). "New Year's Resolutions for Software Quality." *IEEE Software*, 21(1), 12-13.
- Boehm, B. e Port, D. (2001). "Balancing Discipline and Flexibility With the Spiral Model and MBASE." *CROSSTALK - The Journal of Defence Software Engineering*, Dezembro, 23-28.
- Boisot, M. H. (1995). *Information Space*, International Thompson Business, London.

- Booch, G., Rumbaugh, J. e Jacobson, I. (1999). *The Unified Modelling Language - User Guide*, Addison-Wesley.
- Bourque, P. e al., e. (2002). "Fundamental Principles of Software Engineering - A journey." *Journal of Systems and Software*, 62, 59-70.
- Brocka, B. e Brocka, S. (1992). *Quality Management: Implementing the Best Ideas of the Masters*, Business One Irwin.
- Brockers, A., Differding, C. e Trein, G. (1996). "The Role of Software Process Modeling in Planning Industrial Measurement Programs." *Proc. of Metrics ICSE*.
- Broh, R. (1982). *Managing Quality for Higher Projects*, McGraw-Hill.
- Brooking, A. (1997). "The Management of Intellectual Capital." *Journal of Long Range Planning*, 30(3), 364-365.
- Brooks, F. (1987). "No Silver Bullet: Essence and Accidents Software Engineering." *Computer*, 20(7), 10-19.
- BS. (1986). *BS 7830-96, BS 7830: Guide to the Design and Preparation of On-Screen Documentation for Uses of Application Software*, British Standard Institute.
- Butler, T. e Fitzgerald, B. (1999). "A Review and Application of the CSF Concept for Research on the IS Development Process." Information Systems - The Next Generation, L. Brooks and C. Kimble, eds., McGraw-Hill.
- Card, D. (1991). "Understanding Process Improvement." *IEEE Software*, 8(4), 102-103.
- Castells, M. (1996). *The Rise of the Network society*, Blackwell.
- Cavano, P. e McCall, J. (1978). "A Framework for the Measurement of Software Quality." *ACM SQA Conference*, 133-139.
- Charette, R. (1989). *Software Engineering Risk Analysis and Management*, McGraw-Hill.
- Christie, A. (1999). "Simulation in Support of CMM-Based Process Improvement." *Journal of Systems and Software*, 46, 107-112.
- Coad, P., Lefevre, E. e Deluca, J. (1999). *Java Modelling in Colour With UML*, Prentice Hall.
- Cockburn, A. (2002a). *Agile Software Development*, Addison-Wesley.
- Cockburn, A. (2002b). "Agile Software Development Joins the "Would-Be" Crowd." *Cutter IT Journal*, 15(1), 6-12.
- Cohn, M. e Ford, D. (2003). "Introducing an Agile Process to an Organization." *IEEE Computer*, 36(6), 74-78.
- Combelles, A. e De Marco, T. (1998). "Viewpoint, Software in Focus." *ESSI News Project Team*, Março(1).
- Conradi, R. e Fuggetta, A. (2002). "Improving Software Process Improvement." *IEEE Software*, 19(4), 92 - 99.
- Cosgriff, P. (1999). "The Journey to CMM level 5: A time line." *CrossTalk*, 12(5).
- Coyle, G. (1979). *Management System Dynamics*, John Wiley & Sons.
- Coyle, R. (1996). *Systems Dynamics Modelling - A Practical Approach*, Chapman & Hall.
- Craigmyle, M. (1998). "Process Assessment, Using SPICE: The rating framework." *SPICE - The Theory and Practice of Software Process Improvement and Capability Determination*, K. El Emam, J. Drouin, and W. Melo, eds., IEEE Computer society, 123-138.
- Crosby, P. (1979). *Quality is Free*, McGraw-Hill.
- Curtis, B., Kellner, M. e Over, J. (1992). "Process Modelling." *Communications of the ACM*, 35(9), 75-90.

- Curtis, W. (1995). "Building a Cost-Benefit Case for SPI." *7th Software Engineering Process Group*, Boston.
- Daskalantonakis, M. (1992). "A Practical View of Software Measurement and Implementation Experiences Within Motorola." *IEEE Trans. on Software Eng.*, 18(11), 998-1010.
- Davis, A. (2004). "New Year's Resolutions for Software Quality." *IEEE Software*, 21(1), 12-13.
- Davis, A., Bersoff, E. e Comer, E. (1988). "A Strategy for Comp. Alternative Software Development Life Cycle Models." *IEEE Trans. on Software Eng.*, 14(10), 1453-1460.
- Deming, E. (1982). *Out of Crisis*, MIT Press.
- Deming, W. (1993). *The New Economics for Industry, Government, Education*, MIT, Center of Advanced Engineering.
- Diaz, M. e Sligo, J. (1997). "How SPI Helped Motorola." *IEEE Software*, 14(5), 75-81.
- Dion, R. (1993). "Process Improvement and the Corporate Balance Sheet." *IEEE Software*, 10(4), 28-35.
- Drehmer, D. e Dekleva, S. (1993). "Measuring Software Engineering Maturity: A Rasch Calibration." *International Conference on Information Systems*, 191-202.
- Dunaway, D. e Masters, S. (2001). "CMM-Based Appraisal for Internal Process Improvement." *CMU/SEI-2001-TR-033*, SEI.
- Dyba, T. (2000). "An Instrument for Measuring the Key Factors of Success in Software Process Improvement." *Empirical Software Engineering*, 5, 357-390.
- Dyba, T. (2002). "Enabling Software Process Improvement: An Investigation of the Importance of Organizational Issues." *Empirical Software Engineering*, 7, 387-390.
- Easterby-Smith, M., Thorpe, R. e Lowe, A. (1991). *Management Research - An Introduction*, SAGE Publications, London.
- El Emam, K. (1998). "The Internal Consistency of the ISO/IEC 15504 Software Process Capability Scale." *Software Metrics Symposium*, Maryland, 72-81.
- El Emam, K., Fusaro, P. e Smith, B. (1999). "Success Factors and barriers for SPI." *Better Software Practice for Barriers Benefit: Principles and Experience*, C. Tully and R. Messnarz, eds., IEEE Computer Society, 355-371.
- El Emam, K. e Goldenson, D. (1999). "An Empirical Review of Software Process Assessments." *NRC 43610*, NRC-CNRC Canada.
- El Emam, K. e Madhavji, N. (1996). "An Instrument for Measuring the Success of the Requirements Engineering Process in Information Systems Development." *Empirical Software Engineering*, 1(3).
- Emmerich, W., Finkelstein, A., Montangero, C., Antonelli, S., Armitage, S. e Stevens, R. (1999). "Managing Standards Compliance." *IEEE Transactions on Software Engineering*, 25(6), 836 - 851.
- ESPI. (1999). "Risks in SPI Programs." *European Software Process Improvement*.
- Evans, P. (2002). "Software Process Improvement through Process Customization." *Black Diamond Software*.
- Evans, P. (2003). "A Business Case of Software Process Improvement." *Black Diamond Software*.
- Fayad, M. e Laitinen, M. (1997). "Process Assessment Considered Wasteful." *Communications of the ACM*, 40(11), 125-128.
- Fayad, M. e Laitinen, M. (1998). *Transition to software Development Object - Oriented Software Development*, Wiley.

- Ferguson, P., Humphrey, W., Khajenoori, S., MacKe, S. e Matvja, A. (1997). "Results of applying the PSP." *IEEE Computer*, 30(5), 24-31.
- Fitzgerald, B. e O'Kane, T. (1999). "A longitudinal Study of Software Process improvement." *IEEE Software*, 16(3), 37-45.
- Florac, W. e Carleton, A. (1999). *Measuring the Software Process - Statistical Process Control for SPI*, Addison Wesley.
- Forrester, J. (1961). *Industrial Dynamics*, MIT Press.
- Fowler, K. (1997). "SEI CMM Level 5: A Practitioner Perspective." *CrossTalk*, 10(9).
- Fowler, M. (2001). "The new Methodology."
- Fryman, M. (2002). *Quality and Process Improvement*, Delmar - Thomson Learning.
- Gasston, J. e Halloran, P. (1999). "Continuous Software Process Improvement Requires Organisational Learning: An Australian Case Study." *Software Quality Journal*, 8, 37-51.
- Glass, R. (1998). *Software Runaways*, Prentice-Hall.
- Goldenson, D. e Herbsleb, J. (1995). "After the Appraisal: A Systematic Survey of Process Improvement, Its Benefits, and Factors that Influence Success." *CMU/SEI-95-TR-009*, SEI.
- Greenberger, M., Crensen, M. e Crissy, B. (1976). *Models in the Policy Process*, Russel Sage Foundation.
- Guilford, J. (1954). *Psychometric Methods*, McGraw-Hill.
- Haley, T. (1996). "Software Process Improvement at Raytheon." *IEEE Software*, 13(6), 33-41.
- Hall, T., Rainer, A. e Baddoo, N. (2003). "Implementing Software Process Improvement: An Empirical Study." *Software Process Improvement and Practice*, 7, 3-15.
- Hammer, M. e Champy, J. (1993). *Reengineering the Corporation*, Nicholas Breley Publishing.
- Hansen, T., Morten, N. e Tierney, T. (1999). "What's Your Strategy for Managing Knowledge?" *Harvard Business Review*, Março, 106-116.
- Harrington, H. (1991). *Business Process Improvement*, Mc Graw-Hill.
- Harter, D., Krishnan, M. e Slaughtert, S. (2000). "Effects of Process Maturity on Quality, Cycle Time, and Effort in Software Product Development." *Management Science*, 46(4), 451-466.
- Herbsleb, J., Zubrow, D., Siegel, J., Rozum, J. e Carlton, A. (1994). "Software Process Improvement: State of the Payoff." *American Programmer*, 7(9).
- Highsmith, J. (2000). "Adaptative Software Development (ASD) Overview."
- Highsmith, J. (2002a). "What Is Agile Software Development?" *CROSSTALK - The Journal of Defense Software Engineering*.
- Highsmith, J. (2002b). "Today, a New Debate Rages: Agile Software Development versus Rigorous Software Development." *Cutter IT Journal*, 15(1), 1.
- Higuera, R. (1995). "Team Risk Management." *CrossTalk*.
- Hofstede, G. (1984). *Culture's Consequences: International Differences in Work-Related Values*, Sage Publications.
- Hofstede, G. (1991). *Culture and Organizations: Software of the Mind*, McGraw-Hill.
- Horvat, R., Rozman, I. e Gyorkos, J. (2000). "Managing Complexity of SPI in Small Companies." *Software Process Improvement and Practice*, 5, 45-54.
- Huff, K. (1996). "Software Process Modelling." *Software Process*, F. e. Wolf, ed., John Wiley & Sons.
- Humphrey, W. (1988). "Characterizing the Software Process: a Maturity Framework." *IEEE Software*, 5(2), 73-79.



- Humphrey, W. (1989). *Managing the Software Process*, Addison-Wesley.
- Humphrey, W. (1996). "Using a Defined and Measured Personal Software Process." *IEEE Software*, 13(3), 77-88.
- Humphrey, W. (1997). *Introduction to the Personal Software Process*, Addison-Wesley.
- Humphrey, W. (2004). "New Year's Resolutions for Software Quality." *IEEE Software*, 21(1), 12-13.
- IEEE. (1991). *Std 610.12 - Glossary of Software Engineering Terminology*.
- IEEE. (1998). *Std 1219: Standard for Software Maintenance*.
- ISO/IEC. (1994). *8402: Quality Management and Quality Assurance - Vocabulary*.
- ISO/IEC. (1995). "12207 (DIS): 1995 - Information Technology -- Software Life Cycle Processes."
- ISO/IEC. (1998). "14598 -1 (IS): Information Technology -- Software Product Evaluation -- Part 1: General Overview." *N1914*, ISO/IEC JTC1/SC7.
- ISO/IEC. (1999). "9126 - 1 (FDIS): Software Engineering - Product Quality - Part 1: Quality Model." *2228*, ISO/IEC JTC1/SC7.
- ISO/IEC. (2000a). *14764: Software Engineering - Software Maintenance*.
- ISO/IEC. (2000b). *9000: 2000, Quality Management Systems - Fundamentals and Vocabulary*.
- ISO/IEC. (2001). "15939 (FCD): Software Engineering - Software Measurement Process Framework." *N2410*, ISO/IEC JTC1/SC7.
- ISO/IEC. (2002a). "Feedback on ISO 9001:2000." *ISO Management Systems*.
- ISO/IEC. (2002b). "12207: 1995 / Amd 1:2002 - Software Engineering -- Software Life Cycle Processes." *2529r*, ISO/IEC JTC1/SC7.
- ISO/IEC. (2003). "15504 -5 (CD) - Information Technology - Process Assessment - Part 5: An Exemplar Process Assessment Model." *2887*, ISO/IEC.
- ISO/IEC. (2004a). "15504 -1 (FDIS) - Information Technology - Process Assessment - Part 1: Concepts and Vocabulary." ISO/IEC.
- ISO/IEC. (2004b). "15504 -2 - Software Engineering - Process Assessment - Part 2: Performing an Assessment." ISO/IEC.
- ISO/IEC. (2004c). "15504 -3 - Software Engineering - Process Assessment - Part 3: Guidance on Performing an Assessment." ISO/IEC.
- ISO/IEC. (2004d). "15504 -4 - Information Technology - Process Assessment - Part 4: Guidance on Use for Process Improvement and Process Capability Determination." ISO/IEC.
- Jacobson, I. (2002). "A Resounding "Yes" to Agile Processes - But Also to More." *Cutter IT Journal*, 15(1), 18-24.
- Jiang, J. e al., e. (2004). "An Exploration of the Relationship between Software Development Process Maturity and Project Performance." *Information & Management*, 41, 279-288.
- Johansen, J. e Mathiassen, L. (1998). "Lessons Learned in a National SPI Effort." *Euro SPI 98*, Suecia.
- Johnson, D. e Brodman, J. (1997). "Tailoring the CMM for Small Businesses, Small Organizations and Small Projects." *Software Process Newsletter*, 8, 1-5.
- Jones, C. (2003). "Variations in Software Development Practices." *IEEE Software*, 20(6), 22-27.
- Juran, J. (1988). *Juran on Planning for Quality*, Free Press MacMillan.
- Katzenbach, J. e Smith, D. (1993). *The Wisdom of Teams*, Harvard Business School Press.

- Kautz, K. e Thaysen, K. (2001). "Knowledge, Learning and IT Support in a Small Software Company." *European Conference on IS*, Bled, Eslovenia, 696-705.
- Kellner, M., Madachy, R. e Raffo, D. (1999). "Software Process Simulation Modelling: Why? What? How?" *Journal of Systems and Software*, 46, 91 - 105.
- King, J. e Diaz, M. (2002). "How CMM Impacts Quality, Productivity, Rework, and the Bottom Line." *CROSSTALK - The Journal of Defence Software Engineering*.
- Klein, R. (1991). "Achieve Total Quality Management." *Chemical Engineering Progress*, Novembro 1991.
- Kogut, B. e Zander, U. (1995). "Knowledge and the Speed of the Transfer and Imitation of Organizational Capabilities: an Empirical Test." *Organization Science*, 6(1), 76-91.
- Kolmel, B. e Eisenbiegler, J. (2000). "Impact Measurement of SPI-Projects." FZI, Karlsruhe, Alemanha.
- Kuvaja, P. (1999). "BOOTSTRAP 3.0 - A SPICE Conformant Software Process Assessment Methodology." *Software Quality Journal*, 8, 7 - 19.
- Kuvaja, P., Palo, J. e Bicego, A. (1999). "TAPISTRY - A SPI Approach for Small Enterprises." *Software Quality Journal*, 8, 149-156.
- Kuvaja, P., Simila, J., Krzanik, L., Bicego, A., Koch, G. e Saukkonem. (1994). *Software Process Assessment and Improvement. The BOOTSTRAP Approach*, Blackwell Business.
- Leveson, N. (2004). "New Year's Resolutions for Software Quality." *IEEE Software*, 21(1), 12-13.
- Lindberg, K. (1999). "Software Development Perceptions about Software Project Failure: a Case." *The Journal of Systems and Software*, 49(2-3), 177-192.
- Lions, J. (1997). "Arienne 5."  
<http://www.ima.umn.edu/~arnold/disasters/ariane5rep.html>.
- Lipke, W. e Butler, K. (1992). "SPI: A Success Story." *CrossTalk*, 38.
- Lissitz, R. e Green, S. (1975). "Effects of the Number of Scale Points on Reliability: A Monte Carlo Approach." *Journal of applied Psychology*(60), 10-13.
- Lundberg, C. (1989). "Working with Culture." *Journal of Organizational Change Management*, 1, 38-47.
- Madachy, R. (1996). "System Dynamics Modelling of an Inspection-Based Process." *Proceedings of ICSE-1996*, 376-386.
- Marr, B., Gupta, O., Pike, S. e Roos, G. (2003). "Intellectual Capital and Knowledge Management Effectiveness." *Management Decision*, 41(8), 771 - 781.
- Marshall, L. (1997). "Facilitating Knowledge Management and Knowledge Sharing: New Opportunities for Information Professionals." *Online*, 21(5), 92-98.
- Mathiassen, L. e Pourkomeylian, P. (2003). "Managing Knowledge in a Software Organization." *Journal of Knowledge Management*, 7(2), 63 - 80.
- McAdam, R. e McCreedy, S. (1999). "A Critical Review of Knowledge Management Models." *The Learning Organization*, 6(3), 91-100.
- McCall, J., Richards, P. e Walters, P. (1977). *Factors in Software Quality*, NTIS AD-A049--014.
- McDermid, J. (1991). *Software Engineering Reference Book*, Butterworth-Heinemann Ltd.
- McFeely, B. (1996). "IDEAL - A User Guide for Software Process Improvement." *CMU/SEI-96-HB-001*, SEI.
- McGarry, F. (2001). "What is a level 5?" *Proc. 26th Annual NASA SE Workshop*, 83-90.
- McGarry, F. e Decker, B. (2002). "Attaining Level 5 in CMM Process Maturity." *IEEE Software*, 19(6), 87 - 96.

- Mead, N. (2004). "New Year's Resolutions for Software Quality." *IEEE Software*, 21(1), 12-13.
- Messnarz, R. (1997). "A Comparison of BOOTSTRAP and SPICE." *Software Process Newsletter*, 8, 6-8.
- Moneymaker, P. (1995). *Software Life Cycles Guidebook*.
- Montgomery, D. (1996). *Introduction to SPC*, John Wiley & Sons.
- Morgan, G. (1997). *Images of Organization*, Sage.
- Nance, R. e Arthur, J. (2002). *Managing Software Quality*, Springer.
- Ngwenyama, O. e Nielsen, P. (2003). "Competing Values in SPI: An Assumption Analysis of CMM from an Organizational Culture Perspective." *IEEE Transactions on Software Engineering*, 50(1), 100-112.
- Niazi, M., Wilson, D. e Zowghi, D. (2003). "A Maturity Model for the Implementation of Software Process Improvement: an Empirical Study." *Journal of Systems and Software*, 1-18.
- Nielsen, P. e Norbjerg, J. (2001). "Software Process Maturity and Organizational Politics." *IFIP WG 8.2*.
- Nonaka, I. e Takeuchi, K. (1995). *The Knowledge Creating Company: How Japanese Companies Create the Dynamics of Innovation*, Oxford University Press.
- Oldham, L., Putman, G., Peterson, M., Rudd, B. e Tjoland, K. (1999). "Benefits Realized from Climbing the CMM Ladder." *CrossTalk*, 12(5).
- Orci, T. (2000). "Capability Maturity Model for Extra Extra Small Organizations - Level 2." Umea University.
- Orci, T. e Laryd, A. (2000). "Dynamic CMM for Small Organizations - Implementation Aspects." Umea University.
- Orr, K. (2002). "CMM versus Agile Development: Religious Wars and Software Development." *Cutter Consortium - Executive Report*, 3(7).
- Pall, G. (1987). *Quality Process Management*, Prentice-Hall.
- Parnas, B. (2004). "New Year's Resolutions for Software Quality." *IEEE Software*, 21(1), 12-13.
- Paulk, M. (1995). "The Evolution of the SEI's Capability Maturity Model for Software." *Software Process: Improvement and Practice*, 1(1), 3-15.
- Paulk, M. (2002). "Agile Methodologies and Process Discipline." *CROSSTALK - The Journal of Defense Software Engineering*, 15(10), 15-18.
- Paulk, M., Chrissis, M., Weber, C. e Shrum, S. (1997). "A Preview of the Software CMM Version 2." Software Engineering Institute.
- Paulk, M., Weber, C., Curtis, B. e Chrissis, M. (1995). *The Capability Maturity Model: Guidelines for improving the software process*, Addison-Wesley.
- Pettigrew, A. (1979). "On Studying Organizational Cultures." *Administration Science Quarterly*, 24, 570-581.
- Pfleeger, S. (1998). *Software Engineering - Theory and Practice*, Prentice - Hall.
- Pfleeger, S. (2004). "New Year's Resolutions for Software Quality." *IEEE Software*, 21(1), 12-13.
- Pigoski, T. (1997). *Practical software Maintenance: Best Practices for Managing your Software Investment*, Wiley.
- Pike, S. e Roos, G. (2002). "Measuring the Impact of Knowledge Management in Companies." *5th World Congress of Intellectual Capital*, Hamilton.
- Pitterman, B. (2000). "Telcordia Technologies: the Journey to High Maturity." *IEEE Software*, 17(4), 89-96.
- PMI. (2000). *A Guide to the Project Management Body of Knowledge (PMBOK Guide)*, Project Management Institute.

