

# Macoratti.net Análise de Pontos por Função - O Processo de contagem

A seguir apresento uma versão do capítulo sobre o processo de contagem da APF que faz parte de minha monografia para conclusão do curso de especialização na UFRGS.

Este capítulo descreve em detalhes as regras de contagem de APF baseada no *Function Point Counting Practices Manual* – Release 4.1.1, publicado pelo IFPUG em 1999.

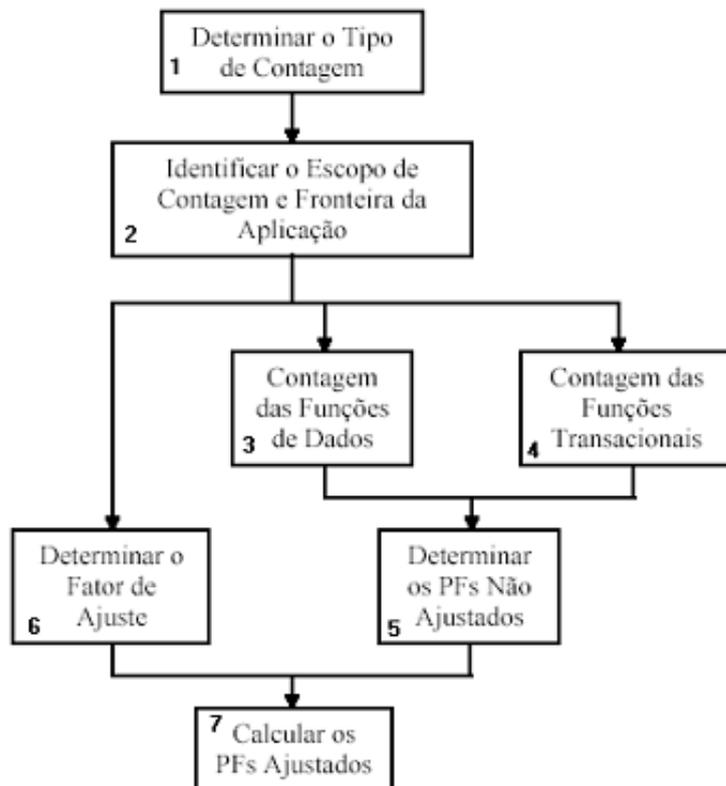


Figura 3.1: Procedimento de Contagem de Pontos de Função (HAZAN, 2001)

Figura 3.1: Procedimento de Contagem de Pontos de Função (HAZAN, 2001)

O procedimento para contagem de pontos de função compreende sete passos mostrados na figura 3.1.

Neste capítulo será detalhado cada passo do processo de contagem.

### 3.1 Determinar o tipo de Contagem

O primeiro passo a ser seguido para a contagem de PF de um projeto de software é determinar o tipo de contagem. Neste passo é estabelecido o tipo de contagem que será usado para medir o projeto de software, tanto no processo como no produto. São possíveis três tipos de contagem.(IFPUG,1999):

1. Contagem de projeto de desenvolvimento;
2. Contagem de projeto de melhoria (manutenção);
3. Contagem de aplicação.(produção)

#### 3.1.1 Projeto de Desenvolvimento

O número de pontos de função de um projeto de desenvolvimento mede a funcionalidade fornecida aos usuários finais quando da primeira instalação do software entregue quanto o projeto estiver pronto. Esta contagem também abrange as funções de conversão de dados que serão precisas para a implantação do software. Como exemplo de função de conversão de dados pode-se citar a necessidade de importar dados de um sistema antigo para o sistema em implantação.(VAZQUEZ,2005)

### 3.1.2 Projeto de melhoria (manutenção)

Em um projeto de melhoria o número de pontos de função mede as modificações para uma aplicação já existente ou seja, as funções adicionais, modificadas ou excluídas do sistema pelo projeto e as funções de conversões de dados.

Após a conclusão e implantação do projeto de melhoria, o número de pontos de função da aplicação deve ser atualizado para refletir as mudanças nas funcionalidades da aplicação. (VAZQUEZ, 2005)

### 3.1.3 Projeto de aplicação

A contagem de pontos de função de uma aplicação refere-se a uma aplicação já instalada e mede a funcionalidade fornecida ao usuário pela aplicação instalada provendo uma medida da atual funcionalidade ganha pelo usuário da aplicação. Ela é iniciada ao final da contagem do projeto de desenvolvimento e atualizado no final do projeto de melhoria. (VAZQUEZ, 2005)

## 3.2 Identificar o escopo da contagem e a fronteira da aplicação

A identificação do escopo visa definir a abrangência da contagem estipulando se a contagem vai se referir a um ou mais sistemas ou a apenas parte de um sistema. Assim, como exemplo, o escopo da contagem de uma aplicação pode abranger todas as funcionalidades disponíveis, algumas funcionalidades específicas ou apenas as funcionalidades usadas pelo usuário. (VAZQUEZ, 2005).

A fronteira da aplicação separa a aplicação que esta sendo contada das aplicações externas indicando o limite entre a aplicação e os demais usuários. A fronteira é definida estabelecendo um limite lógico entre a aplicação que esta sendo contada o usuário e as outras aplicações. Ao identificar a fronteira da aplicação deve-se estabelecer o seguinte: (IFPUG, 1999)

Os limites entre as funções a serem atendidas pela aplicação ou projeto dimensionado e àquelas pertencentes ao ambiente externo;

A propriedade dos dados considerados pelo processo de contagem;

O relacionamento entre os processos, com indicação de onde eles ocorrem.

Segundo o IFPUG, para identificar a fronteira do processo de contagem, devem ser adotadas as seguintes regras:

Deve ser considerado o ponto de vista do usuário, ou seja, o que o usuário pode entender e descrever como função da aplicação.

