

## ANEXO V – GUIA DE CONTAGEM DE PONTO DE FUNÇÃO

### 1. Objetivo

#### 1.1 Finalidade

O objetivo deste documento é apresentar, de forma resumida, um roteiro para a contagem de tamanho funcional de software utilizando a técnica de Análise de Pontos de Função (APF). É utilizado como complemento ao Manual de Práticas e Contagens, versão 4.3.1 (CPM - *Counting Practices Manual Release 4.3.1*) e são definidos os tipos de projetos de manutenção e a sistemática para dimensionar o tamanho de tais projetos, utilizando a métrica Pontos de Função (PF).

A técnica preconizada pelo IFPUG é conhecida como método de CONTAGEM DETALHADA e as técnicas definidas pela NESMA (Netherlands Software Metrics Users Association) são conhecidas como método de CONTAGEM ESTIMATIVA e método de CONTAGEM INDICATIVA. No processo de medição do BNB, observadas divergências conceituais entre as técnicas da NESMA e do IFPUG, prevalecerão as orientações do IFPUG, sendo a contagem executada pelo método de CONTAGEM DETALHADA, adotando-se a complexidade baixa para todas as funcionalidades que não apresentem especificações detalhadas suficientes para análise da contribuição.

#### 1.2 Referências

[Boehm, 2000] BOEHM, B.W. **Software Cost Estimation With COCOMO II**. Prentice Hall, New Jersey, 2000.

[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, McGraw 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.

[PRODEMGE, 2012] PRODEMGE. *Guia de Contagem de Pontos de Função*. Sistema Integrado de Gestão Governamental – ETAPA 01 – Seção IX. 2012.

[Roetzheim, 2005] ROETZHEIM, W. *Estimating and Managing Project Scope for New Development*. CrossTalk, Vol. April, 2005.

[SERPRO, 2008] SERPRO. *Métodos para Estimativa de Projetos de Software Baseado em Pontos de Função*. Relatório do Grupo de Trabalho para Definição da Utilização de Pontos de Função nos Serviços de Desenvolvimento e Manutenção de Sistemas. 2008.

[Sommerville, 2007] SOMMERVILLE, I. *Software Engineering*. Pearson Education Limited, 8th Edition, 2007.

[SISP, 2012] *Roteiro de Métricas de Software do SISP, Versão 2.0*, Ministério do Planejamento, Orçamento e Gestão. 2012.

[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.

## 2. Análise de Pontos de Função (APF)

A métrica PF mede o **tamanho funcional** de um projeto de software, observando as funcionalidades implementadas, considerando a visão do usuário. O tamanho funcional é definido como “tamanho do software derivado pela quantificação dos requisitos funcionais do usuário” [Dekkers, 2003]. A métrica de Pontos de Função é independente da metodologia de desenvolvimento e manutenção e da tecnologia utilizadas.

A Análise de Pontos de Função (APF) é um método padrão para a medição de projetos de desenvolvimento e de manutenção de sistemas, visando estabelecer uma medida de tamanho do software em pontos de função, com base na quantificação das funcionalidades solicitadas e entregues, sob o ponto de vista do usuário. Assim, a APF tem como objetivo medir o que o software faz, por meio de uma avaliação padronizada dos requisitos de negócio do sistema.

Usuário para APF significa:

- Qualquer pessoa que especifique requisitos funcionais; ou
- Qualquer pessoa, sistema ou componente que interaja com o sistema a qualquer momento.

OBS: Em analogia à TUCP, O conceito de ator em um caso de uso é uma boa aproximação do conceito de usuário (que interage com o sistema) para a APF. Um ator não está restrito a ser uma pessoa física, podendo ser outra aplicação ou mesmo algum componente que interaja com o sistema. Se a contagem de pontos de função fosse baseada apenas no conceito de usuário como sendo uma pessoa física, não seria possível medir sistemas que não têm interface com o usuário final. No entanto, a técnica pode ser aplicada também nestas situações.

A Visão do Usuário para APF significa:

- A descrição formal das necessidades de negócio do usuário em sua própria linguagem de negócio.

OBS: A visão do usuário é usada para contar pontos de função, desde que seja uma descrição das suas funções de negócio e também seja aprovada pelo usuário. A visão do usuário é representada por diversos artefatos, alguns deles gerados durante a especificação dos requisitos: documento de visão, especificação de requisitos, casos de uso, modelo de dados (conceitual ou modelo de classes).

É importante observar que oBNB desenvolve também sistemas corporativos, que muitas vezes tem várias categorias de usuários, com visões de negócio distintas. Há o desenvolvimento também de aplicações com arquitetura de serviço (web services) cujos usuários em potencial, muitas vezes, ainda não estão totalmente definidos ou são de várias áreas de negócio. Nestes contextos, para definir a visão do usuário, é importante utilizar a definição do IFPUG para ponto de função:

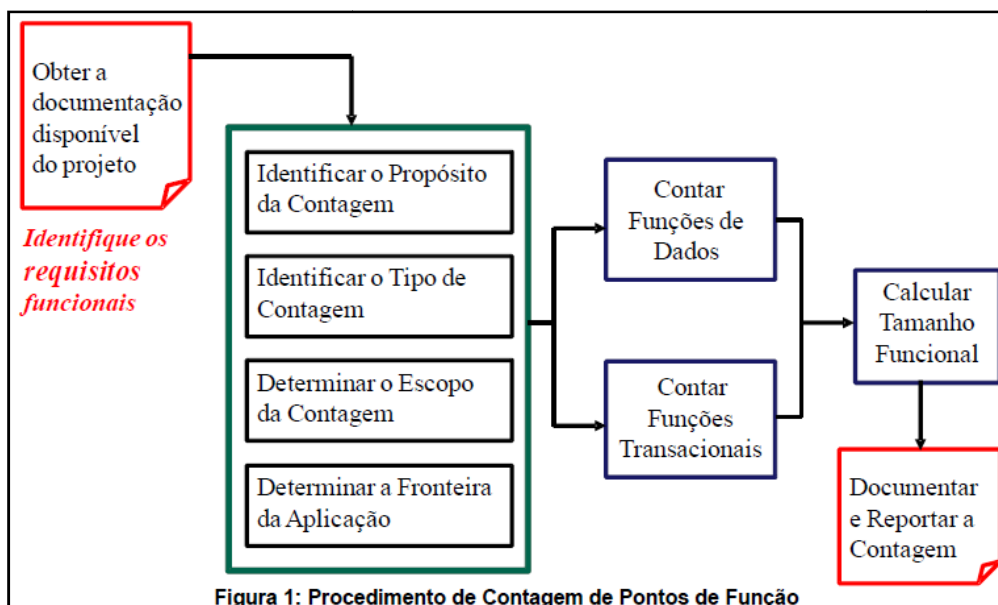
- "Pontos de função medem o tamanho do software através da quantificação da funcionalidade provida ao usuário baseando-se somente em desenhos lógicos e especificações funcionais.