- PMI. (2003). *Organizational Project Management Maturity Model (OPM3)*, Project Management Institute.
- Polanyi, M. (1962). *Personal Knowledge: Towards a Post Critical Philosophy*, Torchbooks, New York.
- Poppendieck, M. (2001). "Lean Programming: Part 2." *Software Development*, Junho.
- Pressman, R. (1997). *Software Engineering - A Practitioner's Approach*, McGraw Hill.
- Quinn, R. e McGrath, M. (1985). "The Transformation of Organizational Cultures: A Competing Values Perspective." *Organizational Culture*, Frost and e. al., eds., Sage, 315-334.
- Quintas, P., Anderson, P. e Filkelstein, S. (1996). "Managing Professional Intellect: Making the Most of the Best." *Harvard Business Review*, 74(Março-Abril), 71-80.
- Quintas, P., Lafrere, P. e Jones, G. (1997). "Knowledge Management: a Strategic Agenda." *Journal of Long Range Planning*, 30(3).
- Raccoon, L. S. B. (1995). "The Chaos Model and the Chaos Life Cycle." *ACM Software Engineering Notes*, 20(1), 55-66.
- Rainer, A. e Hall, T. (2001). "An Analysis of Some "Core Studies" of SPI." *Software Process Improvement and Practice*, 6, 169-187.
- Rainer, A. e Hall, T. (2002). "Key Success for Implementing Software Process Improvement: a Maturity-Based Analysis." *Journal of Systems and Software*, 62, 71-84.
- Rainer, A. e Hall, T. (2003). "A Quantitative and Qualitative Analysis of Factors Affecting Software Processes." *Journal of Systems and Software*, 66, 7-21.
- Randell, B. (1989). "Airbus A320." *The Risk Digest: Forum on Risks to the Public in Computers and Related Systems*, 8(57).
- Ravichandran, T. e Rai, A. (2003). "Structural Analysis of the Impact of Knowledge Creation and Knowledge Embedding on Software Process Capability." *IEEE Transactions on Engineering Management*, 50(3), 270-284.
- Redmill, F. (1997). *Software Projects - Evolutionary vs Big-Bang Delivery*, Wiley.
- Richardson, G. e Pugh, A. (1981). *Introduction to System Dynamics Modelling*, Productivity Press.
- Richardson, I. (2002). "SPI Models: What Characteristics are Required for Small Software Development Companies?" *Software Quality Journal*, 10, 101-114.
- Roberts, E. (1964). *The Dynamics of Research and Development*, Harper & Row.
- Rocha, A. e Vasconcelos, J. (2003). "Reflexão sobre a Aplicação de Modelos do Software Engineering Institute no Diagnóstico e na Melhoria da Maturidade do Processo de Desenvolvimento de Software." *4º CAPSI*, Porto.
- Rodrigues, A. (2000). "Business Modelling: The Road to Business-IT Alignment." *Business-IT Strategies Advisory Service*, 3(6), 1-25.
- Rodrigues, A. e Williams, T. (1997). "Systems Dynamics in Software Project Management: Towards the Development of a Formal Integrated Framework." *European Journal of Information Systems*, 51-66.
- Rombach, H. e Ulery, B. (1989). "Improving Software Maintenance Through Measurement." *Proceedings of IEEE*, 77(4), 581-595.
- Rout, T. e Simms, P. (1998). "Introduction to the SPICE Documents and Architecture." *SPICE - The Theory and Practice of Software Process Improvement and Capability Determination*, K. El Eman, J. Drouin, and W. Melo, eds., IEEE Computer society.
- Rout, T., Tuffley, A., Cahill, B. e Hodgen, B. (2000). "The Rapid Assessment of Software Process Capability." *SPICE 2000*, 47 - 56.

- Royce, W. (1970). "Managing the Development of Large Software Systems." *Wescon - IEEE*.
- Rubin, H., Johnson, M. e Yourdon, E. (1995). "Software Process Flight Simulation - Dynamic Modelling Tools and Metrics." *Information Systems Management*, 40-52.
- Ruiz, M., Ramos, I. e Toro, M. (2002). "A Dynamic Integrated Framework for SPI." *Software Quality Journal*, 10, 181-194.
- Russ, M. e McGregor, J. (2000). "A Software Development Process for Small Projects." *IEEE Software*, 17(5), 96 - 101.
- Sage, A. e Palmer, J. (1990). *Software Systems Engineering*, Wiley.
- Sassenburg, H. (2001). "Assessing Readiness for (Software) Process Improvement." Se-Cure Ag - <http://www.se-cure.ch>.
- Schein, E. (1985). *Organizational Culture and leadership: A Dynamic View*, Jossey-Bass.
- Schneider, K., von Hunnius, J. e Basili, V. (2002). "Experience in Implementing a Learning Software Organization." *IEEE Software*, 19(3), 46 - 49.
- Schwaber, K. (2001). "Will the Real Agile Processes Please Stand Alone." *E-Project Management Advisory Service, Executive Report*, 2(8), 1-22.
- Schwaber, K. e Beedle, M. (2002). *Agile Software Development With Scrum*, Prentice Hall.
- SEI. (2004). "Process Maturity Profile of the Software Community 2003 Year End Update." Software Engineering Institute.
- Siakas, K. e Georgiadou, E. (2002). "Empirical Measurement of the effects of Cultural Diversity on Software Quality Management." *Software Quality Journal*, 10, 169-180.
- Smircich, L. (1983). "Concepts of Culture and Organizational Analysis." *Administration Science Quarterly*, 28, 339-358.
- Sommerville, I. (2001). *Software Engineering*, Addison - Wesley.
- Sommerville, I. e Sawyer, P. (1997). *Requirements Engineering: A Good Practice Guide*, John Wiley and Sons.
- Spence, I. (2002). "Principles of Process Improvement." *The Rational Magazine*.
- SPICE. (1995). "SPICE - Software Process Assessment (v1.0): partes 1 até 9." ISO/IEC.
- SPICE. (1998). "Phase 2 Trials Interim Report." *versão 1.00*.
- SPIRE, T. (1998). *The SPIRE Handbook - Better, Faster, Cheaper Software Development in Small Organizations*.
- Standish. (1995). "Chaos - The State of the Software Industry." Standish Group international.
- Stapleton, J. (1998). *Dynamic Systems Development Method: The Method in Practice*, Addison-Wesley.
- Stewhart, W. (1931). *Economic Control of Quality Manufactured Product*, Van Nostrand.
- Swan, J., Newell, S., Scarbrough, H. e Hislop, D. (1999). "Knowledge Management and Innovation: Networks and Networking." *Journal of Knowledge Management*, 3(4), 262-275.
- SWEBOK. (2001). "Guide to Software Engineering Body of Knowledge (SWEBOK - Trial Version, 0.95)." IEEE.
- Taylor e al., e. (1997). *International Journal of Technology Management*, 11(3), 385-391.
- Taylor, E. S. (1959). "An Interim Report on Engineering Design." Massachusetts Institute of Technology, Cambridge.

- Thayer, R. e Dorfman, M. (1997). *Software Requirements Engineering*, IEEE Computer Society Press.
- Tichy, W. (1998). "Should Computer Scientists Experiment More?" *IEEE Computer*, 31(5), 32-40.
- Trice, H. e Beyer, J. (1984). "Studying Organizational Cultures through Rites and Ceremonials." *Academic Management Review*, 9, 653-669.
- Tricker, R. (2001). *ISO 9001:2000 for Small Businesses*, Butterworth - Heinemann.
- Tvdet, J. e Collofello, J. (1995). "Evaluating the Effectiveness of Process Improvements on the Software Development Cycle Time via System Dynamics Modelling." *IEEE Software*, 318-325.
- Van de Ven, A. e Ferry, D. (1998). *Measuring and Assessing Organizations*, John Wiley & Sons.
- Venzin, M., von Krogh, G. e Roos, J. (1998). "Future Research into Knowledge Management." Knowing firms, R. e. K. von Krogh, ed., Sage Publications.
- Villalon, J., Agustin, G., Gilabert, T., Seco, A., Sanchez, L. e Cota, M. (2002). "Experiences in the Application of Software Process Improvement in SMES." *Software Quality Journal*, 10, 261-273.
- Wang, Y. (2001). "Software Engineering Standards: Review and Perspectives." University Calgary, Alberta, Canada.
- Wheeler, D. e Poling, S. (1998). *Building Continual Improvement*, SPC Press.
- Wiegers, K. (1999a). "A Software Metrics Primer." *Software Development Magazine*, Julho.
- Wiegers, K. (1999b). "Software Process Improvement in Web Time." *IEEE Software*, 16(4), 78 - 86.
- Wiegers, K. e Sturzenberger, D. (2000). "A Modular Software Process Mini-Assessment Method." *IEEE Software*, 17(1), 62 - 69.
- Wolstenholme, E. (1990). *System Enquiry*, John Wiley & Sons.
- Yamamura, G. (1999). "Software Process Satisfied Employees." *IEEE Software*, 16(5), 83-85.
- Yamamura, G. e Wigle, G. (1997). "SEI CMM Level 5: For the Right Reasons." *CrossTalk*, 10(8).
- Zahran, S. (1998). *Software Process Improvement - Pratical Guidelines for Business Success*, Addison Wesley.
- Zelkowitz, M. e Wallace, D. (1998). "Experimental Models for Validating Technology." *IEEE Computer*, 31(5), 23-31.

## Anexos

---

### Índice dos Anexos

Anexo A – Casos Práticos .....	279
Anexo A1 – Empresa Alfa (1999), Resumo dos Resultados .....	279
Anexo A2 – Empresa: Beta (2001), Resumo dos Resultados.....	281
Anexo A3 – Empresa: Gama (2002), Relatório do Diagnóstico .....	283
Anexo B – Selecção das Práticas de Base .....	304
Anexo C – Questionário 2MPspe .....	310
Anexo D – Tradução da Parte 2 da Norma ISO/IEC 15504.....	338

## Anexo A – Casos Práticos

No anexo A1 e A2 apresentam-se alguns dados obtidos do diagnóstico realizado nas empresas Alfa e Beta. No anexo A3 apresenta-se o relatório completo (com excepção dos anexos devido ao seu volume) da iniciativa de melhoria realizada na empresa Gama.

### Anexo A1 – Empresa Alfa (1999), Resumo dos Resultados

#### Resultados da fase do Diagnóstico dos Processos (ISO / IEC 15504):

Nome Processo	%P. base	Nível ISO 15504
CLI.3 Elicitação dos Requisitos	33,3%	0
CLI.4 Suporte	50,0%	0
CLI.5 Serviços ao cliente	33,3%	0
ENG.2 Requisitos	40,0%	0
ENG.3 Desenho	20,0%	0
ENG.4 Implementação do software	60,0%	0
ENG.5 Integração	60,0%	0
ENG.6 Testes	40,0%	0
ENG.7 Manutenção	60,0%	0
GES.1 Gestão do projecto	16,7%	0
GES.2 Gestão da qualidade	16,7%	0
GES.3 Gestão de risco	0,0	0
SUP.1 Documentação	50,0	0
SUP.2 Gestão de configurações	66,6%	0
SUP.3 Garantia de qualidade	16,7%	0
SUP.4 Verificação	0,0%	0
SUP.5 Validação	0,0%	0
SUP.8 Resolução de problemas	0,0%	0
ORG.1 Alinhamento organizacional	33,3%	0
ORG.2 Definição do Processo	0,0%	0
ORG.3 Recursos Humanos	0,0%	0
ORG.6 Reutilização	33,3%	0



**Resultados da fase de validação do diagnóstico (CMM)**

<b>NÍVEL 2: REPETÍVEL</b>	
Gestão de requisitos	33,3%
Planeamento do projecto de software	16,7%
Vigilância e acompanhamento do projecto de software	16,7%
Garantia da qualidade do software (SQA)	33,3%
Gestão de configurações	50,0%
<b>Nível 3: DEFINIDO</b>	
Foco no processo organizacional	0,0%
Definição do processo organizacional	0,0%
Programas de formação	16,7%
Gestão integrada do software	16,7%
Engenharia do produto de software	33,3%
Coordenação inter-grupos	0,0%
Revisões por pares	0,0%

## Anexo A2 – Empresa: Beta (2001), Resumo dos Resultados

### Resultados da fase do Diagnóstico dos Processos (ISO / IEC 15504):

<b>Nome Processo</b>	<b>%P. base</b>	<b>Nível ISO 15504</b>
CLI.3 Elicitação dos Requisitos	50,0	0
CLI.4 Suporte	75,0%	0
CLI.5 Serviços ao cliente	25,0%	0
ENG.2 Requisitos	100,0%	1
ENG.3 Desenho	50,0%	0
ENG.4 Implementação do software	100,0%	1
ENG.5 Integração	50,0%	0
ENG.6 Testes	25,0%	0
ENG.7 Manutenção	25,0%	0
GES.1 Gestão do projecto	75,0%	0
GES.2 Gestão da qualidade	50,0%	0
GES.3 Gestão de risco	0,0	0
SUP.1 Documentação	50,0	0
SUP.2 Gestão de configurações	100,0%	1
SUP.3 Garantia de qualidade	50,0%	0
SUP.4 Verificação	0,0%	0
SUP.5 Validação	0,0%	0
SUP.8 Resolução de problemas	25,0%	0
ORG.1 Alinhamento organizacional	25,0%	0
ORG.2 Definição do Processo	0,0%	0
ORG.3 Recursos Humanos	0,0%	0
ORG.6 Reutilização	75,0%	0

**Resultados da fase de validação do diagnóstico (CMM)**

<b>NÍVEL 2: REPETÍVEL</b>	
Gestão de requisitos	50,0%
Planeamento do projecto de software	75,0%
Vigilância e acompanhamento do projecto de software	75,0%
Garantia da qualidade do software (SQA)	50,0%
Gestão de configurações	100%
<b>Nível 3: DEFINIDO</b>	
Foco no processo organizacional	25,0
Definição do processo organizacional	0,0%
Programas de formação	0,0%
Gestão integrada do software	25,0%
Engenharia do produto de software	50,0%
Coordenação inter-grupos	0,0%
Revisões por pares	0,0%

## **Anexo A3 – Empresa: Gama (2002), Relatório do Diagnóstico**

### **Sumário Executivo**

Este relatório apresenta os resultados da etapa de planeamento do projecto de melhoria do processo de software actualmente a decorrer na Gama. Para cumprir esta etapa utilizou-se a abordagem proposta pelo modelo 2MPspe (Modelo de Melhoria do Processo de desenvolvimento de Software para Pequenas Empresas), executando-se as seguintes tarefas: levantamento do processo, diagnóstico, validação e definição das áreas de melhoria.

Conclui-se através do diagnóstico dos processos que a empresa está no nível 0 (denominado incompleto) da escala proposta pela norma ISO / IEC 15504, verificando-se que apenas o processo de manutenção está no nível 1. Em relação à validação do diagnóstico, conclui-se que a empresa está no nível 1 do CMM, denominado nível inicial. Em relação aos atributos do processo, no nível 2 obteve-se um valor de 32,8% de satisfação e no nível 3 um valor de apenas 13,7% de satisfação dos atributos.

Perante este quadro, definiu-se a meta global de atingir o nível 2 do CMM no prazo máximo de dois anos.

Para atingir a meta proposta, planeou-se uma estratégia com três ciclos, com durações de 11 meses, 6 meses e 5 meses, respectivamente. O 1º ciclo foi formalizado em detalhe e terá três Áreas de Intervenção principais, denominadas, Infra-estrutura do processo, Engenharia do produto e Sistema de qualidade. A AI da infra-estrutura terá uma duração de 4 meses e centra-se na definição do processo, Recursos Humanos e documentação. A área de engenharia do produto terá uma duração prevista de 3 meses e actuará nos processos de análise de requisitos, desenho e implementação do software. A

intervenção no sistema de qualidade terá também uma duração de 3 meses e centra-se na gestão e garantia da qualidade, na verificação e na validação.

O 2º e 3º ciclos foram apenas descritos globalmente, verificando-se que o 2º ciclo intervirá na área de gestão de projectos, nas vertentes de planeamento do projecto e acompanhamento e vigilância do projecto. O 3º ciclo terá como áreas chave a gestão de requisitos e a gestão de configurações.

Conclui-se que este deve ser um projecto de mudança da cultura organizacional, centrado na procura permanente da qualidade e na melhoria contínua dos processos.

## Índice

---

1. Introdução
  2. Método Utilizado
  3. Resultados da Avaliação
  4. Áreas a Melhorar
  5. Plano de Melhoria
  6. Conclusões
- Anexos

## 1. Introdução

Actualmente o desenvolvimento de software e os seus problemas inerentes são vistos na perspectiva do processo, em oposição à abordagem centrada no produto. É aceite pela comunidade de software que, a qualidade de um produto de software é largamente determinada pela qualidade do processo usado no seu desenvolvimento. Definindo-se o processo como o conjunto de actividades, métodos, práticas e transformações que os profissionais utilizam para desenvolver e manter o software e os produtos associados, é necessário considerar três aspectos, a sua definição, a sua aprendizagem e os resultados das actividades dos processos.

Este relatório apresenta os resultados da primeira etapa do projecto de Melhoria do Processo do Software, actualmente em curso na Gama. Esta primeira etapa, denominada etapa de Planeamento, pretende traçar um perfil detalhado da situação actual da organização, possibilitando a definição de prioridades e estratégias de melhoria.

Nesta perspectiva os objectivos principais desta etapa inicial foram:

- Levantamento dos processos da empresa
- Caracterização / diagnóstico dos processos
- Definição do perfil de maturidade
- Caracterização das áreas prioritárias de melhoria
- Definição do plano e estratégias de melhoria

Para concretizar estes objectivos foi utilizado o modelo 2MPspe (Modelo de Melhoria do Processo de desenvolvimento de Software para Pequenas Empresas), baseado na norma ISO / IEC 15504 – “Engenharia de Software, avaliação do processo” e no modelo *Capability Maturity Model* (CMM). A base destes dois modelos é o paradigma *benchmarking*, isto é, a utilização de “boas práticas” definidas através de organizações “excelentes” na área e caracterização comparativa dos processos da organização.

Este relatório descreve na segunda secção o método utilizado para o desenvolvimento do projecto. Segue-se uma apresentação e breve discussão dos resultados obtidos na avaliação dos processos da empresa. A secção quatro apresenta os processos prioritários a melhorar e as estratégias adoptadas e a secção seguinte define o plano de melhoria, atribuindo responsabilidades, definindo a sequência de actividades e clarificando

objectivos. Finalmente a secção seis, descreve as conclusões principais do projecto, deixando sugestões sobre o trabalho futuro e algumas orientações estratégicas sobre a concretização efectiva dos objectivos do projecto.

## **2. Método Utilizado**

A abordagem utilizada foi a proposta pelo modelo 2MPspe. O método de implementação do 2MPspe baseia-se no ciclo PDCA, Plan, Do, Check e Act e para executar a etapa de planeamento aqui descrita foram seguidas quatro fases.

A primeira fase, denominada no 2MPspe por Levantamento do Processo, baseia-se na norma ISO 12207 (ver anexo A), que fornece um referencial para o ciclo de vida do software. Foram analisados os processos primários, que servem os intervenientes principais durante o ciclo de vida do software; os processos de suporte, que apoiam outros processos como parte integrante destes e os processos organizacionais, utilizados globalmente pela organização. Pretendeu-se obter um primeiro conhecimento sobre os processos da empresa e recolher documentação relevante sobre o seu funcionamento. Realizou-se também um levantamento informal da matriz recursos / processos e procedeu-se à selecção dos colaboradores a entrevistar na fase seguinte. Esta fase foi cumprida através da realização de uma reunião entre os três impulsionadores do projecto de melhoria (IPMs).

A fase 2, denominada fase de Diagnóstico dos Processos, teve como base a norma ISO TR 15504 (ver anexo B), modificada e ajustada para as necessidades específicas da empresa. Esta norma define e caracteriza um referencial de avaliação do processo de software. O diagnóstico foi realizado através de questionários realizados aos colaboradores previamente seleccionados e cobriu as cinco categorias de processos propostos pela norma, nomeadamente, cliente – fornecedor, engenharia, suporte, gestão e organização. O resultado desta fase foi um perfil individual da capacidade dos processos da organização.