A fronteira entre aplicações relacionadas deve considerar a funcionalidade das aplicações em termos das funções de negócio identificadas pelo usuário, e não sob o ponto de vista das interfaces necessárias.

Para Vazquez (VAZQUEZ, 2005), as seguintes dicas auxiliam na identificação da fronteira da aplicação:

- Obter uma documentação do fluxo de dados no sistema e desenhar uma fronteira em volta para destacar quais partes são internas e externas à aplicação
- Identificar áreas funcionais pela atribuição de propriedade de certos objetos de análise, como entidades e processos;
- Verificar como o grupo de dados são mantidos;
- Verificar como a aplicação é gerenciada; se é desenvolvida ou mantida na sua totalidade por uma equipe distinta;

A identificação da fronteira da aplicação é um dos passos mais importantes pois se o posicionamento da mesma for feita de maneira incorreta poderá haver uma contagem incorreta dos pontos de função comprometendo o processo.

## 3.3 Contagem das funções de dados

As funções de dados representam as funcionalidades fornecidas ao usuário para atender requisitos internos e externos referentes a dados. São classificadas em arquivos lógicos internos (ALI) e arquivos de Interface

O termo arquivo não significa um arquivo físico no sentido tradicional de processamento de dados mas refere-se a um grupo de dados logicamente relacionado e reconhecido pelo usuário.

Com o objetivo de apoiar a contagem de Arquivos Lógicos Internos e dos Arquivos de Interface Externa o IFPUG complementou os procedimentos e regras do CPM 4.1.1 com um nova seção denominada "Práticas de Contagem" onde situações como dados de código, dados compartilhados e outros casos são avaliados e onde é feita uma abordagem para medir os requisitos de armazenamento em termos de pontos de função. Essa abordagem esta ilustrada na figura 3.2 :(VAZQUEZ 2005)

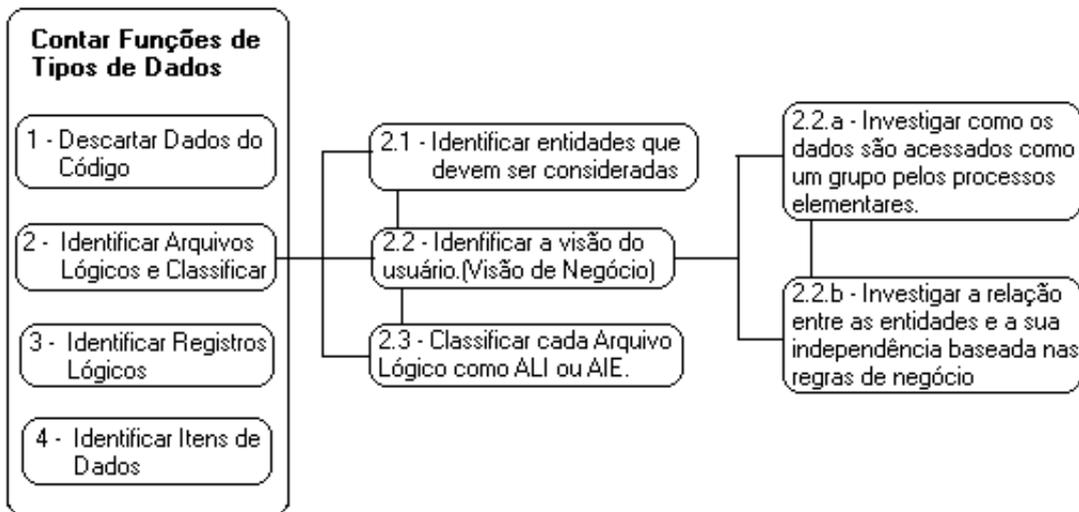


Figura 3.2:

Abordagem prática de contagem ALI e AIE (VAZQUEZ, 2005. p.80.)

Pelo diagrama da abordagem descrita na figura 3.2 verifica-se que os requisitos de armazenamento funcionais e não funcionais de uma aplicação classificados como dados de código não devem ser considerados como ALI, embora possam representar até metade das entidades em um modelo de dados. Devido ao fato de serem implementações de requisitos técnicos não devem influenciar o tamanho funcional da aplicação. As transações existentes para manutenção de dados de código não devem ser consideradas como processos elementares nem os dados de código contados como arquivos referenciados nos processos elementares que os atualizem ou leiam. (VAZQUEZ, 2005)

### 3.3.1 Arquivo Lógico Interno (ALI)

É um grupo logicamente relacionado de dados ou informação de controle cuja manutenção é feita pela própria aplicação. Sua função principal é armazenar dados mantidos dentro da fronteira da aplicação através dos processos da aplicação. Os ALI contribuem para o cálculo de pontos de função com base na sua quantidade e complexidade funcional relativa.

Um grupo logicamente relacionado de dados refere-se a dados relacionados em um nível que o usuário consegue perceber como sendo importante para permitir que a aplicação realiza uma atividade definida. (IFPUG,1999)

As informações de controle são dados usados pela aplicação para garantir total conformidade com os requisitos das funções do negócio definidas pelo usuário.

Manutenção é habilidade de adicionar, alterar ou remover dados de um arquivo através de um processo elementar da aplicação.