Fonte: <http://www.ifpug.org/about/faqs.htm>.

Desta definição, podemos retirar a conclusão de que a técnica de APF IFPUG quantifica funcionalidades disponibilizadas para o usuário, não sendo necessariamente definidas por ele. Obviamente, as funcionalidades especificadas devem atender às necessidades do usuário e devem, sempre que possível, ser aprovadas por ele. Na prática, essa interpretação é necessária para que a técnica de APF possa ser aplicada em contextos onde o usuário final não participa (ou participa apenas indiretamente) da especificação de requisitos, no entanto, aceita como requisitos válidos.

Esta definição também esclarece que a medição é baseada nas especificações funcionais. Para oBNB, isto significa que a especificação de requisitos é a visão do usuário modelada. Assim, é através da interpretação dos itens modelados que a visão do usuário deve ser determinada.

O Manual de Práticas de Contagem (CPM) [IFPUG, 2010b] apresenta as regras de contagem de pontos de função de projetos de desenvolvimento, projetos de melhoria e aplicações implantadas. A Figura 1 ilustra o procedimento de contagem de pontos de função, descrito nas seções seguintes.



No Banco do Nordeste a **Contagem de Pontos de Função** significa medir o tamanho do software por meio do uso das regras de contagem do IFPUG [IFPUG, 2010].

## 2.1 Contagem e Estimativa em Pontos de Função

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]. Existem dois termos que precisam ser diferenciados: 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.

A NESMA (Associação dos Usuários de Métricas de Software da Holanda) reconhece três tipos de “contagem” de pontos de função:

- Contagem de pontos de função detalhada
- Contagem de pontos de função estimativa
- Contagem de pontos de função indicativa

Os métodos estimativo e indicativo para a contagem de pontos de função foram desenvolvidos pela NESMA para permitir que uma contagem de pontos de função seja feita nos momentos iniciais do ciclo de vida de um sistema. A contagem indicativa da NESMA é também conhecida no mundo como “método holandês”.

O Método detalhado é o que está disciplinado na Contagem de Pontos de Função do IFPUG.

A contagem detalhada de pontos de função é a contagem usual de pontos de função e é realizada da seguinte forma:

- 1) Determinam-se todas as funções de todos os tipos (ALI, AIE, EE, SE, CE);
- 2) Determina-se a complexidade de cada função (Baixa, Média, Alta);
- 3) Calcula-se o total de pontos de função não ajustados.

A contagem estimativa é realizada da seguinte forma:

- 1) Determinam-se todas as funções de todos os tipos (ALI, AIE, EE, SE, CE):
  - a. Toda função do tipo dado (ALI, AIE) tem sua complexidade funcional avaliada como Baixa;
  - b. Toda função transacional (EE, SE, CE) é avaliada como de complexidade média.
- 2) Calcula-se o total de pontos de função não ajustados.

**Ponto de atenção:**

- A única diferença em relação à contagem usual de pontos de função é que a complexidade funcional não é determinada individualmente para cada função, mas pré-definida para todas elas.

A contagem indicativa é realizada da seguinte forma:

- 1) Determina-se a quantidade das funções do tipo dado (ALIs e AIEs):
  - a. Calcula-se o total de pontos de função não ajustados da aplicação da seguinte forma:
    - i. Tamanho indicativo (pf) = 35 x número de ALIs + 15 x número de AIEs

**Ponto de atenção:**

- A Indicativa é baseada somente na quantidade de arquivos lógicos existentes (ALIs e AIEs). A contagem indicativa é baseada na premissa de que existem aproximadamente três EEs (para adicionar, alterar, e excluir dados do ALI), duas SEs, e uma CE na média para cada ALI, e aproximadamente uma SE e uma CE para cada AIE.

## 2.2 O processo de medição

### 2.2.1 Propósito da Contagem

Uma medição de tamanho funcional é feita para fornecer uma resposta a um problema do negócio, e é o problema do negócio que determina o propósito.

O propósito:

- Determina o tipo de contagem de ponto de função e o escopo da contagem necessária para obter a resposta ao problema de negócios sob investigação
- Influencia o posicionamento da fronteira entre o software sob análise e o software vizinho; por exemplo, se o Módulo de Pessoal do Sistema de Recursos Humanos está para ser substituído por um pacote, o usuário pode decidir reposicionar a fronteira e considerar o Módulo de Pessoal como uma aplicação separada.

Exemplos de propósito são:

- Fornecer o tamanho funcional de um projeto como uma entrada para o processo de estimativa a fim de determinar o esforço para desenvolver a primeira versão de uma aplicação
- Fornecer o tamanho funcional da base instalada das aplicações para determinar os custos de sustentação por ponto de função
- Fornecer o tamanho funcional de dois pacotes para permitir a comparação de funcionalidades oferecida por cada um.

### 2.2.2 Definições de Tipo de Contagem de Pontos de Função

O tamanho funcional pode ser medido tanto para projetos quanto para aplicações.

O tipo de contagem de ponto de função é determinado com base no propósito, conforme os itens a seguir:

- Contagem de pontos de função de projeto de desenvolvimento

- Contagem de pontos de função de projeto de melhoria
- Contagem de pontos de função de aplicação

#### **a. Projeto de Desenvolvimento**

Um projeto de desenvolvimento é um projeto para desenvolver e fornecer a primeira versão de um software.

#### **b. Projeto de Melhoria**

Um projeto de melhoria é um projeto para desenvolver e entregar manutenção em requisitos funcionais de uma aplicação. O tamanho funcional do projeto de melhoria é uma medida das funcionalidades **adicionadas, alteradas e excluídas** na conclusão de um projeto de melhoria, conforme medido pela contagem dos pontos de função do projeto de melhoria pela atividade de aplicação do método de Medição de Tamanho Funcional (FSM) do IFPUG.

#### **c. Aplicação**

Uma aplicação é uma coleção coesa de procedimentos automatizados e dados apoiando um objetivo de negócio; isto consiste em um ou mais componentes, módulos, ou subsistemas.

**Um tamanho funcional de uma aplicação é uma medida de funcionalidade que uma aplicação oferece ao usuário**, determinado pela contagem de pontos de função da aplicação pela atividade de aplicação do método de Medição de Tamanho Funcional (FSM) do IFPUG.

