



# **Guia de Contagem de Ponto de Função**

## **SERPRO – Ministério da Justiça**

### **Versão 1.0**

**Brasília, outubro de 2015**

# Sumário

<b>1. INTRODUÇÃO</b>	<b>4</b>
<b>2. OBJETIVO</b>	<b>4</b>
<b>3. ESTIMATIVAS E TIPOS DE CONTAGEM DE PROJETOS</b>	<b>5</b>
<b>3.1. CONTAGEM DE PONTOS DE FUNÇÃO PARA PROJETOS ÁGEIS</b>	<b>11</b>
<b>3.2. CONTAGEM ESTIMATIVA DE PONTOS DE FUNÇÃO (CEPF)</b>	<b>12</b>
<b>3.3. ESTIMATIVA DE PRAZO DE PROJETOS DE SOFTWARE</b>	<b>17</b>
<b>4. CONTAGEM DE PONTOS DE FUNÇÃO</b>	<b>20</b>
<b>4.1. PROJETO DE DESENVOLVIMENTO</b>	<b>21</b>
<b>4.2. PROJETO DE MIGRAÇÃO DE DADOS</b>	<b>21</b>
<b>4.3. PROJETO DE MELHORIA</b>	<b>22</b>
<b>4.4. MANUTENÇÃO CORRETIVA</b>	<b>24</b>
<b>4.5. MANUTENÇÃO PREVENTIVA</b>	<b>25</b>
<b>4.6. REDESENVOLVIMENTO DE PROJETOS EM OUTRA PLATAFORMA</b>	<b>25</b>
<b>4.7. ATUALIZAÇÃO DE VERSÃO</b>	<b>26</b>
<b>4.8. MANUTENÇÃO ADAPTATIVA EM REQUISITOS NÃO FUNCIONAIS</b>	<b>27</b>
4.8.1. <i>Manutenção Adaptativa não Arquitetural (Faturamento em PF)</i>	27
4.8.2. <i>Manutenção Adaptativa Arquitetural (Faturamento em Homem-Hora)</i>	28
<b>4.9. DESENVOLVIMENTO E MANUTENÇÃO DE COMPONENTES INTERNOS REUSÁVEIS (NÃO ARQUITETURAIS E FATURADOS EM PF)</b>	<b>29</b>
<b>4.10. DESENVOLVIMENTO OU MANUTENÇÃO DE COMPONENTES ARQUITETURAIS (FATURADOS EM HOMEM-HORA)</b>	<b>30</b>
<b>4.11. MANUTENÇÃO EM INTERFACE DE PROCESSOS ELEMENTARES</b>	<b>31</b>
<b>4.12. APURAÇÃO ESPECIAL</b>	<b>32</b>
4.12.1. <i>Apuração Especial – Base de Dados</i>	32
4.12.2. <i>Apuração Especial – Geração de Relatórios</i>	33
4.12.3. <i>Apuração Especial – Reexecução</i>	33
<b>4.13. ATUALIZAÇÃO DE DADOS DE DOMÍNIO</b>	<b>34</b>
<b>4.14. MANUTENÇÃO EM PÁGINAS ESTÁTICAS DE INTRANET, INTERNET OU PORTAL</b>	<b>34</b>
<b>4.15. MANUTENÇÃO DE DOCUMENTAÇÃO DE SISTEMAS LEGADOS</b>	<b>35</b>
<b>4.16. PROJETO DE TESTE INTEGRADO</b>	<b>35</b>
<b>4.17. PONTOS DE FUNÇÃO DE TESTES FUNCIONAIS</b>	<b>35</b>
<b>4.18. PONTOS DE FUNÇÃO DE TESTES NÃO FUNCIONAIS</b>	<b>37</b>
<b>4.19. PONTOS DE FUNÇÃO DE MASSA DE TESTE PARA HOMOLOGAÇÃO</b>	<b>38</b>
<b>4.20. DESENVOLVIMENTO E MANUTENÇÃO DE HELP</b>	<b>38</b>
<b>5. ATIVIDADES SEM CONTAGEM DE PONTOS DE FUNÇÃO</b>	<b>39</b>
<b>6. CONSIDERAÇÕES PARA CONTAGEM DE PONTOS DE FUNÇÃO</b>	<b>40</b>
<b>6.1 CONSIDERAÇÕES SOBRE MUDANÇA DE REQUISITOS</b>	<b>41</b>
<b>6.2 CONSIDERAÇÕES SOBRE PROJETOS CANCELADOS E SUSPENSOS</b>	<b>45</b>
<b>6.3 FATOR DE CRITICIDADE DE SOLICITAÇÃO DE SERVIÇO</b>	<b>45</b>
<b>7. CONTAGEM DE PONTOS DE FUNÇÃO COM MÚLTIPLAS MÍDIAS E DADOS DE CÓDIGO</b>	<b>46</b>
Cenário 1: Mesmos dados apresentados em tela e impressos	47
Cenário 2: Mesmos dados de saída como dados em arquivo e relatório impresso	48
Cenário 3: Mesmos dados de entrada batch e on-line	48
Cenário 4: Múltiplos canais de entrega da mesma funcionalidade	49
Cenário 5: Relatórios em Múltiplos Formatos	49
<b>7.1. DIMENSIONAMENTO DE DADOS DE CÓDIGO</b>	<b>50</b>
<b>8. GUIA DE CONTAGEM DE PONTOS DE FUNÇÃO DE PROJETOS DE DATAWAREHOUSE</b>	<b>50</b>
<b>8.1. ESCOPO E FRONTEIRA</b>	<b>51</b>
<b>8.2. ENTRADAS EXTERNAS EM PROJETOS DE DATA WAREHOUSE</b>	<b>51</b>
<b>8.3. CONSULTAS EXTERNAS E SAÍDAS EXTERNAS EM PROJETOS DE DATA WAREHOUSE</b>	<b>52</b>
<b>8.4. CONTAGEM DE FUNÇÕES DE DADOS EM PROJETOS DE DATA WAREHOUSE</b>	<b>53</b>
<b>8.5. ALTERAÇÃO DE DADOS DE DIMENSÕES ESTÁTICAS</b>	<b>54</b>
<b>8.6. TABELAS DE VISUALIZAÇÃO – GERAÇÃO DE CUBOS OU CONTEXTO DE ANÁLISE OU UNIVERSO</b>	<b>55</b>
<b>8.7. CONTAGEM DE METADADOS: DESCRIÇÃO DE ATRIBUTOS, MÉTRICAS E PASTAS</b>	<b>55</b>

<b>8.8. REORGANIZAÇÃO DA BANCADA (REPOSICIONAMENTO DE ITENS).....</b>	<b>56</b>
<b>8.9. EVOLUÇÃO DE PÁGINAS ESTÁTICAS EM DATA WAREHOUSE .....</b>	<b>56</b>
<b>8.10. FUNCIONALIDADES DE CONTROLE DO DATA WAREHOUSE .....</b>	<b>56</b>
<b>8.11. MASSA DE DADOS PARA HOMOLOGAÇÃO EM DW .....</b>	<b>56</b>
<b>8.12. DEMANDAS TÍPICAS DE MANUTENÇÃO EVOLUTIVA EM DW.....</b>	<b>57</b>
8.12.1. Criação de Métricas (fórmulas).....	57
8.12.2. Alteração de Campos em tabelas Fato e Dimensão.....	57
8.12.3. Criação, Configuração e Disponibilização de um Filtro .....	58
<b>9. CONCLUSÃO .....</b>	<b>58</b>
<b>REFERÊNCIAS BIBLIOGRÁFICAS .....</b>	<b>59</b>

# 1. Introdução

A métrica de Pontos de Função (PF) tem sido utilizada nas estimativas e dimensionamento de tamanho funcional de projetos de software, devido aos diversos benefícios de utilização da métrica (independência da solução tecnológica utilizada) e às recomendações dos Acórdãos do Tribunal de Contas da União (TCU) e da Instrução Normativa IN04, publicado pela Secretaria Logística e Tecnologia da Informação (SLTI).

O Manual de Práticas de Contagem de Pontos de Função (CPM 4.3) [IFPUG, 2010], publicado pelo *International Function Point Users Group* (IFPUG), define as regras de contagem de Pontos de Função. É importante ressaltar que a métrica de Pontos de Função foi concebida como uma medida de tamanho funcional para projetos de desenvolvimento e de melhoria (manutenção evolutiva) de software. No entanto, os projetos de software não estão limitados a projetos de desenvolvimento e de melhoria. Assim, torna-se essencial a definição de métricas para dimensionar o tamanho de projetos de manutenção de uma maneira objetiva para que estes projetos possam ser avaliados e gerenciados com base em uma métrica.

Além disso, torna-se importante o estabelecimento de métodos para estimar o tamanho funcional dos projetos de software nas fases iniciais do ciclo de vida. Outro ponto a ser destacado é a importância da definição de métodos para geração de estimativas de prazo dos projetos de software.

É importante ressaltar que o Manual de Práticas de Contagem (CPM) é um documento que se destina a mensurar o tamanho funcional de projetos de software, não tendo por objetivo principal suportar contratos de fábrica de software. Desta forma, torna-se necessário criar roteiros complementares para viabilizar o uso de métricas em contratos de software, conforme recomendado pelos Acórdãos do TCU.

## 2. Objetivo

Este documento tem como propósito apresentar um guia de Contagem de Pontos de Função aderente ao Manual de Práticas de Contagem (CPM 4.3), definir os tipos de projetos de manutenção e uma sistemática para dimensionar o tamanho de tais projetos, com base na métrica Pontos de Função. Este guia também apresenta um processo de estimativas com base na métrica Pontos de Função.

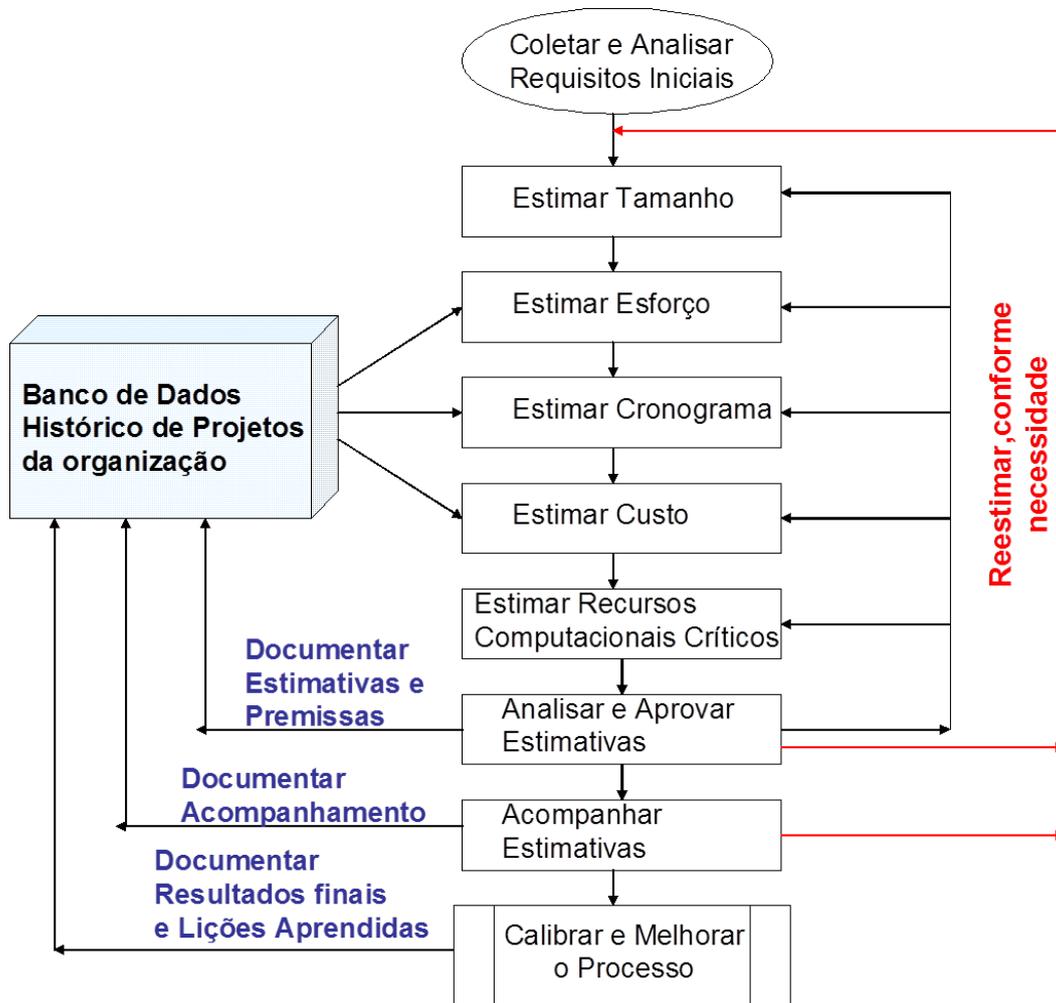
A métrica considerada nesse guia é a de Pontos de Função Não Ajustados, conforme recomendado nos Acórdãos do Tribunal de Contas da União (TCU). O manual de práticas de contagem (CPM 4.3) também define como métrica padrão o Ponto de Função Não Ajustado. **Portanto, não será considerado o Fator de Ajuste nas Contagens e Estimativas de Pontos de Função.**

Este documento encontra-se organizado da seguinte maneira: O Capítulo 1 apresenta a motivação para a elaboração do documento; O Capítulo 2 mostra os objetivos e a organização deste documento; O Capítulo 3 define o processo de estimativas de projetos de software e descreve os tipos de contagem de Pontos de Função realizadas durante o ciclo de vida de um projeto de desenvolvimento ou de manutenção de software, considerando também o processo ágil; O Capítulo 4 apresenta diretrizes para a contagem de Pontos de Função de todos os tipos projetos de software, incluindo projetos de manutenção em requisitos não funcionais não contemplados no CPM; O Capítulo 5 descreve as atividades associadas ao processo de desenvolvimento de software sem contagem de Pontos de Função; O Capítulo 6 apresenta algumas considerações importantes sobre utilização de métricas para dimensionar as mudanças de requisitos, redução de cronograma e orientações específicas para contagem de Pontos de Função de projetos; O Capítulo 7 define um Guia para contagem de Múltiplas Mídias, aderente às diretrizes do IFPUG; O Capítulo 8 apresenta um guia para contagem de Pontos de Função de Projetos de DataWarehouse; Finalmente, o Capítulo 9 conclui o documento apresentando sugestões para trabalhos futuros.