Como exemplos de um ou mais ALIs , dependendo da visão do usuário , têm-se : (IFPUG,1999)

- Dados da aplicação (arquivos mestres como cadastro de clientes ou funcionários);
- Arquivos de dados de segurança da aplicação;
- Arquivos de dados de auditoria;
- Arquivos de mensagem de auxílio;
- Arquivos de mensagens de erro;

- Arquivo de cópia de segurança. Considerado somente se for solicitado pelo usuário para atender requisitos da aplicação.
- Arquivo que sofra manutenção por mais de uma aplicação.
- Não são considerados como ALI:
- Arquivos temporários;
- Arquivos de trabalho;
- Arquivos de classificação;
- Arquivos de cópia de segurança requerido pelo CPD.
- Arquivos introduzidos somente por causa da tecnologia usada. Ex.: arquivos de parâmetro para um software *WFL*, *JCL*, etc.;
- Operações de junção e projeção.
- Arquivos de índices alternativos

Cada Arquivo Lógico Interno deve ser classificado de acordo com sua complexidade funcional relativa, que é baseada no número de Registros Lógicos(RL) e no número de Itens de Dados (ID) do arquivo.(IFPUG, 1999)

### 3.3.1.1 Identificação do Número de Registros Lógicos

Um Registro Lógico é um subgrupo de dados reconhecido pelo usuário dentro de um ALI. Dependendo da visão do usuário um ALI pode ter mais de um Registro Lógico(RL).

Existem dois tipos de subgrupos que podem ser identificados como registros lógicos (IFPUG, 1999):

**Mandatários** – São subgrupos de dados que o usuário deve usar pelo menos uma vez durante o processo elementar de criação de um item num ALI.

**Opcionais** – São subgrupos de dados que o usuário tem opção de usar ou não durante o processo elementar de criação de um item em um ALI.

Regras que devem ser aplicadas para contagem dos registros lógicos:

Conte um registro lógico para cada subgrupo identificado , opcional ou mandatário.

Considere um registro lógico caso o ALI não possua subgrupos.

### 3.3.1.2 #9; Identificação do Número de Itens de Dados

Um item de dados (ID) representa um segmento de um ALI que possui um significado único, não repetitivo e pode ser reconhecido pelo usuário. Representa um campo de dados que formula uma ocorrência de informação completa. (IFPUG, 1999)

Regras de contagem para os itens de dados (IFPUG, 1999):

Contar um item de dados para cada campo único , não repetitivo, reconhecido pelo usuário e mantido em um ALI via execução de um processo elementar.. Ex: Um número contábil ou data que é armazenado em múltiplos campos é contado como um único item de dado.

Quando duas ou mais aplicações mantêm o mesmo ALI, mas cada uma mantém itens de dados separados, contar somente os itens de dados usados por cada aplicação para dimensionar o ALI.

Contar um item de dados para cada parte de dado requisitada pelo usuário para definir um relacionamento com um outro ALI, ou seja, uma chave estrangeira ou uma associação entre objetos.

### 3.3.1.3 Determinar a complexidade de um ALI

Conforme o número de itens de dados (ID) referenciados e o número de registros lógicos (RL) encontrados , um ALI pode ser classificado em simples, médio e complexo de acordo com a tabela 3.1:

Tabela 3.1: Tabela de Complexidade para Arquivo Lógico Interno

Número de Itens de dados(ID)			
1 a 19 ID	20 a 50 ID	51 ou mais ID	
1 RL	SIMPLES	SIMPLES	MÉDIA
2 a 5 RL	SIMPLES	MÉDIA	COMPLEXA
6 RL ou mais	MÉDIA	COMPLEXA	COMPLEXA

Fonte: IFPUG , 1999

### 3.3.2 #9; Arquivo de Interface Externa (AIE)

Um Arquivo de Interface Externa (AIE) é um grupo de dados logicamente relacionados ou informações de controle identificadas pelo usuário, referenciados na aplicação para fins de recuperação de dados cuja manutenção é feita por outra aplicação. Os dados são armazenados fora da fronteira da aplicação. (VAZQUEZ,2005)

Um arquivo de Interface Externa de uma aplicação sempre será contado como um ALI na aplicação de origem.

São considerados AIE , conforme a visão do usuário (IFPUG,1999):

Dados de referência (dados externos usados pela aplicação ,mas que não são usados para manutenção em ALI);

- Arquivos de mensagens de auxílio;
- Arquivos de mensagens de erro.
- Não são considerados AIE:
- Dados recebidos de outra aplicação usados para adicionar, alterar ou remover dados em um ALI;
- Dados cuja manutenção é feita pela aplicação que esta sendo avaliada mas que são acessados e utilizados por outra aplicação;
- Dados formatados e processados para uso por outra aplicação.

Cada Arquivo de Interface Externa (AIE) deve ser classificado de acordo com sua complexidade funcional relativa, que é baseada no número de Registros Lógicos (RL) e no número de Itens de Dados(ID) do arquivo.

#### 3.3.2.1 Identificação do Número de Registros Lógicos

Um Registro Lógico é um subgrupo de dados reconhecido pelo usuário dentro de um AIE. Dependendo da visão do usuário um AIE pode ter mais de um Registro Lógico.

Existem dois tipos de subgrupos que podem ser identificados como registros lógicos (IFPUG, 1999):

**Mandatários** – São subgrupos de dados que o usuário deve usar pelo menos uma vez durante o processo elementar de criação de um item num AIE.

**Opcionais** – São subgrupos de dados que o usuário tem opção de usar ou não durante o processo elementar de criação de um item em um AIE.

Regras que devem ser aplicadas para contagem dos registros lógicos:

- Conte um registro lógico para cada subgrupo identificado , opcional ou mandatário.
- Se não existirem subgrupos contar um registro lógico para cada AIE.

#### 3.3.2.2 #9; Identificação do Número de Itens de Dados

Um item de dados representa um segmento de um Arquivo de Interface Externa que possui um significado único e pode ser reconhecido pelo usuário. Representa um campo de dados que formula uma ocorrência de informação completa.

- As regras de contagem para itens de dados em um AIE são (IFPUG, 1999):
- Contar um item de dados para cada campo único, não repetitivo, reconhecido pelo usuário e referenciado em AIE via execução de um processo elementar.