Ela também é chamada de baseline ou tamanho funcional instalado. Este tamanho fornece uma medida de funções atuais que o aplicativo fornece ao usuário. O número é inicializado quando o projeto de desenvolvimento da contagem de ponto de função é finalizado. É atualizado toda vez que um projeto de melhoria finalizado alterar funções da aplicação.

### **2.2.3 Escopo da Contagem**

O escopo da contagem define o conjunto de Requisitos Funcionais de Usuários para ser incluído na contagem de pontos de função.

O escopo:

- Define o (sub)conjunto do software que está sendo medido;
- É determinado pelo propósito para a realização da contagem de pontos de função;
- Identifica quais funções serão incluídas na medida de tamanho funcional assim como fornecer respostas relevantes para o propósito da contagem.
- Pode incluir mais de uma aplicação

### **2.2.4 Fronteira da Aplicação**

A fronteira da Aplicação é uma interface conceitual entre o software sob estudo e seus usuários.

A fronteira da aplicação:

- Define o que é externo à aplicação;
- Indica a fronteira entre o software que está sendo medido e o usuário;
- Atua como uma 'membrana' através da qual os dados processados pelas transações (EEs, SEs e CEs) passam para dentro e para fora da aplicação;
- Envolve os dados lógicos mantidos pela aplicação (ALLs);
- Auxilia na identificação dos dados lógicos referenciados, mas não mantidos pela aplicação (AIEs);
- Depende da visão externa do negócio do usuário da aplicação. É independente de considerações de técnicas e/ou implementação.

Há algumas dicas que podem ajudar na identificação da fronteira:

- Obter uma documentação do fluxo de dados no sistema e traçar uma linha em volta para destacar quais partes são internas e externas à aplicação;
- Verificar como os grupos de dados são mantidos;
- Identificar áreas funcionais pela atribuição de propriedade de certos objetos de análise, como entidades e processos;
- Identificar os critérios utilizados em outras perspectivas como esforço, custo, duração, defeitos. As fronteiras para a APF e estas outras perspectivas devem ser as mesmas;
- Verificar como a aplicação é gerenciada; se é totalidade por uma equipe única ou equipes distintas;
- Verificar se o software possui ordens de serviços específicas e independentes;
- Se há usuários distintos especificando requisitos para cada parte do software.
- Aplicação com número excessivo de AIEs de outro sistema. Isto indica alto grau de acoplamento com o outro sistema, indicando que provavelmente a "aplicação" é apenas um módulo do outro sistema.
- Manutenções conjuntas: quando um sistema sempre sofre manutenção junto com outro, é também uma indicação de que ele não tem "força" para ser uma aplicação independente.

O conceito de fronteira é tão importante que o BNB estabeleceu um inventário das aplicações da organização. A ideia é identificar e listar todas elas, inclusive com o inventário das funções de cada aplicação, seus ALLs e AIEs. Todas as contagens de pontos de função realizadas devem usar como base as fronteiras estabelecidas previamente. A fronteira é um conceito que dificilmente muda entre as contagens de pontos de função. Ou seja, as diversas medições dos projetos de melhoria na aplicação, terão como base a mesma fronteira.

Para as contagens, é necessário o conhecimento das fronteiras envolvidas e seu tamanho final após a execução do serviço, bem como integrações adicionadas ou removidas.

O BNB entende que as próprias siglas de seus sistemas já delimitam as fronteiras das aplicações existentes, sendo estas descritas na listagem dos sistemas providas em contrato.

Exceções a esta orientação podem acontecer e serão devidamente justificadas e definidas pelo BNB uma vez que há sistemas que por características de desenvolvimento são compostos de módulos que por si só já são outros sistemas.

## 2.3 Cálculo dos Pontos de Função

### 2.3.1 Identificar Funções de Dados e Funções Transacionais

Uma vez identificada a fronteira da aplicação e estabelecido o tipo e o escopo da contagem, o próximo passo é o mapeamento dos requisitos de armazenamento de dados e de funções transacionais para os tipos funcionais da APF. Os detalhes sobre a descrição dos objetos da APF devem ser consultados no CPM 4.3.1. Segue abaixo uma descrição resumida e explicativa de tais objetos considerados pela APF na contagem:

- **Arquivo Lógico Interno (ALI):** é um grupo de dados, logicamente relacionados, reconhecido pelo usuário, mantido por meio de um processo elementar da aplicação que está sendo contada.
- **Arquivo de Interface Externa (AIE):** é um grupo de dados, logicamente relacionados, reconhecido pelo usuário, mantido por meio de um processo elementar de uma outra aplicação e referenciado pela aplicação que está sendo contada. O AIE é obrigatoriamente um ALI de outra aplicação.
- **Entrada Externa (EE):** é um processo elementar que processa dados ou informação de controle que entram pela fronteira da aplicação. Seu objetivo principal é manter um ou mais ALI ou alterar o comportamento do sistema.
- **Consulta Externa (CE):** é um processo elementar que envia dados ou informação de controle para fora da fronteira da aplicação. Seu objetivo principal é apresentar informação para o usuário através da simples recuperação de dados ou informação de controle de ALI ou AIE.
- **Saída Externa (SE):** é um processo elementar que envia dados ou informação de controle para fora da fronteira da aplicação. Seu objetivo principal é apresentar informação para um usuário ou outra aplicação através de um processamento lógico adicional à recuperação de dados ou informação de controle. O processamento lógico deve conter cálculo, ou criar dados derivados, ou manter ALI ou alterar o comportamento do sistema.

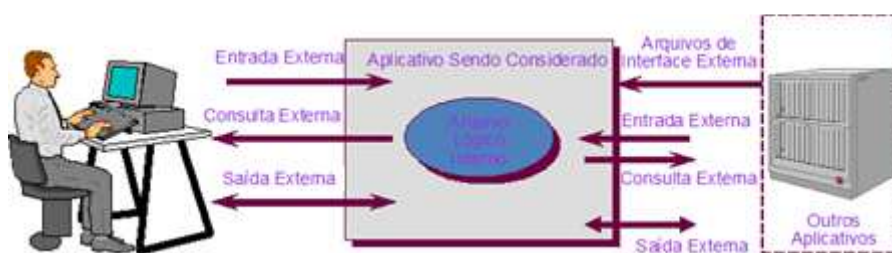


Figura 1: Funcionalidade pela perspectiva do usuário.