### **3. Estimativas e Tipos de Contagem de Projetos**

Este capítulo tem como propósito descrever um processo de estimativas de projetos de software. Nesse contexto, são apresentados: o método Contagem Estimativa de Pontos de Função (CEPF) para estimar o tamanho dos projetos de software, e a fórmula de Capers Jones para estimar os prazos dos projetos. Além disso, são descritos os tipos de contagem de Pontos de Função que devem ser realizadas durante o ciclo de vida de um projeto de desenvolvimento ou de manutenção de software, considerando também o processo de desenvolvimento ágil.

A **Figura 1** ilustra um processo de Estimativas de Projetos de Software, descrito nos parágrafos seguintes.



**Figura 1: Processo de Estimativas de Projetos de Software [Hazan, 2008]**

O principal insumo para o processo de estimativas é o Documento de Visão. O estimador deve analisar as funcionalidades descritas no Documento de Visão do sistema para estimar o tamanho do projeto de software.

O próximo passo é a derivação das estimativas de esforço e custo com base na estimativa de tamanho e nos dados históricos de projetos concluídos, assim como o estabelecimento da estimativa de recursos computacionais críticos. Neste ponto, as principais estimativas foram geradas e precisam ser documentadas. As premissas e suposições utilizadas na geração das estimativas, como por exemplo, complexidade do projeto, plataforma de desenvolvimento, tipo do projeto e percentual de evolução de requisitos, também devem ser documentadas.

A realização das estimativas por um analista de métricas que não atue na equipe de desenvolvimento do projeto constitui uma prática recomendada. O analista de métricas deve analisar também a consistência da documentação utilizada na estimativa. No decorrer do processo de desenvolvimento, as estimativas devem ser acompanhadas conforme o refinamento dos requisitos.

No caso de projetos não ágeis, o projeto deve ser reestimado após a aprovação do modelo de requisitos do projeto. Os insumos principais para a contagem são as especificações de casos de uso ou especificações de tema (e modelo de dados, caso disponível) do sistema. Esta contagem é denominada Contagem de Referência. Nesse ponto, deve-se estimar o prazo do projeto e reestimar o custo e esforço do projeto.

No caso de projetos ágeis, são realizadas contagens de Pontos de Função das Releases (estimadas e realizadas).

Quando o projeto é concluído, deve-se aferir e documentar o tamanho, prazo, custo e esforço realizados, assim como outros atributos relevantes do projeto, visando a coleta de dados para a melhoria do processo de estimativas. Esta contagem é denominada Contagem de Pontos de Função Final. As lições aprendidas também devem ser documentadas [Hazan, 2008].

No caso de projetos não ágeis, as estimativas e contagens de Pontos de Função devem ser realizadas em três marcos do processo de software, a saber:

- **Estimativa inicial:** realizada após o fechamento do escopo do projeto. É baseada no Documento de Visão ou pré especificação. Constitui uma boa prática a previsão de evolução de requisitos. Sugere-se o uso de um percentual para essa evolução, que não é fixo para todos os projetos e que deve ser analisado com base na completude da especificação inicial (Documento de Visão ou pré especificação) e na complexidade do projeto em questão;

Nessa etapa é importante destacar os seguintes conceitos na área de estimativas: Uma **Estimativa** é obtida por meio de uma atividade técnica, utilizando métodos de estimativas. Não deve sofrer interferências políticas. A **Meta** é um desejo, em função de necessidades de negócio, estabelecida politicamente. Um **Compromisso** é um acordo da gerência com as equipes técnicas para alcançar uma meta [Parthasarathy,2007]. Em um cenário ideal os resultados da estimativa atendem as

metas de negócio. Quando este cenário não é real, é fundamental a redução de escopo do projeto, de modo que a meta se adapte aos resultados da estimativa.

O SERPRO deverá fornecer a estimativa inicial em PF do escopo negociado para a Demanda dentro do prazo máximo estabelecido na tabela a seguir, a contar a partir da abertura da Demanda:

Tipo de Demanda(OS)	Prazo para entrega da Estimativa de Tamanho (em dias úteis)
Corretiva (*)	05
Apuração especial ou manutenção de sistemas consideradas de pequeno porte ou de até 50 Pontos por Função - PF	05
Demais demandas de manutenção, desenvolvimento de software ou migração de dados	20

Tabela 1: Prazos para entrega de Estimativas de PF

- **Contagem de Pontos de Função de Referência (para projetos não ágeis):** realizada após o aceite dos requisitos. Esta Contagem de Pontos de Função é gerada com base na Especificação dos Casos de Uso ou dos Temas e Regras de Negócio da aplicação e modelo de dados, caso estes estejam disponíveis. Deve-se ressaltar que este guia não utiliza percentual para tratar mudanças de requisitos na Contagem de Referência. A documentação dessa contagem deve ser detalhada e rastreável para os requisitos.

As mudanças em requisitos funcionais ou não funcionais que ocorrerem após a contagem de referência devem ser avaliadas em uma análise de impacto, considerando a reestimativa de tamanho, prazo, esforço e custo do projeto. Em caso de mudanças de requisitos deve ser gerada uma contagem de Pontos de Função de retrabalho.

Os marcos de contagem dos projetos ágeis são descritos na seção 3.1.

O detalhamento da contagem consiste no seguinte:

- **Funções de Dados:** descrever os Registros Lógicos; descrever os Tipos de Dados até justificar a complexidade da função de dados. Por exemplo: em um

Arquivo Lógico Interno com complexidade baixa não é obrigatória a descrição dos Tipos de Dados. Para os Arquivos de Interface Externa, a fronteira deve ser documentada conforme o exemplo: AIE: Usuários (Senha Rede); AIE: Município (TOM); AIE: Processo (SIEF Processos).

- **Funções Transacionais:** descrever os Arquivos Referenciados; descrever os Tipos de Dados até justificar a complexidade da função de transação. Documentar a rastreabilidade de requisitos, indicando a qual requisito funcional a função está associada.

- **Contagem de Pontos de Função Final:** realizada após a homologação da aplicação. Esta contagem leva em consideração as funcionalidades efetivamente entregues para o usuário pelo projeto.

### **Contagem de PF para Faturamento**

O faturamento dos projetos deve levar em consideração o tamanho do projeto em Pontos de Função, preço base do Ponto de Função, Fator Criticidade da demanda, Peso dos Blocos contratados e Item Tecnológico.

O Fator criticidade (FC) é de 1,35 e deve ser aplicado para demandas com necessidade de alocação de recursos para atendimento da demanda no prazo estipulado pelo cliente. Será adotado, para demandas que devem ser atendidas em finais de semana, feriados e fora do horário comercial e mediante negociação entre as partes.

O Item Tecnológico (IT) é aplicado de acordo com a tecnologia utilizada na construção da demanda, conforme contrato. Seguem os grupos tecnológicos:

<b>Grupo</b>	<b>Descrição</b>	<b>Implicação do Valor do PF Impacto Tecnológico (IT) no Preço Base do PF</b>
Tecnologia A	JAVA Demoiselle V 2.0	1,00
Tecnologia B	PL/SQL	0,77
Tecnologia C	Mobile	1,46
Tecnologia D	Portal (Zope/Plone, Liferay, Joomla)	1,69
Tecnologia E	Data Warehouse, Workflow e Data	1,73

Grupo	Descrição	Implicação do Valor do PF Impacto Tecnológico (IT) no Preço Base do PF
	Discovery.	
Tecnologia F	Noosfero (Ruby on Rails)	1,97
Tecnologia G	Webservices (Projetos e Componentes) e Middleware (Projetos de Componentes)	2,68
Tecnologia H	Geoprocessamento e Georreferenciamento	4,33

Tabela 2: Grupos Tecnológicos

O Grupo Tecnologia G (Webservices e Middleware) deverá ser utilizado para projetos que envolvam desenvolvimento/manutenção de webservices e/ou Middleware, que fornecem serviços para outro(s) sistema(s) e representam uma fronteira ou aplicação à parte. Quando uma funcionalidade utilizada dentro da fronteira for implementada com solução webservice, deverá ser adotado o Grupo Tecnológico conforme a linguagem utilizada no projeto de desenvolvimento/manutenção.

Nas demandas com utilização de mais de uma Tecnologia, a contagem de Pontos de Função deve ser realizada separadamente por tecnologia em planilhas de contagem de Pontos de Função distintas. Caso haja dificuldade de ordem técnica para identificar a fronteira entre as tecnologias, deve prevalecer, para fins de faturamento, a tecnologia que requer o maior esforço de construção e o fator de impacto tecnológico correspondente.

Os blocos do processo de desenvolvimento e manutenção de projetos de software são os seguintes: Especificação, Construção e Implantação, para projetos não ágeis; e Release e Implantação da Release, para projetos ágeis. Seguem os Pesos dos Blocos (PB):

BLOCO	PESO DO BLOCO (PB)
<b>Projetos Não Ágeis</b>	
Especificação do <i>Software</i>	25%
Construção do <i>Software</i>	70%
Implantação do <i>Software</i>	5%
<b>Projetos Ágeis</b>	
Release	95%
Implantação da Release	5%

Tabela 3: Blocos do Processo de Desenvolvimento de Software

### **3.1. Contagem de Pontos de Função para Projetos Ágeis**

Nesta seção, constam diretrizes para contagem de Pontos de Função para os projetos que utilizam o Processo Ágil. Essas diretrizes visam facilitar a adoção da métrica Pontos de Função para medir o tamanho do software e facilitar o processo de faturamento, sem onerar a gestão do projeto com o controle de mudanças.

No caso de projetos ágeis, deve ser realizada uma estimativa inicial de Pontos de Função com base no Documento de Visão. Após a inception ou o planejamento da release, deve ser realizada uma estimativa inicial de Pontos de Função do escopo tratado na inception ou no planejamento da release. No final da release, após a entrega da release, deve ser realizada uma Contagem de Pontos de Função Final da Release Aceita.

#### **Contagem de PF para Faturamento**

A inception e o planejamento da release (este último, caso ultrapasse 4 horas de duração) serão faturados à parte como serviços de consultoria.

Também deverão ser utilizadas as Tabelas de Grupos Tecnológicos (Tabela 2) e de Blocos do Processo de Desenvolvimento de Software (Tabela 3) nas fórmulas de cálculo para faturamento dos projetos ágeis. A Tabela de Prazos para entrega de Estimativas de PF (Tabela 1) também é válida para a Estimativa Inicial da Demanda com base no Documento de Visão (DV).

O valor de faturamento da etapa Release será calculado somente ao final da Release quando forem planejadas até 3 Sprints. Para as Releases planejadas com mais de 3 Sprints, haverá faturamento parcial ao final de cada conjunto de 3 *Sprints*. O faturamento deverá ser realizado com base nas metas de negócio atingidas (registradas na ata do evento de validação de cada Sprint envolvida). A contagem de Pontos de Função para o Faturamento Parcial deve considerar apenas as funcionalidades concluídas e validadas pelo cliente nas reuniões de validação das Sprints. Não deve ser aplicado o Fator Mudança (FM) nesta contagem de Pontos de Função.

## 3.2. Contagem Estimativa de Pontos de Função (CEPF)

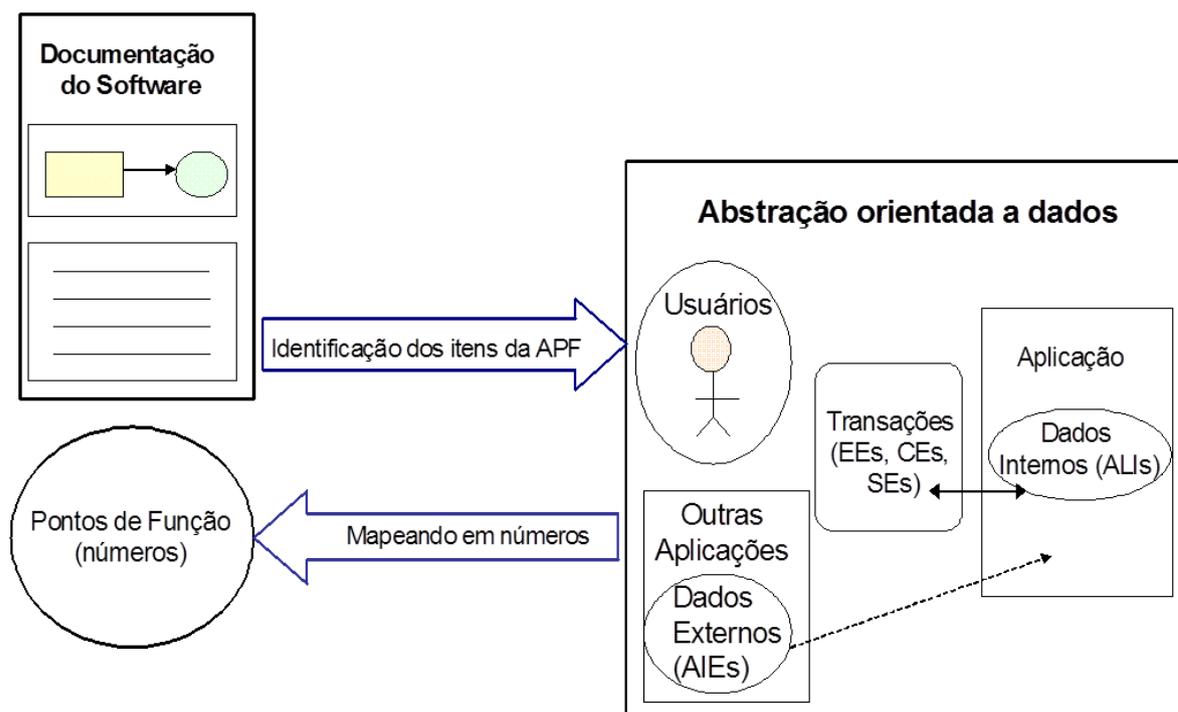
Antes de definir o método de estimativas – Contagem Estimativa de Pontos de Função (CEPF) é importante destacar que “estimar significa utilizar o mínimo de tempo e esforço para se obter um valor aproximado dos Pontos de Função do projeto de software investigado” [Meli, 1999]. Desta forma, é recomendável sempre fazer uma distinção entre os termos e conceitos: Contagem de Pontos de Função e Estimativa de Pontos de Função.

- **Contagem de Pontos de Função:** significa medir o tamanho do software por meio do uso das regras de contagem do IFPUG [IFPUG, 2010];

- **Estimativa de Pontos de Função:** significa fornecer uma avaliação aproximada do tamanho de um software utilizando métodos diferentes da Contagem de Pontos de Função do IFPUG.