Quando duas ou mais aplicações referenciam o mesmo AIE, mas cada uma referenciam itens de dados separados, contar somente os itens de dados usados por cada aplicação para dimensionar o AIE.

Contar um item de dados para cada parte de dado requisitada pelo usuário para definir um relacionamento com um outro AIE, ou seja, uma chave estrangeira ou uma associação entre objetos. Ex.: Se um AIE é composto por mais de uma tabela em um Banco de dados relacional, as chaves usadas para relacionar as tabelas são contadas apenas uma vez.

### 3.3.2.3 Determinar a complexidade de um AIE

Conforme o número de itens de dados referenciados e o número de registros lógicos encontrados, um AIE pode ser classificado em simples, médio e complexo de acordo com a tabela 3.2

Tabela 3.2: Tabela de complexidade para Arquivo de Interface Externa

Número de Itens de dados (ID)			
1 a 19 ID	20 a 50 ID	51 ou mais ID	
1 RL	SIMPLES	SIMPLES	MÉDIA
2 a 5 RL	SIMPLES	MÉDIA	COMPLEXA
6 RL ou mais	MÉDIA	COMPLEXA	COMPLEXA

Fonte: IFPUG, 1999

### 3.3.3 Determinação da contribuição dos ALI e AIE

Após determinar o número de ALI e AIE e suas respectivas complexidades pode-se calcular a contribuição de cada um utilizando a seguinte tabela do IFPUG:

Tabela 3.3: Tabela de contribuição dos PF não ajustados das funções de dados

Tipo de Função	SIMPLES	MÉDIA	COMPLEXA
Arquivo Lógico Interno(ALI)	7 PF	10 PF	15 PF
Arquivo de Interface Externa(AIE)	5 PF	7 PF	10 PF

Fonte: IFPUG, 1999

Desta forma, pela tabela acima, um ALI de complexidade MÉDIA contribui com 10 pontos de função não ajustados, já um AIE de complexidade MÉDIA contribui com 7 pontos de função.

### 3.4 Contagem das funções do tipo Transação

As funções transacionais representam as funcionalidades fornecidas ao usuário para atender suas necessidades de processamento de dados pela aplicação. São classificadas em Entradas Externas, Saídas Externas e Consultas Externas.(VAZQUEZ,2005)

A identificação dos tipos de funções transacionais nem sempre é fácil, de forma que o IFPUG criou uma tabela que resume os objetivos primários de cada tipo de função transacional para facilitar sua identificação

Tabela 3.4 : Resumo dos objetivos primários das funções de transação

Tipo de Função Transação			
EE	SE	CE	
Alterar o comportamento do sistema	IP	F	NP
Manter um ou mais ALIs	IP	F	NP
Apresentar Informações	F	IP	IP

Fonte: IFPUG, 1999

Onde têm-se:

IP - intenção primária da função de transação

F - Uma operação que pode ser feita pela função , mas não é sua intenção primária. Ela pode existir ou não.

NP – Não é permitida a função transacional.

EE – Entrada Externa

SE – Saida Externa

CE – Consulta Externa

### 3.4.1 Entradas Externas

Uma entrada externa é um processo elementar que processa dados ou informações de controle recebidos de fora da fronteira da aplicação e cujo objetivo principal é manter um ou mais Arquivos Lógicos Internos (ALI) e/ou alterar o comportamento do sistema. Desta forma uma EE provoca uma inclusão , exclusão e/ou alteração nos dados dos ALI. (VAZQUEZ, 2005)

As EE representam o fluxo de informação de fora para dentro da fronteira da aplicação , ou seja, transações originadas do usuário ou de outros sistemas e que representam alguma forma de entrada de dados no sistema.

Uma EE processa dados ou informações de controle que entram pela fronteira da aplicação. Esses dados, através de um processo lógico único, atualizam ALI, Informações de controle podem ou não atualizar diretamente. Uma EE é considerada única para uma aplicação se possuir um formato diferente das demais ou se precisar de uma lógica de processamento diferente de outras EE que tenham o mesmo formato.

O conceito de formato pode ser compreendido como um conjunto de dados arrumados em uma ordem única dentro da aplicação e a lógica de processamento é visto como o conjunto de críticas, cálculos, algoritmos e referências/utilização de ALI ou AIE, requeridos pelo usuário para realizar um processo elementar.

São exemplos de EE: Operações de inclusões e alterações de registros em arquivos da aplicação, Janela que permite adicionar, excluir e alterar registros em arquivos.

Não são exemplos de EE : Menus, Telas de Login, telas de filtro de relatórios e consultas, Múltiplos métodos de se executar uma mesma lógica de entrada

O processo de identificação de EE requer que seja analisados todos os processos elementares que recebem dados de fora da aplicação e que fazem a atualização de um ou mais ALIs segundo as seguintes regras( IFPUG, 1999):

Os dados ou informações de controle devem ser recebidos de fora da fronteira da aplicação;

Se a entrada de dados pela fronteira não for uma informação de controle que modifique o comportamento do sistema , deve se manter no mínimo um AIE;

Para o processo elementar identificado a ser contado como uma EE, pelo menos uma das três opções a seguir devem ser satisfeitas:

- A lógica de processamento deve ser única e diferente das demais entradas externas;
- O conjunto de dados elementares identificados é distinto dos conjuntos identificados por outras EE;
- Os ALIs mantidos e os AIEs referenciados são distintos dos utilizados por outras EE;

Cada EE deve ser classificada conforme sua complexidade funcional relativa que é baseada no número de

Arquivos Referenciados (ALI e AIE) e no número de itens de dados (ID).