Primeiramente, os requisitos funcionais iniciais documentados nos documentos de visão, *outlines* de casos de uso ou em qualquer especificação inicial do sistema do usuário são mapeados nos 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). 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)



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 completo em si mesmo, independente e deixar a aplicação em um estado consistente [IFPUG, 2010]. Em outras palavras, os processos elementares são funções transacionais independentes, isto é, funções sequenciais pertencem a um mesmo processo elementar e funções independentes constituem processos elementares diferentes.

Para a correta identificação de diversos Processos Elementares de uma funcionalidade, é importante avaliá-las a partir da perspectiva do negócio, onde se deve verificar quais funcionalidades são completas e reconhecidas pelos usuários do negócio. Desta forma não se confunde passos da funcionalidade com os Processos Elementares.

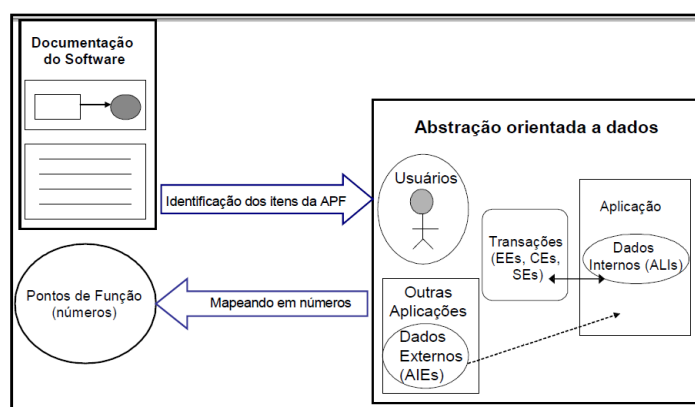


Figura 2: Modelo Lógico da Análise de Pontos de Função

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, 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 Simples. É importante ressaltar que se o estimador identificar mais de um Registro Lógico no Arquivo Lógico Interno recomenda-se utilizar a complexidade Média.

As tabelas 1, 2 e 3 relacionam os itens para identificação da complexidade funcional do ALIs, AIEs, EEs, CEs e SEs.

ALI / AIE	Tipos de Dados		
	< 20	20 - 50	> 50
1	Baixa	Baixa	<b>Média</b>
2 - 5	Baixa	<b>Média</b>	Alta
>5	<b>Média</b>	Alta	Alta

Tabela 1: Itens pra identificação da complexidade funcional para ALI e AIE

EE	Tipos de Dados		
	< 5	5 - 15	> 15
< 2	Baixa	Baixa	<b>Média</b>

2	Baixa	<b>Média</b>	Alta
>2	<b>Média</b>	Alta	Alta

**Tabela 2: Itens pra identificação da complexidade funcional para EE – Entrada Externa**

SE / CE	Tipos de Dados		
	< 6	6 - 19	> 19
< 2	Baixa	Baixa	<b>Média</b>
2 – 3	Baixa	<b>Média</b>	Alta
>3	<b>Média</b>	Alta	Alta

**Tabela 3: Itens pra identificação da complexidade funcional para SE e CE – Saída e Consultas Externas**

**Ponto de atenção:**

- Enquanto na TUCP a complexidade é influenciada pelo número de transações que atendem as regras de negócio, a complexidade de um Processo Elementar é influenciada pela quantidade de relacionamentos com os ALIs, AIEs, Tipos de Registros e Tipos de Dados.

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

**Tabela 4** – Resumo de Contagem dos Arquivos Lógicos Internos (ALIs): Banco de Dados Lógico da Aplicação (tabelas e arquivos mantidos pela aplicação). Para maior detalhamento da contagem de ALIs, observar as regras contidas no CPM 4.3.1.

**Considerações:** Identifique os grupos de dados lógicos de aplicação nos modelos de dados ou diagrama de classes ou a partir dos requisitos funcionais, descritos nos documentos de requisitos (Documento de Visão, Relação de Casos de Uso, etc.). Não considere arquivos físicos, arquivos de índices, arquivos de trabalho e tabelas de relacionamento sem atributos próprios (tabelas que existem para quebrar o relacionamento nxn e apenas transportam as chaves estrangeiras). As entidades fracas também não são consideradas 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 são de complexidade **Baixa**.

Nº ALIs de Complexidade Baixa	x 7PF
Nº ALIs de Complexidade Média	x 10PF
Nº ALIs de Complexidade Alta	x 15PF

**Tabela 4: Identificação dos Arquivos Lógicos Internos da Aplicação**

**Tabela 5**– Resumo de 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). Para maior detalhamento da contagem de AIEs, observar as regras contidas no CPM 4.3.1.

**Considerações:** Identifique os grupos de dados lógicos de outras aplicações referenciados pela aplicação que está sendo estimada. Frequentemente, a referência 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 e entidades fracas.

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 AIEs de Complexidade Baixa	x 5PF
Nº AIEs de Complexidade Média	x 7PF
Nº AIEs de Complexidade Alta	x 10PF

**Tabela 5: Identificação dos Arquivos de Interface Externa da Aplicação**

**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.

Caso não haja conhecimento sobre o processo elementar (funcionalidade analisada), considere as Entradas Externas identificadas com complexidade **Média**.

Nº EEs de Complexidade Baixa	x 3PF
Nº EEs de Complexidade Média	x 4PF
Nº EEs de Complexidade Alta	x 6PF

**Tabela 6: Identificação das Entradas Externas da Aplicação**

**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:** Está sendo desenvolvida 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.

Caso não haja conhecimento sobre o processo elementar (funcionalidade analisada), considere as Consultas Externas com complexidade **Média**.

Nº CEs de Complexidade Baixa	x 3PF
Nº CEs de Complexidade Média	x 4PF
Nº CEs de Complexidade Alta	x 6PF

**Tabela 7: Identificação das Consultas Externas da Aplicação**

**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:** Está sendo desenvolvida 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. Caso negativo, ou seja, se estas funções não forem identificadas como Saídas Externas verifique o Manual de Práticas de Contagem (CPM) [IFPUG, 2010b].

Nº SEs de Complexidade Baixa	x 4PF
Nº SEs de Complexidade Média	x 5PF
Nº SEs de Complexidade Alta	x 7PF

**Tabela 8: Identificação das Saídas Externas da Aplicação**

A Estimativa de tamanho do projeto em PFs deve ser gerada totalizando-se os PFs obtidos nas **Tabelas 4, 5, 6, 7 e 8**.