O método CEPF visa estimar o tamanho em PF de maneira simplificada, com base nos requisitos funcionais identificados no Documento de Visão ou Pré especificação do Projeto. As funcionalidades são mapeadas para os tipos funcionais da Análise de Pontos de Função: Arquivo Lógico Interno (ALI), Arquivo de Interface Externa (AIE), Entrada Externa (EE), Consulta Externa (CE) e Saída Externa (SE) (Figura 2).

Posteriormente, os Pontos de Função são associados a cada função identificada, baseando-se nas tabelas de complexidade e de contribuição funcional do CPM.



**Figura 2: Modelo Lógico da Análise de Pontos de Função**

O estimador deve realizar uma leitura no documento inicial de requisitos, buscando informações relevantes para a identificação de processos elementares. O processo elementar é definido como a menor unidade de atividade significativa para o usuário. O processo elementar deve ser uma transação completa em si mesma, independente, e deixar a aplicação em um estado consistente [IFPUG, 2010]. Em outras palavras, os processos elementares são funções transacionais independentes.

Uma vez identificado o processo elementar, o estimador deve buscar o entendimento deste para classificá-lo em Entrada Externa, Consulta Externa ou Saída Externa. Adicionalmente, o estimador deve descobrir os dados associados ao processo elementar, visando a determinação da complexidade funcional da função identificada. Caso não seja possível a identificação da complexidade da funcionalidade em questão, recomenda-se a utilização da complexidade Média. Na análise do processo elementar também são identificados os grupos de dados lógicos da aplicação, que são classificados como Arquivos Lógicos Internos ou Arquivos de Interface Externa. Caso não seja possível a identificação da complexidade da função de dados em questão, recomenda-se a utilização da complexidade Baixa. Caso seja identificado mais de um

Registro Lógico em um Arquivo Lógico Interno ou Arquivo de Interface Externa, recomenda-se utilizar a complexidade Média.

A seguir são apresentadas dicas para ajudar no mapeamento dos requisitos funcionais da aplicação nos tipos funcionais da APF. As funcionalidades especificadas para o projeto, contidas no Documento de Visão devem ser enquadradas em uma das seguintes tabelas:

<b>Nº ALIs Baixa:</b>	<b>X 7 PF</b>
<b>Nº ALIs Média:</b>	<b>X 10 PF</b>
<b>Nº ALIs Alta:</b>	<b>X 15 PF</b>
<b>Total PF da Tabela 1:</b>	

**Tabela 4** - Contagem dos Arquivos Lógicos Internos (ALIs): Banco de Dados Lógico da Aplicação (tabelas e arquivos mantidos pela aplicação).

**Considerações:** Identifique os grupos de dados lógicos de aplicação a partir dos requisitos funcionais, descritos no Documento de Visão ou pré especificação. Não contar arquivos físicos, arquivos de índices, arquivos de trabalho e tabelas de relacionamento sem atributos próprios (tabelas que existem para quebrar o relacionamento nxm e apenas transportam as chaves estrangeiras). As entidades fracas também não são contadas como ALI, podem ser consideradas Registro Lógico de um ALI. Se possível, tente descobrir os atributos lógicos, campos reconhecidos pelo usuário, e subgrupos de dados existentes para obter a complexidade funcional, segundo as regras de contagem do CPM. Caso não seja possível, a experiência tem mostrado que a maioria dos ALIs dos sistemas possuem complexidade **Baixa**.

<b>Nº AIEs Baixa:</b>	<b>X 5 PF</b>
<b>Nº AIEs Média:</b>	<b>X 7 PF</b>
<b>Nº AIEs Alta:</b>	<b>X 10 PF</b>
<b>Total PF da Tabela 2:</b>	

**Tabela 5** - Contagem de Arquivos de Interface Externa (AIEs): Banco de Dados de outras Aplicações, **apenas referenciados** pela aplicação que está sendo estimada (tabelas e arquivos mantidos por outra aplicação).

**Considerações:** Identifique os grupos de dados lógicos de outras aplicações referenciados pela aplicação que está sendo estimada. Frequentemente, o referenciamento de dados ocorre para a validação de informações em cadastros ou consultas. Algumas vezes, relatórios ou consultas referenciam dados externos de outras aplicações, também considerados AIEs. Não são considerados arquivos físicos, arquivos de índice, arquivos de trabalho, tabelas de relacionamento sem atributos próprios. As entidades fracas não são contadas como AIE, podem ser consideradas Registro Lógico de um AIE. Um AIE é obrigatoriamente um ALI de uma outra aplicação. Geralmente, os AIEs dos sistemas possuem a classificação de complexidade **Baixa** porque são considerados para a determinação da complexidade funcional do AIE apenas os atributos referenciados pela aplicação que está sendo contada.

Nº EEs Baixa:	X 3 PF
Nº EEs Média:	X 4 PF
Nº EEs Alta:	X 6 PF
<b>Total PF da Tabela 3:</b>	

**Tabela 6** - Contagem de Entradas Externas (EEs): Funcionalidades que mantêm os Arquivos Lógicos Internos (ALIs) ou alteram o comportamento da aplicação.

**Considerações:** Identifique as funcionalidades de manutenção de dados. Conte separadamente a inclusão, alteração e exclusão de dados, isto é, cada função independente de inclusão ou alteração ou exclusão deve ser contada separadamente. A aplicação possui funções de entrada de dados que alteram o comportamento dela, por exemplo: processamentos *batch*, ou processamento de informações de controle? Caso positivo, estas funções também devem ser identificadas como Entradas Externas. Se você não possui conhecimento sobre o processo elementar (funcionalidade analisada), considere a Entrada Externa identificada com complexidade **Média**.

<b>Nº CEs Baixa:</b>	<b>X 3 PF</b>
<b>Nº CEs Média:</b>	<b>X 4 PF</b>
<b>Nº CEs Alta:</b>	<b>X 6 PF</b>
<b>Total PF da Tabela 4:</b>	

**Tabela 7** - Contagem de Consultas Externas (CEs): funcionalidades que apresentam informações para o usuário **sem** a utilização de cálculos ou algoritmos. São os processos elementares do tipo “lê - imprime”, “lê - apresenta dados”, incluindo consultas, relatórios, geração de arquivos pdf, xls, *downloads*, entre outros.

**Considerações:** Você está desenvolvendo uma função para apresentar informações para o usuário: uma consulta, relatório, *browse*, *listbox*, *download*, geração de um arquivo, geração de arquivo psd, xls? Esta função **não** possui cálculos ou algoritmos para derivação dos dados referenciados nem altera um Arquivo Lógico Interno, nem muda o comportamento do sistema? Caso positivo, estas funções devem ser identificadas como Consultas Externas. Se você não possui conhecimento sobre o processo elementar (funcionalidade analisada), considere as Consultas Externas com complexidade **Média**.

<b>Nº SEs Baixa:</b>	<b>X 4 PF</b>
<b>Nº SEs Média:</b>	<b>X 5 PF</b>
<b>Nº SEs Alta:</b>	<b>X 7 PF</b>
<b>Total PF da Tabela 5:</b>	

**Tabela 8** - Contagem de Saídas Externas (SEs): Funcionalidades que apresentam informações para o usuário **com** utilização de cálculos ou algoritmos para derivação de dados ou atualização de Arquivos Lógicos Internos ou mudança de comportamento da aplicação. São as consultas ou relatórios com totalização de dados, relatórios estatísticos, gráficos, geração de arquivos com atualização *log*, *downloads* com cálculo de percentual, entre outros.

**Considerações:** Você está desenvolvendo uma funcionalidade para apresentar informações para o usuário: uma consulta ou relatório com totalização de dados,

etiquetas de código de barras, gráficos, relatórios estatísticos, *download* com percentual calculado, geração de arquivo com atualização de *log*? Caso positivo, estas funções devem ser identificadas como Saídas Externas. Observe que esta função *deve* ter cálculos ou algoritmos para processar os dados referenciados nos arquivos lógicos ou atualizar campos (normalmente indicadores) nos arquivos ou mudar o comportamento da aplicação ou criar dados derivados. Caso não haja conhecimento sobre o processo elementar (funcionalidade) analisado, considere a Saída Externa com complexidade **Média**.

A Estimativa de tamanho do projeto em PFs deve ser gerada totalizando-se os PFs obtidos nas **Tabelas 4, 5, 6, 7 e 8**. A fórmula de contagem ou de estimativa de Pontos de Função para Projetos de Desenvolvimento é a seguinte:

$$\text{PF\_Desenvolvimento} = \text{PF\_Incluído} + \text{PF\_Conversão}$$

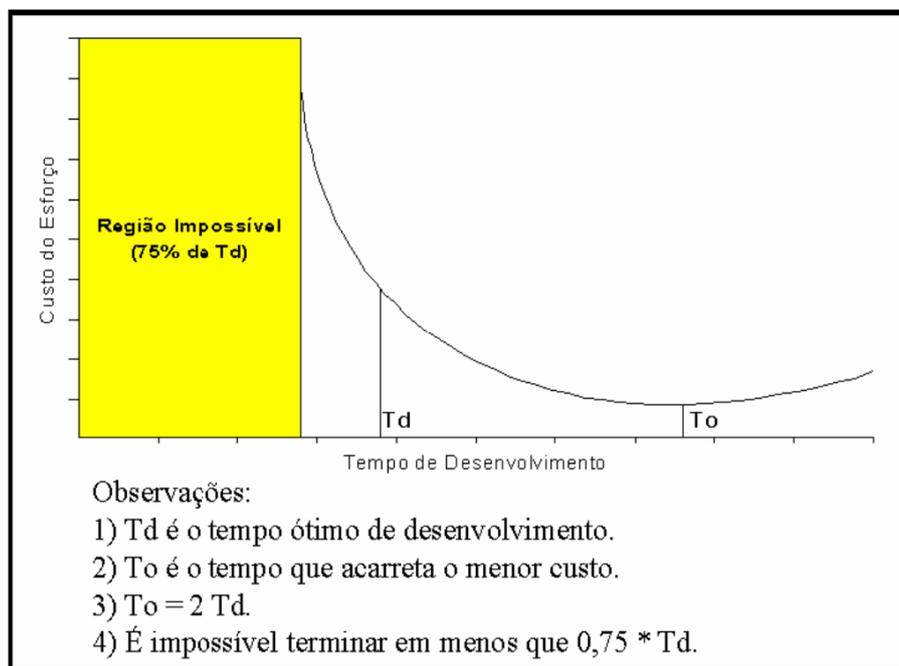
**Observação 1:** PF\_Conversão: Pontos de Função associados às funcionalidades de conversão de dados dos projetos. Exemplos de funções de conversão incluem: migração ou carga inicial de dados para popular as novas tabelas criadas no sistema e relatórios associados à migração de dados.

**Observação 2:** Em projetos de redesenvolvimento de sistemas em outra plataforma, a migração de dados pode ser tratada como um novo projeto de desenvolvimento, especialmente em casos de migração de dados de Bancos hierárquicos para relacionais. Desta forma, não serão contadas as funções de conversão de dados. Observe que serão dois projetos de desenvolvimento, o desenvolvimento do sistema propriamente dito e o desenvolvimento do projeto de migração de dados.

### 3.3. Estimativa de Prazo de Projetos de Software

As estimativas de prazo não são lineares com o tamanho do projeto. O melhor tempo de desenvolvimento, no qual há uma melhor relação custo x benefício de alocação de recursos e menor prazo de desenvolvimento, com base no tamanho de um projeto, tem o uso recomendado por esse guia. Jones [Jones, 2007] propõe uma

fórmula para o cálculo do melhor tempo de desenvolvimento, denominado Td e de Região Impossível (RI) de desenvolvimento, onde a adição de mais recursos ao projeto não implicará em redução no prazo (Figura 3).



**Figura 3: Relação entre a Estimativa de Prazo e de Esforço**

Observe que a curva (Figura 3) mostra que quanto menor o prazo almejado para a conclusão do projeto, maior será o esforço requerido e conseqüentemente maior o custo do projeto. O aumento do esforço para reduzir o prazo acontece através da realização de horas-extras e da inclusão de pessoal adicional. No entanto, a redução de prazo tem um limite, como demonstra a região impossível.

O método utilizado para estimar o prazo dos projetos (Td) é baseado na fórmula de Capers Jones [Jones, 2007]. Posteriormente, pretende-se implantar o modelo CO-COMO II para obtenção de mais de uma estimativa de prazo para o projeto. A fórmula de Capers Jones estima o prazo, baseando-se no tamanho do projeto em Pontos de Função, da seguinte maneira:

$$Td = V^t$$

Onde:

**Td:** prazo de desenvolvimento/manutenção em meses

**V:** tamanho do projeto em Pontos de Função

**t:** o expoente t é definido de acordo com a Tabela 9

Tipo de demanda	Expoente t
Sistema Comum: Portais ou Projetos de Migração de Dados, Sistemas Web, Cliente/Servidor	0,35
Sistema Mobile, Sistemas Cliente/Servidor ou Desktop ou Web, com alta complexidade arquitetural e integrações	0,37
Sistemas Gerenciais, Workflow, Data Warehouse, Georreferenciamento, Geoprocessamento	0,39
Programa/Produto Comercial, Componentes, Webservices, Middleware, Framework	0,40

**Tabela 9: Expoente t por tipo de Projeto**

É importante destacar que o método só funciona para projetos com mais de 100 PFs.

A fórmula apresenta o resultado (Td) em meses. Para efeito de contagem dos dias úteis de demandas a partir de 100 PF, o prazo Td deve ser multiplicado por 21.

Nas Demandas que tiverem em seu escopo mais de um tipo da tabela acima, deve prevalecer, para fins de referência de prazo, o tipo que requerer o maior expoente, ou poderá ser negociado entre as partes um expoente intermediário.

Caso seja necessário um prazo menor que o Td calculado, recomenda-se propor um processo de desenvolvimento incremental, priorizando funcionalidades em cada iteração de acordo com a necessidade.

Caso, ainda assim, a estimativa não atenda às necessidades do cliente, pode ser negociada uma redução de prazo, mediante aplicação do Fator Criticidade de 35%.

Não é recomendada a redução de prazo, devido ao alto risco. Deve-se buscar priorizar funcionalidades trabalhando com o processo incremental.

Caso o projeto seja menor que 100 PF, seguem os prazos de referência em dias úteis na tabela abaixo:

Tamanho da Demanda	Apuração Especial	Manutenções, Desenvolvimento e Migração de Dados
Até 10 PF	10 dias	15 dias
De 11 a 25 PF	17 dias	35 dias
De 26 a 50 PF	34 dias	58 dias
De 51 a 99 PF	66 dias	104 dias
A partir de 100 PF	Conforme técnica de estimativa de Carpers Jones	

**Tabela 10: Prazos para demandas menores que 100 PF**

Os prazos definidos compreendem somente as atividades do **SERPRO**. A data de início da contagem de prazo, o tempo total de elaboração dos produtos e a duração das atividades do **SERPRO** e do **Ministério da Justiça** serão os definidos no planejamento da Demanda, que poderá ser negociado entre as partes em virtude da complexidade de cada Demanda.