#### 3.4.1.1 Identificação do Número de Arquivos Referenciados

Um arquivo referenciado é qualquer ALI que foi consultado ou atualizado pelo processo ou qualquer AIE que foi consultado sendo que o número de Arquivos Referenciados é a soma dos ALI e AIE atualizados ou consultados na Entrada Externa.(IFPUG,1999)

Regras de contagem para os Arquivos referenciados em uma EE (IFPUG, 1999):

- Contar um AR para cada ALI mantido;
- Contar um AR para cada ALI ou AIE lido durante o processo de EE;
- Contar somente um AR para cada ALI que seja mantido e lido durante o processo da EE.

#### 3.4.1.2 Identificação do Número de Itens de Dados

Um item de dado é um campo único , não repetido, identificado pelo usuário e que é atualizado em um ALI pela EE.

Cada item de dado atualizado em um ALI pela EE deve ser computado , considerando o seguinte (IFPUG, 1999) :

- Contar um item de dados para cada campo único , não repetitivo, reconhecido pelo usuário e mantido em um ALI via execução de um processo elementar
- Campo recuperado ou derivado pelo sistema e armazenado em um ALI, durante um processo elementar de uma EE que não cruzar a fronteira da aplicação não deve ser contado.
- Linhas de comando ou teclas de função que provêm a capacidade para definir a ação a ser tomada pela EE.
- Campos não informados pelo usuário, mas que são atualizados em um ALI por uma EE.
- Mensagem de erro ou confirmação ligadas aos processos lógicos executados pela EE.

#### 3.4.1.3 Determinar a complexidade de uma EE

A complexidade funcional de uma EE é determinada em função da quantidade de ALIs e AIEs referenciados e do número de itens de dados referenciados e pode ser classificada em simples, média e complexa conforme tabela 3.5.

Tabela 3.5 : Tabela de complexidade de Entradas Externas

Número de Itens de Dados(ID)			
1 a 4 ID	5 a 15 ID	16 ou mais ID	
0 ou 1 arquivo referenciado	SIMPLES	SIMPLES	MÉDIA
2 AR	SIMPLES	MÉDIA	COMPLEXA
3 ou mais AR	MÉDIA	COMPLEXA	COMPLEXA

Fonte : IFPUG , 1999

#### 3.4.2 Saídas Externas

Uma saída externa(SE) é um processo elementar que envia dados ou informações de controle para fora da fronteira da aplicação. Seu objetivo é exibir informações recuperadas através de processamento lógico , isto é , processamento que envolva cálculos ou criação de dados derivados e não apenas uma simples recuperação de dados. Uma Saída Externa pode manter um ALI ou alterar o comportamento do sistema. Representam assim atividades do sistema que transformam dados dos ALI e geram resultados que são exibidos ao usuário. (VAZQUEZ,2005)

O processamento lógico é definido como o conjunto de críticas, cálculos, algoritmos e referência/acesso a arquivos requisitados pelo usuário que visa completar um processo elementar. (IFPUG,1999)

O processo elementar é a menor conjunto de atividades significativa para o usuário final em termos de função de negócio.

Dados derivados são dados que exigem um processamento diferente de um acesso, recuperação ou edição de dados de um ALI e/ou AIE.

A identificação de uma saída externa pode ser feita pela identificação de todos os processos e informações de controle que enviam dados para fora da fronteira da aplicação. Satisfeita esta condição pode-se considerar uma saída externa (IFPUG , 1999) :

Dados transferidos para outra aplicação : dados de um ALI que são formatados (arrumados em uma ordem única) e processados para uso por uma aplicação externa.

- Relatórios : Cada relatório produzido pela aplicação pode ser considerado uma SE. Para relatórios de formato idênticos mas que necessitam de lógicas de processamento ou cálculos distintos devem ser considerados duas saídas externas.
- Relatórios on-line : Saída de dados on-line que não seja a parte de saída de uma consulta Externa.
- Formatos Gráficos : Contados da mesma forma como saída em formato texto, isto é , cada formato gráfico diferente é contado como uma saída externa.

Gerador de relatórios : Cada relatório de uma saída desenvolvida para o usuário via gerador de relatório deve ser considerado como uma saída externa.

Não devem ser considerados como saídas externas:

- Telas de Ajuda;
- Literais;
- Data, hora, controles de paginação , etc.;
- Relatórios múltiplos com a mesma lógica e formato
- Relatórios criados pelo usuário de forma dinâmica pelo usuário usando um linguagem como SQL.

Cada saída externa deve ser classificada de acordo com sua complexidade funcional e é baseada no número de Arquivos Referenciados e no número de itens de dados.

#### *3.4.2.1 Identificação do Número de arquivos referenciados*

Um arquivo referenciado é qualquer AIE que foi lido ou ALI que foi lido ou mantido pelo processamento da Saída Externa sendo que o número de Arquivos Referenciados é a soma dos ALI e AIE atualizados ou consultados na Saída Externa.

Regras de contagem para AR em uma Saída Externa( IFPUG, 1999):

- Contar um AR para cada Ali mantido durante o processo elementar;
- Contar um AR para cada ALI ou AIE lido durante o processo elementar;
- Contar somente um AR quando um ALI é mantido e lido pelo processo elementar da SE.

#### *3.4.2.2 Identificação do Número de Itens de Dados*

Um item de dado é um campo único, não repetitivo, identificado pelo usuário que é apresentado pela saída externa.

Um item de dado deverá ser identificado conforme as seguintes regras (IFPUG, 1999):

- Contar um item de dado para cada campo não repetido reconhecido pelo usuário e que sua entrada seja feita pela fronteira da aplicação. Se o item de dados entra e sai pela fronteira da aplicação ele será computado apenas uma vez;

- Contar um item de dado pela capacidade da aplicação enviar mensagens de resposta para fora da fronteira, indicando erro ou sucesso no processamento;
- Um campo recuperado ou derivado pelo sistema e armazenado em um ALI, durante um processo elementar de uma SE, que não cruzar a fronteira da aplicação não deve ser contado.