### 2.3.2 Tipos de Entidades de Dados

Para a Análise de Pontos de Função, os requisitos de armazenamento, funcionais e não funcionais são classificados em três tipos de dados:

- Dados do Negócio
- Dados de Referência
- Dados de Código

Os **Dados de Negócio** podem também ser referenciados como Dados Centrais do Usuário (“Core User Data”) ou Objetos de Negócio. Estes tipos de dados refletem a informação necessária para ser guardada e recuperada pela área funcional tratada pela aplicação. Dados de Negócio normalmente representam um percentual significativo das entidades identificadas. São tipicamente armazenados nos ALIs e AIEs.

Os **Dados de Referência** são armazenados para auxiliar as regras de negócio, para a manutenção dos Dados de Negócio. Normalmente os Dados de Referência representam um pequeno percentual das entidades identificadas.

Os **Dados de Código** fornecem uma lista de valores válidos que um atributo descritivo pode ter. Tipicamente os atributos dos Dados de Códigos são Códigos, Descrição e/ou outros atributos padrão descrevendo o código. Sem uma descrição, o código nem sempre pode ser entendido claramente.

Os **Dados de Código** são dados que surgem em resposta a requisitos técnicos como: normalização de dados, garantia da integridade de dados ou melhoria na entrada de dados. Em geral são dados essencialmente estáticos que possuem poucos atributos, tipicamente código e descrição. Estes dados não contribuem para o tamanho funcional do software, nem as transações que os manipulam. (FATTO)

Os **Dados de Negócio** e os **Dados de Referências** são duas categorias de entidades que são normalmente identificadas para satisfazer os Requisitos Funcionais do Usuário, e como tais, estas entidades serão analisadas para contagem como arquivos lógicos. Os **Dados de Código** são categorias de dados que existem normalmente para satisfazer requisitos técnicos mais propriamente do que requisitos funcionais.

As transações que existem exclusivamente para a manutenção de **Dados de Código** não devem ser consideradas processos elementares, assim como os **Dados de Código** não devem ser contados como arquivos referenciados nos processos elementares que os leiam e/ou atualizem.

<b>Características</b>	<b>Dados de Negócio</b>	<b>Dados de Referência</b>	<b>Dados de Código</b>
Obrigatório para a operação da área funcional do usuário.	<b>X</b>	<b>X</b>	
Dados são obrigatórios para a área funcional, mas armazenado opcionalmente como um arquivo de dados.			<b>X</b>
Identificável por pelo menos um usuário do negócio.	<b>X</b>	<b>X</b>	
Não identificado como parte dos requisitos funcionais; Geralmente identificado como requisitos técnicos;			<b>X</b>
Mantido por um usuário:	<b>de Negócio</b>	<b>Administrador</b>	<b>de Suporte</b>
Normalmente criado quando a aplicação é instalada pela primeira vez e mantido de forma intermitente.		<b>X</b>	
Armazenam Dados Centrais do Usuário para auxiliar as transações do negócio	<b>X</b>		
Armazenam dados para padronizar e facilitar atividades do negócio e transações do negócio;			<b>X</b>
Armazena os dados para auxiliar nas atividades centrais do usuário		<b>X</b>	
Muito dinâmico – operações normais do negócio fazem com que eles sejam regularmente referenciados, incluídos, alterados e excluídos rotineiramente.	<b>X</b>		
Pouco dinâmico – ocasionalmente alterado em resposta a mudanças no ambiente das áreas funcionais, processos funcionais externos e/ou regras de negócio.		<b>X</b>	
Essencialmente estático – apenas alterado em resposta a mudanças na maneira que se opera o negócio;			<b>X</b>
Reportável	<b>X</b>		
Transações processando Dados de Negócio frequentemente necessitam acessar os Dados de Referência		<b>X</b>	
Transações do negócio acessam Dados de Código para melhorar casos de entradas de dados, melhorar a consistência de dados, garantir integridade de dados, etc.;			<b>X</b>

## 2.4 Exemplos de situações particulares do BNB

Esta seção tem como propósito apresentar as diretrizes de Contagem de Pontos de Função utilizadas no BNB para casos e situações particulares presentes no dia a dia das atividades de desenvolvimento de software.

### 2.4.1 Fator de Ajuste

O BNB **não** utiliza a avaliação do fator de ajustes em suas medições, sendo fixado em 1,00. Ou seja, as características gerais do sistema, apresentadas no manual de práticas de contagem do IFPUG **não** afetam o tamanho dos pontos de função não-ajustados obtidos em suas contagens, fazendo com que seus valores sejam sempre iguais aos dos Pontos de Função Ajustados.

### 2.4.2 Data Warehouse e BI

Mesmo com características de desenvolvimento muito peculiares e a falta de referências, os elementos presentes são contados sem dificuldade quanto à interpretação das regras do CPM. A principal intenção de um Data Warehouse é consolidar dados em uma aplicação, originados de várias outras aplicações, dentro e/ou fora da organização.

Os dados podem ser armazenados como são recebidos, processados na entrada via ETL (extração, transformação, e carga) para derivação de novas informações.

A maior parte das funções de carga é processada em lotes e há poucas funções de manutenção de dados on-line.

Considerações:

#### a) ODS – Depósitos de Dados Operacionais

O ODS é um banco de dados projetado para integrar dados de múltiplas fontes visando facilitar operações, análise e reporte;  
A integração geralmente envolve a "purificação" dos dados, resolução de redundâncias e business rule enforcement;  
Contém dados com o maior nível de detalhes e contrasta com dados agregados normalmente armazenados nas tabelas do DW.

Via de regra não conta a carga no ODS, porque a ferramenta já faz isso. Depois, se ele for utilizado por algum DM, será um AIE.

#### b) DW – Data Warehouse

O DW é uma coleção lógica de informação de várias bases de dados operacionais e históricos usada para criar inteligência de negócio, suportando atividades de análise de negócio e tomada de decisão.

#### c) Tabelas Fato e Dimensão

Caracterizam, em conjunto, os ALI's da aplicação.

Medição: Cada tabela dimensional é um TR do ALI principal, que é a tabela Fato.

As tabelas Fato armazenam fatos básicos (medições) de um processo de negócio, obtidos a partir da ODS, que representam um determinado período;

As tabelas Dimensão contêm campos usados para restringir e agrupar dados em consultas. Categoriza dados para usuários responderem questões de negócio;

**Ponto de atenção:**

- Caso a quantidade de TR's seja indeterminada, considere um ALI e três EE na medição

**d) ETL – Extrair, Transformar e Carregar**

Processo:

- 1) Extração dos dados de fontes externas;
- 2) Transformá-los para adequar às necessidades de negócio;
- 3) Carregá-los no DW;

Medição:

- 1) Uma EE para cada origem;  
Obs: Se tiver dados de métrica com origens diferentes, contar uma EE para cada origem de dado de métrica.
- 2) Dados diretamente extraídos, transformados e carregados pelo processo ETL constituem TD's da transação.
- 3) Um ALI para cada Fato. Caso tenham várias Fatos só com organizações diferentes, contendo os mesmos dados, só conta 1 ALI. Ex: carga mensal e semanal.
- 4) Um AIE para cada dimensão, DM externos, origens externas.
- 5) Para o DM corporativo (S990), que será usado por outros sistemas, 1ALI para o DM... Cargas e dimensões serão EE. Dimensões serão ALI.
- 6) ALI Fato, os TR serão o somatório das dimensões mais cargas.