A execução da demanda, bem como a contagem do seu prazo, somente se iniciarão após a aprovação da estimativa pelo **Ministério da Justiça**.

Para renegociar os prazos acordados de uma Demanda, **Ministério da Justiça** e **SERPRO** deverão comunicar formalmente à outra parte, antes do seu vencimento, uma proposta de novo prazo com a devida justificativa.

## 4. Contagem de Pontos de Função

Esta seção tem como propósito apresentar as fórmulas de contagem de Pontos de Função de projetos de desenvolvimento, descrever os diversos tipos de projetos de manutenção e mostrar métricas baseadas em Pontos de Função para dimensionar tais projetos, visto que o manual de práticas de contagem - CPM - não contempla projetos de manutenção (*maintenance*), apenas o de Melhoria (*enhancement*).

Quanto à documentação de projetos de manutenção pequenos (menores que 100 PF) [Jones, 2007], deve-se registrar a solicitação do cliente (Solicitação do Serviço ou Solicitação de Mudança), documentar os requisitos da demanda em questão de forma detalhada e atualizar a documentação da aplicação impactada pela demanda, visando apoiar a contagem de Pontos de Função da demanda. É importante também documentar as estimativas e a contagem de Pontos de Função.

## 4.1. Projeto de Desenvolvimento

Um Projeto de Desenvolvimento tem como objetivo construir e entregar a primeira versão de uma aplicação de software. A contagem de Pontos de Função de um projeto de desenvolvimento é definida pelo CPM de acordo com a fórmula abaixo:

$$\text{PF\_Desenvolvimento} = \text{PF\_Incluído} + \text{PF\_Conversão}$$

**Observação 1:** PF\_Conversão: Pontos de Função associados às funcionalidades de conversão de dados dos projetos. As funções de migração e conversão de dados são processos elementares contidos em um projeto de desenvolvimento necessários para a sua implantação, que têm por objetivo: migração de dados oriundos de outros sistemas ou tabelas, com ou sem transformação; carga inicial de dados para popular as novas tabelas ou novos campos em tabelas já existentes; atualização de dados legados para manter consistência com o projeto de melhoria; relatórios de exceção, erros, conversão ou de controle necessários para garantir a integridade dos dados que estão sendo convertidos.

**Observação 2:** Em projetos de redesenvolvimento de sistemas em outra plataforma, com mudança de Banco de Dados hierárquico para relacional, a migração de dados deve ser tratada como um novo projeto de desenvolvimento. Desta forma, não serão contadas as funções de conversão de dados.

## 4.2. Projeto de Migração de Dados

Conforme mencionado na observação 2 da seção anterior, em aderência ao Roteiro de Métricas do SISP, este Guia recomenda a supressão do PF\_CONVERSÃO das fórmulas de contagem de Pontos de Função de Desenvolvimento e de Melhoria, em casos de mudança plataforma com alteração de Base de Dados hierárquico para relacional, e o tratamento das funções de migração de dados como projetos separados de migração de dados.

Os projetos de migração de dados devem ser contados como um novo projeto de desenvolvimento de um sistema, contemplando minimamente: os ALIs mantidos pela migração, as Entradas Externas (considerando as cargas de dados nos ALIs), os AIEs de outras fronteiras usados na validação de dados durante as cargas e, caso seja

solicitado pelo usuário relatórios gerenciais das cargas, estes serão contados como Saídas Externas.

### **4.3. Projeto de Melhoria**

O Projeto de Melhoria, também denominado de projeto de melhoria funcional, ou manutenção evolutiva, está associado às mudanças em requisitos funcionais da aplicação, ou seja, a inclusão de novas funcionalidades, alteração ou exclusão de funcionalidades em aplicações implantadas.

Segundo o padrão IEEE Std 1229 [IEEE 1229], esta manutenção seria um tipo de manutenção adaptativa, definida como: modificação de um produto de software concluído após a entrega para mantê-lo funcionando adequadamente em um ambiente com mudanças. O projeto de melhoria é considerado um tipo de projeto de manutenção adaptativa com mudanças em requisitos funcionais da aplicação, ou seja com funcionalidades incluídas, alteradas ou excluídas na aplicação, segundo o CPM 4.3.

Este documento separa o projeto de melhoria, quando as mudanças são associadas aos requisitos funcionais e a manutenção adaptativa quando as mudanças estão associadas aos requisitos não funcionais da aplicação.

Um projeto de melhoria consiste em demandas de criação de novas funcionalidades (grupos de dados ou processos elementares), demandas de exclusão de funcionalidades (grupos de dados ou processos elementares) e demandas de alteração de funcionalidades (grupos de dados ou processos elementares) em aplicações implantadas em produção.

Uma função de dados (Arquivo Lógico Interno ou Arquivo de Interface Externa) é considerada alterada, quando a alteração contemplar mudanças de tipos de dados, inclusão ou exclusão de tipos de dados, ou mudança de tamanho (número de posições) ou tipo de campo (por exemplo: mudança de numérico ou alfanumérico), sendo que esta ocorre por mudança de regra de negócio do usuário.

Uma função transacional (Entrada Externa, Consulta Externa e Saída Externa) é considerada alterada, quando a alteração contemplar:

- Mudança de tipos de dados em uma função existente;

- Mudança de arquivos referenciados;
- Mudança de lógica de processamento, segundo as ações das lógicas e processamento do CPM 4.3:

A **Lógica de Processamento** é definida como requisitos especificamente solicitados pelo usuário para completar um processo elementar. Esses requisitos devem incluir as seguintes ações:

- Validações são executadas;
- Fórmulas matemáticas e cálculos são executados;
- Valores equivalentes são convertidos;
- Dados são filtrados e selecionados através da utilização de critérios;
- Condições são analisadas para verificar quais são aplicáveis;
- Um ou mais ALLs são atualizados;
- Um ou mais ALLs e AIEs são referenciados;
- Dados ou informações de controle são recuperados;
- Dados derivados são criados através da transformação de dados existentes, para criar dados adicionais;
- O comportamento do sistema é alterado;
- Preparar e apresentar informações para fora da fronteira;
- Receber dados ou informações de controle que entram pela fronteira da aplicação;
- Dados são reordenados;

A contagem ou estimativa de Pontos de Função de projetos de manutenção evolutiva deve seguir a fórmula de *enhancement* do CPM 4.3, considerando Fatores de Impacto para funcionalidades Excluídas:

$$\text{PF\_MELHORIA} = \text{PF INCLUIDO} + (0,75 \times \text{PF ALTERADO}) + (0,40 \times \text{PF EXCLUÍDO}) + \text{PF\_CONVERSÃO}$$

Definições:

PF INCLUÍDO ≡ Pontos de Função associados às novas funcionalidades que farão parte da aplicação.

PF ALTERADO ≡ Pontos de Função associados às funcionalidades existentes na aplicação que serão alteradas no projeto de manutenção.

PF EXCLUÍDO = Pontos de Função associados às funcionalidades existentes na aplicação que serão excluídas no projeto de manutenção.

PF CONVERSÃO = Pontos de Função associados às funcionalidades de conversão de dados dos projetos de melhoria. As funções de migração e conversão de dados são processos elementares contidos em um projeto de desenvolvimento necessários para a sua implantação, que têm por objetivo: migração de dados oriundos de outros sistemas ou tabelas, com ou sem transformação; carga inicial de dados para popular as novas tabelas ou novos campos em tabelas já existentes; atualização de dados legados para manter consistência com o projeto de melhoria; relatórios de exceção, erros, conversão ou de controle necessários para garantir a integridade dos dados que estão sendo convertidos.

**Observação 1:** Em caso de projetos não desenvolvidos pelo SERPRO, sem documentação atualizada, as funções alteradas serão contadas como PF\_INCLUÍDO.

**Observação 2:** Em caso de projetos legados desenvolvidos pelo SERPRO, sem documentação atualizada, as funções alteradas serão contadas como PF\_INCLUÍDO.

#### 4.4. Manutenção Corretiva

O padrão IEEE [IEEE,1998] define um tipo de manutenção corretiva, denominado de Manutenção Emergencial como “manutenção corretiva não programada executada para manter o sistema em estado operacional”. A manutenção corretiva altera o software para correção de defeitos.

Entende-se por defeito o funcionamento inadequado de sistemas, tais como:

- A) Funcionamento do sistema em desacordo com a especificação de requisitos;
- B) Erros de recuperação e/ou consolidação de dados;
- C) Erros em cargas ou extrações;
- D) Erros de cálculo;
- E) Erros no conteúdo de sistemas ou sítios;
- F) Problemas da arquitetura do sistema;
- G) Funcionamento descontínuo do sistema;
- H) Outros defeitos identificados e reconhecidos pelas partes.

Não serão consideradas demandas de Manutenção Corretiva se o defeito reportado se referir a um comportamento do sistema que está de acordo com os requisitos e

regras especificados. Neste caso, a manutenção necessária que implique alteração nos requisitos atuais será considerada Manutenção Evolutiva.

Uma vez não constatado o problema apontado, ou este for decorrente de regras de negócio implementadas, ou de utilização incorreta das funcionalidades, a Demanda será classificada como Análise de Solução (Consultoria).

Em caso de sistemas desenvolvidos pelo SERPRO, a contagem de Pontos de Função destas demandas será zero, ou seja, demandas de manutenção corretiva não terão ônus para o Ministério da Justiça. Em caso de absorção de sistemas pelo SERPRO, os projetos de manutenção corretiva serão classificados como projetos de melhoria.

#### **4.5. Manutenção Preventiva**

Refere-se a manutenções nos sistemas em produção desenvolvidos pelo SERPRO, caracterizadas por alterações no sentido de reduzir a probabilidade de ocorrência de falhas e interrupções de serviços em execução, que tornem o código dos programas mais facilmente corrigíveis, adaptados e melhorados.

A identificação e realização de tais manutenções são sem ônus para o **Ministério da Justiça** e independem de solicitação formal. Porém, sua realização deverá ser previamente aprovada.

Em caso de absorção de sistemas pelo SERPRO ou manutenções preventivas ônus Ministério da Justiça, os projetos de manutenção preventiva serão classificados como projetos de manutenção adaptativa em requisitos não funcionais.

#### **4.6. Redesenvolvimento de Projetos em outra Plataforma**

São considerados nesta categoria projetos que precisam ser migrados para outra plataforma. Os projetos com mudança linguagem de desenvolvimento, por exemplo, um sistema em COBOL que precisa ser re-desenvolvido em JAVA, serão considerados como novos projetos de desenvolvimento.

$$PF = PF\_INCLUÍDO + PF\_CONVERSÃO$$

Em caso de mudança de plataforma - Banco de Dados, demandas de redesenvolvimento de sistemas para executar em um outro Sistema Gerenciador de Banco de Dados, deve-se verificar se mudança é de Banco de Dados Hierárquico para Relacional ou de Relacional para Relacional.

Em casos de **mudança de banco hierárquico para relacional**, deve-se considerar como um novo projeto de desenvolvimento, seguindo a fórmula abaixo:

$$PF\_REDESENVOLVIMENTO\_BD\_HIERÁRQUICO = PF\_INCLUÍDO$$

Em projetos de redesenvolvimento de sistemas em outra plataforma, com mudança de Banco de Dados hierárquico para relacional, a migração de dados deve ser tratada como um novo projeto de desenvolvimento. Desta forma, não serão contadas as funções de conversão de dados. Observe que serão dois projetos de desenvolvimento, o desenvolvimento do sistema propriamente dito e o desenvolvimento do projeto de migração de dados. A contagem de PF de projetos de migração de dados está descrita na Seção 4.2.

Em casos de **mudança de banco relacional para relacional**, então deve ser utilizada a seguinte fórmula:

$$PF\_REDESENVOLVIMENTO\_BD\_RELACIONAL = (PF\_ALTERADO \times 0,50) + PF\_CONVERSÃO$$

Nos Projetos de redesenvolvimento de Banco de Dados Relacional para outro Relacional, recomenda-se tratar o PF\_CONVERSÃO dentro do mesmo projeto.

## 4.7. Atualização de Versão

São consideradas nesta categoria demandas para uma aplicação existente ou parte de uma aplicação existente executar em versões mais atuais de browsers (ex: versão atual do Internet Explorer, Firefox,...), sistemas operacionais, ou de linguagens de programação (ex: versão mais atual do JAVA), ou do Banco de Dados (ex: versão mais atual do Oracle). Estas demandas são classificadas como Manutenção Adaptativa Arquitetural (faturamento em homem-hora).

## 4.8. Manutenção Adaptativa em Requisitos Não Funcionais

Seguindo os conceitos da IEEE, existem vários tipos de Manutenção Adaptativa. Quando há mudança em requisitos funcionais, estes projetos são denominados de projetos de Melhoria, descritos na seção 4.3. Esta seção visa apresentar demandas de manutenções adaptativas associadas às mudanças em requisitos não funcionais da aplicação, ou seja, adaptação de Funcionalidades a Requisitos Não Funcionais.

São consideradas nesta categoria as demandas de manutenção adaptativa associadas à adequação de funcionalidades às mudanças de regras de negócio, legislação ou requisitos de usabilidade **que não se enquadram nas funções alteradas do Projeto de Melhoria**, seguindo as regras de contagem do CPM. Observe que tais solicitações envolvem aspectos não funcionais, sem alteração em requisitos funcionais.

### 4.8.1. Manutenção Adaptativa não Arquitetural (Faturamento em PF)

As manutenções adaptativas em requisitos não funcionais que não envolvem alterações arquiteturais e tecnológicas, em função da evolução do ciclo de vida de tecnologia, serão faturadas em PF. Seguem exemplos abaixo:

- Aumentar a quantidade de linhas por página em um relatório;
- Incluir paginação em relatórios;
- Limitar quantidade de linhas por página em uma Consulta existente;
- Permitir exclusões múltiplas em uma funcionalidade que antes só possibilitava a exclusão de um item;
- Adaptação da funcionalidade para possibilitar a chamada por um Webservice ou para outro tipo de integração com outros sistemas;
- Replicação de funcionalidade: chamar uma consulta existente em outra tela da aplicação;
- Alteração na aplicação para adaptação às alterações realizadas na interface com rotinas de integração com outros softwares, por exemplo, alteração em sub-rotinas chamadas por este software;

- Modificar o servidor a ser acessado em uma funcionalidade de download de arquivo.