Como seria de prever os resultados da fase de levantamento foram utilizados no desenho e adaptação dos questionários utilizados no diagnóstico. Os questionários foram preenchidos pelos dois directores da área de desenvolvimento (simultaneamente com o papel de impulsionadores internos do projecto de melhoria), por quatro chefes de projecto, pelo director comercial e pela encarregada do controlo final da qualidade.

A fase 3, denominada no 2MPspe, como fase de Validação do Diagnóstico, teve como base o modelo de avaliação da maturidade CMM (Capability Maturity Model), desenvolvido pelo Software Engineering Institute (SEI). O CMM (ver anexo C) é o modelo de melhoria do processo mais divulgado em termos mundiais e tem como *output* principal a definição do nível de maturidade da organização. A estratégia seguida por este modelo consiste na utilização de um questionário, proposto pelo SEI, para avaliar as Áreas Chave do Processo características de cada nível de maturidade. Considera-se uma área chave do processo como um *cluster* de actividades relacionadas sendo caracterizada pelos objectivos, pela obrigação e capacidade de executar, pelas actividades, pelas métricas utilizadas e pela verificação da implementação.

Os objectivos desta fase foram a confirmação dos resultados do diagnóstico, a definição do nível de maturidade da organização e o desenvolvimento “precoce” de um possível conjunto de áreas de melhoria. Para executar esta fase os IPMs preencheram o questionário referente às áreas chave do processo do nível 2 e 3.

Finalmente a fase 4, denominada de Definição das Áreas de Melhoria pretendeu analisar os resultados obtidos e desenvolver um plano de acção, apresentando as áreas chave de melhoria e as estratégias adoptadas para atingir os objectivos. Numa abordagem inicial realizou-se a caracterização dos impactos esperados na organização, devido a melhorias independentes na capacidade dos processos. Esta avaliação dos impactos possibilitou a definição das prioridades de melhorias e foi o ponto de partida para a definição das estratégias a utilizar em cada processo individualmente. Esta fase terminou com a definição do cronograma de acção e a caracterização quantitativa dos objectivos do plano de melhoria.

A estratégia adoptada nesta fase consistiu na realização de uma reunião de trabalho entre os IPMs, realizada nas instalações da empresa.

A próxima secção apresenta os resultados obtidos após o tratamento dos questionários relativos às fases de diagnóstico dos processos e validação.

### **3. Resultados da Avaliação**

Pretende-se nesta secção apresentar e comentar brevemente os resultados obtidos na avaliação dos processos da empresa. Nesta perspectiva apresenta-se o nível global da capacidade da organização e o perfil de capacidade dos processos, obtidos através dos questionários e calculados utilizando o CMM e a norma ISO 15504.

A figura 1 apresenta os valores obtidos para cada processo, em relação à percentagem de satisfação das práticas de base existentes na empresa e ao nível de cada processo. Estes valores baseiam-se na aplicação da norma ISO TR 15504. Os processos estão agrupados em categorias, nomeadamente: cliente – fornecedor (cli), processos com impacto directo no cliente; engenharia (eng), processos que directamente especificam, implementam ou mantêm o software; suporte (sup), processos que podem ser utilizados por qualquer outro processo - incluindo outros processos de suporte – em vários pontos no ciclo de vida de software; gestão (ges), actividades empregues por qualquer colaborador encarregue de gerir um ou mais processos e organização (org) que são os processos relacionados com o ambiente organizacional.

<b>Nome Processo</b>	<b>%P. base</b>	<b>Nível ISO 15504</b>
CLI.3 Elicitação dos Requisitos	27,8%	0
CLI.4 Suporte	53,3%	0
CLI.5 Serviços ao cliente	33,3%	0
ENG.2 Requisitos	50,0%	0
ENG.3 Desenho	6,7%	0
ENG.4 Implementação do software	29,2%	0
ENG.5 Integração	41,7%	0
ENG.6 Testes	58,3%	0
ENG.7 Manutenção	100,0%	1
GES.1 Gestão do projecto	60,6%	0
GES.2 Gestão da qualidade	14,3%	0
GES.3 Gestão de risco	4,8%	0
SUP.1 Documentação	92,9%	0
SUP.2 Gestão de configurações	72,2%	0
SUP.3 Garantia de qualidade	58,3%	0
SUP.4 Verificação	0,0%	0
SUP.5 Validação	0,0%	0
SUP.8 Resolução de problemas	0,0%	0
ORG.1 Alinhamento organizacional	40,0%	0
ORG.2 Definição do Processo	12,5%	0
ORG.3 Recursos Humanos	0,0%	0
ORG.6 Reutilização	58,3%	0

*Figura1 – Resultados da fase do Diagnóstico dos Processos (ISO TR 15504)*

Analisando a figura 1, conclui-se que a empresa está no nível zero (denominado incompleto), da escala proposta pelo ISO TR 15504. O nível incompleto indica que a empresa globalmente ou não executa os processos ou executa-os de uma forma incompleta.

Em relação aos processos analisados individualmente, verifica-se que apenas o processo de manutenção está no nível 1 e todos os outros foram classificados no nível 0. Esta característica indica que há uma falha geral em atingir os objectivos do processo de

software, verificando-se que os produtos de trabalho resultantes do processo não são facilmente identificáveis.

A figura 2 apresenta os resultados da avaliação das áreas chave do processo do nível 2 e 3 do CMM.

As perguntas do questionário do CMM (ver anexo C) analisam as actividades, métricas e verificação de cada área chave do processo. A coluna dos resultados apresenta a percentagem de respostas positivas para perguntas relacionadas com cada área chave do processo.

<b>NÍVEL 2: REPETÍVEL</b>	
Gestão de requisitos	27,8%
Planeamento do projecto de software	47,6%
Vigilância e acompanhamento do projecto de software	42,9%
Garantia da qualidade do software (SQA)	29,2%
Gestão de configurações	16,7%
<b>Nível 3: DEFINIDO</b>	
Foco no processo organizacional	4,8%
Definição do processo organizacional	0,0%
Programas de formação	21,4%
Gestão integrada do software	11,1%
Engenharia do produto de software	27,8%
Coordenação inter-grupos	14,3%
Revisões por pares	16,7%

*Figura 2 – Resultados da fase de validação do diagnóstico (CMM)*

Analisando a figura 2 conclui-se que a empresa está no nível 1 do CMM, denominado inicial. Este nível caracteriza-se por uma forte dependência no esforço individual dos profissionais, verificando-se também que poucos processos estão definidos. A organização opera sem procedimentos formais e as estimações de custo e os planos dos projectos não são efectivos. As ferramentas não se encontram bem integradas com os processos e não são uniformemente aplicadas.

Verifica-se ainda que a média de respostas positivas no nível 2 é 32,8% e a média de respostas positivas no nível 3 é de apenas 13,7%. A empresa estará no nível 2 do CMM quando tiver uma percentagem de respostas positivas de 100% e as actividades e documentos forem identificáveis.

A secção seguinte apresenta as prioridades de melhoria e as estratégias e acções específicas necessárias para concretizar as melhorias propostas.

#### **4. Áreas a Melhorar**

Nesta secção faz-se uma análise detalhada dos resultados obtidos na avaliação dos processos. Desenvolve-se o cruzamento dos dados recolhidos nas duas avaliações (Diagnóstico e Validação) de modo a definir os processos prioritários a melhorar. A definição das prioridades inclui a caracterização das estratégias específicas e acções a executar para a concretização efectiva das melhorias.

Perante o panorama actual, descrito no capítulo anterior, a meta global deste projecto de melhoria foi claramente definida, **a Gama pretende atingir o nível 2 do CMM**. Naturalmente é necessário pesar esta meta global, com objectivos mais “urgentes” e com maior impacto e visibilidade em termos da empresa, dos próprios colaboradores e dos clientes finais. Convém referir que dados retirados da literatura especializada, apresentam como duração média de um projecto de melhoria do nível 1 para o nível 2 do CMM, o valor de 26 meses.

O CMM apresenta claramente os tipos de melhoria esperados em relação à previsibilidade, controlo e efectividade, conforme a maturidade de uma organização aumenta. Em primeiro lugar a diferença entre os resultados alvo e os resultados actuais diminui, tornando os projectos mais previsíveis. Em relação ao controlo, verifica-se uma diminuição da variabilidade dos resultados actuais em relação aos resultados alvo. Finalmente uma melhoria na efectividade significa que os resultados alvo melhoram com o aumento da maturidade da organização.

A passagem do nível de maturidade 1 para o nível de maturidade 2 do CMM, implica uma actuação nas cinco áreas chave do processo do nível 1. Vamos então analisar cada uma das áreas chave e definir as carências principais e pontos de actuação.

O objectivo da gestão de requisitos é estabelecer um entendimento comum entre o cliente e a equipa do projecto de software, acerca dos requisitos do cliente que serão implementados pelo projecto de software. Estes requisitos serão a base para o planeamento e gestão do projecto de software. Em relação a esta área chave do processo a empresa obteve um resultado de 27,8 %, verificando-se que as falhas principais se situam ao nível da utilização do processo definido, do desenvolvimento de métricas coerentes e ao nível da garantia da qualidade do processo.

A área chave do processo denominada planeamento do projecto de software tem como objectivo o estabelecimento de planos razoáveis para a execução das actividades de engenharia de software e para a gestão do projecto. Envolve o desenvolvimento de estimativas do trabalho a ser realizado, estabelecendo os compromissos necessários e definindo o plano para a execução do projecto. O resultado verificado nesta área chave foi 47,6 %. Embora as actividades sejam executadas notam-se falhas na utilização / definição de um processo e documentos “standard”, bem como na definição de métricas e na verificação do processo.

O objectivo da vigilância e acompanhamento do projecto é possibilitar uma visibilidade adequada sobre o andamento do projecto, de maneira à gestão poder tomar as acções correctivas necessárias quando o desempenho se desvie significativamente do estabelecido nos planos. O resultado obtido nesta área foi 42,9%, verificando-se a necessidade de definir as actividades e formalizar a documentação, bem como estabelecer as métricas e definir as actividades de verificação da qualidade.

O objectivo da garantia da qualidade do software (SQA) é possibilitar uma visibilidade apropriada sobre o processo usado no projecto de software e sobre os produtos resultantes. Envolve a revisão e auditoria dos produtos do software e das suas actividades, de maneira a inferir se estão de acordo com as normas e procedimentos estabelecidos. O resultado obtido foi 29,2 %, verificando-se falhas na definição das

actividades, no estabelecimento das métricas e na garantia da qualidade do processo. Actualmente este processo refere-se apenas à garantia da qualidade do produto final.

Finalmente a área chave do processo denominada gestão de configurações tem como objectivo estabelecer e manter a integridade dos produtos do software ao longo de todo o ciclo de vida do projecto. A gestão de configurações envolve a identificação da configuração do software (p.ex., produtos de trabalho do software seleccionados e suas descrições) num dado instante de tempo, sistematicamente controlando as alterações e mantendo a integridade e a reconstituição da configuração por todo o ciclo de vida. O resultado obtido foi 16,7 % mas verifica-se que esta actividade está apenas centrada no código.

A figura 3 apresenta um quadro resumo das necessidades de melhoria principais, obtido a partir do modelo CMM.

Área Chave do Processo	Actividades	Documentos	Recursos	Métricas	Verificação
Gestão de Requisitos	Utilizar	Utilizar	Responsabilizar	Definir	Definir
Planeamento do projecto	Definir	Definir	Formação	Definir	Definir
Acompanhamento do projecto	Definir	Definir	Formação	Definir	Definir
Garantia da qualidade do software	Executar	Definir	Responsabilizar	Definir	Definir
Gestão de configurações	Definir	Definir	Responsabilizar	Definir	Definir

*Figura 3 – Resumo das necessidades de melhoria*

Para a definição das prioridades de melhoria, caracterizou-se o impacto nos objectivos globais da organização de uma melhoria realizada em cada um dos processos. A figura 4 apresenta o resultado desta avaliação dos impactos. Utilizou-se a seguinte escala: 0 – pouco impacto; 1 – algum impacto e 2 – muito impacto.

<b>Nome Processo</b>	<b>Impacto</b>
CLI.3 Elicitação dos Requisitos	0
CLI.4 Suporte	0
CLI.5 Serviços ao cliente	1
ENG.2 Requisitos	2
ENG.3 Desenho	2
ENG.4 Implementação do software	2
ENG.5 Integração	1
ENG.6 Testes	1
ENG.7 Manutenção	0
GES.1 Gestão do projecto	1
GES.2 Gestão da qualidade	2
GES.3 Gestão de risco	1
SUP.1 Documentação	2
SUP.2 Gestão de configurações	1
SUP.3 Garantia de qualidade	2
SUP.4 Verificação	2
SUP.5 Validação	2
SUP.8 Resolução de problemas	0
ORG.1 Alinhamento organizacional	1
ORG.2 Definição do Processo	2
ORG.3 Recursos Humanos	2
ORG.6 Reutilização	0

*Figura 4 – Avaliação do impacto das melhorias*

A análise da figura 4 torna evidente a importância dos processos com um impacto 2. Nesta perspectiva agrupou-se estes processos em áreas de intervenção e decidiu-se então definir três áreas de intervenção (AI), nomeadamente:

1. **Infra-estrutura do processo**, constituída pelos processos ORG.2 – Definição do processo, ORG.3 – Recursos Humanos e SUP.1 – Documentação;
2. **Engenharia do produto**, constituída pelos processos ENG.2 – Requisitos, ENG.3 – Desenho e ENG.4 – Implementação do software;
3. **Sistema de Qualidade**, composta pelos processos GES.2 – Gestão da Qualidade, SUP.3 – Garantia da Qualidade, SUP.4 – Verificação e SUP.5 – Validação.



As áreas de intervenção definidas terão uma abrangência aplicacional diferente em relação à empresa e as estratégias a adoptar para a sua melhoria serão também diversas. No caso da AI, Infra-estrutura do processo, o impacto é global uma vez que será necessário caracterizar o processo de definição, formalizando as actividades, técnicas e ferramentas a utilizar e permitindo estabelecer um conjunto de definições do processo que suportarão o desempenho estável e repetível dos processos de engenharia e de gestão. Será necessário também definir o processo de Recursos Humanos, realizando a gestão de recursos humanos e caracterizando as necessidades de formação e finalmente estabelecer o processo de documentação, permitindo desenvolver e manter os documentos necessários para registar toda a informação produzida pelas actividades. Espera-se com esta intervenção caracterizar as diversas perspectivas dos processos permitindo assim ultrapassar os problemas relacionados com a integração das diversas áreas e sempre que possível estabelecendo uma infra-estrutura automatizada para suportar os processos.

Em relação à AI de Engenharia do produto o impacto estará relacionado com a área de desenvolvimento. Será necessário formalizar as actividades, caracterizar os documentos, definir métricas para os processos e estabelecer o processo de verificação (regras e responsáveis). A intervenção centrar-se-á nos processos de requisitos, desenho e implementação do software. Esta AI deverá tomar em consideração a necessidade de desenvolver processos adaptáveis aos dois tipos de projectos principais existentes na empresa, nomeadamente, novos produtos e projectos de manutenção. Intervindo nestes processos espera-se um aumento da confiança e satisfação dos profissionais da empresa e simultaneamente uma clarificação e responsabilização sobre as tarefas e actividades do desenvolvimento.

Finalmente a AI do Sistema de qualidade, terá também um impacto global em termos da organização, tornando-se necessário definir as actividades, atribuir responsabilidades e definir métricas. Esta AI está intimamente ligada a uma mudança cultural, com reflexo em todos os colaboradores da empresa. Em termos mais específicos, esta área de intervenção é subdividida em quatro processos, nomeadamente: a gestão da qualidade que envolve o estabelecimento do foco na monitorização da qualidade do produto e do processo, tanto no nível do processo como no nível organizacional; a garantia da qualidade do software que deverá garantir que os produtos do trabalho satisfazem os

requisitos especificados e cumprem os planos estabelecidos; a verificação que confirmará que cada produto do software e/ou serviço está em conformidade com os requisitos e finalmente a validação que deverá confirmar que os requisitos do produto final foram atingidos e assim o produto é útil e será aceite pelo mercado. Espera-se com esta intervenção alterar a abordagem actual em que a qualidade está apenas centrada no produto final, passando para uma perspectiva total, em que a procura, gestão e garantia da qualidade engloba todos os processos e recursos da empresa.

É importante referir nesta fase o alinhamento entre estas três áreas de intervenção e as áreas chave do processo (KPA) do nível 2 do CMM, numa perspectiva de atingir o nível 2 do CMM de uma forma faseada e apoiada. Verifica-se então que a AI infra-estrutura do processo irá afectar todas as KPAs, uma vez que trata da definição, documentação e Recursos Humanos dos processos. A AI sistema de qualidade engloba claramente a KPA garantia da qualidade do software e a AI engenharia do produto, embora seja uma KPA do nível 3, sofrerá uma intervenção devido à sua visibilidade interna e impacto em termos da organização.

Para definir os objectivos de uma forma quantitativa é necessário referir que para uma área chave do processo (tal como definida no CMM) ser satisfeita é necessário que o(s) processo(s) correspondente(s) no ISO 15504 seja(m) avaliado(s) no nível 3. Um processo no nível 3 na escala ISO 15504, denomina-se estabelecido, verificando-se que o processo é executado e gerido, utilizando um processo standard baseado em “bons” princípios de engenharia de software. Este nível caracteriza-se pela satisfação dos atributos de desempenho do processo (nível 1), gestão do desempenho e gestão dos produtos do trabalho (nível 2) e definição do processo e recursos do processo do nível 3.

## **5. Plano de Melhoria**

Nesta secção apresenta-se o plano de melhoria, detalhando e concretizando em termos de um cronograma, dos recursos e dos objectivos específicos, os resultados

apresentados na secção anterior. A meta global deste projecto de melhoria foi então definido como sendo, atingir o nível 2 do CMM dentro de 24 meses.

O projecto terá início no dia 1/10/2002 e a data prevista para o fim é 1/10/2004. Preconiza-se a utilização de 3 ciclos (ver figura 5) e no fim de cada ciclo deverá realizar-se uma nova avaliação para verificar o cumprimento dos objectivos e ajustar as metas do ciclo seguinte conforme os resultados obtidos. O primeiro ciclo terá uma duração de 11 meses, o 2º ciclo terá uma duração de 6 meses e o 3º ciclo uma duração esperada de 5 meses. Os *timings* poderão ser ajustados conforme o decorrer do projecto, mas espera-se uma maior dificuldade no primeiro ciclo, pois existirá uma “mudança cultural” de grande amplitude.

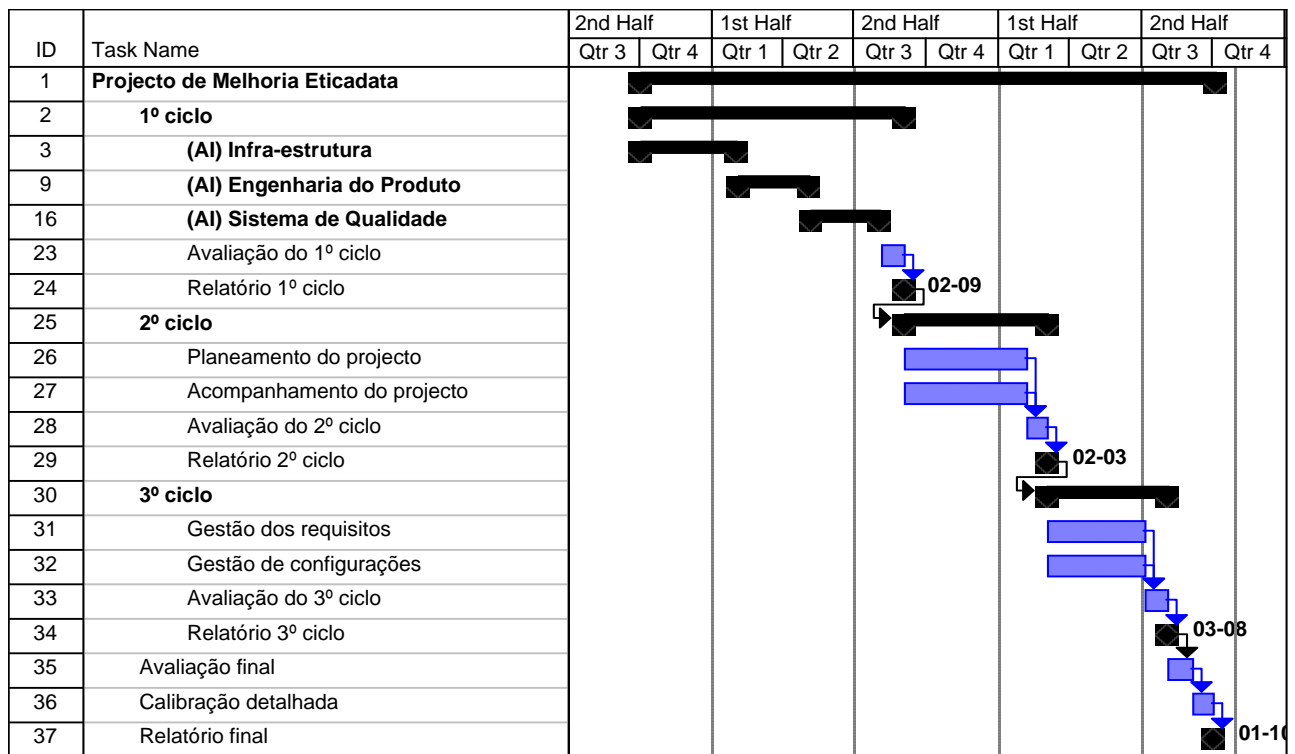


Figura 5 – Plano de Melhoria (perspectiva global)

Em relação aos objectivos específicos, o primeiro ciclo intervirá na infra-estrutura do processo, na engenharia do produto e no sistema de qualidade. O segundo ciclo terá com área principal de acção a gestão de projectos, nas vertentes do planeamento e acompanhamento do projecto e finalmente o terceiro ciclo intervirá nos processos de gestão de requisitos e gestão de configurações.