**Ponto de atenção:**

- A transformação e carga de dados são secundárias à intenção primária e fazem parte de um conjunto indivisível. Ou seja, não conte uma EE de Extração, uma EE de Transformação, e uma EE de Carregamento.
- Quando a dimensão for dado de código, não será contada.

**e) Data Mart**

O Data Mart (DM) representa uma estrutura em fronteira diferente ao DW.

É um ponto central de um modelo estrela, sendo usualmente acessado por diversos relatórios e aplicações. Podendo ser um resultado da seleção e agregação (soma, média, máxima, percentual) de Fatos do DW, gerado como resultado da combinação de Fatos do DW (número de operações por empregado, margem por produto, classe de risco) ou DM cujos dados não podem ser mais derivados a partir das Fatos do DW;

Medição:

Para cada Data Mart, deve ser medido um ALI para os dados armazenados e um EE para os dados processados para gerá-lo.

**Ponto de atenção:**

- A princípio, como o Data Mart se utiliza das mesmas dimensões do DW, nenhum ALI adicional deve ser considerado.

#### f) Relatórios/Consultas

Considerando o comportamento das consultas como relatórios, temos os tipos:

- a) Relatórios Fixos ou Pré-formatados; \* Contados tradicionalmente. No BNB, esses relatórios são gerados usando ferramentas específicas.
- b) Relatórios OLAP – Online Analytical Processing que permitem os usuário manipularem e analisarem um volume significativo de dados sob múltiplas perspectivas;

Medição:

- a) Identificar o requisito e contar uma CE ou SE, como relatório normal.
- b) Não será contado, pois são gerados por uma ferramenta.

#### g) Cubo

A **geração do Cubo** (também denominado Universo ou Contexto de Análise) deve ser contada como uma SE. Esse tipo de tabela, normalmente, é utilizada para consumo por outras aplicações ou pelo próprio Datamart.

### 2.4.3 Trilha de Auditoria

Trilha de auditoria é um conjunto de registros de auditoria definidos e regulamentados em documento específico. O documento BNB\_Diretrizes\_Segurança, devidamente versionado e mantido pela Área de Segurança do BNB, publicado e disponibilizado para consulta, é parte integrante dos Requisitos Suplementares de Sistemas Aplicativos do BNB.

Normalmente a gravação de trilha de auditoria ocorre para transações pré-definidas da aplicação. Não sendo, portanto, um requisito específico e particular dessas transações, mas sim requisitos gerais da aplicação seguindo uma política geral do BNB. Uma evidência disto, é que esse requisito não estará presente nos casos de uso que especificam as transações, mas em uma parte geral da especificação de requisitos, normalmente os Requisitos Suplementares. Logo, não há transação adicional a ser considerada pela gravação da trilha de auditoria; assim como não haverá nestas transações a contagem de um arquivo referenciado para os dados de auditoria.

Para que a Trilha de Auditoria seja considerada um ALI do sistema e seja efetivamente contado, ela deve ser relevante para o negócio e ser solicitada formalmente pelo Gestor da aplicação. Sugere-se que seja avaliado o desenvolvimento de uma funcionalidade para o usuário realizar a consulta a esses dados e incorporá-la ao sistema. Cabe ao Gerente de Projetos esclarecer ao Gestor sobre os custos relacionados a essas demandas de extrações eventuais de dados do ALI, que podem superar os custos da construção e incorporação de uma funcionalidade de negócio.

O histórico, na maioria das vezes, é considerado registro lógico do ALI relacionado, devendo ser solicitado pelo gestor e deve haver no sistema funcionalidades de consulta a tais dados.

Exemplos específicos do BNB:

- 1) Usando o componente “Trilha de Auditoria” conforme estabelecido e fornecido pelo BNB:
  - Se for uma funcionalidade nova, que usará a Trilha de Auditoria, não será acrescentado nada na contagem do sistema.
  - Se for adicionar a Trilha de Auditoria a uma funcionalidade já existente,



deverá ser contada a alteração da transação que a utilizará, adicionando os TDs envolvidos na chamada da trilha de auditoria. Não será contado nenhum ALI a mais.

- 2) Desenvolvendo uma Trilha de Auditoria personalizada:
  - Caso o usuário peça uma trilha de auditoria com algumas regras específicas, deve-se contar a Trilha de Auditoria como uma funcionalidade de negócio. Um ALI para a Trilha personalizada e a alteração da transação que a utilizará, adicionando os TDs envolvidos na chamada da trilha de auditoria.

#### 2.4.4 Log da Aplicação

**Log de dados** é uma expressão utilizada para descrever o processo de registro de eventos relevantes dentro de um sistema. Os logs relativos aos processos elementares de uma determinada aplicação são denominados “log de aplicação”.

Os logs são muitas vezes reconhecidos pelo usuário, mas caso não sejam, o analista deve esclarecer sobre a importância deste arquivo para o negócio da aplicação, bem como da consulta aos seus registros.

Para os logs mantidos como “imagem antes” e “imagem depois”, deve ser contada a “imagem antes” como um TD e a “imagem depois” como outro TD.

O log de aplicação será único para toda a aplicação sendo contada.

O log de aplicação será considerado como arquivo referenciado, para cada processo elementar em que for tratado.

Esta orientação não se aplica a log de sistema operacional.

Qualquer manutenção em funções de transação já implementadas, para tratamento de log, deverá ser considerada como Projeto de Melhoria.

#### 2.4.5 Sistemas on-line e batch

Os sistemas on-line são aqueles que apresentam uma interface de acessos e de trabalho com os usuários de negócio, de forma independente das plataformas e linguagens de programação envolvidas.

- 1) Algumas funcionalidades se apresentam em figura de “combos”. Se o combo de consulta trazer dado de entidade do negócio, não for um combo estático, deve ser contado como Consulta Externa.
- 2) Identificar as views referenciadas para Tabelas de Outros Sistemas.  
Ex: A Visão de Clientes do S400 (Ativos e Inativos) **V030CLIE** do SINC acessa DDES400..T400PESS com 15 campos. Porém para o SINC apenas 06 campos são relevantes. Logo, contabilizamos 1 AIE com apenas 06 Tipos de Dados.