O processo para identificar as SE deve verificar o processamento lógico do processo elementar conforme as seguintes regras (IFPUG, 1999):

- Se existe pelo menos uma fórmula matemática ou cálculo;
- Se cria dados derivados;
- Se mantém pelo menos um ALI;
- Se muda o comportamento do sistema.

### 3.4.2.3 Determinar a complexidade de uma SE

A complexidade funcional de uma SE é determinada em função do número de itens de dados e da quantidade de arquivos referenciados (ALI + AIE) podendo ser classificada como simples, média e complexa conforme a tabela 3.6.

Tabela 3.6 : Tabela de complexidade de Saídas Externas

Número de Itens de Dados (ID)			
1 a 5 ID	6 a 19 ID	20 ou mais ID	
0 ou 1 AR	SIMPLES	SIMPLES	MÉDIA
2 a 3 AR	SIMPLES	MÉDIA	COMPLEXA
4 ou mais AR	MÉDIA	COMPLEXA	COMPLEXA

Fonte: IFPUG , 1999

### 3.4.3 Consultas Externas

Uma consulta Externa representa a necessidade de processamento de consultas da aplicação sendo uma combinação de entrada/saída de dados onde uma entrada de dados causa uma recuperação e saída de dados correspondente. A lógica de processamento não deve conter fórmula matemática ou cálculo nem criar dados derivados ou atualizar nenhum ALI.

As consultas Externas são requisições de informações que para serem satisfeitas precisam que sejam combinados parâmetros de entradas e saídas permitindo a recuperação da informação solicitada pelo usuário. (Raquel , 2005)

Dados derivados são dados que exigem um processamento diferente de um acesso, recuperação ou edição de dados de um ALI e/ou AIE.

A identificação de uma CE é feita pela identificação dos processos onde uma entrada está associada a uma recuperação e exibição de dados sem derivação e sem atualização de um ALI. As seguintes regras devem ser satisfeitas para um processo elementar ser contado como uma consulta Externa:

O processamento elementar recupera dados ou informações de controle de um ALI ou AIE;

- A lógica do processo elementar não pode conter cálculos;
- A lógica do processo elementar não cria dados derivados;
- A lógica do processo elementar não mantém nenhum ALI;
- A lógica do processo elementar não altera o comportamento do sistema.

Pode-se citar como exemplos de CE:

Um processo de recuperação de dados que seleciona dados com base em uma entrada fornecida;

- Telas de Logon;
- Telas de Help;
- Telas de alteração/remoção que mostram o que será alterado ou removido antes de sua efetivação.
- Tela de menus que permitem informar parâmetros para a consulta na tela escolhida.
- Não são consideradas CE:
- Telas de Menus que oferecem somente funcionalidade de seleção de telas;
- Dados derivados;
- Documentação On-Line;
- Sistema de Teste;
- Sistema Tutoriais;
- Relatórios e consultas que contenham cálculo ou gerem dados derivados.

#### 3.4.3.1 Identificação do Número de arquivos referenciados

Um arquivo referenciado é um arquivo ALI ou AIE lido pela Consulta Externa.

Regras de contagem para um AR em uma CE (IFPUG, 1999) :

- Contar um AR para cada ALI lido;
- Contar um AR para cada AIE lido.

#### 3.4.3.2 Identificação do Número de Itens de Dados

Um item de dado é um campo único , não repetitivo, reconhecido pelo usuário. Todos os itens de dados que servem de parâmetro no processamento da consulta Externa devem ser contados.

Regras de contagem de itens de dados para CE (IFPUG, 1999):

- Contar um item de dados para cada campo reconhecido pelo usuário e não repetitivo que atravessa a fronteira da aplicação e é requisitado para definir quando , o que ou como os dados serão recuperados ou gerados pelo processo elementar;
- Contar um item de dados para cada campo reconhecido pelo usuário e não repetitivo que sai pela fronteira da aplicação;
- Se um item de dados entre e sai da aplicação deve ser contado somente uma vez;
- Contar um item de dados pela capacidade da aplicação enviar uma mensagem de resposta para fora da fronteira da aplicação indicando um erro ocorrido ou confirmando que um processo terminado ou deve continuar.
- Contar um item de dados pela habilidade da aplicação em definir que uma ação a ser feita, mesmo havendo diversos métodos de chamar o mesmo processo lógico;

Não devem ser contados : literais, paginação, variáveis ou sinalizações geradas pelo sistema.

Um item de dado adicional deve ser computado caso sejam requeridas mensagens de erro ou campos de confirmação associados à parte de entrada da Consulta Externa.

#### 3.4.3.3 Determinar a complexidade de uma CE

A complexidade funcional de uma CE é determinada em função do número de itens de dados e arquivos referenciados conforme a tabela 3.7:

Tabela 3.7: Tabela de complexidade de Consultas Externas

Número de Itens de dados(ID)	

1 a 5 ID	6 a 19 ID	20 ou mais ID	
0 ou 1 AR	SIMPLES	SIMPLES	MÉDIA
2 a 3 AR	SIMPLES	MÉDIA	COMPLEXA
4 ou mais AR	MÉDIA	COMPLEXA	COMPLEXA

Fonte : IFPUG, 1999

### 3.5 Cálculo de Pontos de Função não Ajustados ou Brutos

Após definir a fronteira da aplicação , o tipo de contagem e reconhecer as funções de dados e de transação pode-se calcular os pontos de função não ajustados, ou brutos, multiplicando-se o total de ALI, AIE, EE, SE, e CE pela respectiva complexidade conforme a tabela 3.8.(IFPUG, 1999)

Tabela 3.8 : Contribuição das Funções de dados e de transação na contagem dos PFs não ajustados