A avaliação dos ciclos (tarefas 23, 28 e 33) terá uma duração de 1 mês e será realizada pelos IPMs, consistindo na realização das fases de diagnóstico dos processos e validação proposta pelo 2MPspe, de modo a obter informação quantitativa sobre as melhorias registadas e corrigir objectivos com as “lições obtidas”. Os relatórios dos ciclos (tarefas 24, 29 e 34) e o relatório final (tarefa 37) serão elaborados pelos IPMs externos. A calibração detalhada (tarefa 36) consistirá numa correcção final dos problemas detectados nos processos sujeitos a intervenção. Esperando-se assim atingir rigorosamente todas as KPAs do nível 2 do CMM. Por fim a avaliação final (tarefa 35), terá a duração de 1 mês, será realizada pelos IPMs com a contribuição de alguns colaboradores e consistirá nas fases de diagnóstico e validação do 2MPspe, mas numa perspectiva global. Perante os resultados será tomada a decisão sobre a certificação ISO 9000: 2000 ou CMM.

O cronograma detalhado para o 1º ciclo é apresentado na figura 6. Inicia-se pela AI da infra-estrutura uma vez que irá influenciar directamente o desenvolvimento das outras áreas de intervenção. A 2ª fase passará por uma intervenção no processo de desenvolvimento, esperando-se um forte impacto no produto final. Esta selecção traduz uma maior apetência para a mudança dos profissionais do desenvolvimento. Finalmente a 3ª AI será no sistema de qualidade, com um impacto global, mas onde se espera aplicar alguns dos “ensinamentos” obtidos nas AI anteriores de forma a minorar as possíveis resistências.

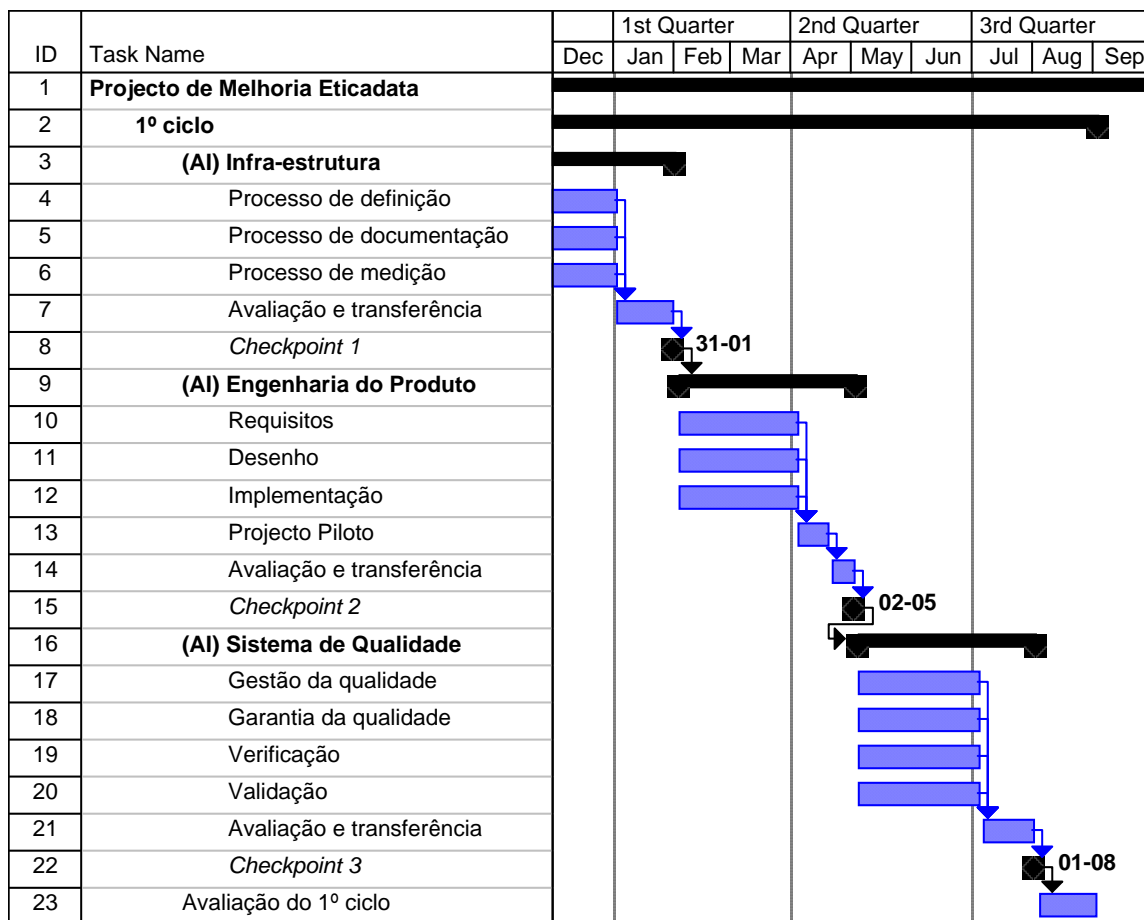


Figura 6 – Cronograma detalhado do 1º ciclo

Em relação à primeira AI (infra-estrutura do processo) está planeada uma duração de 4 meses, com início no dia 1/10/2002. O objectivo específico é passar para o nível 1 do ISO 15504 os processos ORG.2, ORG.3 e SUP.1. Esta actividade de definição, Recursos Humanos e documentação do processo será desenvolvida em *part-time* pelos dois directores da área de desenvolvimento (impulsionadores internos do projecto). A avaliação e transferência (tarefa 7) contará também com a participação do impulsionador externo do projecto. Esta AI revela-se crucial para todo o projecto, uma vez que servirá de suporte e influenciará todo o seu desenvolvimento. A sua maior dificuldade deriva da pouca visibilidade inicial. A passagem para o nível 1 obriga a que o processo seja executado, o que em termos práticos consistirá em, caracterizar o processo de definição e desenvolver a actividade para os processos da empresa (tarefa 4), caracterizar as métricas necessárias para suportar uma gestão efectiva dos processos (tarefa 6) e estabelecer o processo que permite desenvolver e manter os documentos necessários para registar a informação produzida pelas actividade (tarefa 5).

Em relação à AI denominada engenharia do produto está planeada uma duração de 3 meses, com início no dia 1/2/2003. O objectivo específico é passar os processos ENG.2, ENG.3 e ENG.4 para o nível 3 da norma ISO 15504 (nível estabelecido), numa perspectiva de cumprir a KPA correspondente no CMM. Estas tarefas serão supervisionadas pelos IPMs internos, mas a sua execução será distribuída pelos chefes de projecto da empresa. A definição dos processos (tarefas 10, 11 e 12) terá uma duração de dois meses, verificando-se em seguida a aplicação dos processos “redefinidos” num projecto piloto (tarefa 13) com a duração de 3 semanas. Por fim a avaliação e transferência (tarefa 14) terá a duração de 1 semana e contará com a colaboração do IPM externo. As expectativas em relação a esta AI são bastante elevadas, uma vez que se espera um grande impacto quer nos profissionais do desenvolvimento, quer no produto final. Em termos práticos a passagem para o nível 3 do ISO 15504 implica atingir os seguintes atributos do processo: desempenho do processo, gestão do desempenho, gestão dos produtos do trabalho, definição do processo e recursos do processo. Estes atributos característicos dos três primeiros níveis deverão ser satisfeitos nas áreas de requisitos, desenho e implementação do software.

Finalmente, em relação à última área de intervenção, relacionada com o sistema de qualidade, está planeada uma duração de 3 meses, com início previsto para o dia 3/5/2003. O objectivo específico é passar os processos GES.2, SUP.3, SUP.4 e SUP.5 para o nível 3 do ISO 15504. As tarefas de definição destes processos terão uma reunião inicial entre os IPMs (internos e externo) e um colaborador que deverá estar alocado a 100%, com o intuito de clarificar objectivos, formalizar as tarefas e acertar “pontos de vista”. Em seguida as tarefas serão executadas pelo colaborador em *full-time* com a supervisão dos IPMs internos. Estas tarefas (17, 18, 19 e 20 na figura 6) terão uma duração prevista de 2 meses. A avaliação e transferência (tarefa 21) contará com a participação dos IPMs, do colaborador em *full-time* e outros colaboradores séniores. Tal como já foi referido esta terceira AI terá um forte impacto na cultura organizacional, sendo de esperar alguma resistência inicial.

A avaliação do 1º ciclo (tarefa 23) terá a duração de 1 mês e será executada pelos IPMs, estando prevista a entrega do relatório final para o dia 2/9/2003. Esta tarefa consistirá na realização do diagnóstico e validação propostos pelo 2MPspe.

Nesta secção detalhou-se o plano de melhoria, apresentando o cronograma, os recursos e os objectivos específicos de cada fase. A próxima secção apresentará algumas conclusões, aspectos a ponderar e trabalho futuro.

## 6. Conclusões

*“An important first step in addressing software problems is to treat the entire software task as a process that can be controlled, measured and improved” [Humphrey, 1989]*

A primeira etapa deste projecto de melhoria, denominada etapa de planeamento, pretendeu detalhar a situação actual da empresa, permitindo a definição de prioridades e estratégias de melhoria. Considera-se que esta primeira etapa foi concretizada com sucesso, agora é necessário executar o plano, verificar as melhorias e finalmente actuar.

Este projecto deve ser apenas o início de um trabalho constante de melhoria. Uma procura da qualidade total e da satisfação real do cliente. Claramente, a empresa deve procurar atingir o nível 3 do CMM e deve ver o seu esforço reconhecido através da certificação. Este projecto é o “pontapé de saída”, é necessário que os problemas do dia a dia não façam esmorecer o ânimo.

A meta inicial de atingir o nível 2 do CMM, está faseada em 3 ciclos com a duração de 2 anos e o 1º ciclo (descrito neste relatório) está dividido em 3 Áreas de Intervenção, com duração de 11 meses. O projecto necessitará de um acompanhamento rigoroso, de uma verificação planeada e de um *feedback* constante.

Considera-se importante salientar que os recursos humanos são um factor fundamental em qualquer projecto de melhoria/mudança e por isso é necessário que todos os colaboradores sintam a sua contribuição nos novos processos definidos. Certamente o projecto é ambicioso, mas espera-se um retorno compensador a curto / médio prazo, em termos da organização, dos seus colaboradores e do cliente final.

**Anexos do Relatório**

(Não são apresentados devido à sua extensão)

Anexo 1 – Levantamento do processo do software (ISO 12207)	a1
Anexo 2 – Diagnóstico dos processos (ISO 15504)	a22
Anexo 3 – Validação do diagnóstico do processo (CMM)	a54



## Anexo B – Selecção das Práticas de Base

Neste anexo, PT refere-se a Produtos de Trabalho

	Aceite	Não viável	Muito dispendioso	Não alinhado c. obj. estrat.	Não definido
<b>Processo Gestão de Requisitos</b>					
Práticas de Base (1 a 6)	X				
PT – Avaliação da necessidade do produto			X		
PT – Relatório de controlo da alterações	X				
PT – Requisitos do cliente	X				
<b>Processo Análise Requisitos do Software (ENG.21)</b>					
PB – Especificar Requisitos do software	X				
PB – Determinar impacto no ambiente operacional		X	X		
PB – Desenvolver critérios para teste do software	X				
PB – Assegurar consistência		X			
PB – Avaliar e actualizar requisitos	X				
PB – Desenvolver versão base dos requisitos	X				
PT – Relatório de controlo de alterações	X				
PT – Registo da comunicação		X			
PT – Registo de acompanhamento		X			
PT – Resultado da análise		X			
PT – Requisitos de interface	X				
PT – Requisitos do software	X				
<b>Processo Concepção do software (ENG.22)</b>					
PB – Descrever Arquitectura do software	X <sup>1</sup>				
PB – Definir interfaces	X				
PB – Desenvolver documentação preliminar do utilizador		X			
PB – Definir requisitos dos testes		X			
PB – Desenvolver concepção detalhada	X				
PB – Avaliar e verificar concepção do software	X				
PB – Assegurar consistência	X				
PT – Concepção da base de dados	X				
PT – Concepção de alto nível do software	X				
PT – Concepção detalhada do software	X				
PT – Registo da consistência	X				
<b>Processo Construção do software (ENG.23)</b>					
Práticas de Base (1 a 4)	X				
PT – Dados de teste	X <sup>2</sup>				
PT – Concepção dos testes	X				
PT – Casos de teste	X				
PT – Unidades de software	X				
PT – Resultados dos testes	X				
PT – Registo da consistência	X				
<b>Processo Integração do software (ENG.24)</b>					
Práticas de Base (1 a 7)	X				
PT – Item de software	X				
PT – Procedimentos de teste de integração (plano, concepção, regressão e casos de teste)	X				

<sup>1</sup> PB 1 – Descrever a arquitectura do software e as interfaces entre os elementos no M<sub>a</sub>2MP.

<sup>2</sup> PT – Procedimentos de teste das unidades de software no M<sub>a</sub>2MP.

	Aceite	Não viável	Muito dispendioso	Não alinhado c. obj. estrat.	Não definido
PT – Produto integrado	X				
PT – Registo dos itens de teste		X <sup>3</sup>			
PT – Log dos testes					X
PT – Resultados dos testes	X				
PT – Lista de construção		X			
PT – Registo de consistência	X				
<b>Processo Teste do software (ENG.25)</b>					
Práticas de Base (1 a 4)	X				
PT – Item de configuração	X				
PT – relatório de incidente	X				
PT – Manual do cliente		X			
PT – Procedimentos de teste do software (plano, regressão, casos de teste)	X				
PT – Log dos testes					X
PT – Relatório dos testes	X				
PT – Manual do utilizador (novo)	X				
<b>Processo Manutenção do software (ENG.3)</b>					
Práticas de Base (1 a 6)	X				
PT – Item de configuração	X				
PT – Relatório de controlo / acompanhamento da mudança	X				
PT – Medição de campo		X			
PT – Estratégia de manutenção (aceitação, instalação, regressão, teste, informação da <i>release</i> )	X				
PT – Soluções temporárias		X			
PT – Registo		X			
PT – Estratégia das <i>releases</i>		X			
PT – Resultados dos testes	X				
PT – Registo de consistência	X				
<b>Processo Documentação (SUP.1)</b>					
PB – Desenvolver a estratégia de gestão da documentação				X	
PB – Estabelecer normas para os documentos	X				
PB – Especificar os requisitos dos documentos	X				
PB – Identificar os documentos a produzir	X				
PB – Desenvolver documentos	X				
PB – Verificar documentos		X			
PB – Distribuir documentos	X				
PB – Manter documentos	X				
PT – Relatório de controlo / acompanhamento de mudanças	X				
PT – Política				X	
PT – Registo de aceitação	X				
PT – Registo de entrega	X				
PT – Registo de revisão		X			
PT – Produto de trabalho	X				
<b>Processo Gestão de Configurações (SUP.2)</b>					
PB – Desenvolver a estratégia de gestão de configurações			X		
PB – Identificar itens de configuração	X				
PB – Estabelecer estrutura de ficheiros, directorias e hierarquias			X		
PB – Estabelecer estratégia de gestão de ramificações			X		

<sup>3</sup> Só com ferramentas

	Aceite	Não viável	Muito dispendioso	Não alinhado c. obj. estrat.	Não definido
PB – Estabelecer versão base interna e para entrega	X				
PB – Manter a descrição dos itens de configuração	X				
PB – Controlar modificações e versões	X				
PB – Manter o histórico dos itens de configuração	X				
PB – Relatar o estado da configuração	X				
PB – Verificar que a informação sobre o item está completa e consistente			X		
PB – Gerir cópias de segurança, armazenamento, arquivo, manuseamento e entrega dos itens	X				
PT – Item de configuração	X				
PT – Guia de manuseamento e armazenamento	X				
PT – Plano de gestão de configurações		X			
PT – Produto integrado	X				
PT – Registo do estado do progresso	X				
<b>Processo Garantia da Qualidade (SUP.3)</b>					
PB – Desenvolver a estratégia de garantia da qualidade	X				
PB – Definir registos de qualidade do projecto	X				
PB – Assegurar a qualidade das actividades do processo e dos produtos do trabalho	X				
PB – São produzidos e armazenados registos das actividades de garantia	X				
PB – Resolução de problemas	X				
PT – Relatório de problemas	X				
PT – Política da garantia da qualidade		X			
PT – Procedimentos	X				
PT – Descrição do processo	X				
PT – Normas	X				
PT – Critério de qualidade	X				
<b>Processo Gestão do Projecto (GES.1)</b>					
Práticas de Base (1 a 11, excepto 5 e 7)	X				
PB – Definir necessidade para aptidões e conhecimento		X			
PB – Identificar e monitorizar interfaces do projecto		X			
PT – Estimativas de custo	X <sup>4</sup>				
PT – Estimativas	X				
PT – Relatório de problemas		X			
PT – Medições do projecto	X				
PT – Plano do projecto	X <sup>5</sup>				
PT – Plano de formação		X			
PT – Modelo do ciclo de vida	X				
PT – Registo de comunicação			X		
PT – Registo do estado do progresso	X <sup>6</sup>				
PT – Registo de revisão	X <sup>2</sup>				
PT – Registo de acção correctiva	X <sup>2</sup>				
PT – Escalonamento	X <sup>1</sup>				
PT – WBS	X <sup>1</sup>				
PT – Rede PERT	X <sup>1</sup>				
PT – Relatório do estado do projecto	X <sup>2</sup>				
<b>Processo Gestão do Risco (GES.2)</b>					
PB – Estabelecer o âmbito da gestão de riscos				X	
PB – Identificar os riscos	X				

<sup>4</sup> PT – Estimativas no M<sub>a</sub>2MP

<sup>5</sup> PT – Plano do projecto no M<sub>a</sub>2MP

<sup>6</sup> PT – Relatório do estado do projecto no M<sub>a</sub>2MP

	Aceite	Não viável	Muito dispendioso	Não alinhado c. obj. estrat.	Não definido
PB – Analisar os riscos e determinar prioridades	X				
PB – Definir estratégias de gestão de risco		X			
PB – Definir e executar acções para reduzir o risco	X				
PB – Monitorizar riscos	X				
PB – Tomar acções correctivas	X				
PT – Medição do risco	X <sup>7</sup>				
PT – Plano de gestão do risco	X <sup>8</sup>				
PT – Plano de mitigação do risco	X				
PT – Registo de revisão		X			
PT – Pedido de acção sobre risco					X
PT – Registo de acção correctiva	X <sup>1</sup>				
PT – Sistema de acompanhamento	X <sup>1</sup>				
PT – Relatório de análise dos riscos	X <sup>9</sup>				
PT – Relatório do estado dos riscos	X				
<b>Processo Gestão da Qualidade (GES.3)</b>					
PB – Estabelecer objectivos de qualidade	X				
PB – Estabelecer estratégia global	X				
PB – Definir critérios de qualidade	X				
PB – Definir um sistema de gestão de qualidade	X				
PB – Avaliar até que ponto os objectivos de qualidade são atingidos	X				
PB – Executar acções correctivas	X				
PB – Receber <i>feedback</i>		X			
PT – Objectivos de qualidade	X				
PT – Medição da qualidade	X				
PT – Plano da qualidade	X				
PT – Política da qualidade	X				
PT – Registo da revisão	X				
PT – Registo da acção correctiva	X				
PT – Análise dos resultados	X				
PT – Critérios de aceitação	X				
<b>Processo Gestão de Recursos Humanos (ORG.11)</b>					
PB – Identificar aptidões e competências necessárias	X				
PB – Definir critérios de avaliação	X				
PB – Recrutar pessoal qualificado	X				
PB – desenvolver aptidões e competências	X				
PB – Definir a organização da equipa de projecto		X			
PB – <i>Empower</i> Equipas de projecto		X			
PB – Manter interacção entre equipas		X			
PB – Avaliar desempenho do pessoal	X				
PB – Fornecer <i>feedback</i> sobre o desempenho	X				
PB – Manter registos do pessoal	X				
PT – Aprovação do compromisso		X			
PT – Análise das necessidades de recursos humanos	X <sup>10</sup>				
PT – Avaliação do desempenho do pessoal	X <sup>11</sup>				
PT – Medição da avaliação do desempenho do pessoal	X				
PT – Plano de recursos humanos	X <sup>1</sup>				
PT – Plano de formação	X				
PT – Política do pessoal	X <sup>1</sup>				