Nestes casos, a aferição do tamanho em Pontos de Função da funcionalidade ou das funcionalidades alteradas considera 75% do PF\_Alterado, segundo a fórmula abaixo:

$$PF\_ADAPTATIVA = PF\_ALTERADO \times 0,75$$

#### 4.8.2. Manutenção Adaptativa Arquitetural (Faturamento em Homem-Hora)

As manutenções adaptativas em requisitos não funcionais referentes a modificações na solução de software e que portanto envolvem alterações arquiteturas e tecnológicas, com ou sem impacto em funcionalidades, em função da evolução do ciclo de vida de tecnologia, serão faturadas em homem-hora. Seguem exemplos abaixo:

- Atualização de plataforma;
- Alterações de arquitetura e/ou infraestrutura, ou adaptações em componentes arquiteturas;
- Intervenções técnicas para melhoria de desempenho ou segurança, alterações de política de tratamento de dados, alterações de ferramenta, tais como compiladores, *frameworks* ou banco de dados, alterações de ferramenta OLAP ou ETL;
- Evoluções para prover acessibilidade, ou seja, o uso por portadores de necessidades especiais.
- Evoluções para utilização de mecanismos arquiteturas especiais, como, por exemplo, aplicação de mecanismo XSD (Xml Schema Definition);
- Migração de versão de banco de dados, de servidor web e de aplicação, de sistema operacional.

Recomenda-se que demandas de Manutenção Adaptativa Arquitetural não sejam agrupadas com outros tipos de demandas faturadas em PF.

## 4.9. Desenvolvimento e Manutenção de Componentes Internos Reusáveis (Não Arquiteturais e Faturados em PF)

Em alguns casos é demandado desenvolvimento ou manutenção em componentes internos de uma aplicação, que são reusados por várias outras funcionalidades dessa mesma aplicação. Por exemplo, suponha uma mudança em uma rotina de validação de um CPF ou mudança de rotina de validação NIRE do sistema CNPJ, usada em várias funcionalidades de cadastro. Se considerarmos o método de contagem de projetos de melhoria do CPM, seriam contadas todas as funcionalidades impactadas por esta mudança. Seguem alguns exemplos de Manutenção de Componentes Internos Reusáveis:

- ⑩ Alteração de valores de elementos internos de configuração que afetem o comportamento ou a apresentação do sistema de forma geral, tais como arquivos com mensagens de erro, arquivos de configuração de sistema, páginas de estilo, arquivos de internacionalização;
- ⑩ Mudança em item de um menu de um sistema Web que aparece em todas as telas da aplicação. A contagem pode ser realizada considerando o componente “Apresentar Menu”.

Este tópico também contempla a contagem do desenvolvimento ou manutenção de componentes internos baseados apenas em requisitos não funcionais em projetos de desenvolvimento ou de manutenção, que não estejam relacionados com mudanças arquiteturais ou tecnológicas. Seguem alguns exemplos abaixo:

- ⑩ Desenvolvimento/manutenção de instalador/desinstalador de software;
- ⑩ Disponibilização de Help de Sistema.

O Componente Interno Reusável, que será implementado e testado, será considerado um processo elementar independente e contado e tratado como uma funcionalidade, assim um componente de apresentação de dados deve ser contado como uma Consulta Externa, o Arquivo Referenciado será o grupo lógico de dados apresentado, mesmo que este seja físico. Um componente que apresente dados e tenha um algorit-

mo de cálculo será tratado como Saída Externa. Um componente que a função seja atualizar um arquivo de configuração interna será classificado como Entrada Externa, o Arquivo Referenciado será o arquivo físico de configuração atualizado.

Além disso, as funcionalidades da aplicação que necessitem de teste devem ser acordadas entre Ministério da Justiça e SERPRO e dimensionadas por meio da métrica Pontos de Função de Testes.

Segue a fórmula de cálculo de Manutenção de Componentes Internos Reusáveis:

$$PF\_COMPONENTE = PF\_Função\_Incluída \text{ ou } PF\_Função\_Alterada$$

#### **4.10. Desenvolvimento ou Manutenção de Componentes Arquiteturais (Faturados em Homem-Hora)**

Refere-se ao desenvolvimento ou manutenção de componente arquitetural para atender requisitos não-funcionais, sem funcionalidade associada, considerando inclusive a preparação e execução dos testes não-funcionais associados.

Como exemplo, temos desenvolvimento/manutenção de Componentes webservice/middleware não funcionais que possuem como função a integração entre dois sistemas.

Recomenda-se que demandas de Desenvolvimento/Manutenção de Componentes Arquiteturais não sejam agrupadas com outros tipos de demandas faturadas em PF.

As demandas de desenvolvimento ou manutenção componentes específicos, tais como: middleware, webservices, instaladores de software devem ser tratadas como uma fronteira de aplicação à parte, seguindo as orientações do IFPUG, visto que consistem uma demanda específica. As funcionalidades do componente, funções de dados e transacionais, devem ser dimensionadas em Pontos de Função. Além disso, devem ser contados os testes não funcionais associados ao desenvolvimento ou manutenção do componente. Deverá ser utilizado o Grupo Tecnologia G (Webservices e Middleware) na fórmula de cálculo dos valores destas demandas para faturamento.

## 4.11. Manutenção em Interface de Processos Elementares

A manutenção em Interface, denominada na literatura de manutenção cosmética, é associada às demandas de alterações de interface, por exemplo: fonte de letra, cores de telas, logotipos, mudança de botões na tela, inclusão de botões ou links em telas para chamada de funcionalidades existentes (navegação), mudança de posição de campos ou texto em uma tela ou relatório, manutenção em texto estático em telas ou relatórios.

Também se enquadram neste tipo de manutenção as mudanças de texto em mensagens específicas de uma funcionalidade, tais como: erro, validação, aviso, alerta, confirmação de cadastro ou conclusão de processamento; Mudança em texto estático de e-mail enviado para o usuário em uma funcionalidade de cadastro. A demanda deve ser contada como manutenção em interface na funcionalidade de cadastro; Alteração de título de um relatório; Alteração de labels de uma tela de consulta.

Nestes casos, a aferição do tamanho em Pontos de Função da funcionalidade ou das funcionalidades alteradas considera 20% do PF\_Alterado de uma função transacional de mais baixa complexidade (3 PF), independentemente da complexidade da funcionalidade alterada, de acordo com a seguinte fórmula:

$$\text{PF\_INTERFACE} = 0,6 \times \text{Quantidade de Processos Elementares Alterados}$$

Além da manutenção, a documentação da aplicação deve ser atualizada. Seguem exemplos de atualização de documentação:

- Sistemas onde as mensagens estão documentadas utilizando o artefato EMS (Especificação de Mensagens do Sistema), definido no PSDS, sendo necessário alterar este artefato de documentação.
- Sistemas que possuem protótipos, a alteração cosmética no sistema em produção também deve ser realizada no protótipo.

Essa modalidade de manutenção não contempla a redocumentação da aplicação.

## 4.12. Apuração Especial

São funcionalidades executadas apenas uma vez para: corrigir problemas de dados incorretos na base dados das aplicações ou atualizar dados em bases de dados de aplicações; gerar um relatório específico ou arquivo para o usuário por meio de recuperação de informações nas bases da aplicação

Algumas demandas de Apuração Especial não estão associadas à especificamente uma única fronteira. Estas demandas de Apuração Especial envolvendo várias fronteiras devem ser tratadas como um novo projeto de desenvolvimento. Por exemplo, uma demanda de geração de um relatório de contribuinte com dados calculados envolvendo leitura de dados em vários sistemas do IRPF. Esta demanda deve ser contada da seguinte maneira: os dados lidos devem ser contados como Arquivos de Interface Externa e o relatório com dados calculados deve ser contado como uma Saída Externa.

### 4.12.1. Apuração Especial – Base de Dados

Este tipo de apuração especial é um projeto que inclui a geração de procedimentos para atualização da base de dados. Deve-se destacar que estas funções são executadas apenas uma vez, não fazendo parte da aplicação, visando a correção de dados incorretos na base de dados da aplicação ou atualização em função de modificação da estrutura de dados, como por exemplo a inclusão do indicador de matriz - sim ou não - para um CNPJ. Nestes casos, considera-se a contagem de Pontos de Função das funcionalidades desenvolvidas. Geralmente, estas funcionalidades são classificadas como Entradas Externas. Nesse caso, como artefato de homologação da demanda, deve ser gerado um relatório para validação do usuário.

É importante ressaltar que as funções de dados associadas aos dados atualizados não devem ser contadas, considerando que não há mudanças nas estruturas dos Arquivos Lógicos.

PF_AESP_BD = PF_INCLUÍDO
--------------------------

A realização de consulta prévia para confirmação das atualizações de uma Apuração Especial será contada como rotina independente da Apuração Especial que a motivou. Esta consulta será contada como Apuração Especial – Geração de Relatórios.

#### **4.12.2. Apuração Especial – Geração de Relatórios**

Este tipo de apuração especial é um projeto que inclui a geração de relatórios em uma ou mais mídias para o usuário. Em alguns casos, são solicitadas extrações de dados e envio dos dados para outros sistemas. Caso neste envio de dados sejam requisitadas atualizações no sistema de origem, então estas funções são Saídas Externas, devido à atualização do Arquivo Lógico Interno.

Deve-se destacar que estas funções são executadas apenas uma vez, não fazendo parte da aplicação. Nestes casos, considera-se contagem de Pontos de Função das funcionalidades desenvolvidas. Frequentemente, estas funcionalidades são classificadas como Saídas Externas. Também podem ser classificadas como Consultas Externas, caso não possuam cálculos ou criação de dados derivados.

É importante ressaltar que as funções de dados associadas aos dados atualizados não devem ser contadas, considerando que não há mudanças nas estruturas dos Arquivos Lógicos.

PF_AESP_RELATÓRIO = PF_INCLUÍDO
---------------------------------

#### **4.12.3. Apuração Especial – Reexecução**

Em alguns casos, o Ministério da Justiça pode ter interesse em executar uma apuração especial mais de uma vez. Nestes casos, deve ser solicitado formalmente ao SERPRO o armazenamento do script executado. Desta forma, se for solicitada a reexecução de uma apuração especial, esta deve ser dimensionada com aplicação de um fator considerando 15% na contagem de Pontos de Função da apuração especial em questão, da seguinte maneira:

$$\text{PF\_REEXECUÇÃO\_APURAÇÃO} = \text{PF\_INCLUÍDO} \times 0,15$$

#### 4.13. Atualização de Dados de Domínio

Em alguns casos, as demandas de atualização em base de dados estão associadas a atualizações em um único registro em uma Tabela de Domínio do tipo Code Data. Por exemplo, suponha a atualização de uma Tabela de Situação de Projetos (Andamento, Concluído, Cancelado) para inclusão da Situação “Suspenso”. Então, esta demanda será classificada como Atualização de Dados de Domínio (PF\_Domínio) e contada como 10% da contagem de Pontos de Função de uma Entrada Externa de complexidade baixa, ou seja 0,3 PF. No exemplo acima, EE: Atualizar Situação de Projetos (AR:1 Situação do Projeto; TD: 1 Situação do Projeto) – Baixa – 3 PF x 10% = 0,3 PF. Segue a fórmula de cálculo:

$$\text{PF\_Domínio} = \text{PF\_INCLUÍDO} \times 0,10$$

#### 4.14. Manutenção em Páginas Estáticas de *Intranet*, *Internet* ou *Portal*

Nesta seção, são tratadas manutenções específicas em páginas estáticas de Portais, Intranets, Websites, Menus ou tela principal de aplicações cliente servidor. A demanda consiste na alteração dessas estruturas não contadas como funções transacionais pelo CPM, tais como: alteração de página de estilo, criação de página web estática, atualização de menu estático, atualização de texto ou banner estáticos em páginas web estática existentes. Em caso de alteração de Banner, menu ou outro componente que se repete em várias páginas, considerando-se que não ocorra na mudança no código da página, tratar a demanda como PF\_Componentes. Estas demandas são tratadas como desenvolvimento de consultas. Cada página é contada como uma consulta. Considera-se 20% dos Pontos de Função de uma Consulta Externa de complexidade baixa, segundo a fórmula abaixo:

$$\text{PF\_Páginas} = 0,6 \times \text{Quantidade de Páginas Alteradas ou Incluídas}$$

#### 4.15. Manutenção de Documentação de Sistemas Legados

Nesta seção, são tratadas demandas de documentação ou atualização de documentação de sistemas legados. Observe que o desenvolvedor deve realizar uma Engenharia Reversa da aplicação para gerar a documentação. Para este tipo de projeto, caso a demanda seja apenas a documentação de requisitos, devem ser considerados 25% dos Pontos de Função da aplicação em questão, conforme a fórmula abaixo:

$$PF\_Documentação = PF\_ALTERADO \times 0,25$$

Caso a demanda seja a geração de artefatos de documentação de todas as fases do processo de desenvolvimento, deve-se considerar um percentual mais alto de 30% a 50%, dependendo dos artefatos a serem gerados. As premissas utilizadas devem ser acordadas entre o SERPRO e o cliente e documentadas no documento de estimativas do projeto.

#### 4.16. Projeto de Teste Integrado

Nesta seção, são tratadas as demandas específicas do cliente de realização de Testes Integrados, baseados em processos de negócio. Estas demandas consistem no teste do comportamento de funções de dados e transacionais de forma integrada com outros sistemas. Trata-se de uma demanda de teste em requisitos não funcionais.

A contagem de PF destas demandas consiste no seguinte:

- Calcular o tamanho em Pontos de Função de todas as funções de dados ou transacionais envolvida no teste.

$$PFI = PF\_Funcionalidades\_Apenas\_Testadas \times 0,25$$

#### 4.17. Pontos de Função de Testes Funcionais

Muitas vezes, em projetos de manutenção o conjunto de funções de dados e funções transacionais a serem testadas é maior do que a quantidade de funções a serem implementadas, i.e., além das funcionalidades que são afetadas diretamente

pelo projeto de manutenção, outras precisam ser testadas [NESMA, 2009]. O tamanho das funções a serem testadas deve ser aferido em Pontos de Função de Teste Funcional (PFT). Não considerar as funcionalidades incluídas, alteradas ou excluídas do projeto de manutenção na contagem de Pontos de Função de Teste Funcional.

A contagem de PFT deve considerar o seguinte [NESMA, 2009]:

- Determinar o tamanho em Pontos de Função de cada função de dados ou transacional envolvida no teste.
- Calcular o tamanho em Pontos de Função de todas as funções de dados ou transacionais envolvidas no teste.