**Ponto de atenção:**

Nem toda view deve ser contada como AIE. Se a view referenciar uma Tabela do próprio sistema, não se considera nem como AIE, nem como ALI.

- 3) Há casos de sistemas que se integram a outros sistemas por componentes reusáveis/WebServices. Desta forma, identificar esses componentes e quais informações são consultadas.

Ex: O componente do S253 oferece um serviço com 20 informações. O SINC utiliza esse serviço, porém para o Sistema apenas 3 informações são relevantes. Logo, contabilizamos 1 AIE com apenas 3 Tipos de Dados.

**Ponto de atenção:**

- Para um sistema que utiliza um serviço, não é necessário identificar as transações, nem a complexidade desse serviço.
- Se o serviço fizer uma consulta a uma base externa, conta-se um AIE.
- O que deve ser contado são as quantidades de campos que saem e entram na fronteira.
- Estas observações valem para um serviço desenvolvido fora da fronteira da aplicação.

- 4) Alguns sistemas tem continuidade de processos em códigos que estão diretamente ligados ao banco de dados, nas figuras de TRIGGERS e/ou PROCEDURES. Exatamente por serem continuidade elas são interpretadas como passos de algum processo elementar.

- 5) Alguns sistemas tem continuidade de processos no que chamamos de janela noturna de processamento, são os processos batch. Alguns outros sistemas tem atuação exclusivamente no processamento noturno, sem portanto, a interface do usuário de negócio. As APLICAÇÕES, os JOBS e os STEPs são as principais ferramentas para caracterizar e controlar estes processos na janela noturna. Algumas considerações devem ser feitas:

- a) Há Jobs que são continuidade de um processo on-line e existem por limitações técnicas e não por necessidade de negócio. Como por exemplo, rotinas que se iniciam em um momento com o usuário de negócio e que pela necessidade de um “fechamento” para atualização e divulgação para rede de agências são encerradas pela execução de um JOB. A contagem desse tipo de job já foi feita no momento de identificação e contagem do processo on-line, o momento original da funcionalidade. Dessa forma o Job é considerado somente como um passo para a conclusão da funcionalidade, não tendo uma existência independente.

- I. O usuário interage com o sistema pelo on-line, logo um processo batch que completa um processo do on-line, não deve ser contado, pois é acessório e considerado um passo do processo total.
- II. Relatórios gerados por batch e disponíveis no on-line não devem ser contados. Arquivo VSAM (com ordenação de dados na base, por exemplo) não deve ser contado.

- b) Para uma identificação efetiva da independência de um JOB, considerar:

- I. A funcionalidade Fechamento do Dia – não é funcionalidade do usuário. Não contar.
- II. Verificar comportamento / dependências da aplicação (Aplicação é um conjunto de Jobs programados na ferramenta CONTROL-M). Caso existam muitos Jobs, analisar:
  - Identificar quais são as tarefas de negócio executadas no processamento batch.

- Se um deles cancelar e for necessário iniciar por algum JOB anterior, provavelmente todos farão parte de uma única transação.
  - Se um JOB cancelar e iniciar por esse, provavelmente serão transações diferentes.
  - A existência de poucos Jobs em uma aplicação, normalmente, significa que seja uma única transação.
  - Execuções eventuais não serão contadas, pois são re-execuções de transações no on-line ou finalização do on-line.
  - Se não significar uma funcionalidade para o usuário, não é transação, por exemplo, Transformar Relatório em arquivo.
- c) Nas demais situações que o JOB tem existência independente, ele é a origem e a conclusão de uma funcionalidade, a contagem se dá de forma natural, envolvendo a fronteira do sistema, suas funções de dados e suas funções transacionais.

**Ponto de atenção:**

- Não são contados arquivos de movimento recebidos (ou temporários) de outra aplicação para manter um ALI (Exemplos: arquivos de remessa e de retorno), no entanto, os processos de carga e de geração desses arquivos podem ser funções do tipo transação.
- O arquivo de movimento é simplesmente um relatório gerado em formato do arquivo do sistema operacional.
- Um arquivo é temporário, se durante sua existência não há consulta, exclusão, inclusão ou alteração por outro processo, no mundo do usuário.
- Relatórios gerados para o usuário serão contados como CE/SE. Se forem vários relatórios com dados semelhantes e com agrupamentos distintos, contar um único Processo Elementar.

#### 2.4.5.1 Exemplos de Processos Elementares

Os sistemas on-line são aqueles que apresentam uma interface de acessos e de trabalho com os usuários de negócio, de forma independente das plataformas e linguagens de programação envolvidas.

##### 1) Consultas Externas (CE)

- a. Informações em formato gráfico
- b. Drop-downs que recuperem dados de ALIs ou AIEs
- c. Menus gerados dinamicamente com base na configuração da aplicação.
- d. Telas de login sem criptografia

##### 2) Entradas Externas (EE)

- a. Transações que recebem dados externos utilizados na manutenção de arquivos lógicos internos
- b. Janelas ou Telas que permitem incluir, alterar e excluir registros. (OBS: Se a janela ou tela executa as 3 funções, logo contamos 3 Entradas Externas.)
- c. Processamento em lote (Batch) de atualização de bases cadastrais a partir de

arquivos de movimento.

- d. Em geral o nome da transação possui termos característicos como: incluir, alterar, excluir, gravar, importar, etc.

### 3) Saídas Externas (SE)

- a. Relatórios com totalização de dados
- b. Relatórios que também atualizam arquivos
- c. Consultas com cálculos ou apresentação de dados derivados
- d. Arquivos de movimento ou remessa gerados para outra aplicação.
- e. Informações em formato gráfico
- f. Telas de login com criptografia

#### Ponto de atenção:

- Drop-downs estáticos não são contados nem como CEs

## 2.4.6 Contagem de Pontos de Função com Múltiplas Mídias

Esta seção tem como propósito apresentar as diretrizes de Contagem de Pontos de Função utilizadas no BNB 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].

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 e adotadas no BNB segue a abordagem de *single 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. Sendo assim, nos casos de consultas em .pdf, .doc, .xls e consultas idênticas em tela e papel serão consideradas uma única funcionalidade e estimadas e contadas como *single instance*.