<sup>7</sup> PT – Registo de controlo do risco no M<sub>a</sub>2MP

<sup>8</sup> PT – Plano de gestão do risco no M<sub>a</sub>2MP

<sup>9</sup> PT – Relatório de análise do risco no M<sub>a</sub>2MP

<sup>10</sup> PT – Plano de recursos humanos no M<sub>a</sub>2MP

<sup>11</sup> PT – Avaliação do desempenho do pessoal no M<sub>a</sub>2MP

	Aceite	Não viável	Muito dispendioso	Não alinhado c. obj. estrat.	Não definido
PT – registo de comunicação		X			
PT – Registo de revisão do desempenho		X			
PT – Registo do pessoal	X				
PT – Registo da formação	X				
PT – Critérios de desempenho	X				
<b>Processo Formação (ORG.12)</b>					
PB – Desenvolver estratégia de formação		X			
PB – Identificar necessidades de formação	X				
PB – Desenvolver ou adquirir formação	X				
PB – Preparar a execução da formação	X				
PB – Dar formação	X				
PB – Manter registo da formação	X				
PB – Avaliar efectividade da formação	X				
PB – Avaliar formação globalmente		X			
PT – Material de formação	X				
PT – Plano de formação	X				
PT – Registo de formação	X				
PT – Estratégia de formação		X			
<b>Processo Gestão do Conhecimento (ORG.13)</b>					
PB – Estabelecer o sistema de gestão do conhecimento	X				
PB – Criar uma rede de contribuintes do conhecimento		X			
PB – Desenvolver uma estratégia de gestão do conhecimento	X				
PB – Dessiminar os <i>assets</i> de conhecimento	X <sup>12</sup>				
PB – Melhorar os <i>assets</i> de conhecimento	X				
PT – Registo da comunicação		X			
PT – Repositório do conhecimento	X				
PT – Estratégia da gestão do conhecimento	X				
<b>Processo Estabelecimento do processo (ORG.2)</b>					
PB – Definir objectivos	X				
PB – Definir e documentar os processos	X				
PB – Identificar expectativas de desempenho	X				
PB – Estabelecer políticas organizacionais		X			
PB – Estabelecer orientações para colocar à medida	X				
PB – Manter dados sobre o processo	X				
PB – Manter descrições dos processos normalizados	X				
PT – Dados de desempenho do processo	X <sup>13</sup>				
PT – Política		X			
PT – Política de qualidade		X			
PT – Descrição do processo	X				
PT – Registo de qualidade	X				
PT – Registo de revisão	X				
PT – Análise de resultados	X <sup>1</sup>				
PT – Repositório	X				
<b>Processo Suporte ao Cliente (ORG.3)</b>					
PB – Estabelecer suporte ao cliente	X				
PB – Fornecer formação e documentação aos utilizadores	X				
PB – Monitorizar desempenho		X			
PB – Determinar nível de satisfação do cliente	X				
PB – Realizar o <i>benchmark</i> da satisfação do cliente		X			
PB – Comunicar satisfação do cliente	X				

<sup>12</sup> PB 3 – Gerir os *assets* do conhecimento no M<sub>a</sub>2MP.

<sup>13</sup> PT – Análise de resultados no M<sub>a</sub>2MP.

	Aceite	Não viável	Muito dispendioso	Não alinhado c. obj. strat.	Não definido
PT – Dados de <i>benchmarking</i>		X			
PT – Dados da satisfação do cliente	X				
PT – Relatório de problemas	X				
PT – Survey da satisfação do cliente		X			
PT – Medição do nível de serviço	X				
PT – Registo de formação	X				

## **Anexo C – Questionário 2MPspe**

**Questionário do 2MPspe**  
(versão 3)

**Diagnóstico dos Processos do Software**

Pedro Abreu Ribeiro  
Departamento de Sistemas de Informação  
Escola de Engenharia  
Universidade do Minho  
Junho 2004



## **Índice**

### Identificação

1. Introdução
  2. Orientações para o preenchimento
    - 2.1. Método para responder ao questionário
    - 2.2. Escala de classificação das actividades (Práticas de Base)
    - 2.3. Escala de classificação dos atributos do processo
  3. Descrição dos Processos
  4. Questões sobre os processos
- Anexo – Indicadores dos atributos

<b>Identificação</b>
----------------------

<b>Nome da empresa:</b> _____.	
<b>Nome*:</b> _____ _____.	
<b>Email:</b> _____.	
<b>Habilitações:</b> _____.	
<b>Idade:</b> _____.	
<b>Cargo na empresa:</b> _____ <b>Anos na empresa:</b> _____.	
<b>Responsabilidades:</b> _____ _____.	
<b>Projecto actual</b> _____.	<b>Data de preenchimento:</b> ____ / ____ / ____.
<b>Responsabilidades no projecto:</b> _____ _____ _____ _____.	<b>Tempo para preencher:</b> _____ Minutos.
	<b>Rubrica:</b> _____

\* Os dados deste questionário são confidenciais. Serão tratados e apresentados em conjunto com os restantes questionários.

## 1. Introdução

Este questionário tem como objectivo suportar o diagnóstico das práticas actualmente utilizadas na organização. Nesta perspectiva serão analisados os processos para verificar a sua execução, a sua capacidade e o conhecimento dos factores de contexto. Este diagnóstico faz parte de um esforço mais amplo de melhoria do processo de desenvolvimento de software. Espera-se uma resposta consciente de todos os profissionais de forma a realizar um diagnóstico correcto da situação actual e permitindo assim a obtenção de benefícios para a organização e para os colaboradores num curto prazo.

## 2. Orientações para o preenchimento

### 2.1. Método para responder ao questionário

**Para cada processo:**

1. Responder a todas as questões sobre a execução das actividades do processo (práticas de base)
2. Verificar os documentos identificáveis
3. Tendo como apoio 1. e 2., responder à questão sobre o atributo da capacidade AC1.1
4. Responder às questões sobre os atributos da capacidade (ACs) e os atributos do conhecimento (AKs). Utilizar como suporte os indicadores no anexo 1.

### 2.2. Escala de classificação das actividades (Práticas de Base):

**(Não) Não executada:** Ou não é executada, ou não produz produtos identificáveis.

**(Sim) Executada:** A actividade é executada e produz produtos identificáveis.

### 2.3. Escala de classificação dos atributos do processo

Para a medição dos atributos da capacidade (ACs) e atributos do conhecimento (AKs), utiliza-se a seguinte escala:

**N** - Não atingido: não há evidência que o atributo é atingido.

**P** - Parcialmente atingido: o atributo é parcialmente atingido.

**L** - Largamente atingido: o atributo é atingido de modo significativo.

**C** - Completamente atingido: o atributo é completamente atingido.

### 3. Descrição dos Processos

Apresenta-se em seguida uma descrição sucinta dos processos

**Categoria do processo:** Engenharia (ENG)

**Processos:**

#### **ENG.1 – Gestão de Requisitos**

O propósito do processo de Gestão dos requisitos é recolher, processar e acompanhar as necessidades e requisitos do cliente durante o ciclo de vida do produto.

#### **ENG.2 – Desenvolvimento**

##### **ENG.21 – Análise dos Requisitos do Software**

O propósito do processo de análise de requisitos do software é estabelecer os requisitos do software.

##### **ENG.22 – Concepção do Software**

O propósito do processo de concepção do software é definir uma solução para o software que acomode os requisitos e possa ser testada em relação a eles.

##### **ENG.23 – Construção do Software**

O propósito do processo de construção do software é produzir unidades de software executáveis que implementem correctamente a concepção do software.

##### **ENG.24 – Integração do Software**

O propósito do processo de integração do software é combinar as unidades de software produzindo itens integrados consistentes com a especificação do software<sup>293</sup>.

##### **ENG.25 – Teste do Software**

O propósito do processo de testes do software é confirmar que o produto de software integrado cumpre os requisitos definidos.

#### **ENG.3 – Manutenção do Software**

O propósito do processo de manutenção do software é modificar o produto depois da entrega para corrigir falhas, melhorar o desempenho ou adaptá-lo a um ambiente modificado.

**Categoria do processo:** Suporte (SUP)

**Processos:**

#### **SUP.1 – Documentação**

O propósito do processo de Documentação é desenvolver e manter a informação armazenada sobre o software, produzida por um processo.

#### **SUP.2 – Gestão de Configurações**

O propósito do processo de gestão de configurações é estabelecer e manter a integridade de todos os produtos de trabalho de um processo ou projecto e disponibilizá-los a todos os intervenientes.

#### **SUP.3 – Garantia da Qualidade**

O propósito do processo de garantia da qualidade é fornecer garantia que os produtos de trabalho e processos cumprem os requisitos especificados e implícitos e aderem aos planos estabelecidos.

---

<sup>293</sup> documento resultante do processo de concepção de software

**Categoria do processo:** Gestão (GES)

**Processos:**

**GES.1 – Gestão do Projecto**

O propósito do processo da gestão do projecto é identificar, estabelecer, coordenar e monitorizar as actividades, tarefas e recursos necessários para um projecto produzir o produto e/ou serviço, no contexto das restrições e requisitos do projecto.

**GES.2 – Gestão do Risco**

O propósito do processo de gestão de risco é identificar, gerir e mitigar de modo contínuo os riscos.

**GES.3 – Gestão da Qualidade**

O propósito do processo de gestão da qualidade é monitorizar a qualidade dos produtos e serviços, no nível da organização e do projecto de modo a assegurar a satisfação do cliente.

**Categoria do processo:** Organizacionais (ORG)

**Processos:**

**ORG.1 – Recursos Humanos**

**ORG.11 – Gestão de Recursos Humanos**

O propósito do processo de Gestão de Recursos Humanos é fornecer a organização e os projectos com indivíduos que possuam aptidões e conhecimento para executar os seus papéis efectivamente e para trabalharem em conjunto como uma equipa coesa.

**ORG.12 – Formação**

O propósito do processo de formação é desenvolver os aptidões e conhecimento dos indivíduos de modo a que possam executar os seus papéis efectivamente.

**ORG.13 – Gestão do Conhecimento**

O propósito do processo de gestão do conhecimento é assegurar que o conhecimento, informação e aptidões dos profissionais são coligidas, partilhadas, reutilizadas e melhoradas através da organização.

**ORG.2 – Estabelecimento do Processo**

O propósito do processo de estabelecimento do processo é estabelecer um conjunto de processos organizacionais para todos os processos do ciclo de vida do software de forma a serem aplicados às suas actividades do negócio.

**ORG.3 – Suporte ao Cliente**

O propósito do processo de suporte ao cliente é estabelecer e manter um nível de serviço aceitável, realizando assistência e consultoria ao cliente para suportar a utilização efectiva do produto.

**4. Questões sobre os processos**

PROCESSO ENG.1 Gestão de Requisitos				
<b>Actividades executadas:</b>				
<ul style="list-style-type: none"> <li>• ENG.1.PB1 – Os requisitos e pedidos do cliente são obtidos e definidos? Sim <input type="checkbox"/> Não <input type="checkbox"/></li> <li>• ENG.1.PB2 – As expectativas do cliente são compreendidas? Sim <input type="checkbox"/> Não <input type="checkbox"/></li> <li>• ENG.1.PB3 – É obtida a aprovação sobre os requisitos? Sim <input type="checkbox"/> Não <input type="checkbox"/></li> <li>• ENG.1.PB4 – É estabelecida a versão base dos requisitos do cliente? Sim <input type="checkbox"/> Não <input type="checkbox"/></li> <li>• ENG.1.PB5 – São geridos os pedidos de alterações realizados pelo cliente? Sim <input type="checkbox"/> Não <input type="checkbox"/></li> <li>• ENG.1.PB6 – É disponibilizada informação ao cliente sobre o estado dos requisitos? Sim <input type="checkbox"/> Não <input type="checkbox"/></li> </ul>				
<b>Documentos a identificar:</b> Relatório de controlo de alterações <input type="checkbox"/> e requisitos do cliente <input type="checkbox"/>				
	<b>N</b>	<b>P</b>	<b>L</b>	<b>C</b>
<b>AC1.1</b> O processo <b>Gestão dos Requisitos</b> utiliza as práticas de base adequadas e resulta nos produtos de trabalho identificados? <i>(Indicadores: Práticas de Base e Documentos)</i>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<b>AC1.51</b> Os intervenientes do processo <b>Gestão dos Requisitos</b> estão identificados e reconhecem a importância da sua execução? <i>(Indicadores: 1.51.PG.1 e 1.51.PG.2)</i>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<b>AC2.1</b> A <b>Gestão dos Requisitos</b> é gerida de forma a produzir os produtos de trabalho com os recursos previstos e dentro do prazo? <i>(Indicadores: 2.1.PG.1, 2.1.PG.2, 2.1.PG.3, 2.1.PG.4 e 2.1.PG.5)</i>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<b>AC2.2</b> A <b>Gestão dos Requisitos</b> é gerida para produzir produtos de trabalho documentados, controlados e verificados, em linha com os objectivos de qualidade definidos? <i>(Indicadores: 2.2.PG.1, 2.2.PG.2, 2.2.PG.3 e 2.2.PG.4)</i>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<b>AC3.1</b> A <b>Gestão dos Requisitos</b> é baseada num processo normalizado? <i>(Indicadores: 3.1.PG.1, 3.1.PG.2, 3.1.PG.3, 3.1.PG.4 e 3.1.PG.5)</i>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<b>AC3.2</b> A <b>Gestão dos Requisitos</b> utiliza recursos humanos adaptados e com aptidões e uma infra-estrutura do processo, de modo a permitir a transferência efectiva como um processo definido? <i>(Indicadores: 3.2.PG.1, 3.2.PG.2, 3.2.PG.3, 3.2.PG.4 e 3.2.PG.5)</i>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<b>AK1.1</b> A <b>Gestão dos Requisitos</b> utiliza uma abordagem à Gestão de Conhecimento explicitamente identificada e aplicada? <i>(Indicadores: 1.1.PT.1, 1.1.PT.2, 1.1.PT.3, e 1.1.PT.4)</i>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<b>AK1.2</b> – O tipo de cultura organizacional predominante no processo <b>Gestão dos Requisitos</b> está explicitamente identificado e aplicado? <i>(Indicadores: 1.2.PT.1, 1.2.PT.2 e 1.2.PT.3)</i>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<b>Comentários:</b>				

PROCESSO ENG.21 Análise dos Requisitos do Software				
<b>Actividades executadas:</b> <ul style="list-style-type: none"> <li>• ENG.21.PB1 – Os requisitos do software são especificados? Sim <input type="checkbox"/> Não <input type="checkbox"/></li> <li>• ENG.21.PB2 – São desenvolvidos critérios para o teste de aceitação do software? Sim <input type="checkbox"/> Não <input type="checkbox"/></li> <li>• ENG.21.PB3 – São avaliados e actualizados os requisitos do software? Sim <input type="checkbox"/> Não <input type="checkbox"/></li> <li>• ENG.21.PB4 – É desenvolvida uma versão base dos requisitos do software? Sim <input type="checkbox"/> Não <input type="checkbox"/></li> </ul>				
<b>Documentos a identificar:</b> Relatório de controlo de alterações <input type="checkbox"/> , requisitos de interface <input type="checkbox"/> e requisitos do software <input type="checkbox"/>				
	<b>N</b>	<b>P</b>	<b>L</b>	<b>C</b>
<b>AC1.1</b> O processo <b>Análise dos Requisitos do Software</b> utiliza as práticas de base adequadas e resulta nos produtos de trabalho identificados? <i>(Indicadores: Práticas de Base e Documentos)</i>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<b>AC1.51</b> Os intervenientes do processo <b>Análise dos Requisitos do Software</b> estão identificados e reconhecem a importância da sua execução? <i>(Indicadores: 1.51.PG.1 e 1.51.PG.2)</i>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<b>AC2.1</b> A <b>Análise dos Requisitos do Software</b> é gerida de forma a produzir os produtos de trabalho com os recursos previstos e dentro do prazo? <i>(Indicadores: 2.1.PG.1, 2.1.PG.2, 2.1.PG.3, 2.1.PG.4 e 2.1.PG.5)</i>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<b>AC2.2</b> A <b>Análise dos Requisitos do Software</b> é gerida para produzir produtos de trabalho documentados, controlados e verificados, em linha com os objectivos de qualidade definidos? <i>(Indicadores: 2.2.PG.1, 2.2.PG.2, 2.2.PG.3 e 2.2.PG.4)</i>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<b>AC3.1</b> A <b>Análise dos Requisitos do Software</b> é baseada num processo normalizado? <i>(Indicadores: 3.1.PG.1, 3.1.PG.2, 3.1.PG.3, 3.1.PG.4 e 3.1.PG.5)</i>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<b>AC3.2</b> A <b>Análise dos Requisitos do Software</b> utiliza recursos humanos adaptados e com aptidões e uma infra-estrutura do processo, de modo a permitir a transferência efectiva como um processo definido? <i>(Indicadores: 3.2.PG.1, 3.2.PG.2, 3.2.PG.3, 3.2.PG.4 e 3.2.PG.5)</i>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<b>AK1.1</b> A <b>Análise dos Requisitos do Software</b> utiliza uma abordagem à Gestão de Conhecimento explicitamente identificada e aplicada? <i>(Indicadores: 1.1.PT.1, 1.1.PT.2, 1.1.PT.3, e 1.1.PT.4)</i>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<b>AK1.2</b> – O tipo de cultura organizacional predominante no processo <b>Análise dos Requisitos do Software</b> está explicitamente identificado e aplicado? <i>(Indicadores: 1.2.PT.1, 1.2.PT.2 e 1.2.PT.3)</i>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<b>Comentários:</b>				



PROCESSO ENG.22 Concepção do Software				
<b>Actividades executadas:</b> <ul style="list-style-type: none"> <li>• ENG.22.PB1 – É descrita a arquitectura do software e os interfaces entre os elementos? Sim <input type="checkbox"/> Não <input type="checkbox"/></li> <li>• ENG.22.PB2 – É desenvolvida a concepção detalhada do software? Sim <input type="checkbox"/> Não <input type="checkbox"/></li> <li>• ENG.22.PB3 – A especificação do software é avaliada e especificada? Sim <input type="checkbox"/> Não <input type="checkbox"/></li> <li>• ENG.22.PB4 – É assegurada a consistência com os requisitos do software? Sim <input type="checkbox"/> Não <input type="checkbox"/></li> </ul>				
<b>Documentos a identificar:</b> Especificação da Base de Dados <input type="checkbox"/> , especificação de alto nível do software <input type="checkbox"/> , especificação detalhada do software <input type="checkbox"/> e registo de consistência <input type="checkbox"/>				
	<b>N</b>	<b>P</b>	<b>L</b>	<b>C</b>
<b>AC1.1</b> O processo <b>Concepção do Software</b> utiliza as práticas de base adequadas e resulta nos produtos de trabalho identificados? <i>(Indicadores: Práticas de Base e Documentos)</i>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<b>AC1.51</b> Os intervenientes do processo <b>Concepção do Software</b> estão identificados e reconhecem a importância da sua execução? <i>(Indicadores: 1.51.PG.1 e 1.51.PG.2)</i>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<b>AC2.1</b> A <b>Concepção do Software</b> é gerida de forma a produzir os produtos de trabalho com os recursos previstos e dentro do prazo? <i>(Indicadores: 2.1.PG.1, 2.1.PG.2, 2.1.PG.3, 2.1.PG.4 e 2.1.PG.5)</i>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<b>AC2.2</b> A <b>Concepção do Software</b> é gerida para produzir produtos de trabalho documentados, controlados e verificados, em linha com os objectivos de qualidade definidos? <i>(Indicadores: 2.2.PG.1, 2.2.PG.2, 2.2.PG.3 e 2.2.PG.4)</i>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<b>AC3.1</b> A <b>Concepção do Software</b> é baseada num processo normalizado? <i>(Indicadores: 3.1.PG.1, 3.1.PG.2, 3.1.PG.3, 3.1.PG.4 e 3.1.PG.5)</i>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<b>AC3.2</b> A <b>Concepção do Software</b> utiliza recursos humanos adaptados e com aptidões e uma infra-estrutura do processo, de modo a permitir a transferência efectiva como um processo definido? <i>(Indicadores: 3.2.PG.1, 3.2.PG.2, 3.2.PG.3, 3.2.PG.4 e 3.2.PG.5)</i>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<b>AK1.1</b> A <b>Concepção do Software</b> utiliza uma abordagem à Gestão de Conhecimento explicitamente identificada e aplicada? <i>(Indicadores: 1.1.PT.1, 1.1.PT.2, 1.1.PT.3, e 1.1.PT.4)</i>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<b>AK1.2</b> – O tipo de cultura organizacional predominante no processo <b>Concepção do Software</b> está explicitamente identificado e aplicado? <i>(Indicadores: 1.2.PT.1, 1.2.PT.2 e 1.2.PT.3)</i>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<b>Comentários:</b>				