A contagem do Ponto de Função de Testes Funcionais (PFT) deve ser feita de acordo com a fórmula abaixo:

$$\text{PFT} = \text{PF\_Funcionalidades\_Apenas\_Testadas} \times 0,15$$

É importante ressaltar que as funções testadas consideradas no PFT devem ser identificadas pelo SERPRO e documentadas. Observe que estas funções farão parte do escopo do projeto de manutenção, sendo consideradas para efeito de estimativa de esforço e faturamento junto ao cliente.

Se no teste for identificada a necessidade de alteração em alguma funcionalidade, esta fará parte de um projeto de melhoria. Para efeito de contagem de PF, a funcionalidade será contada apenas como PF\_Alterado de um projeto de melhoria, sendo desconsiderado o Ponto de Função de Testes (PFT).

**Observação:** Em alguns projetos de desenvolvimento ou de manutenção podem ocorrer demandas de testes em outras fronteiras. Nestes casos, há demandas de teste de funcionalidades de outras aplicações. Desta forma, será contado Ponto de Função de Testes para estas funcionalidades de outras fronteiras.

#### 4.18. Pontos de Função de Testes Não Funcionais

Em alguns projetos podem ocorrer demandas de testes não funcionais. O teste não funcional tem como propósito testar os atributos de um componente ou sistema que não se relacionam à funcionalidade, por exemplo: confiabilidade, eficiência, usabilidade,

manutenibilidade e portabilidade<sup>1</sup>. Os principais tipos de testes não funcionais que podem ser realizados nos sistemas, dentre outros, são os seguintes: Testes de performance – para verificar o tempo de resposta da aplicação; Testes de stress – para verificar a quantidade de usuários simultâneos suportados pela aplicação; Teste de integração – para verificar se os componentes, juntos, executam conforme está descrito nas especificações. Estes testes não são focados em “o quê” os componentes fazem, mas se eles "se comunicam" conforme especificado no desenho do sistema; Teste de segurança – para verificar o atendimento dos requisitos não funcionais de segurança; Teste de portabilidade – para verificar o comportamento da aplicação em vários ambientes (é comum a demanda para testar um sistema em vários ambientes). As atividades de execução de testes não funcionais devem ser realizadas por especialistas em testes e dimensionadas de acordo com a seguinte fórmula:

$$PF\_Testes\_Não\_Funcionais = PF\_Funcionalidades\_Testadas \times 0,25$$

Devem ser consideradas apenas as funções transacionais impactadas pela demanda do teste não funcional.

É importante ressaltar que PF é uma métrica de tamanho funcional. Desta forma, deve-se ter cuidado com essa métrica definida por analogia. Em alguns casos, o teste em requisitos não funcionais pode gerar um esforço muito grande em relação a essa métrica. Então, deve-se buscar a remuneração em homem-hora com justificativa por meio de Parecer Técnico emitido pela Área de Métricas do SERPRO em acordo com o cliente.

**Observação:** Os testes não funcionais referentes a manutenções adaptativas arquiteturais, descritas na Seção 4.8.2, e a desenvolvimento ou manutenção de componentes arquiteturais, descritos na Seção 4.10, serão contemplados nessas respectivas demandas, faturadas em homem-hora.

---

<sup>1</sup> Fonte: Standard glossary of terms used in Software Testing Version 1.1 (September/2005) - International Software Testing Qualification Board; Editor : Erik van Veenendaal (The Netherlands)

#### 4.19. Pontos de Função de Massa de Teste para Homologação

Em alguns projetos podem ocorrer demandas de geração de massa de testes para homologação. Estas atividades consistem em carga de dados no Banco de Dados do Ambiente de Homologação. A realização destas atividades é similar à Conversão de Dados. Desta forma, o dimensionamento deve ser realizado da seguinte maneira: Contar uma Entrada Externa para cada Arquivo Lógico Interno ou Arquivo de Interface Externa carregado, desconsiderando os Registros Lógicos. A EE terá sempre apenas um Arquivo Referenciado (o Arquivo Lógico carregado) e os Tipos de Dados (TDs) serão os campos das tabelas carregados. Segue a fórmula de cálculo:

$$PF\_Massa\_Testes = PF\_Incluído\_Carga\_Dados$$

#### 4.20. Desenvolvimento e Manutenção de Help

Em alguns projetos há demandas específicas de desenvolvimento e/ou manutenção de help de funções (help de campo, help de tela ou relatório e help do sistema). O conteúdo do help deve ser elaborado pelo cliente. O dimensionamento destas demandas deve ser realizado de acordo com a seguinte fórmula:

$$PF\_HELP = PF\_funcionalidades\_impactadas \times 0,05$$

Não considerar contagem de função de dados, nem massa de testes para homologação.

Caso a demanda seja **apenas** de Help de sistema, consistindo na disponibilização de texto enviado pelo usuário em um link de Help, então a demanda será contada como Manutenção de Componentes, sendo considerada uma Consulta Externa de complexidade baixa.

Em casos de projetos de manutenção, considerando-se uma alteração apenas no Help de uma funcionalidade, então este help será contado como publicação de página estática. Observe que se a funcionalidade e o help forem alterados a demanda será contada como projeto de melhoria e Ponto de Função de Help. Se apenas a funcionalidade for alterada, sem impacto no Help, não será contado Ponto de Função de Help. Se o Help for um arquivo, a atualização de dados do Arquivo de Help será tratada como um Projeto de Atualização de Dados de Domínio.

## 5. Atividades Sem Contagem de Pontos de Função

O processo de desenvolvimento de soluções possui várias atividades que devem ser consideradas como um projeto separado, cujo esforço pode ser remunerado em homem-hora ou outro item faturável, conforme Contrato. Algumas dessas atividades são:

- **Especificação de Requisitos de Negócio:** esta atividade é a primeira atividade a ser executada em uma demanda de projeto de desenvolvimento e/ou de manutenção. O objetivo desta atividade é gerar o Documento de Visão da demanda.
- **Análise de Viabilidade:** Serviço de apoio destinado à análise de viabilidade e análise de impacto de soluções e ambientes de TI definidos pelo **Ministério da Justiça**.
- **Apoio à Especificação:** Serviço de apoio destinado à análise de regras de negócio a serem implementadas em soluções de TI, à escrita de história de usuário ou ao apoio ao *Product Owner* na execução das atividades de sua responsabilidade. Estas demandas não possuem contagem de PF associada.
- **Análise de Solução:** consiste em atividades de verificação e esclarecimentos técnicos, bem como de negócio (arquitetura, modelos de dados, integrações, funcionalidades, regras de negócio) em solução de TI contratada pelo Ministério da Justiça e mantida pelo SERPRO.
- **Administração e Modelagem de Dados:** este serviço requer uma equipe de ADs com um número de profissionais definido junto ao Cliente, dedicada para atender as demandas associadas à definição e manutenção do modelo de dados de negócio do cliente. É importante ressaltar que as atividades de banco de dados associadas ao projeto de desenvolvimento ou de manutenção, por exemplo, preparação de ambiente (testes, homologação, implantação), desempenhadas pelos DBAs da equipe de desenvolvimento, já estão consideradas dentro da contagem de PF do projeto de software, não cabendo cobrança adicional.

- **Suporte à Homologação:** Esclarecimento de dúvidas sobre os requisitos do sistema, participação de analistas do SERPRO para suporte ao evento de homologação, podem ser tratados como uma demanda à parte.
- **Treinamento para Implantação:** são demandas de treinamentos presencial ou à distância sobre utilização do sistema a ser implantado para os gestores de solução do cliente e usuários. O esforço deste serviço é considerado separadamente da estimativa de esforço derivada da contagem de PF.
- **Atividades de consultoria aos usuários de soluções de BI:** tange à construção de relatórios, utilização de objetos e as diversas aplicações que compõem o serviço do Processo OLAP durante a utilização das soluções.
- **Análise e/ou absorção de conhecimento e/ou implantação de sistemas desenvolvidos por terceiros:** análise em termos da solução de sistema, infraestrutura, dos códigos-fonte e demais componentes.
- **Elaboração de soluções de design:** Criação de Marcas; Definição de Arquitetura de Informação para sítios e portais; Criação de Interfaces Visuais especializadas para sítios, sistemas e aplicativos; Criação de Manuais de Interface para sistemas; Criação de Bibliotecas de Elementos Visuais para frameworks; Consultorias de Identidade, Usabilidade e, Acessibilidade.

## 6. Considerações para Contagem de Pontos de Função

Este capítulo apresenta considerações especiais sobre o dimensionamento em Pontos de Função de mudanças de requisitos, projetos suspensos, projetos cancelados e fator criticidade.

## 6.1 Considerações sobre Mudança de Requisitos

A Contagem de Pontos de Função para as mudanças de requisitos, denominada Contagem de Pontos de Função de Retrabalho, se aplica **apenas** aos Projetos Não Ágeis. A contagem de Pontos de Função de projetos ágeis não considera a contagem de Pontos de Função de Retrabalho. Os projetos ágeis possuem um Fator Mudança (FM) aplicado na Contagem de Pontos de Função da Release.

Em projetos de desenvolvimento e manutenção de software são comuns as mudanças de requisitos no decorrer do projeto, conforme o usuário e o desenvolvedor adquirem mais conhecimento sobre o sistema [Sommerville, 2007]. O CPM denomina este fenômeno de Scope Creep. Como os requisitos não podem ser congelados, então temos que gerenciá-los de forma efetiva.

Este Guia recomenda o uso de percentual para evolução de requisitos na estimativa inicial baseada no Documento de Visão ou pré especificação. Sugere-se o uso do percentual de 35% para tratar a evolução de requisitos. Este percentual não é fixo para todos os projetos, deve ser analisado com base na completeza da especificação inicial (Documento de Visão) e na complexidade do projeto em questão.

Uma mudança de requisito anterior à implantação do projeto gera retrabalho para a equipe de desenvolvimento, aumentando assim o esforço e o custo do projeto. Neste guia, as demandas de mudança de requisitos serão dimensionadas como PF\_RETRABALHO, contadas separadamente do projeto de desenvolvimento ou de manutenção. O dimensionamento final de um projeto deve levar em consideração a contagem de Pontos de Função final do projeto e as contagens de Pontos de Função de retrabalho associados à demanda em questão.

O método de contagem de mudança de requisitos, ou PF\_Retrabalho, descrito neste guia tem os seguintes pressupostos:

- A demanda de mudança de requisitos gera uma contagem de Pontos de Função (PF\_Retrabalho) à parte da contagem do projeto de desenvolvimento ou manutenção e que deve considerar as funcionalidades antes da mudança;

- A contagem do projeto de desenvolvimento ou manutenção deverá ser atualizada a cada demanda de mudança de requisitos, visando refletir a funcionalidade após a mudança.
- Para fins de planejamento ou de faturamento, o tamanho total do projeto será obtido da seguinte forma:

$$\text{PF\_Projeto} = \text{Contagem\_Encerramento\_Projeto} + \Sigma (\text{PF\_Retrabalho})$$

A contagem de PF\_Retrabalho leva em consideração o bloco do processo de desenvolvimento, no qual a mudança foi solicitada. Ou seja, é necessário aplicar o percentual dos Blocos concluídas sobre a contagem de Pontos de Função das mudanças de requisitos.

Para tratar o dimensionamento das mudanças de requisitos torna-se importante definir a distribuição de esforço, visando definir o valor agregado ao projeto após cada Bloco. A Tabela 11 apresenta o percentual de esforço para projetos de desenvolvimento e manutenção de software por Bloco.

<b>Bloco</b>	<b>Percentual de esforço do Bloco</b>
Especificação do Software	25%
Construção do Software	70%
Implantação do Software	5%

**Tabela 11: Distribuição de Esforço por Bloco**

O cálculo de PF do requisito original deve considerar o tamanho funcional do requisito, o percentual dos Blocos realizados e o fator de impacto do tipo de retrabalho. O Percentual dos Blocos será tratado da seguinte forma:

- Mudanças antes do início da construção do Software serão faturadas mediante o percentual relativo ao bloco de especificação.
- Mudanças após a construção do Software serão faturadas mediante o percentual do bloco de construção.

A contagem de PF\_Retrabalho leva em conta as seguintes características:

- Requisito original: é o requisito do projeto de desenvolvimento ou manutenção original, que pode incluir, alterar ou excluir funcionalidades de uma aplicação.
- Tipo de mudança do requisito: é a natureza da mudança de requisitos no projeto em andamento, que pode acrescentar um requisito, alterar um requisitos definido ou desistir de um requisito (retirar do escopo do projeto).

A tabela abaixo resume os percentuais que devem ser aplicados sobre as funções alteradas (considerando o tamanho antes a mudança) para obtenção de PF\_Retrabalho.

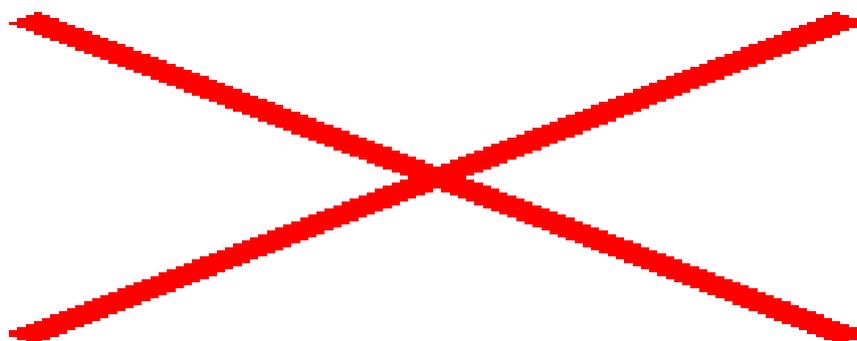


Tabela 12: Percentuais para PF\_Retrabalho

Em caso de mudanças adaptativas em requisitos não funcionais, assim como as mudanças em funcionalidades, deve ser aplicado um Fator Retrabalho de 75% na contagem de PF Retrabalho.

Segue a fórmula de cálculo do PF\_Retrabalho:

$$\text{PF\_Retrabalho} = \text{Contagem\_PF\_Requisito\_Original} \times \text{Fator\_Retrabalho} \times \text{Bloco}$$

Em caso de mudanças cosméticas em requisitos, manutenção de interface, na contagem de PF Retrabalho a funcionalidade deve ser contada com 0,6 PFs (20% do PF de uma funcionalidade com complexidade mais baixa). Além disso, deve ser considerado o percentual do Bloco. Desta forma, a contagem de PF é de 0,6 PF (PF\_Requisito Original x Fator\_Retrabalho) multiplicado pelo percentual do Bloco.

Seguem alguns exemplos:

Por exemplo, suponha uma mudança de requisitos no Sistema de Gerenciamento de Projetos, onde o Caso de Uso 2: Relatório de Atividades, precisará de um novo filtro.