COMPLEXIDADE			
SIMPLES	MÉDIA	COMPLEXA	
ALI	7	10	15
AIE	5	7	10
EE	3	4	6
SE	4	5	7
CE	3	4	6

Fonte : IFPUG, 1999

O cálculo dos pontos de função não ajustados é feito da seguinte forma (HAZAN , 2001):

Para cada um dos cinco tipos de função (ALI, AIE , EE, SE e CE), são contados os totais de pontos de função (TPF) segundo a expressão:

$$TPF_i = \sum_{j=1}^3 N_{C_{i,j}} * C_{i,j}$$

Onde:

a)  $N_{C_{i,j}}$  é o número de funções do tipo  $i$  ( $i$  variando de 1 a 5, segundo os tipos de função existentes: ALI, AIE, EE, SE e CE) que foram classificados na complexidade  $j$  ( $j$  variando de 1 a 3 , segundo os valores de complexidade: simples , média e complexa)

b)  $C_{i,j}$  é o valor da contribuição da complexidade  $j$  no cálculo dos pontos de função  $i$ , conforme a tabela acima.

O total de pontos de função não ajustados (PFNA) é dado pela soma dos pontos das tabelas de função:

$$PFNA = \sum_{i=1}^5 TPF_i$$

onde  $i$  varia de 1 a 5, conforme os tipos de função existentes (AIL, AIE, EE, SE, CE)

### 3.6 Determinação do valor do fator de Ajuste

A técnica de Análise por Pontos de Função considera que outros fatores afetam o tamanho funcional de um sistema. Estes fatores estão relacionados com características da aplicação. No cálculo dos PF brutos não é

levada em conta a tecnologia usada nem os requisitos não funcionais. Por este motivo é calculado o valor do fator de ajuste (VFA) que é baseado em 14 características gerais de sistema. Abaixo têm-se as 14 características (detalhes no ANEXO A)

Tabela 3.9: Características gerais do sistema

	<b>Características gerais do sistema</b>
<b>C1</b>	Comunicação de dados
<b>C2</b>	Processamento distribuído
<b>C3</b>	Performance
<b>C4</b>	Utilização de Equipamento
<b>C5</b>	Volume de transações
<b>C6</b>	Entrada de dados on-line
<b>C7</b>	Eficiência do Usuário Final
<b>C8</b>	Atualização On-Line
<b>C9</b>	Processamento complexo
<b>C10</b>	Reutilização de código
<b>C11</b>	Facilidade de Implantação
<b>C12</b>	Facilidade Operacional
<b>C13</b>	Múltiplos Locais
<b>C14</b>	Facilidade de mudanças

Fonte: IFPUG, 1999

Para se obter o **VFA** deve-se calcular o nível de influência de cada uma das 14 características , baseado em um equação que atribui valores em uma escala que varia de nenhuma influência até a influência mais forte, isto é , de 0 a 5 conforme exibida na tabela 3.10:

Tabela 3.10 : Níveis de influência das características gerais do sistema

Níveis ou Graus de Influência
0 – Nenhuma Influência
1 – Influência Mínima
2 – Influência Moderada
3 – Influência Média
4 – Influência Significativa
5 – Grande Influência

Fonte : IFPUG, 1999

O fator de ajuste influência os pontos de função não ajustados em +/- 35% , o que dá um intervalo de variação da ordem de 0,65 a 1,35. Ele é responsável pela correção das distorções ocorridas na etapa de contagem das funções de dados e funções transacionais. Na prática , esta correlação , na maioria dos casos, varia entre -10% e + 10%.

Para cada uma das 14 características atribui-se um valor de 0 (nenhuma influência) a 5 (forte influência) que indica quanto uma característica tem influência no sistema. Os 14 graus de influência (GI) são então somados o

que resulta no grau de influência total (GIT),:

$$\text{GIT} = \sum_{i=1}^{14} \text{GI}_i$$

O valor do fator de ajuste (VFA) é calculado pela seguinte fórmula:

$$\text{VFA} = (\text{GIT} * 0,01) + 0,65$$

Se o fator de ajuste de valor é igual a 1,00 , a influência total das características gerais do sistema é neutra. Nesta situação , a contagem dos pontos de função ajustados equivale a contagem de pontos de função não ajustados.

Devido a sua subjetividade, a utilização do fator de ajuste tornou-se opcional ao final do ano de 2002 como medida para aceitação dos pontos de função do IFPUG como um método padrão de medida funcional, pois várias características estão relacionadas com requisitos não funcionais da aplicação.(VAZQUEZ, 2005)

### 3.7 Determinação os Pontos de Função Ajustados

O último passo da contagem de pontos de função é o cálculo dos pontos de função ajustados que engloba os três tipos de contagem : projeto de desenvolvimento , de melhoria e aplicação. Nesta etapa são usados três tipos de fórmulas diferentes , como descritas no manual do IFPUG, dependendo do tipo de cálculo que se deseja realizar.

#### 3.7.1 Cálculo de Pontos de Função para um projeto de desenvolvimento

O projeto de desenvolvimento apresenta três componentes em termos de funções :

Funcionalidades da aplicação incluídas pelos usuários como requisitos – "Compreendem as funções usadas depois da instalação do sistema. Elas existem para satisfazer as necessidades de saída do negócio do usuário." (IFPUG, 1999)

Funcionalidades de conversão incluídas pelos usuários como requisitos – "Compreendem funcionalidades providas somente na instalação do sistema. Elas existem para converter dados ou proporcionar outros requisitos estabelecidos pelo usuário e necessários à conversão." (IFPUG, 1999)

Valor do fator de ajuste da aplicação – "Compreende a determinação das 14 características gerais do sistema em desenvolvimento, para avaliar a complexidade funcional da aplicação". (IFPUG, 1999)

Fórmula para cálculo:

$$\text{DFP} = (\text{UFP} + \text{CFP}) * \text{VAF}$$

Onde :

DFP – Número de pontos de função de desenvolvimento;

UFP – Número de pontos de função brutos apurados;

CFP – Número de pontos de função adicionados por processos de conversão de dados;

VAF – Valor do fator de ajuste.