<b>PROCESSO ENG.23 Construção do Software</b>				
<b>Actividades executadas:</b>				
<ul style="list-style-type: none"> <li>• ENG.23.PB1 – São desenvolvidos os procedimentos para a verificação das unidades? Sim <input type="checkbox"/> Não <input type="checkbox"/></li> <li>• ENG.23.PB2 – São produzidas as unidades de software? Sim <input type="checkbox"/> Não <input type="checkbox"/></li> <li>• ENG.23.PB3 – É estabelecida a ligação com a concepção? Sim <input type="checkbox"/> Não <input type="checkbox"/></li> <li>• ENG.23.PB4 – As unidades de software são verificadas? Sim <input type="checkbox"/> Não <input type="checkbox"/></li> </ul>				
<b>Documentos a identificar:</b> Procedimentos de teste das unidades <input type="checkbox"/> , unidades de software <input type="checkbox"/> , resultados dos testes <input type="checkbox"/> e registo de consistência <input type="checkbox"/>				
	<b>N</b>	<b>P</b>	<b>L</b>	<b>C</b>
<b>AC1.1</b> O processo <b>Construção do Software</b> utiliza as práticas de base adequadas e resulta nos produtos de trabalho identificados? <i>(Indicadores: Práticas de Base e Documentos)</i>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<b>AC1.51</b> Os intervenientes do processo <b>Construção do Software</b> estão identificados e reconhecem a importância da sua execução? <i>(Indicadores: 1.51.PG.1 e 1.51.PG.2)</i>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<b>AC2.1</b> A <b>Construção do Software</b> é gerida de forma a produzir os produtos de trabalho com os recursos previstos e dentro do prazo? <i>(Indicadores: 2.1.PG.1, 2.1.PG.2, 2.1.PG.3, 2.1.PG.4 e 2.1.PG.5)</i>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<b>AC2.2</b> A <b>Construção do Software</b> é gerida para produzir produtos de trabalho documentados, controlados e verificados, em linha com os objectivos de qualidade definidos? <i>(Indicadores: 2.2.PG.1, 2.2.PG.2, 2.2.PG.3 e 2.2.PG.4)</i>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<b>AC3.1</b> A <b>Construção do Software</b> é baseada num processo normalizado? <i>(Indicadores: 3.1.PG.1, 3.1.PG.2, 3.1.PG.3, 3.1.PG.4 e 3.1.PG.5)</i>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<b>AC3.2</b> A <b>Construção do Software</b> utiliza recursos humanos adaptados e com aptidões e uma infra-estrutura do processo, de modo a permitir a transferência efectiva como um processo definido? <i>(Indicadores: 3.2.PG.1, 3.2.PG.2, 3.2.PG.3, 3.2.PG.4 e 3.2.PG.5)</i>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<b>AK1.1</b> A <b>Construção do Software</b> utiliza uma abordagem à Gestão de Conhecimento explicitamente identificada e aplicada? <i>(Indicadores: 1.1.PT.1, 1.1.PT.2, 1.1.PT.3, e 1.1.PT.4)</i>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<b>AK1.2</b> – O tipo de cultura organizacional predominante no processo <b>Construção do Software</b> está explicitamente identificado e aplicado? <i>(Indicadores: 1.2.PT.1, 1.2.PT.2 e 1.2.PT.3)</i>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<b>Comentários:</b>				

PROCESSO ENG.24 Integração do Software				
<b>Actividades executadas:</b>				
<ul style="list-style-type: none"> <li>• ENG.24.PB1 – A estratégia de integração do software é desenvolvida? Sim <input type="checkbox"/> Não <input type="checkbox"/></li> <li>• ENG.24.PB2 – Os testes para os itens integrados são desenvolvidos? Sim <input type="checkbox"/> Não <input type="checkbox"/></li> <li>• ENG.24.PB3 – Os itens de software são integrados? Sim <input type="checkbox"/> Não <input type="checkbox"/></li> <li>• ENG.24.PB4 – Os itens integrados são testados? Sim <input type="checkbox"/> Não <input type="checkbox"/></li> <li>• ENG.24.PB5 – É assegurada a consistência com a especificidade do software? Sim <input type="checkbox"/> Não <input type="checkbox"/></li> <li>• ENG.24.PB6 – Quando necessário, são realizados os testes de regressão dos itens de software? Sim <input type="checkbox"/> Não <input type="checkbox"/></li> <li>• ENG.24.PB7 – O produto de software é integrado? Sim <input type="checkbox"/> Não <input type="checkbox"/></li> </ul>				
<b>Documentos a identificar:</b> Itens do software <input type="checkbox"/> , procedimentos de teste de integração <input type="checkbox"/> , produto integrado <input type="checkbox"/> , resultados dos testes <input type="checkbox"/> e registo de consistência <input type="checkbox"/>				
	<b>N</b>	<b>P</b>	<b>L</b>	<b>C</b>
<b>AC1.1</b> O processo <b>Integração do Software</b> utiliza as práticas de base adequadas e resulta nos produtos de trabalho identificados? <i>(Indicadores: Práticas de Base e Documentos)</i>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<b>AC1.51</b> Os intervenientes do processo <b>Integração do Software</b> estão identificados e reconhecem a importância da sua execução? <i>(Indicadores: 1.51.PG.1 e 1.51.PG.2)</i>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<b>AC2.1</b> A <b>Integração do Software</b> é gerida de forma a produzir os produtos de trabalho com os recursos previstos e dentro do prazo? <i>(Indicadores: 2.1.PG.1, 2.1.PG.2, 2.1.PG.3, 2.1.PG.4 e 2.1.PG.5)</i>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<b>AC2.2</b> A <b>Integração do Software</b> é gerida para produzir produtos de trabalho documentados, controlados e verificados, em linha com os objectivos de qualidade definidos? <i>(Indicadores: 2.2.PG.1, 2.2.PG.2, 2.2.PG.3 e 2.2.PG.4)</i>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<b>AC3.1</b> A <b>Integração do Software</b> é baseada num processo normalizado? <i>(Indicadores: 3.1.PG.1, 3.1.PG.2, 3.1.PG.3, 3.1.PG.4 e 3.1.PG.5)</i>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<b>AC3.2</b> A <b>Integração do Software</b> utiliza recursos humanos adaptados e com aptidões e uma infra-estrutura do processo, de modo a permitir a transferência efectiva como um processo definido? <i>(Indicadores: 3.2.PG.1, 3.2.PG.2, 3.2.PG.3, 3.2.PG.4 e 3.2.PG.5)</i>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<b>AK1.1</b> A <b>Integração do Software</b> utiliza uma abordagem à Gestão de Conhecimento explicitamente identificada e aplicada? <i>(Indicadores: 1.1.PT.1, 1.1.PT.2, 1.1.PT.3, e 1.1.PT.4)</i>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<b>AK1.2</b> – O tipo de cultura organizacional predominante no processo <b>Integração do Software</b> está explicitamente identificado e aplicado? <i>(Indicadores: 1.2.PT.1, 1.2.PT.2 e 1.2.PT.3)</i>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<b>Comentários:</b>				

PROCESSO ENG.25 Testes do Software				
<p><b>Actividades executadas:</b></p> <ul style="list-style-type: none"> <li>• ENG.25.PB1 – São desenvolvidos os testes para o produto integrado? Sim <input type="checkbox"/> Não <input type="checkbox"/></li> <li>• ENG.25.PB2 – O produto integrado é testado? Sim <input type="checkbox"/> Não <input type="checkbox"/></li> <li>• ENG.25.PB3 – Sempre que necessário são realizados os testes de regressão do software? Sim <input type="checkbox"/> Não <input type="checkbox"/></li> <li>• ENG.25.PB4 – O produto para entrega é preparado? Sim <input type="checkbox"/> Não <input type="checkbox"/></li> </ul>				
<p><b>Documentos a identificar:</b> Itens de configuração <input type="checkbox"/>, relatórios de incidente <input type="checkbox"/>, procedimentos de teste <input type="checkbox"/>, relatórios dos testes <input type="checkbox"/> e manual do utilizador <input type="checkbox"/></p>				
	<b>N</b>	<b>P</b>	<b>L</b>	<b>C</b>
<p><b>AC1.1</b> O processo <b>Testes do Software</b> utiliza as práticas de base adequadas e resulta nos produtos de trabalho identificados? <i>(Indicadores: Práticas de Base e Documentos)</i></p>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<p><b>AC1.51</b> Os intervenientes do processo <b>Testes do Software</b> estão identificados e reconhecem a importância da sua execução? <i>(Indicadores: 1.51.PG.1 e 1.51.PG.2)</i></p>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<p><b>AC2.1</b> Os <b>Testes do Software</b> são geridos de forma a produzir os produtos de trabalho com os recursos previstos e dentro do prazo? <i>(Indicadores: 2.1.PG.1, 2.1.PG.2, 2.1.PG.3, 2.1.PG.4 e 2.1.PG.5)</i></p>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<p><b>AC2.2</b> Os <b>Testes do Software</b> são geridos para produzir produtos de trabalho documentados, controlados e verificados, em linha com os objectivos de qualidade definidos? <i>(Indicadores: 2.2.PG.1, 2.2.PG.2, 2.2.PG.3 e 2.2.PG.4)</i></p>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<p><b>AC3.1</b> Os <b>Testes do Software</b> são baseados num processo normalizado? <i>(Indicadores: 3.1.PG.1, 3.1.PG.2, 3.1.PG.3, 3.1.PG.4 e 3.1.PG.5)</i></p>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<p><b>AC3.2</b> Os <b>Testes do Software</b> utilizam recursos humanos adaptados e com aptidões e uma infra-estrutura do processo, de modo a permitir a transferência efectiva como um processo definido? <i>(Indicadores: 3.2.PG.1, 3.2.PG.2, 3.2.PG.3, 3.2.PG.4 e 3.2.PG.5)</i></p>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<p><b>AK1.1</b> Os <b>Testes do Software</b> utilizam uma abordagem à Gestão de Conhecimento explicitamente identificada e aplicada? <i>(Indicadores: 1.1.PT.1, 1.1.PT.2, 1.1.PT.3, e 1.1.PT.4)</i></p>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<p><b>AK1.2</b> – O tipo de cultura organizacional predominante no processo <b>Testes do Software</b> está explicitamente identificado e aplicado? <i>(Indicadores: 1.2.PT.1, 1.2.PT.2 e 1.2.PT.3)</i></p>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<p><b>Comentários:</b></p>				

PROCESSO ENG.3 Manutenção do Software				
<b>Actividades executadas:</b>				
<ul style="list-style-type: none"> <li>• ENG.3.PB1 – É desenvolvida a estratégia de manutenção? Sim <input type="checkbox"/> Não <input type="checkbox"/></li> <li>• ENG.3.PB2 – Estão estabelecidos procedimentos de resposta a pedidos de modificações do utilizador? Sim <input type="checkbox"/> Não <input type="checkbox"/></li> <li>• ENG.3.PB3 – Os problemas e alterações do utilizador são analisados? Sim <input type="checkbox"/> Não <input type="checkbox"/></li> <li>• ENG.3.PB4 – As modificações são implementadas e testadas? Sim <input type="checkbox"/> Não <input type="checkbox"/></li> <li>• ENG.3.PB5 – É realizado o <i>upgrade</i> do software? Sim <input type="checkbox"/> Não <input type="checkbox"/></li> <li>• ENG.3.PB6 – É retirado o produto obsoleto? Sim <input type="checkbox"/> Não <input type="checkbox"/></li> </ul>				
<b>Documentos a identificar:</b> Item de configuração <input type="checkbox"/> , relatório de controlo / acompanhamento de modificações <input type="checkbox"/> , documento com estratégia de manutenção <input type="checkbox"/> , resultados dos testes <input type="checkbox"/> e registo de consistência <input type="checkbox"/>				
	<b>N</b>	<b>P</b>	<b>L</b>	<b>C</b>
<b>AC1.1</b> O processo <b>Manutenção do Software</b> utiliza as práticas de base adequadas e resulta nos produtos de trabalho identificados? <i>(Indicadores: Práticas de Base e Documentos)</i>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<b>AC1.51</b> Os intervenientes do processo <b>Manutenção do Software</b> estão identificados e reconhecem a importância da sua execução? <i>(Indicadores: 1.51.PG.1 e 1.51.PG.2)</i>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<b>AC2.1</b> A <b>Manutenção do Software</b> é gerida de forma a produzir os produtos de trabalho com os recursos previstos e dentro do prazo? <i>(indicadores: 2.1.PG.1, 2.1.PG.2, 2.1.PG.3, 2.1.PG.4 e 2.1.PG.5)</i>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<b>AC2.2</b> A <b>Manutenção do Software</b> é gerida para produzir produtos de trabalho documentados, controlados e verificados, em linha com os objectivos de qualidade definidos? <i>(Indicadores: 2.2.PG.1, 2.2.PG.2, 2.2.PG.3 e 2.2.PG.4)</i>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<b>AC3.1</b> A <b>Manutenção do Software</b> é baseada num processo normalizado? <i>(Indicadores: 3.1.PG.1, 3.1.PG.2, 3.1.PG.3, 3.1.PG.4 e 3.1.PG.5)</i>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<b>AC3.2</b> A <b>Manutenção do Software</b> utiliza recursos humanos adaptados e com aptidões e uma infra-estrutura do processo, de modo a permitir a transferência efectiva como um processo definido? <i>(Indicadores: 3.2.PG.1, 3.2.PG.2, 3.2.PG.3, 3.2.PG.4 e 3.2.PG.5)</i>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<b>AK1.1</b> A <b>Manutenção do Software</b> utiliza uma abordagem à Gestão de Conhecimento explicitamente identificada e aplicada? <i>(Indicadores: 1.1.PT.1, 1.1.PT.2, 1.1.PT.3, e 1.1.PT.4)</i>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<b>AK1.2</b> – O tipo de cultura organizacional predominante no processo <b>Manutenção do Software</b> está explicitamente identificado e aplicado? <i>(Indicadores: 1.2.PT.1, 1.2.PT.2 e 1.2.PT.3)</i>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<b>Comentários:</b>				

PROCESSO SUP.1 Documentação				
<b>Actividades executadas:</b>				
<ul style="list-style-type: none"> <li>• SUP.1.PB1 – São estabelecidas normas para os documentar? Sim <input type="checkbox"/> Não <input type="checkbox"/></li> <li>• SUP.1.PB2 – São especificados os requisitos dos documentos? Sim <input type="checkbox"/> Não <input type="checkbox"/></li> <li>• SUP.1.PB3 – São identificados os documentos a produzir? Sim <input type="checkbox"/> Não <input type="checkbox"/></li> <li>• SUP.1.PB4 – São desenvolvidos os documentos? Sim <input type="checkbox"/> Não <input type="checkbox"/></li> <li>• SUP.1.PB5 – Os documentos são distribuídos? Sim <input type="checkbox"/> Não <input type="checkbox"/></li> <li>• SUP.1.PB6 – Os documentos são mantidos? Sim <input type="checkbox"/> Não <input type="checkbox"/></li> </ul>				
<b>Documentos a identificar:</b> Relatório de controlo / acompanhamento de mudanças <input type="checkbox"/> , registo de aceitação <input type="checkbox"/> , registo de entrega <input type="checkbox"/> e produto de trabalho <input type="checkbox"/>				
	<b>N</b>	<b>P</b>	<b>L</b>	<b>C</b>
<b>AC1.1</b> O processo <b>Documentação</b> utiliza as práticas de base adequadas e resulta nos produtos de trabalho identificados? <i>(Indicadores: Práticas de Base e Documentos)</i>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<b>AC1.51</b> Os intervenientes do processo <b>Documentação</b> estão identificados e reconhecem a importância da sua execução? <i>(Indicadores: 1.51.PG.1 e 1.51.PG.2)</i>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<b>AC2.1</b> A <b>Documentação</b> é gerida de forma a produzir os produtos de trabalho com os recursos previstos e dentro do prazo? <i>(Indicadores: 2.1.PG.1, 2.1.PG.2, 2.1.PG.3, 2.1.PG.4 e 2.1.PG.5)</i>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<b>AC2.2</b> A <b>Documentação</b> é gerida para produzir produtos de trabalho documentados, controlados e verificados, em linha com os objectivos de qualidade definidos? <i>(Indicadores: 2.2.PG.1, 2.2.PG.2, 2.2.PG.3 e 2.2.PG.4)</i>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<b>AC3.1</b> A <b>Documentação</b> é baseada num processo normalizado? <i>(Indicadores: 3.1.PG.1, 3.1.PG.2, 3.1.PG.3, 3.1.PG.4 e 3.1.PG.5)</i>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<b>AC3.2</b> A <b>Documentação</b> utiliza recursos humanos adaptados e com aptidões e uma infra-estrutura do processo, de modo a permitir a transferência efectiva como um processo definido? <i>(Indicadores: 3.2.PG.1, 3.2.PG.2, 3.2.PG.3, 3.2.PG.4 e 3.2.PG.5)</i>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<b>AK1.1</b> A <b>Documentação</b> utiliza uma abordagem à Gestão de Conhecimento explicitamente identificada e aplicada? <i>(Indicadores: 1.1.PT.1, 1.1.PT.2, 1.1.PT.3, e 1.1.PT.4)</i>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<b>AK1.2</b> – O tipo de cultura organizacional predominante no processo <b>Documentação</b> está explicitamente identificado e aplicado? <i>(Indicadores: 1.2.PT.1, 1.2.PT.2 e 1.2.PT.3)</i>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<b>Comentários:</b>				

PROCESSO SUP.2 Gestão de Configurações				
<b>Actividades executadas:</b>				
<ul style="list-style-type: none"> <li>• SUP.2.PB1 – Os itens de configuração são identificados? Sim <input type="checkbox"/> Não <input type="checkbox"/></li> <li>• SUP.2.PB2 – São estabelecidas as versões internas e para entrega? Sim <input type="checkbox"/> Não <input type="checkbox"/></li> <li>• SUP.2.PB3 – A descrição dos itens de configuração é mantida? Sim <input type="checkbox"/> Não <input type="checkbox"/></li> <li>• SUP.2.PB4 – São controladas as modificações e versões? Sim <input type="checkbox"/> Não <input type="checkbox"/></li> <li>• SUP.2.PB5 – É mantido o histórico dos itens de configuração? Sim <input type="checkbox"/> Não <input type="checkbox"/></li> <li>• SUP.2.PB6 – São emitidos relatórios sobre o estado da configuração? Sim <input type="checkbox"/> Não <input type="checkbox"/></li> <li>• SUP.2.PB7 – Cópias de segurança, armazenamento, arquivo, manuseamento, e entrega dos itens são geridos? Sim <input type="checkbox"/> Não <input type="checkbox"/></li> </ul>				
<b>Documentos a identificar:</b> Item de configuração <input type="checkbox"/> , guia de manuseamento e armazenamento <input type="checkbox"/> , produto integrado <input type="checkbox"/> e registo do estado do progresso <input type="checkbox"/>				
	<b>N</b>	<b>P</b>	<b>L</b>	<b>C</b>
<b>AC1.1</b> O processo <b>Gestão de Configurações</b> utiliza as práticas de base adequadas e resulta nos produtos de trabalho identificados? <i>(Indicadores: Práticas de Base e Documentos)</i>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<b>AC1.51</b> Os intervenientes do processo <b>Gestão de Configurações</b> estão identificados e reconhecem a importância da sua execução? <i>(Indicadores: 1.51.PG.1 e 1.51.PG.2)</i>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<b>AC2.1</b> A <b>Gestão de Configurações</b> é gerida de forma a produzir os produtos de trabalho com os recursos previstos e dentro do prazo? <i>(Indicadores: 2.1.PG.1, 2.1.PG.2, 2.1.PG.3, 2.1.PG.4 e 2.1.PG.5)</i>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<b>AC2.2</b> A <b>Gestão de Configurações</b> é gerida para produzir produtos de trabalho documentados, controlados e verificados, em linha com os objectivos de qualidade definidos? <i>(Indicadores: 2.2.PG.1, 2.2.PG.2, 2.2.PG.3 e 2.2.PG.4)</i>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<b>AC3.1</b> A <b>Gestão de Configurações</b> é baseada num processo normalizado? <i>(Indicadores: 3.1.PG.1, 3.1.PG.2, 3.1.PG.3, 3.1.PG.4 e 3.1.PG.5)</i>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<b>AC3.2</b> A <b>Gestão de Configurações</b> utiliza recursos humanos adaptados e com aptidões e uma infra-estrutura do processo, de modo a permitir a transferência efectiva como um processo definido? <i>(Indicadores: 3.2.PG.1, 3.2.PG.2, 3.2.PG.3, 3.2.PG.4 e 3.2.PG.5)</i>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<b>AK1.1</b> A <b>Gestão de Configurações</b> utiliza uma abordagem à Gestão de Conhecimento explicitamente identificada e aplicada? <i>(Indicadores: 1.1.PT.1, 1.1.PT.2, 1.1.PT.3, e 1.1.PT.4)</i>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<b>AK1.2</b> – O tipo de cultura organizacional predominante no processo <b>Gestão de Configurações</b> está explicitamente identificado e aplicado? <i>(Indicadores: 1.2.PT.1, 1.2.PT.2 e 1.2.PT.3)</i>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<b>Comentários:</b>				