Segue a contagem de PF do requisito em questão:

**[Caso de Uso 2] Relatório de Atividades – SE – Média – 5 PF**

Em uma análise de impacto da mudança foi constatado que este requisito encontra-se no final da Homologação, ou seja no final do Bloco de Construção. Desta forma, o tamanho desta mudança deve ser calculado da seguinte maneira:

Bloco	Esforço
Especificação do Software	25%
Construção do Software	70%

Total: 95%

O tamanho da mudança de requisitos ou PF\_Retrabalho é de 3,56 PF (5 PF x 95% x 75%).

O requisito alterado será considerado na planilha de contagem de PF do projeto em questão, supondo que este será entregue ao cliente sem passar por novas alterações. Desta forma, a contagem de Retrabalho é de 3,56 PF e o PF do requisito entregue ao cliente é de 5 PF. Portanto, o tamanho total desta funcionalidade é de 8,56 PFs.

Supondo uma demanda adaptativa de inclusão de paginação no Relatório de Atividades, mencionado no exemplo anterior. A contagem de PF Retrabalho seria:  $PF\_Retrabalho = 5 \times 95\% \times 75\% = 3,56$  PFs

Supondo uma demanda de alteração do título do Relatório de Atividades, mencionado no exemplo anterior. A contagem de PF Retrabalho seria:  $PF\_Retrabalho = 0,6 \times 95\% = 0,57$  PFs.

Em caso de exclusão de requisitos (desistência de uma funcionalidade), a contagem de PF de retrabalho deve considerar a atividade de construção do requisito em questão e o processo de retirada do requisito da baseline do projeto. A exclusão do requisito deve considerar o percentual de PF\_Excluído 40%. Desta forma, a contagem PF\_Retrabalho de um novo requisito excluído deve aplicar o fator de 140% na contagem de PF da funcionalidade em questão, multiplicado pelo percentual do Bloco.

Desta forma, suponha a demanda de exclusão do Relatório de Atividades. A contagem de PF Retrabalho seria:  $PF\_Retrabalho = 5 \times 140\% \times 95\% = 6,65$  Pfs.

## 6.2 Considerações sobre Projetos Cancelados e Suspensos

Em alguns casos, devido às mudanças no ambiente do cliente, uma demanda ou parte de um projeto de desenvolvimento ou de manutenção pode ser cancelado pelo cliente. Nestes casos, o tamanho funcional das funcionalidades canceladas será aferido por meio da contagem de Pontos de Função das funcionalidades canceladas.

A aferição do tamanho funcional das funcionalidades canceladas deve considerar o percentual de conclusão do processo de software, considerando os Blocos.

A suspensão da demanda implicará negociação entre as partes para o encerramento do projeto com o escopo realizado. Neste caso, haverá faturamento do que tenha sido realizado até o momento. A contagem de PF ocorrerá da mesma forma que os projetos cancelados.

## 6.3. Fator de Criticidade de Solicitação de Serviço

Em função da criticidade e da necessidade de alocação de recursos extras para atendimento da demanda no prazo estipulado pelo cliente, será adotado um Fator de Criticidade de 1,35 (um vírgula trinta e cinco), que deverá ser multiplicado pelo tamanho funcional da demanda considerada crítica, de modo a remunerar adequadamente o aumento do esforço de atendimento. **Este fator é considerado para demandas que devem ser atendidas em finais de semana, feriados e fora do horário comercial. Entende-se como horário comercial o horário de 08:00 às 18:00 h, horário de Brasília.**

Segue a fórmula de cálculo:

$$PF\_CRITICIDADE = PF\_DEMANDA \times 1,35$$

## 7. Contagem de Pontos de Função com Múltiplas Mídias e Dados de Código

Esta seção tem como propósito apresentar as diretrizes de Contagem de Pontos de Função utilizadas no SERPRO em relação ao tema Múltiplas Mídias. Esta abordagem é reconhecida pelo IFPUG. As definições apresentadas têm como base o artigo “*Considerations for Counting with Multiple Media*” Release 1.0 publicado pelo IFPUG [IFPUG, 2009]. Também é abordado neste capítulo algumas diretrizes para mensuração de Dados de Código e Code Data.

Considerando-se a contagem de PF de funcionalidades entregues em mais de uma mídia, a aplicação das regras de contagem de Pontos de Função definidas no CPM tem levado a duas abordagens alternativas, a saber: *single instance* e *multiple instance*.

A abordagem *single instance* considera que a entrega de uma função transacional em múltiplas mídias não deve ser utilizada na identificação da unicidade da função.

A abordagem *multiple instance* leva em consideração que a mídia utilizada na entrega da funcionalidade é uma característica de identificação da unicidade da função. Assim, funcionalidades únicas são reconhecidas no contexto da mídia na qual elas são requisitadas para operar.

É importante enfatizar que o IFPUG reconhece ambas abordagens *single instance* e *multiple instance* para a aplicação das regras definidas no CPM. A determinação de da contagem de PF seguindo a abordagem *multiple instance* ou *single instance* depende da avaliação da Área de Métricas da organização.

A seguir são descritos os termos comuns definidos pelo IFPUG [IFPUG, 2009]:

- **Canal:** também refere-se a mídia. Múltiplos canais é sinônimo de múltiplas mídias.
- **Mídia:** descreve a maneira que os dados ou informações se movimentam para dentro e para fora de uma fronteira de aplicação, por exemplo, apresentação de dados em tela, impressora, arquivo, voz. Este termo é

utilizado para incluir, dentre outros: diferentes plataformas técnicas e formatos de arquivos como diferentes mídias.

- **Múltiplas Mídias:** quando a mesma funcionalidade é entregue em mais de uma mídia. Frequentemente, somente uma mídia é requisitada para um usuário específico em um determinado momento, por exemplo consulta de extrato bancário via internet como oposto a consulta de extrato bancário via terminal do banco.

- **Multimídia:** quando mais de uma mídia é necessária para entregar a função, por exemplo, uma nova notícia publicada na Internet que é apresentada em vídeo e texto. Observe que a notícia completa só é apresentada para o usuário se ele ler o texto e assistir o vídeo.

- **Abordagem *Single Instance*:** esta abordagem não reconhece que a mídia utilizada na entrega da função transacional é uma característica de diferenciação na identificação da unicidade da função transacional. Se duas funções entregam a mesma funcionalidade usando mídias diferentes, elas são consideradas a mesma funcionalidade em uma contagem de Pontos de Função.

- **Abordagem *Multiple Instance*:** esta abordagem especifica que o tamanho funcional é obtido no contexto de objetivo da contagem, permitindo uma função de negócio ser reconhecida no contexto das mídias que são requisitadas para a funcionalidade ser entregue. A abordagem *multiple instance* reconhece que a mídia para entrega constitui uma característica de diferenciação na identificação da unicidade da função transacional.

Os cenários descritos nas seções seguintes não representam uma lista completa de situações de múltiplas mídias. O entendimento destes exemplos facilitará o entendimento de outros cenários envolvendo múltiplas mídias. Este guia deve ser atualizado considerando a publicação de novas diretrizes do IFPUG e novos cenários que emergirão nos projetos.

### **Cenário 1: Mesmos dados apresentados em tela e impressos**

Neste cenário, uma aplicação apresenta uma informação em uma consulta em tela. A mesma informação pode ser impressa caso requisitado pelo usuário na tela em questão.

Nesses casos, utiliza-se a abordagem *single instance*, considerando que dados idênticos sendo apresentados em tela e relatório impresso devem ser contados como uma única função. Caso as lógicas de processamento da consulta em tela e do relatório em papel sejam distintas, o processo elementar não é único e portanto a funcionalidade será contada duas vezes.

### **Cenário 2: Mesmos dados de saída como dados em arquivo e relatório impresso**

Uma aplicação grava dados em um arquivo de saída e imprime um relatório com informações idênticas as gravadas no arquivo.

Nesses casos, utiliza-se a abordagem *single instance* considerando que os dados impressos e os dados apresentados no arquivo de saída sejam idênticos e que a ferramenta de desenvolvimento apoie a geração dessas múltiplas saídas. Assim, apenas uma funcionalidade será incluída na contagem de Pontos de Função. Caso as lógicas de processamento da geração do arquivo de saída e do relatório em papel sejam distintas, o processo elementar não é único e portanto a funcionalidade será contada duas vezes. E ainda, se a geração das múltiplas saídas não seguirem o padrão da ferramenta de desenvolvimento e tiverem que ser customizadas para o cliente, então será utilizada a abordagem *multiple instance*. Caso seja necessária a geração de mais de um tipo de arquivo, cada tipo de arquivo gerado será considerada uma funcionalidade distinta.

Observe que a abordagem *multiple instance* considera que dados idênticos estão sendo entregues em mais de um tipo de mídia e a contagem de PF incluirá todas as instâncias de tipos de mídia. Neste cenário, duas funções são contadas – geração arquivo e apresentação dos dados impressos.

### **Cenário 3: Mesmos dados de entrada batch e on-line**

Uma informação pode ser carregada na aplicação por meio de dois métodos: arquivo batch e entrada on-line. O processamento do arquivo batch executa validações durante o processamento. O processamento on-line também executa validações das informações.

Deve-se utilizar a abordagem *multiple instance* que conta duas funcionalidades: a entrada de dados batch e a entrada de dados on-line. Geralmente, a lógica de

processamento utilizada nas validações em modo batch é diferente da lógica de processamento das validações nas entradas de dados on-line. Portanto, serão contadas duas funcionalidades.

#### **Cenário 4: Múltiplos canais de entrega da mesma funcionalidade**

Uma funcionalidade deve ser disponibilizada em múltiplos canais, por exemplo consulta de dados em página Web e consulta de dados no telefone celular.

Deve-se utilizar a abordagem *multiple instance* que conta duas funcionalidades: a consulta de dados na Web e a consulta de dados via celular. Considera-se que a funcionalidade é desenvolvida duas vezes para os dois canais. Algumas vezes, são até projetos de desenvolvimento distintos, um projeto relativo ao sistema Web e outro para o sistema via celular. Portanto, serão contadas duas funcionalidades.

#### **Cenário 5: Relatórios em Múltiplos Formatos**

Um relatório deve ser entregue em diferentes formatos, por exemplo, em um arquivo *html* e um formato de valores separados por vírgula.

Nestes casos, conforme sugerido na abordagem *multiple instance*, considera-se a ferramenta utilizada na geração dos relatórios. Se a equipe de desenvolvimento precisar desenvolver o relatório nos dois formatos na ferramenta em questão, serão contadas duas funcionalidades. Porque, a lógica de processamento de análise de condições para verificar quais são aplicáveis é identificada. No entanto, se a ferramenta de desenvolvimento suportar um gerador de relatórios que o usuário visualize o relatório em tela e o gerador permita ao usuário imprimir o relatório, salvar em *html* ou salvar no formato de valores separados por vírgula, então a funcionalidade será contada apenas uma vez, observando que a funcionalidade será da ferramenta e não da aplicação.

### **7.1. Dimensionamento de Dados de Código**

As Tabelas com atributos de Código e Descrição devem ser analisadas com muito cuidado. Caso estas tabelas não sejam mantidas pela aplicação, o que geralmente acontece, estas são classificadas como Code Data (Dados de Código) e, portanto, não são contadas. Os Code Data são implementações de requisitos técnicos,

requisitos não funcionais para melhorar a manutenibilidade da aplicação, não sendo contemplados em uma Contagem de Pontos de Função.

Entretanto, as Tabelas com atributos de código e descrição que armazenam dados de negócio, ou seja possuem manutenção de dados por processos elementares da aplicação, serão tratadas como Entidades de Negócio e contadas como Arquivos Lógicos Internos.

Desta forma, as Tabelas com atributos de código e descrição serão tratadas da seguinte maneira:

- **Dados de Código:** Tabelas Código – Descrição ou qualquer outra entidade com dados estáticos, sem funcionalidades de manutenção requisitadas pelo Ministério da Justiça. A manutenção de tais entidades de dados é por meio de apuração especial. Estas Tabelas não são contadas.
- **Dados de Negócio:** Tabelas Código- Descrição ou qualquer outra entidade com dados dinâmicos, com funcionalidades de manutenção requisitadas pelo Ministério da Justiça, por meio de Especificações de Casos de Uso aprovadas. Estas Tabelas são contadas como Arquivo Lógico Interno ou Registro Lógico, observando as regras de contagem de Funções de dados do CPM. As funcionalidades de manutenção e de consulta a estes dados, descritas no Documento de Requisitos aprovado, também serão contadas de acordo com as regras de contagem de Funções transacionais do CPM.

## **8. Guia de Contagem de Pontos de Função de Projetos de DataWarehouse**

Este capítulo apresenta um guia para Contagem de Pontos de Função de Projetos de DataWarehouse.

### **8.1. Escopo e Fronteira**

O Escopo da Contagem define quais são as funcionalidades objeto de determinada contagem. O escopo da contagem é determinado pelo propósito da contagem, e identifica os sistemas, as aplicações ou seus componentes que serão dimensionados. Um escopo de contagem pode conter mais de uma aplicação. No entanto, a contagem

de Pontos de Função é realizada separadamente considerando cada fronteira de aplicação. No contexto de Contagem de PF de Data Warehouse, o escopo da contagem abrange: o Projeto de Desenvolvimento ou Manutenção do Data Warehouse e o Projeto de Melhoria nas aplicações que irão fornecer os dados para o Data Warehouse, ou seja, o DW é uma fronteira única e as aplicações de origem são outras fronteiras.

## **8.2. Entradas Externas em Projetos de Data Warehouse**

Em Projetos de Data Warehouse geralmente existem funcionalidades de cargas de dados nas tabelas. Estas tabelas são denominadas de tabelas fato e tabelas dimensão em um modelo multidimensional em um diagrama estrela. As funcionalidades de carga de dados são classificadas como Entradas Externas

Geralmente, os dados do DW provenientes de outras aplicações, denominadas de aplicações de origem dos dados, são armazenados em uma base de dados temporária, denominada Data Staging Area (DSA). Assim, os dados são importados da aplicação de origem para a DSA e então, em outro processo de integração, importa os dados da DSA para as tabelas Fato e Dimensão do DW. Observe que a utilização da DSA é uma solução técnica, portanto não tem contagem de Pontos de Função. No entanto, é importante ressaltar que em alguns casos, o usuário deseja realizar consultas e emitir relatórios diretamente dos dados da DSA. Nesses casos, as funcionalidades da DSA serão consideradas na contagem de Pontos de Função, ou seja, os dados da DSA serão contados como Arquivos Lógicos Internos, as cargas de dados serão contadas como Entradas Externas e as consultas e relatórios serão contados como Consultas Externas ou Saídas Externas.