#### 3.7.2 Cálculo de Pontos de Função para um projeto de Melhoria

Segundo o IFPUG o conceito de melhoria envolve apenas manutenções evolutivas na aplicação, ou seja, alterações feitas na aplicação para atender aos novos requisitos de negócio do usuário. Não são levadas em conta manutenções corretivas e preventivas. (VAZQUEZ, 2005)

Um projeto de melhoria consiste de três componentes em termos de funções:

Funcionalidades da aplicação incluídas como requisitos pelo usuário para o projeto: Funções incluídas, alteradas ou excluídas pelo projeto de melhoria;

Funcionalidades de Conversão : "Consiste dos pontos de função entregues por causa de qualquer funcionalidade

de conversão requerida pelo usuário". (IFPUG, 1999)

Valor do fator de ajuste da aplicação – Dois valores são considerados , segundo o manual :

- Valor do fator de ajuste ANTES do início do projeto de melhoria (VAFB)
- Valor do fator de ajuste DEPOIS da conclusão do projeto de melhoria (VAFA)

Deve ser considerada que uma função do tipo dado (ALI ou AIE) foi alterada quando ela foi modificada em sua estrutura com alguma inclusão , alteração ou exclusão de campos ou atributos.

Uma função do tipo transação é considerada alterada quando há alteração em um dos itens a seguir : (VAZQUEZ, 2005)

- Tipos de dados – Se houve inclusão , alteração ou exclusão da função.
- Arquivos referenciados – Se foram incluídos , excluídos ou alterados da função.
- Lógica de processamento – Se qualquer lógica for incluída , alterada ou excluída.

Fórmula para cálculo :

$$EFP = [(ADD + CHGA + CFP) * VAFA] + (DEL * VAFB)$$

Onde:

EFP – Número de pontos de função do projeto de melhoria;  
 ADD – Número de pontos de função não ajustados das funções incluídas pelo projeto de melhoria;  
 CHGA – Número de pontos de função não ajustados das funções modificadas depois das modificações;  
 CFP - Número de pontos de função não ajustados adicionados pela conversão;  
 VAFA – Valor do fator de ajuste da aplicação depois do projeto de melhoria;  
 DEL - Número de pontos de função não ajustados das funções excluídas pelo projeto de melhoria;  
 VAFB – Valor do fator de ajuste da aplicação antes do projeto de melhoria.

### 3.7.3 Cálculo de Pontos de Função para uma aplicação

Para calcular os pontos de função de uma aplicação existem duas fórmulas que são utilizadas :

Fórmula para Contagem Inicial – representa todas as funcionalidades requeridas pelo usuário de uma aplicação instalada. As funções da conversão de dados não devem ser computadas no tamanho da aplicação entregue pois elas existirão somente para o processo de implantação do aplicativo (VAZQUEZ, 2005).

$$AFP = ADD * VAF$$

Onde:

AFP – Número de pontos de função ajustados da aplicação;  
 ADD – Número de pontos de função não ajustados das funções instaladas;  
 VAF – Valor do fator de ajuste da aplicação.

- Fórmula usada após o projeto de melhoria – Após a conclusão de um projeto de melhoria os pontos de função devem ser atualizados para refletir as mudanças na aplicação. Novamente as funções de conversão de dados não devem ser computadas pois elas não fazem parte da aplicação (VAZQUEZ, 2005).

$$AFP = [(UFPB + ADD + CHGA) - (CHGB + DEL)] * VAFA$$

Onde:

AFP – Número de pontos de função ajustados da aplicação  
 UFPB – Número de pontos de função não ajustados da aplicação antes do projeto de melhoria;  
 ADD – Número de pontos de função não ajustados das funções incluídas pelo projeto de melhoria;  
 CHGA – Número de pontos de função não ajustados das funções modificadas depois do seu término;  
 CHGB – Número de pontos de função não ajustados das funções modificadas antes do seu término;  
 DEL - Número de pontos de função não ajustados das funções excluídas pelo projeto de melhoria;  
 VAFA – Valor do fator de ajuste da aplicação depois do projeto de melhoria.

**Abreviaturas usadas:**

- AIE - Arquivos de Interface Externa
- ALI - Arquivos Lógicos Internos
- ALR - Arquivo Lógico Referenciado
- APF - Análise por Pontos de Função
- CE - Consultas Externas
- EE - Entrada Externa
- ID - Item de dados
- IFPUG - International Function Point User Group
- PF - Pontos de Função
- PFA - Pontos de Função Ajustados
- PFNA - Pontos de Função Não Ajustados
- RL - Registros Lógicos
- SE - Sidas Externas

**Referências:**

HAZAN, Cláudia - Implantação de um Processo de medições de software – agosto , 2002 . disponível em <http://www.bfpug.com.br/artigos.htm>. acessado em: out. 2005

HAZAN, Cláudia - **Análise de Pontos por Função** – agosto , 2001 . disponível em <http://www.inf.ufes.br/~falbo/download/aulas/es-g/2005-1/APF.pdf>. acessado em: out. 2005

[IFPUG99] THE INTERNATIONAL FUNCTION POINT USERS GROUP, **Princeton Junction. Function Point Counting Practices Manual release 4.1.1** [s.l.], 1999.

VAZQUEZ, C. E.;SIMÕES, G. S; ALBERT, R. M. **Análise de Pontos de Função – Medição, Estimativas e Gerenciamento de Projetos de Software**. 2005. 3.ed. São Paulo- Editora Erica

---

José Carlos Macoratti