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

- **Canal:** também se refere a mídia. Múltiplos canais são sinônimos 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 *MultipleInstance*:** esta abordagem especifica que o tamanho funcional é obtido no contexto do objetivo da contagem, permitindo uma função de negócio ser reconhecida no contexto das mídias que são requisitadas para a funcionalidade ser entregue. A abordagem *multipleinstance* reconhece que a mídia para entrega constitui uma característica de diferenciação na identificação da unicidade da função transacional.

A seguir cenários com exemplos do comportamento de adoção no BNB:

Cenário	Descrição	Abordagem
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.	Adota-se a abordagem <i>single instance</i> , 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 ( <i>multipleinstance</i> ). Neste caso, duas funções são contadas: apresentação de dados em tela e apresentação de dados impressos.
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 às gravadas no arquivo.	Deve-se utilizar a abordagem <i>single instance</i> 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. Além disso, 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 <i>multipleinstance</i> .
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	Adota-se a abordagem <i>multipleinstance</i> , 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

	validações durante o processamento, da mesma forma que o processamento da entrada on-line também executa validações das informações.	processamento das validações nas entradas de dados on-line.
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 <i>multipleinstance</i> , que conta duas funcionalidades: consulta de dados na Web e consulta de dados via celular. Considera-se que a funcionalidade é desenvolvida duas vezes, uma para cada canal de saída. Algumas vezes, são até projetos de desenvolvimento distintos, um projeto relativo ao sistema Web e outro para o sistema via celular. Caso o projeto seja claro o suficiente para dizer que o desenvolvimento é o mesmo, deverá ser utilizada a abordagem <i>single instance</i> .
Relatório em múltiplos formatos	Um relatório deve ser entregue em diferentes formatos, por exemplo: um arquivo html e um arquivo com valores separados por vírgula (.csv).	Nestes casos, 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 se contará apenas uma vez, observando que a funcionalidade será da ferramenta e não da aplicação.

## 2.4.7 Projetos e Manutenções na plataforma LIFERAY

Esta seção tem como propósito apresentar as diretrizes de Contagem de Pontos de Função utilizadas no BNB em situações peculiares de desenvolvimento e manutenção de sistemas baseados na plataforma LIFERAY. A característica marcante do desenvolvimento na plataforma consiste no uso de componentes inerentes à ferramenta. A persistência de dados em estruturas específicas e proprietárias também deve ser mencionada.

Com base na avaliação e análise das características de sistema desenvolvido na plataforma LIFERAY, foram definidas as seguintes recomendações:

1. A contagem deverá ser feita pela identificação das funções de dados e transacionais reconhecidas na aplicação.

2. Para a contagem de aplicação já desenvolvida, o analista deve navegar pelas telas e identificar as funções de dados e funções transacionais envolvidas.
3. Desta forma, serão identificados os ALIs, AIEs e Processos Elementares. Isso feito teremos o tamanho em Pontos de Função da aplicação.

OBS: A contagem nos moldes apresentados corresponde ao mesmo procedimento usado na contagem de aplicação tradicional nas plataformas tradicionais adotadas no BNB. A diferença nessa abordagem está na forma de contratação, descrita no Guia de Contratação.

#### 2.4.8 Arquivos “temporários” que são utilizados por outras rotinas

Esta seção tem como propósito apresentar as diretrizes de Contagem de Pontos de Função utilizadas no BNB em situações peculiares de sistemas mainframe. Em determinadas rotinas, arquivos são criados para fins de performance. Esses arquivos são considerados Arquivos Temporários. Porém, muitos desses arquivos são utilizados como fonte de dados para outras rotinas durante sua vida útil.

Nesses casos, temos as seguintes considerações:

4. Para a rotina original, não se contam os arquivos temporários;
5. Para as rotinas que utilizam esses arquivos, conta-se o processo elementar (CE/SE), e os campos efetivamente utilizados do arquivo serão tipos de dados.
6. Esses arquivos intermediários não são considerados ALIs.

### 2.5 Medição das Manutenções

A APF não mede todos os tipos de manutenção possíveis em um software, restringindo-se apenas aquelas que alteram os requisitos de negócio/funcionais. Estas são chamadas pelo IFPUG de Manutenções Adaptativas. A terminologia utilizada pelo IFPUG é apresentada na tabela abaixo:

IFPUG	Significado
Manutenções Adaptativas	Alterações de requisitos funcionais
Manutenções Perfectivas	Alterações de requisitos não funcionais

A terminologia utilizada pelo BNB é apresentada na tabela abaixo:

BNB	Significado	Contagem
Manutenção Evolutiva	Alteração de requisitos <b>funcionais</b>	APF
Manutenção Adaptativa	Alteração de requisitos <b>não funcionais</b>	APF – Adaptada
Manutenção de Interface ou Manutenção Cosméticas	Alteração de “ <b>aparência</b> ”	APF – Adaptada
Manutenção Perfectiva	Alteração de requisitos <b>não funcionais</b>	APF – Adaptada
Manutenção Corretiva	Alteração decorrente de Falhas / Defeitos / Comportamentos inadequados	Não se aplica

No caso das manutenções mensuráveis, a técnica sempre medirá a **funcionalidade completa**, da forma como foi entregue ao usuário, independente da extensão da manutenção. Ou seja, mede-se a função que foi alterada, não o quanto ela foi alterada.

Pela maneira como o IFPUG definiu a medição da manutenção de uma função (medir a

função toda que será entregue), faz com que, em muitos casos, alterações muito pequenas tenham **o mesmo tamanho** de uma manutenção extensa na mesma função. ***Portanto, para que não haja desperdício de recursos é fundamental que se discipline uma maneira mais racional na demanda por manutenções.***

Quando se está num contexto em que diversas manutenções serão atendidas ao longo do tempo, o esforço ou o custo derivado da medição pela APF para uma manutenção específica pode ser **super** ou **subdimensionado**. Mas ao analisar o conjunto de diversas manutenções em um horizonte de tempo maior (ao menos um ano), estas distorções tendem a compensar se o parâmetro de preço (R\$/PF) ou taxa de entrega (HH/PF) forem bem estabelecidos.

### **3. Processo de Revisão do Guia de Contagem**

#### **3.1 Revisão para Correção de Inconsistências e Situações não previstas**

A revisão deste Guia será feita sempre que forem verificadas inconsistências entre uma definição do CPM e uma regra constante deste documento e situações não previstas neste Guia. Para situações não previstas neste Guia, dever-se-á recorrer à equipe de contagem da área de métricas do BNB que decidirá pela atualização deste guia.

#### **3.2 Revisão para Adoção de Novas Versões do CPM**

A adoção de nova versão do CPM como referência para este Guia de Contagem não será imediata à sua publicação pelo IFPUG.