As funcionalidades associadas às cargas de dados inicial (full) e incremental (delta) em Tabelas Fato, Dimensão ou Agregação (Estrela de Performance) são contadas como Entradas Externas.

## **8.3. Consultas Externas e Saídas Externas em Projetos de Data Warehouse**

Freqüentemente, em projetos de DW existem funcionalidades que geram arquivos de dados consolidados nas aplicações de origem (aplicações que fornecem os

dados para o DW). Essas funcionalidades de extração de dados da aplicação de origem serão contadas como Saídas Externas ou Consultas Externas na fronteira da aplicação de origem em um Projeto de Melhoria.

Em alguns casos, o DW acessa diretamente o Banco de Dados das aplicações de origem, por meio de alguma ferramenta. Observe que nesses casos não há transferência de dados para o Banco de Dados do DW. Assim, os dados do Sistema de Origem são contados como Arquivos de Interface Externa e as Consultas como Consultas ou Saídas Externas.

Em Aplicações de Data Warehouse existem requisitos para geração de relatórios (relatório ou consulta de bancada) ou gráficos (painéis, dashboards) usando a ferramenta OLAP. Cada relatório ou gráfico requisitado pelo usuário e implementado pela equipe de desenvolvimento será contado como Consulta ou Saída Externa, devendo-se ainda verificar se atendem os critérios de determinação da unicidade do CPM.

Os Relatórios gerados pelo usuário por meio da ferramenta OLAP não são contados, porque não constituem um requisito do usuário para a equipe de desenvolvimento.

A geração do Contexto de Análise também deve ser contada como Saída Externa.

#### **8.4. Contagem de Funções de Dados em Projetos de Data Warehouse**

Em um modelo de dados multidimensional, Esquema Estrela, são reconhecidos dois tipos de entidades: Tabelas Fato e Tabelas Dimensão.

As Tabelas Dimensão mantidas por um ou mais processos de ETL devem ser contadas como um Arquivo Lógico Interno.

Deve-se observar a quantidade de níveis hierárquicos na Dimensão e contar um registro lógico para cada nível hierárquico. Por exemplo, suponha que a dimensão setor sempre está relacionada à dimensão departamento, ou seja não se associa diretamente à fato empregado. Desta forma, a dimensão setor é uma hierarquia da dimensão Departamento e deve ser contada como um Registro Lógico do ALI:

Departamento/Setor. Caso não existam níveis hierárquicos nem subgrupos de dados dentro da dimensão, considere apenas um Registro Lógico para a Dimensão.

Frequentemente, a atualização de dados da Dimensão ocorre por adição de dados, conta-se uma Entrada Externa para atualizar as informações de cada Dimensão.

Deve-se ressaltar que a carga inicial de dados nas tabelas Dimensão também deve ser contada separadamente como uma Entrada Externa. Caso exista uma funcionalidade para exclusão de dados, esta será contada como Entrada Externa. Em geral, conta-se uma Entrada Externa para cada Registro Lógico da Tabela Dimensão. Algumas vezes, as tabelas Dimensão não são mantidas por carga, possuindo dados estáticos. Nesses casos, a Dimensão não deve ser contada como Arquivo Lógico Interno, nem como Registro Lógico. Essas tabelas são classificadas como Dados de Código (Code Data).

As tabelas Fato são contadas como Arquivos Lógicos Internos. Deve ser contada uma Entrada Externa para a carga de dados na Tabela Fato, e, de modo análogo ao da Dimensão, a carga inicial de dados é contada separadamente como uma Entrada Externa.

O DW pode ter como fonte de dados vários sistemas. Assim, os dados de uma Tabela Fato ou de uma Tabela Dimensão podem ser carregados de vários sistemas de origem. Geralmente, o processamento dos dados de cada arquivo proveniente desses sistemas é diferente dos demais. Portanto, conta-se um Arquivo Lógico Interno para a Tabela Fato ou Tabela Dimensão e uma Entrada Externa para cada carga de dados de um sistema de origem distinto, observando o critério de unicidade das Entradas Externas do Manual de Práticas de Contagem (CPM). Desta forma, se as Entradas Externas tiverem os mesmos Arquivos Referenciados, os mesmos Tipos de Dados e a mesma lógica de processamento, então deve-se contar apenas uma vez.

Caso exista leitura de dados de outras aplicações para validação de informações durante as cargas de dados, essas tabelas que são Arquivos Lógicos Internos de outras aplicações e são apenas lidas pelo DW, serão contadas como Arquivos de Interface Externa.

As Dimensões de outros Data Warehousing referenciadas na geração do Contexto de Análise também são consideradas como Arquivos de Interface Externa.

Algumas vezes, o usuário requer a combinação de tabelas Fatos gerando uma outra Tabela Fato ou uma estrutura de agregação ou estrela de performance, visando apoiar a geração de consultas. Em alguns casos, a estrutura de agregação pode ser formada por uma Tabela Fato e Tabelas Dimensão. A estrutura de agregação é contada como Arquivo Lógico Interno e a carga de dados é contada como uma Entrada Externa.

Em Datawarehouse também pode ser encontrada uma estrutura de agregação de Tabelas Dimensão denominada Placa. A Placa deve ser contada como um Arquivo Lógico Interno. A carga de dados é contada como uma Entrada Externa.

Caso exista uma extração de dados específica para as estruturas de agregação, seja ela do tipo estrela de performance ou placa. A extração de dados deve ser contada como Consulta Externa ou Saída Externa.

## 8.5. Alteração de Dados de Dimensões Estáticas

A inclusão ou alteração de dados nas dimensões estáticas, em projetos de manutenção, serão contadas da seguinte forma:

$$PF\_Dimensão\_Estática = 0,3 PF \times Qtd \text{ Dimensões Alteradas}$$

## 8.6. Tabelas de Visualização – Geração de Cubos ou Contexto de Análise ou Universo

Esse tipo de tabela normalmente é utilizada para consumo por outras aplicações ou pelo próprio Datamart (temas). A geração do contexto de análise deve ser contada como uma Saída Externa por Tabela Fato, considerando a estrela, ou seja, a Tabela Fato e as Dimensões. Os Arquivos Referenciados serão as Tabela Fato e cada Tabela Dimensão, identificada como Arquivo Lógico Interno, e os tipos de dados serão os atributos de todos os Arquivos Referenciados (Tabela Fato e Dimensão) e as métricas associadas. As Dimensões de outros Data Warehousing referenciadas na geração do Contexto de Análise, contadas como Arquivos de Interface Externa, também devem ser consideradas como Arquivos Referenciados da SE: Geração de Contexto de Análise.

Nos projetos de melhoria que possuem como requisitos alteração de métricas existentes ou criação de novas métricas em uma Tabela Fato deve ser contada a funcionalidade de Geração de Contexto de Análise como PF\_Alterado.

## **8.7. Contagem de Metadados: Descrição de Atributos, Métricas e Pastas**

As demandas para descrever atributos, métricas e pastas relacionadas a uma tabela fato ou atributos e pastas associados a uma tabela Dimensão serão contadas como PF\_METADADOS, onde PF\_METADADOS é igual à 0,2 x contagem da tabela Fato ou Dimensão. Ainda que uma dimensão tenha mais de um registro lógico, a contagem desse ALI será feita apenas uma vez.

$$\text{PF\_Metadados} = 0,2 \text{ PF} \times \text{Contagem PF do ALI}$$

Caso uma dimensão seja dados de código, conta-se 0,3 pontos de função.

$$\text{PF\_Metadados\_Dimensão\_Estática} = 0,3 \text{ PF por Dimensão}$$

## **8.8. Reorganização da bancada (reposicionamento de itens)**

A demanda reorganização pode estar associada à qualquer métrica, atributo ou filtro. As demandas para reorganização de objetos da bancada serão contadas como 0,6 pontos de função.

$$\text{PF\_Reorganização\_de\_Bancada} = 0,6 \text{ PF} \times \text{Qtd de Itens Reorganizados}$$

## **8.9. Evolução de Páginas Estáticas em Data Warehouse**

A contagem de páginas estáticas será contada conforme a seção manutenção de páginas estáticas em Internet, Intranet e Portal (4.10).

## 8.10. Funcionalidades de Controle do Data Warehouse

Como um dos propósitos do Data Warehouse é o de disponibilizar dados históricos, as funções de limpeza de dados são usualmente incorporadas na área de controle do DW, por exemplo guardar 60 meses de dados históricos. Esta função de limpeza é contada como uma Entrada Externa.

Os dados utilizados para gerenciar o DW podem ser, por exemplo, datas nas quais uma funcionalidade inclui dados em uma tabela fato a partir dos dados de um sistema de origem, a quantidade de registros adicionados, a quantidade de registros rejeitados, ou parâmetros utilizados para o processamento. Os processos elementares da aplicação devem ler e editar esses metadados. Essas funções não são identificadas pelo usuário final, no entanto, esses mecanismos de controle devem ser criados para o DW, sendo considerados pelo perfil administrador. Assim, essas funcionalidades devem ser contadas.

## 8.11. Massa de Dados para Homologação em DW

Em alguns projetos de Datawarehouse podem ocorrer demandas de geração de massa de testes para homologação. Estas atividades consistem em carga de dados no Banco de Dados do Ambiente de Homologação. A contagem de Pontos de Função deve ser realizada da seguinte maneira: Contar uma Entrada Externa para cada Tabela Fato, Dimensão ou Agregação carregada. A EE terá sempre apenas um Arquivo Referenciado (o Arquivo Lógico carregado) e os Tipos de Dados (TDs) serão os campos das tabelas carregados. Segue a fórmula de cálculo:

$$PF\_Massa\_Testes = PF\_Incluído\_Carga\_Dados$$

## 8.12. Demandas Típicas de Manutenção Evolutiva em DW

Esta seção apresenta demandas típicas de projetos de Manutenção Evolutiva para DataWarehouse.

### 8.12.1. Criação de Métricas (fórmulas)

As métricas (fórmulas) são atributos lógicos associados às Tabelas Fatos ou Tabela Dimensão, e são criadas com a geração do contexto de análise da Tabela Fato. Assim, caso o usuário solicite a criação de uma nova métrica, a contagem de PF será a seguinte:

SE: Geração do Contexto de Análise da Tabela – Fato

Arquivos Referenciados: Tabela Fato e suas tabelas dimensões.

Tipos de Dados: todos os campos da Tabela Fato, Dimensão e Métricas.

### **8.12.2. Alteração de Campos em tabelas Fato e Dimensão**

É importante ressaltar que caso seja solicitada alteração em campos ou criação de campos em Tabelas Fato, a contagem será a seguinte:

CE/SE: Extração de Dados do Sistema de Origem

ALI: Tabela Fato

EE: Atualização de Dados da Tabela Fato

EE: Carga de Dados na Tabela Fato

SE: Geração de Contexto de Análise

É importante ressaltar que caso seja solicitada alteração em campos ou criação de campos em Tabelas Dimensão, a contagem será a seguinte:

CE/SE: Extração de Dados do Sistema de Origem

ALI: Tabela Dimensão

EE: Atualização de Dados da Tabela Dimensão

EE: Carga de Dados na Tabela Dimensão

SE: Geração de Contexto de Análise

Devem ser contadas todas as gerações de contexto de análise que referenciem a tabela Dimensão alterada.

As funções acima serão classificadas como PF\_ALTERADO.

### 8.12.3. Criação, Configuração e Disponibilização de um Filtro

Quando for solicitada a criação de um filtro de segurança para um DW, não serão recontados os relatórios de bancada, a demanda completa será contada como 3 Pontos de Função.

PF\_Filtro de Segurança = **3 PONTOS DE FUNÇÃO por demanda**  
(limitada a 10 filtros)

Quando for solicitada a criação de um filtro de relatório para um DW, não serão recontados os relatórios de bancada, a demanda completa será contada como 3 pontos de função.

PF\_Filtro de Relatório = **3 PONTOS DE FUNÇÃO por demanda**

## 9. Conclusão

Este documento apresentou um guia para o dimensionamento de tamanho de todos os tipos de projetos de software, visando a aderência às diretrizes dos Órgãos de Controle. O tamanho funcional é aferido com base na métrica de Pontos de Função Não Ajustados como unidade de medida, conforme recomendado nos Acórdãos do Tribunal de Contas da União (TCU).

Como trabalho futuro recomenda-se a revisão e atualização deste guia sempre que se verificar inconsistência entre alguma definição do IFPUG, publicada em versões futuras do CPM ou em *White Paper*, ou quando for detectado um novo tipo de serviço associado ao desenvolvimento de software não previsto neste trabalho.

## Referências Bibliográficas

- [Hazan, 2008] HAZAN, C. ***Análise de Pontos de Função: Uma Aplicação nas Estimativas de Tamanho de Projetos de Software***. Engenharia de Software Magazine, Edição 2, Devmedia, pp. 25-31.
- [IEEE,1998] IEEE Computer Society. ***IEEE Standard for Software Maintenance***. IEEE Std 1219, 1998.

- [Meli, 1999] MELI, R.; SANTILLO, L. **Function Point Estimation Methods: A Comparative Overview**. Proceedings of FESMA 99, Amsterdam, Netherlands, October 1999, pp. 271-286.
- [IEEE,1998] IEEE Computer Society. **IEEE Standard for Software Maintenance**. IEEE Std 1219, 1998.
- [IFPUG,2009] IFPUG. **Considerations for Counting with Multiple Media**. Release 1.0, September, 2009.
- [IFPUG,2010] IFPUG. **Counting Practices Manual**. Version 4.3, January, 2010.
- [Jones, 2007] JONES, C. **Estimating Software Costs**. Second Edition, Mc Graw Hill, 2007.
- [NESMA, 2009] NESMA. **Function Point Analysis for Software Enhancement Guidelines**. Version 2.2.1, 2009
- [Parthasarathy,2007] PARTHASARATHY, M. A. **Practical Software Estimation: function point methods for insourced and outsourced projects**. Addison Wesley, New York, 2007.
- [Roetzheim, 2005] ROETZHEIM, W. **Estimating and Managing Project Scope for New Development**. CrossTalk, Vol. April, 2005.
- [Sommerville, 2007] SOMMERVILLE, I. **Software Engineering**. Pearson Education Limited, 8th Edition, 2007.
- [Vazquez, 2010] VAZQUEZ, C. E.; SIMÕES, G. S.; ALBERT, R. M. **Análise de Pontos de Função: Medição, Estimativas e Gerenciamento de Projetos de Software**. 9ª Edição. Editora Érica, São Paulo.