PROCESSO SUP.3 Garantia da Qualidade				
<b>Actividades executadas:</b>				
<ul style="list-style-type: none"> <li>• SUP.3.PB1 – A estratégia de Garantia da Qualidade é desenvolvida? Sim <input type="checkbox"/> Não <input type="checkbox"/></li> <li>• SUP.3.PB2 – São definidos registos de qualidade do projecto? Sim <input type="checkbox"/> Não <input type="checkbox"/></li> <li>• SUP.3.PB3 – É assegurada a qualidade das actividades do processo e dos produtos de trabalho? Sim <input type="checkbox"/> Não <input type="checkbox"/></li> <li>• SUP.3.PB4 – São produzidos e armazenados os registos das actividades de Garantia de Qualidade? Sim <input type="checkbox"/> Não <input type="checkbox"/></li> <li>• SUP.3.PB5 – Os problemas encontrados são resolvidos? Sim <input type="checkbox"/> Não <input type="checkbox"/></li> </ul>				
<b>Documentos a identificar:</b> Procedimentos de Garantia da Qualidade <input type="checkbox"/> , descrição do processo <input type="checkbox"/> , normas <input type="checkbox"/> , critérios de qualidade <input type="checkbox"/> e relatório de problemas <input type="checkbox"/>				
	<b>N</b>	<b>P</b>	<b>L</b>	<b>C</b>
<b>AC1.1</b> O processo <b>Garantia da Qualidade</b> utiliza as práticas de base adequadas e resulta nos produtos de trabalho identificados? <i>(Indicadores: Práticas de Base e Documentos)</i>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<b>AC1.51</b> Os intervenientes do processo <b>Garantia da Qualidade</b> estão identificados e reconhecem a importância da sua execução? <i>(Indicadores: 1.51.PG.1 e 1.51.PG.2)</i>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<b>AC2.1</b> A <b>Garantia da Qualidade</b> é gerida de forma a produzir os produtos de trabalho com os recursos previstos e dentro do prazo? <i>(Indicadores: 2.1.PG.1, 2.1.PG.2, 2.1.PG.3, 2.1.PG.4 e 2.1.PG.5)</i>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<b>AC2.2</b> A <b>Garantia da Qualidade</b> é gerida para produzir produtos de trabalho documentados, controlados e verificados, em linha com os objectivos de qualidade definidos? <i>(Indicadores: 2.2.PG.1, 2.2.PG.2, 2.2.PG.3 e 2.2.PG.4)</i>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<b>AC3.1</b> A <b>Garantia da Qualidade</b> é baseada num processo normalizado? <i>(Indicadores: 3.1.PG.1, 3.1.PG.2, 3.1.PG.3, 3.1.PG.4 e 3.1.PG.5)</i>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<b>AC3.2</b> A <b>Garantia da Qualidade</b> utiliza recursos humanos adaptados e com aptidões e uma infra-estrutura do processo, de modo a permitir a transferência efectiva como um processo definido? <i>(Indicadores: 3.2.PG.1, 3.2.PG.2, 3.2.PG.3, 3.2.PG.4 e 3.2.PG.5)</i>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<b>AK1.1</b> A <b>Garantia da Qualidade</b> utiliza uma abordagem à Gestão de Conhecimento explicitamente identificada e aplicada? <i>(Indicadores: 1.1.PT.1, 1.1.PT.2, 1.1.PT.3, e 1.1.PT.4)</i>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<b>AK1.2</b> – O tipo de cultura organizacional predominante no processo <b>Garantia da Qualidade</b> está explicitamente identificado e aplicado? <i>(Indicadores: 1.2.PT.1, 1.2.PT.2 e 1.2.PT.3)</i>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<b>Comentários:</b>				



<b>PROCESSO GES.1 Gestão do Projecto</b>				
<b>Actividades executadas:</b>				
<ul style="list-style-type: none"> <li>• GES.1.PB1 – O âmbito do trabalho é definido? Sim <input type="checkbox"/> Não <input type="checkbox"/></li> <li>• GES.1.PB2 – O ciclo de vida do projecto é definido? Sim <input type="checkbox"/> Não <input type="checkbox"/></li> <li>• GES.1.PB3 – São desenvolvidas as estimativas? Sim <input type="checkbox"/> Não <input type="checkbox"/></li> <li>• GES.1.PB4 – As actividades do projecto são definidas? Sim <input type="checkbox"/> Não <input type="checkbox"/></li> <li>• GES.1.PB5 – O cronograma do projecto é definido? Sim <input type="checkbox"/> Não <input type="checkbox"/></li> <li>• GES.1.PB6 – É estabelecido o plano do projecto? Sim <input type="checkbox"/> Não <input type="checkbox"/></li> <li>• GES.1.PB7 – O projecto é implementado? Sim <input type="checkbox"/> Não <input type="checkbox"/></li> <li>• GES.1.PB8 – Os parâmetros do projecto são monitorizados? Sim <input type="checkbox"/> Não <input type="checkbox"/></li> <li>• GES.1.PB9 – O progresso do projecto é revisto? Sim <input type="checkbox"/> Não <input type="checkbox"/></li> <li>• GES.1.PB10 – Os desvios são corrigidos? Sim <input type="checkbox"/> Não <input type="checkbox"/></li> </ul>				
<b>Documentos a identificar:</b> Estimativas <input type="checkbox"/> , plano do projecto <input type="checkbox"/> , modelo do ciclo de vida <input type="checkbox"/> , medições do projecto <input type="checkbox"/> e relatório do estado do projecto <input type="checkbox"/>				
	<b>N</b>	<b>P</b>	<b>L</b>	<b>C</b>
<b>AC1.1</b> O processo <b>Gestão do Projecto</b> utiliza as práticas de base adequadas e resulta nos produtos de trabalho identificados? <i>(Indicadores: Práticas de Base e Documentos)</i>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<b>AC1.51</b> Os intervenientes do processo <b>Gestão do Projecto</b> estão identificados e reconhecem a importância da sua execução? <i>(Indicadores: 1.51.PG.1 e 1.51.PG.2)</i>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<b>AC2.1</b> A <b>Gestão do Projecto</b> é gerida de forma a produzir os produtos de trabalho com os recursos previstos e dentro do prazo? <i>(indicadores: 2.1.PG.1, 2.1.PG.2, 2.1.PG.3, 2.1.PG.4 e 2.1.PG.5)</i>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<b>AC2.2</b> A <b>Gestão do Projecto</b> é gerida para produzir produtos de trabalho documentados, controlados e verificados, em linha com os objectivos de qualidade definidos? <i>(Indicadores: 2.2.PG.1, 2.2.PG.2, 2.2.PG.3 e 2.2.PG.4)</i>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<b>AC3.1</b> A <b>Gestão do Projecto</b> é baseada num processo normalizado? <i>(Indicadores: 3.1.PG.1, 3.1.PG.2, 3.1.PG.3, 3.1.PG.4 e 3.1.PG.5)</i>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<b>AC3.2</b> A <b>Gestão do Projecto</b> utiliza recursos humanos adaptados e com aptidões e uma infra-estrutura do processo, de modo a permitir a transferência efectiva como um processo definido? <i>(Indicadores: 3.2.PG.1, 3.2.PG.2, 3.2.PG.3, 3.2.PG.4 e 3.2.PG.5)</i>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<b>AK1.1</b> A <b>Gestão do Projecto</b> utiliza uma abordagem à Gestão de Conhecimento explicitamente identificada e aplicada? <i>(Indicadores: 1.1.PT.1, 1.1.PT.2, 1.1.PT.3, e 1.1.PT.4)</i>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<b>AK1.2</b> – O tipo de cultura organizacional predominante no processo <b>Gestão do Projecto</b> está explicitamente identificado e aplicado? <i>(Indicadores: 1.2.PT.1, 1.2.PT.2 e 1.2.PT.3)</i>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<b>Comentários:</b>				

PROCESSO GES.2 Gestão do Risco				
<p><b>Actividades executadas:</b></p> <ul style="list-style-type: none"> <li>• GES.2.PB1 – Os riscos são identificados? Sim <input type="checkbox"/> Não <input type="checkbox"/></li> <li>• GES.2.PB2 – Os riscos são analisados e são atribuídas prioridades? Sim <input type="checkbox"/> Não <input type="checkbox"/></li> <li>• GES.2.PB3 – São definidas e executadas acções para reduzir os riscos? Sim <input type="checkbox"/> Não <input type="checkbox"/></li> <li>• GES.2.PB4 – Os riscos são monitorizados? Sim <input type="checkbox"/> Não <input type="checkbox"/></li> <li>• ENG.1.PB5 – São executadas acções correctivas quando falha a mitigação? Sim <input type="checkbox"/> Não <input type="checkbox"/></li> </ul>				
<p><b>Documentos a identificar:</b> Registo de controlo de risco (medição, acção correctiva, acompanhamento) <input type="checkbox"/>, plano de gestão do risco <input type="checkbox"/> e relatório dos riscos <input type="checkbox"/></p>				
	<b>N</b>	<b>P</b>	<b>L</b>	<b>C</b>
<p><b>AC1.1</b> O processo <b>Gestão do Risco</b> utiliza as práticas de base adequadas e resulta nos produtos de trabalho identificados? <i>(Indicadores: Práticas de Base e Documentos)</i></p>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<p><b>AC1.51</b> Os intervenientes do processo <b>Gestão do Risco</b> estão identificados e reconhecem a importância da sua execução? <i>(Indicadores: 1.51.PG.1 e 1.51.PG.2)</i></p>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<p><b>AC2.1</b> A <b>Gestão do Risco</b> é gerida de forma a produzir os produtos de trabalho com os recursos previstos e dentro do prazo? <i>(Indicadores: 2.1.PG.1, 2.1.PG.2, 2.1.PG.3, 2.1.PG.4 e 2.1.PG.5)</i></p>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<p><b>AC2.2</b> A <b>Gestão do Risco</b> é gerida para produzir produtos de trabalho documentados, controlados e verificados, em linha com os objectivos de qualidade definidos? <i>(Indicadores: 2.2.PG.1, 2.2.PG.2, 2.2.PG.3 e 2.2.PG.4)</i></p>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<p><b>AC3.1</b> A <b>Gestão do Risco</b> é baseada num processo normalizado? <i>(Indicadores: 3.1.PG.1, 3.1.PG.2, 3.1.PG.3, 3.1.PG.4 e 3.1.PG.5)</i></p>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<p><b>AC3.2</b> A <b>Gestão do Risco</b> utiliza recursos humanos adaptados e com aptidões e uma infra-estrutura do processo, de modo a permitir a transferência efectiva como um processo definido? <i>(Indicadores: 3.2.PG.1, 3.2.PG.2, 3.2.PG.3, 3.2.PG.4 e 3.2.PG.5)</i></p>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<p><b>AK1.1</b> A <b>Gestão do Risco</b> utiliza uma abordagem à Gestão de Conhecimento explicitamente identificada e aplicada? <i>(Indicadores: 1.1.PT.1, 1.1.PT.2, 1.1.PT.3, e 1.1.PT.4)</i></p>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<p><b>AK1.2</b> – O tipo de cultura organizacional predominante no processo <b>Gestão do Risco</b> está explicitamente identificado e aplicado? <i>(Indicadores: 1.2.PT.1, 1.2.PT.2 e 1.2.PT.3)</i></p>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<p><b>Comentários:</b></p>				

PROCESSO GES.3 Gestão da Qualidade				
<b>Actividades executadas:</b>				
<ul style="list-style-type: none"> <li>• GES.3.PB1 – Os objectivos de qualidade são estabelecidos? Sim <input type="checkbox"/> Não <input type="checkbox"/></li> <li>• GES.3.PB2 – A estratégia global é estabelecida? Sim <input type="checkbox"/> Não <input type="checkbox"/></li> <li>• GES.3.PB3 – Os critérios de qualidade são definidos? Sim <input type="checkbox"/> Não <input type="checkbox"/></li> <li>• GES.3.PB4 – Está estabelecido um sistema de gestão da qualidade? Sim <input type="checkbox"/> Não <input type="checkbox"/></li> <li>• ENG.3.PB5 – É avaliado o cumprimento dos objectivos de qualidade? Sim <input type="checkbox"/> Não <input type="checkbox"/></li> <li>• ENG.3.PB6 – São executadas acções correctivas? Sim <input type="checkbox"/> Não <input type="checkbox"/></li> </ul>				
<b>Documentos a identificar:</b> Objectivos de qualidade <input type="checkbox"/> , medição de qualidade <input type="checkbox"/> , plano de qualidade <input type="checkbox"/> , política de qualidade <input type="checkbox"/> , registo de revisão <input type="checkbox"/> , registo de acção correctiva <input type="checkbox"/> , análise de resultados <input type="checkbox"/> e critérios de aceitação <input type="checkbox"/>				
	<b>N</b>	<b>P</b>	<b>L</b>	<b>C</b>
<b>AC1.1</b> O processo <b>Gestão da Qualidade</b> utiliza as práticas de base adequadas e resulta nos produtos de trabalho identificados? <i>(Indicadores: Práticas de Base e Documentos)</i>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<b>AC1.51</b> Os intervenientes do processo <b>Gestão da Qualidade</b> estão identificados e reconhecem a importância da sua execução? <i>(Indicadores: 1.51.PG.1 e 1.51.PG.2)</i>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<b>AC2.1</b> A <b>Gestão da Qualidade</b> é gerida de forma a produzir os produtos de trabalho com os recursos previstos e dentro do prazo? <i>(indicadores: 2.1.PG.1, 2.1.PG.2, 2.1.PG.3, 2.1.PG.4 e 2.1.PG.5)</i>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<b>AC2.2</b> A <b>Gestão da Qualidade</b> é gerida para produzir produtos de trabalho documentados, controlados e verificados, em linha com os objectivos de qualidade definidos? <i>(Indicadores: 2.2.PG.1, 2.2.PG.2, 2.2.PG.3 e 2.2.PG.4)</i>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<b>AC3.1</b> A <b>Gestão da Qualidade</b> é baseada num processo normalizado? <i>(Indicadores: 3.1.PG.1, 3.1.PG.2, 3.1.PG.3, 3.1.PG.4 e 3.1.PG.5)</i>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<b>AC3.2</b> A <b>Gestão da Qualidade</b> utiliza recursos humanos adaptados e com aptidões e uma infra-estrutura do processo, de modo a permitir a transferência efectiva como um processo definido? <i>(Indicadores: 3.2.PG.1, 3.2.PG.2, 3.2.PG.3, 3.2.PG.4 e 3.2.PG.5)</i>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<b>AK1.1</b> A <b>Gestão da Qualidade</b> utiliza uma abordagem à Gestão de Conhecimento explicitamente identificada e aplicada? <i>(Indicadores: 1.1.PT.1, 1.1.PT.2, 1.1.PT.3, e 1.1.PT.4)</i>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<b>AK1.2</b> – O tipo de cultura organizacional predominante no processo <b>Gestão da Qualidade</b> está explicitamente identificado e aplicado? <i>(Indicadores: 1.2.PT.1, 1.2.PT.2 e 1.2.PT.3)</i>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<b>Comentários:</b>				

PROCESSO ORG.11 Gestão de Recursos Humanos				
<b>Actividades executadas:</b>				
<ul style="list-style-type: none"> <li>• ORG.11.PB1 – São identificados aptidões e competências necessários? Sim <input type="checkbox"/> Não <input type="checkbox"/></li> <li>• ORG.11.PB2 – São definidos critérios de avaliação do desempenho? Sim <input type="checkbox"/> Não <input type="checkbox"/></li> <li>• ORG.11.PB3 – É recrutado pessoal qualificado? Sim <input type="checkbox"/> Não <input type="checkbox"/></li> <li>• ORG.11.PB4 – São dadas oportunidades para o desenvolvimento das aptidões e competências? Sim <input type="checkbox"/> Não <input type="checkbox"/></li> <li>• ORG.11.PB5 – O desempenho do pessoal é avaliado? Sim <input type="checkbox"/> Não <input type="checkbox"/></li> <li>• ORG.11.PB6 – É dado <i>feedback</i> ao pessoal sobre o desempenho? Sim <input type="checkbox"/> Não <input type="checkbox"/></li> <li>• ORG.11.PB7 – São mantidos registos do pessoal? Sim <input type="checkbox"/> Não <input type="checkbox"/></li> </ul>				
<b>Documentos a identificar:</b> Plano de Recursos Humanos <input type="checkbox"/> , avaliação do desempenho do pessoal <input type="checkbox"/> , plano de formação <input type="checkbox"/> , registos do pessoal <input type="checkbox"/> , registos de formação <input type="checkbox"/> e critérios de desempenho <input type="checkbox"/>				
	<b>N</b>	<b>P</b>	<b>L</b>	<b>C</b>
<b>AC1.1</b> O processo <b>Gestão de Recursos Humanos</b> utiliza as práticas de base adequadas e resulta nos produtos de trabalho identificados? <i>(Indicadores: Práticas de Base e Documentos)</i>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<b>AC1.51</b> Os intervenientes do processo <b>Gestão de Recursos Humanos</b> estão identificados e reconhecem a importância da sua execução? <i>(Indicadores: 1.51.PG.1 e 1.51.PG.2)</i>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<b>AC2.1</b> A <b>Gestão de Recursos Humanos</b> é gerida de forma a produzir os produtos de trabalho com os recursos previstos e dentro do prazo? <i>(Indicadores: 2.1.PG.1, 2.1.PG.2, 2.1.PG.3, 2.1.PG.4 e 2.1.PG.5)</i>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<b>AC2.2</b> A <b>Gestão de Recursos Humanos</b> é gerida para produzir produtos de trabalho documentados, controlados e verificados, em linha com os objectivos de qualidade definidos? <i>(Indicadores: 2.2.PG.1, 2.2.PG.2, 2.2.PG.3 e 2.2.PG.4)</i>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<b>AC3.1</b> A <b>Gestão de Recursos Humanos</b> é baseada num processo normalizado? <i>(Indicadores: 3.1.PG.1, 3.1.PG.2, 3.1.PG.3, 3.1.PG.4 e 3.1.PG.5)</i>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<b>AC3.2</b> A <b>Gestão de Recursos Humanos</b> utiliza recursos humanos adaptados e com aptidões e uma infra-estrutura do processo, de modo a permitir a transferência efectiva como um processo definido? <i>(Indicadores: 3.2.PG.1, 3.2.PG.2, 3.2.PG.3, 3.2.PG.4 e 3.2.PG.5)</i>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<b>AK1.1</b> A <b>Gestão de Recursos Humanos</b> utiliza uma abordagem à Gestão de Conhecimento explicitamente identificada e aplicada? <i>(Indicadores: 1.1.PT.1, 1.1.PT.2, 1.1.PT.3, e 1.1.PT.4)</i>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<b>AK1.2</b> – O tipo de cultura organizacional predominante no processo <b>Gestão de Recursos Humanos</b> está explicitamente identificado e aplicado? <i>(Indicadores: 1.2.PT.1, 1.2.PT.2 e 1.2.PT.3)</i>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<b>Comentários:</b>				

PROCESSO ORG.12 Formação				
<b>Actividades executadas:</b>				
<ul style="list-style-type: none"> <li>• ORG.12.PB1 – As necessidades de formação são identificadas? Sim <input type="checkbox"/> Não <input type="checkbox"/></li> <li>• ORG.12.PB2 – A formação é desenvolvida ou adquirida? Sim <input type="checkbox"/> Não <input type="checkbox"/></li> <li>• ORG.12.PB3 – A execução da formação é preparada? Sim <input type="checkbox"/> Não <input type="checkbox"/></li> <li>• ORG.12.PB4 – É dada formação ao pessoal? Sim <input type="checkbox"/> Não <input type="checkbox"/></li> <li>• ORG.12.PB5 – São mantidos registos sobre a formação? Sim <input type="checkbox"/> Não <input type="checkbox"/></li> <li>• ORG.12.PB6 – É avaliada a efectividade da formação? Sim <input type="checkbox"/> Não <input type="checkbox"/></li> </ul>				
<b>Documentos a identificar:</b> Plano de formação <input type="checkbox"/> , material de formação <input type="checkbox"/> e registo da formação <input type="checkbox"/>				
	<b>N</b>	<b>P</b>	<b>L</b>	<b>C</b>
<b>AC1.1</b> O processo <b>Formação</b> utiliza as práticas de base adequadas e resulta nos produtos de trabalho identificados? <i>(Indicadores: Práticas de Base e Documentos)</i>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<b>AC1.51</b> Os intervenientes do processo <b>Formação</b> estão identificados e reconhecem a importância da sua execução? <i>(Indicadores: 1.51.PG.1 e 1.51.PG.2)</i>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<b>AC2.1</b> A <b>Formação</b> é gerida de forma a produzir os produtos de trabalho com os recursos previstos e dentro do prazo? <i>(Indicadores: 2.1.PG.1, 2.1.PG.2, 2.1.PG.3, 2.1.PG.4 e 2.1.PG.5)</i>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<b>AC2.2</b> A <b>Formação</b> é gerida para produzir produtos de trabalho documentados, controlados e verificados, em linha com os objectivos de qualidade definidos? <i>(Indicadores: 2.2.PG.1, 2.2.PG.2, 2.2.PG.3 e 2.2.PG.4)</i>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<b>AC3.1</b> A <b>Formação</b> é baseada num processo normalizado? <i>(Indicadores: 3.1.PG.1, 3.1.PG.2, 3.1.PG.3, 3.1.PG.4 e 3.1.PG.5)</i>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<b>AC3.2</b> A <b>Formação</b> utiliza recursos humanos adaptados e com aptidões e uma infra-estrutura do processo, de modo a permitir a transferência efectiva como um processo definido? <i>(Indicadores: 3.2.PG.1, 3.2.PG.2, 3.2.PG.3, 3.2.PG.4 e 3.2.PG.5)</i>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<b>AK1.1</b> A <b>Formação</b> utiliza uma abordagem à Gestão de Conhecimento explicitamente identificada e aplicada? <i>(Indicadores: 1.1.PT.1, 1.1.PT.2, 1.1.PT.3, e 1.1.PT.4)</i>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<b>AK1.2</b> – O tipo de cultura organizacional predominante no processo <b>Formação</b> está explicitamente identificado e aplicado? <i>(Indicadores: 1.2.PT.1, 1.2.PT.2 e 1.2.PT.3)</i>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<b>Comentários:</b>				

PROCESSO ORG.13 Gestão do Conhecimento				
<b>Actividades executadas:</b>				
<ul style="list-style-type: none"> <li>• ORG.13.PB1 – Está estabelecido o sistema de Gestão do Conhecimento? Sim <input type="checkbox"/> Não <input type="checkbox"/></li> <li>• ORG.13.PB2 – Está desenvolvida uma estratégia de Gestão do Conhecimento? Sim <input type="checkbox"/> Não <input type="checkbox"/></li> <li>• ORG.13.PB3 – Os assets de Gestão do Conhecimento são geridos? Sim <input type="checkbox"/> Não <input type="checkbox"/></li> </ul>				
<b>Documentos a identificar:</b> Repositório de conhecimento <input type="checkbox"/> e estratégia de conhecimento <input type="checkbox"/>				
	<b>N</b>	<b>P</b>	<b>L</b>	<b>C</b>
<b>AC1.1</b> O processo <b>Gestão do Conhecimento</b> utiliza as práticas de base adequadas e resulta nos produtos de trabalho identificados? <i>(Indicadores: Práticas de Base e Documentos)</i>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<b>AC1.51</b> Os intervenientes do processo <b>Gestão do Conhecimento</b> estão identificados e reconhecem a importância da sua execução? <i>(Indicadores: 1.51.PG.1 e 1.51.PG.2)</i>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<b>AC2.1</b> A <b>Gestão do Conhecimento</b> é gerida de forma a produzir os produtos de trabalho com os recursos previstos e dentro do prazo? <i>(Indicadores: 2.1.PG.1, 2.1.PG.2, 2.1.PG.3, 2.1.PG.4 e 2.1.PG.5)</i>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<b>AC2.2</b> A <b>Gestão do Conhecimento</b> é gerida para produzir produtos de trabalho documentados, controlados e verificados, em linha com os objectivos de qualidade definidos? <i>(Indicadores: 2.2.PG.1, 2.2.PG.2, 2.2.PG.3 e 2.2.PG.4)</i>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<b>AC3.1</b> A <b>Gestão do Conhecimento</b> é baseada num processo normalizado? <i>(Indicadores: 3.1.PG.1, 3.1.PG.2, 3.1.PG.3, 3.1.PG.4 e 3.1.PG.5)</i>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<b>AC3.2</b> A <b>Gestão do Conhecimento</b> utiliza recursos humanos adaptados e com aptidões e uma infra-estrutura do processo, de modo a permitir a transferência efectiva como um processo definido? <i>(Indicadores: 3.2.PG.1, 3.2.PG.2, 3.2.PG.3, 3.2.PG.4 e 3.2.PG.5)</i>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<b>AK1.1</b> A <b>Gestão do Conhecimento</b> utiliza uma abordagem à Gestão de Conhecimento explicitamente identificada e aplicada? <i>(Indicadores: 1.1.PT.1, 1.1.PT.2, 1.1.PT.3, e 1.1.PT.4)</i>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<b>AK1.2</b> – O tipo de cultura organizacional predominante no processo <b>Gestão do Conhecimento</b> está explicitamente identificado e aplicado? <i>(Indicadores: 1.2.PT.1, 1.2.PT.2 e 1.2.PT.3)</i>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<b>Comentários:</b>				

PROCESSO ORG.2 Estabelecimento do Processo				
<b>Actividades executadas:</b>				
<ul style="list-style-type: none"> <li>• ORG.2.PB1 – Os objectivos dos processos normalizados são definidos? Sim <input type="checkbox"/> Não <input type="checkbox"/></li> <li>• ORG.2.PB2 – Os processos estão documentados? Sim <input type="checkbox"/> Não <input type="checkbox"/></li> <li>• ORG.2.PB3 – As expectativas de desempenho são identificadas? Sim <input type="checkbox"/> Não <input type="checkbox"/></li> <li>• ORG.2.PB4 – Estão estabelecidas orientações para colocar à medida? Sim <input type="checkbox"/> Não <input type="checkbox"/></li> <li>• ORG.2.PB5 – Os dados sobre os processos são mantidos? Sim <input type="checkbox"/> Não <input type="checkbox"/></li> <li>• ORG.2.PB6 – As descrições dos processos são mantidas? Sim <input type="checkbox"/> Não <input type="checkbox"/></li> </ul>				
<b>Documentos a identificar:</b> Descrição dos processos <input type="checkbox"/> , registo de qualidade <input type="checkbox"/> , registo de revisão <input type="checkbox"/> , análise de resultados <input type="checkbox"/> e repositório dos processos <input type="checkbox"/>				
	<b>N</b>	<b>P</b>	<b>L</b>	<b>C</b>
<b>AC1.1</b> O processo <b>Estabelecimento do Processo</b> utiliza as práticas de base adequadas e resulta nos produtos de trabalho identificados? <i>(Indicadores: Práticas de Base e Documentos)</i>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<b>AC1.51</b> Os intervenientes do processo <b>Estabelecimento do Processo</b> estão identificados e reconhecem a importância da sua execução? <i>(Indicadores: 1.51.PG.1 e 1.51.PG.2)</i>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<b>AC2.1</b> O <b>Estabelecimento do Processo</b> é gerido de forma a produzir os produtos de trabalho com os recursos previstos e dentro do prazo? <i>(Indicadores: 2.1.PG.1, 2.1.PG.2, 2.1.PG.3, 2.1.PG.4 e 2.1.PG.5)</i>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<b>AC2.2</b> O <b>Estabelecimento do Processo</b> é gerido para produzir produtos de trabalho documentados, controlados e verificados, em linha com os objectivos de qualidade definidos? <i>(Indicadores: 2.2.PG.1, 2.2.PG.2, 2.2.PG.3 e 2.2.PG.4)</i>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<b>AC3.1</b> O <b>Estabelecimento do Processo</b> é baseada num processo normalizado? <i>(Indicadores: 3.1.PG.1, 3.1.PG.2, 3.1.PG.3, 3.1.PG.4 e 3.1.PG.5)</i>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<b>AC3.2</b> O <b>Estabelecimento do Processo</b> utiliza recursos humanos adaptados e com aptidões e uma infra-estrutura do processo, de modo a permitir a transferência efectiva como um processo definido? <i>(Indicadores: 3.2.PG.1, 3.2.PG.2, 3.2.PG.3, 3.2.PG.4 e 3.2.PG.5)</i>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<b>AK1.1</b> O <b>Estabelecimento do Processo</b> utiliza uma abordagem à Gestão de Conhecimento explicitamente identificada e aplicada? <i>(Indicadores: 1.1.PT.1, 1.1.PT.2, 1.1.PT.3, e 1.1.PT.4)</i>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<b>AK1.2</b> – O tipo de cultura organizacional predominante no processo <b>Estabelecimento do Processo</b> está explicitamente identificado e aplicado? <i>(Indicadores: 1.2.PT.1, 1.2.PT.2 e 1.2.PT.3)</i>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<b>Comentários:</b>				

<b>PROCESSO ORG.3 Suporte ao Cliente</b>				
<b>Actividades executadas:</b>				
<ul style="list-style-type: none"> <li>• ORG.3.PB1 – É estabelecido o suporte ao cliente? Sim <input type="checkbox"/> Não <input type="checkbox"/></li> <li>• ORG.3.PB2 – É fornecida formação e documentação aos utilizadores? Sim <input type="checkbox"/> Não <input type="checkbox"/></li> <li>• ORG.3.PB3 – É determinado o nível de satisfação do cliente? Sim <input type="checkbox"/> Não <input type="checkbox"/></li> <li>• ORG.3.PB4 – É comunicada a satisfação do cliente? Sim <input type="checkbox"/> Não <input type="checkbox"/></li> </ul>				
<b>Documentos a identificar:</b> Dados sobre satisfação do cliente <input type="checkbox"/> , relatório de problemas <input type="checkbox"/> , medição do nível de serviço <input type="checkbox"/> e registo de formação <input type="checkbox"/>				
	<b>N</b>	<b>P</b>	<b>L</b>	<b>C</b>
<b>AC1.1</b> O processo <b>Suporte ao Cliente</b> utiliza as práticas de base adequadas e resulta nos produtos de trabalho identificados? <i>(Indicadores: Práticas de Base e Documentos)</i>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<b>AC1.51</b> Os intervenientes do processo <b>Suporte ao Cliente</b> estão identificados e reconhecem a importância da sua execução? <i>(Indicadores: 1.51.PG.1 e 1.51.PG.2)</i>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<b>AC2.1</b> O <b>Suporte ao Cliente</b> é gerido de forma a produzir os produtos de trabalho com os recursos previstos e dentro do prazo? <i>(Indicadores: 2.1.PG.1, 2.1.PG.2, 2.1.PG.3, 2.1.PG.4 e 2.1.PG.5)</i>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<b>AC2.2</b> O <b>Suporte ao Cliente</b> é gerido para produzir produtos de trabalho documentados, controlados e verificados, em linha com os objectivos de qualidade definidos? <i>(Indicadores: 2.2.PG.1, 2.2.PG.2, 2.2.PG.3 e 2.2.PG.4)</i>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<b>AC3.1</b> O <b>Suporte ao Cliente</b> é baseada num processo normalizado? <i>(Indicadores: 3.1.PG.1, 3.1.PG.2, 3.1.PG.3, 3.1.PG.4 e 3.1.PG.5)</i>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<b>AC3.2</b> O <b>Suporte ao Cliente</b> utiliza recursos humanos adaptados e com aptidões e uma infra-estrutura do processo, de modo a permitir a transferência efectiva como um processo definido? <i>(Indicadores: 3.2.PG.1, 3.2.PG.2, 3.2.PG.3, 3.2.PG.4 e 3.2.PG.5)</i>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<b>AK1.1</b> O <b>Suporte ao Cliente</b> utiliza uma abordagem à Gestão de Conhecimento explicitamente identificada e aplicada? <i>(Indicadores: 1.1.PT.1, 1.1.PT.2, 1.1.PT.3, e 1.1.PT.4)</i>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<b>AK1.2</b> – O tipo de cultura organizacional predominante no processo <b>Suporte ao Cliente</b> está explicitamente identificado e aplicado? <i>(Indicadores: 1.2.PT.1, 1.2.PT.2 e 1.2.PT.3)</i>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<b>Comentários:</b>				



## Anexo – Indicadores dos atributos

Descrevem-se em seguida os indicadores que deverão ser utilizados para responder às questões das ACs e AKs. A única excepção é o AC1.1 que usa como indicadores as práticas de base e os produtos de trabalho.

- AC1.51 - *Atributo de valorização do processo*
  - 1.51.PG.1 – *Identificar os intervenientes do processo*, não só os intervenientes directos (executores), mas também responsáveis, consultores e outros intervenientes indirectos.
  - 1.51.PG.2 – *Reconhecer a importância da execução*, todos os intervenientes estão conscientes da contribuição do processo para os objectivos do negócio.
- AC2.1 – *Atributo de gestão da execução*.
  - 2.1.PG.1 – *Identificar os objectivos de desempenho*. Devem incluir: qualidade dos artefactos; tempo de ciclo e utilização de recursos.
  - 2.1.PG.2 – *Planear o desempenho do processo*, incluindo atribuição de responsabilidades, desenvolvimento do plano, definição de prazos e planeamento de revisões.
  - 2.1.PG.3 – *Monitorizar e controlar o desempenho do processo*, incluindo execução de acções quando os resultados não são atingidos, ajustamentos no plano e no cronograma, caso sejam necessários.
  - 2.1.PG.4 – *Atribuir e utilizar recursos*, incluindo identificação e disponibilização de recursos e disponibilização da informação necessária para executar o processo.
  - 2.1.PG.5 – *Gestão dos interfaces entre os intervenientes*. Os intervenientes foram identificados no nível 1.5 e é necessário atribuir responsabilidades e gerir as ligações.
- AC2.2 – *Atributo de gestão do produto de trabalho*.
  - 2.2.PG.1 – *Definir os requisitos dos produtos de trabalho*, incluindo critérios de qualidade e critérios para aprovação e revisão.
  - 2.2.PG.2 – *Definir requisitos de documentação e controlo*, incluindo a identificação e definição das dependências entre os produtos de trabalho.
  - 2.2.PG.3 – *Identificar, documentar e controlar os produtos de trabalho*. Realizar estas actividades de acordo com os requisitos, incluindo a disponibilização do produto de trabalho através de mecanismos de acesso adequados.
  - 2.2.PG.4 – *Rever ajustar os produtos de trabalho*. Quando necessário, realizar estas actividades de modo a que os produtos de trabalho cumpram os requisitos definidos.
- AC3.1 – *Atributo de definição do processo*.
  - 3.1.PG.1 – *Definir o processo normalizado*, incluindo as necessidades e contexto para a transferência, orientações para colocar à medida o processo normalizado e procedimentos para suportar a implementação do processo.
  - 3.1.PG.2 – *Determinar a sequência e interacção entre os processos*.

- 3.1.PG.3 – *Identificar autoridade, papéis, responsabilidades e competências, necessárias para a execução do processo.*
- 3.1.PG.4 – *Identificar os requisitos para a infra-estrutura e para o ambiente de trabalho, incluindo métodos, ferramentas, hardware, redes e outros.*
- 3.1.PG.5 – *Determinar métodos adaptados para monitorizar o processo, incluindo os critérios para aferir a efectividade do processo e os dados necessários para monitorizar os critérios.*
- AC3.2 – *Atributo de transferência do processo.*
  - 3.2.PG.1 – *Transferir um processo definido, através da colocação à medida do processo normalizado.*
  - 3.2.PG.2 – *Atribuir recursos e disponibilizar informação, de modo a possibilitar a transferência do processo definido. Devem ser atribuídos e comunicados os papéis, responsabilidades e autoridade.*
  - 3.2.PG.3 – *Assegurar as competências necessárias, incluindo a identificação das competências apropriadas e a disponibilização da formação adequada.*
  - 3.2.PG.4 – *Fornecer a infra-estrutura adequada, incluindo a disponibilização, utilização e manutenção da infra-estrutura.*
  - 3.2.PG.5 – *Coligir e analisar dados sobre o desempenho do processo, permitindo compreender o comportamento e efectividade do processo definido.*
- AK1.1 – *Atributo de Gestão do Conhecimento*
  - 1.1.PT.1 – *Identificar o perfil de gestão de conhecimento, no nível dos indivíduos, das equipas e específico do processo.*
  - 1.1.PT.2 – *Documentar o impacto dos papéis no processo, incluindo os interfaces entre os diversos perfis.*
  - 1.1.PT.3 – *Alinhar os perfis de gestão de conhecimento.*
- AK1.2 – *Atributo de Gestão da Cultura Organizacional*
  - 1.1.PT.4 – *Fornecer a infra-estrutura adequada, incluindo a disponibilização, implementação e manutenção da infra-estrutura adequada para o tipo de conhecimento identificado.*
  - 1.2.PT.1 – *Identificar o tipo de cultura organizacional predominante.*
  - 1.2.PT.2 – *Documentar o impacto da cultura organizacional no processo.*
  - 1.2.PT.3 – *Definir e manter um sistema de acompanhamento da dinâmica da cultura organizacional.*

## **Anexo D – Tradução da Parte 2 da Norma ISO/IEC 15504**

A parte 2 da norma ISO/IEC 15504 apresenta os seguintes requisitos sobre o processo de avaliação.

### “4.2. Processo de avaliação

4.2.1 A avaliação deve ser conduzida de acordo com um processo de avaliação documentado, capaz de atingir o propósito da avaliação

4.2.2 O processo de avaliação documentado deve conter no mínimo as seguintes actividades:

- a) Planeamento. Um plano para avaliação deve ser desenvolvido e documentado, incluindo, no mínimo:
  - 1) as entradas necessárias especificadas nesta parte da ISO/IEC 15504;
  - 2) as actividades a serem executadas na condução da avaliação;
  - 3) recursos e prazos atribuídos a estas actividades;
  - 4) identificação e definição de responsabilidades dos participantes na avaliação;
  - 5) critérios para verificar que os requisitos da norma internacional foram cumpridos;
  - 6) descrição das saídas planeadas da avaliação.
  
- b) Recolha de dados. Os dados necessários para avaliar os processos no âmbito da avaliação (ver 4.4.2 c)) e informação adicional (ver 4.4.2 j)) devem ser recolhidos de uma forma sistemática aplicando, no mínimo, o seguinte:
  - 1) a estratégia e as técnicas para a selecção, recolha e análise dos dados e a justificação das classificações devem ser explicitamente identificados e devem ser demonstrados;

- 2) deve ser estabelecida correspondência entre os processos das unidades organizacionais, especificados no âmbito da avaliação, e os elementos do modelo de avaliação do processo;
  - 3) cada processo identificado no âmbito da avaliação, deve ser avaliado com base em evidência objectiva;
  - 4) a evidência objectiva recolhida para cada atributo de cada processo avaliado deve ser suficiente para cumprir o propósito e âmbito da avaliação;
  - 5) a identificação da evidência objectiva recolhida deve ser armazenada e mantida para fornecer a base para verificar as classificações;
- c) Validação dos dados. Os dados recolhidos devem ser validados para:
- 1) confirmar que a evidência recolhida é objectiva;
  - 2) assegurar que a evidência objectiva é suficiente e representativa para cobrir o âmbito e propósito da avaliação;
  - 3) assegurar que os dados, como um todo são consistentes.
- d) Classificação dos atributos do processo. Deve ser atribuída uma classificação a cada atributo do processo, com base nos dados validados.
- 1) o conjunto de atributos do processo deve ser armazenado como o perfil do processo para cada unidade organizacional definida;
  - 2) durante a avaliação, o conjunto de indicadores de avaliação definido no modelo de avaliação do processo deve ser utilizado para suportar o julgamento dos avaliadores para classificar os atributos do processo de modo a fornecer a base para a repetibilidade entre avaliações;
  - 3) o processo de tomada de decisão utilizado para derivar o julgamento das classificações deve ser armazenado;
  - 4) deve ser mantida a consistência entre a classificação de um atributo e a evidência objectiva utilizada na determinação dessa classificação;
  - 5) para cada atributo do processo classificado, deve ser armazenado o relacionamento entre os indicadores e a evidência objectiva.

- e) Elaboração do relatório. Os resultados da avaliação incluindo, no mínimo, as saídas especificadas na secção 4.5, devem ser documentadas e comunicadas ao patrocinador da avaliação.”

[ISO/IEC 2001]