

Aula 32

Professor:

Geraldo Xexéo
UFRJ

Conteúdo:

**Análise de Pontos
de Função**

Pergunta comum

- ➡ Como calcular o preço de um software que eu/minha empresa vai vender?
- ➡ Como eu devo cobrar para desenvolver um software para um amigo?

Resposta Correta

- ➡ O preço do software não é cobrado em função do software propriamente dito, mas sim do valor de mercado e das relações comerciais entre as partes.

Outras Perguntas

- ➡ Qual o preço do sistema?
- ➡ Qual o custo do sistema?
- ➡ Qual o esforço para desenvolver o sistema?
- ➡ Quantas pessoas serão necessárias para fazer esse software?
- ➡ Em quanto tempo o sistema ficará pronto?
- ➡ Quantas linhas de código tem o sistema?
- ➡ Quantos pontos de função tem o sistema?
- ➡ Qual o tamanho do sistema?
- ➡ Que recursos são necessários para o sistema?

Perguntas Com Respostas

- ➡ Quanto tempo eu vou gastar fazendo esse software?
- ➡ Quantas pessoas vou precisar para fazer esse software?
- ➡ Quanto eu/minha empresa vou/vamos gastar?

Preço x Custo

Preço

- Mercado

- Cliente

Custo

- Esforço

- Tempo

- Custos da Empresa

Esforço ...

- ➡ O principal fator de custo no desenvolvimento de software é o gasto com pessoal
- ➡ Uma das principais preocupações da Engenharia de Software é determinar qual será a quantidade de pessoas e o tempo por elas dedicado a um projeto.
- ➡ Para isso usamos o conceito de Esforço que representa a quantidade de trabalho realizado, medido em pessoa-mês,
 - ➡ o trabalho feito por uma pessoa em um mês.

... Esforço

- ➡ Assim, podemos dizer que um sistema precisa de 4 pessoas-mês para ser realizado, ou seja, que uma pessoa trabalhando 4 meses ou 4 pessoas trabalhando um mês.
- ➡ Acontece que sistemas de informação são um pouco como bebês: não podemos ter a gestação de um bebê com nove mães em um mês.
- ➡ Na verdade, Boehm achou uma relação entre o esforço necessário e o tempo necessário para fazer um sistema, e conseqüentemente o tamanho médio da equipe.

Porém...

- ➡ Tanto o cliente quanto você/sua empresa podem negociar em torno do conceito de funcionalidade entregue
 - ➡ Porém, é importante lembrar que certas funcionalidades tem mais valor do que outras

O que é possível?

- ➡ Medir um produto de software
- ➡ Prever o esforço e o tempo necessário para produzi-lo
 - ▬ Dentro de condições específicos
 - ▬ Dentro de um histórico específico
 - ▬ Dentro de uma margem de erro

Medir?

- ➡ Para indicar o tamanho total
 - Dentro de condições específicos
- ➡ Para indicar o progresso
- ➡ Para indicar a qualidade

Métricas de Projeto

➡ Primeiro uso: nas estimativas

— Nos contratos

➡ Monitoração e controle

Tipos de Medidas

➡ Diretas

➡ Indiretas

➡ Medidas orientadas a tamanho

➡ Medidas orientadas a função

Medidas Orientadas a Tamanho

- ➡ Tamanho em KLOC
- ➡ Erros por KLOC
- ➡ Defeitos por KLOC
- ➡ Custo por KLOC
- ➡ Documentação por KLOC
- ➡ Erros/pessoa-mês
- ➡ LOC por pessoa-mês
- ➡ Custo/página de documentação

KLOC????

➡ Kilo Lines Of Code

➡ Depende da Linguagem

- ▢ Mesmo se usarmos SLOC

- ▢ Busca resolver o problema da instrução válida

Hello World em C

```
#include <stdio.h>
```

```
int main(void) {
```

```
    printf("Hello World");
```

```
    return 0;
```

```
}
```


Hello World em Cobol



```
000100 IDENTIFICATION DIVISION.  
000200 PROGRAM-ID. HELLOWORLD.  
000300  
000400*  
000500 ENVIRONMENT DIVISION.  
000600 CONFIGURATION SECTION.  
000700 SOURCE-COMPUTER. RM-COBOL.  
000800 OBJECT-COMPUTER. RM-COBOL.  
000900  
001000 DATA DIVISION.  
001100 FILE SECTION.  
001200  
100000 PROCEDURE DIVISION.  
100100  
100200 MAIN-LOGIC SECTION.  
100300 BEGIN.  
100400 DISPLAY " " LINE 1 POSITION 1 ERASE EOS.  
100500 DISPLAY "Hello world!" LINE 15 POSITION 10.  
100600 STOP RUN.  
100700 MAIN-LOGIC-EXIT.  
100800 EXIT.
```

Hello World in Z80

```
ORG 32768  
ENT
```

```
LD IY, #5C3A  
RES 0, (IY+02)  
RES 1, (IY+01)
```

```
LD HL, HELLO  
LD A,22  
RST #10  
LD A,0  
RST #10  
LD A,0  
RST #10
```

```
LOOP LD A,(HL)  
PUSH AF  
PUSH HL  
AND #7F  
RST #10  
POP HL  
INC HL  
POP AF  
BIT 7,A  
JR Z, LOOP
```

```
LD A,13  
RST #10  
LD HL, HELLO  
JR LOOP  
RET
```

```
HELLO DEFM /Hello World/  
DEFB 161
```

Discutindo LOCs

- ➡ Fácil de contar
- ➡ Dependente de linguagem
- ➡ Ruim para linguagens não procedurais
- ➡ Exigem estimar KLOCs muito cedo
- ➡ Podem penalizar programadores melhores

Linhas de Código

- ➡ São muito boas para criar curvas de desenvolvimento/custo
- ➡ Porém, quando o projeto está pronto!
- ➡ Os métodos que usam linhas de código se baseiam no histórico de projetos

O que queremos?

- ➡ Uma medida que possamos compreender, ensinar e verificar
- ➡ Uma medida repetitiva
- ➡ Uma medida que possamos utilizar para avaliar o valor do produto
- ➡ Uma medida que sirva para estimar

Análise de Pontos de Função

- ➡ Métricas normalizadas pela funcionalidade entregue ao usuário
- ➡ Baseadas em relações empíricas levantadas por Albrecht em 1979, revisado em 1983.
- ➡ Evoluíram, como prática de medição, devido ao FPUG

Ponto de Função

- ➡ Um Ponto de Função (PF) é uma medida abstrata e relativa que conta "o número de funções de negócio entregues ao usuário".
- ➡ Um relatório simples, por exemplo, pode medir "4 Pontos de Função".
 - ➡ Da mesma forma que um "metro" ou "um litro", Pontos de Função só fazem sentido quando comparados com um padrão.
 - ➡ Assim, um sistema com 1.000 PF entrega o dobro de funcionalidade de que um sistema com 500 PF.

Vantagens Reconhecidas

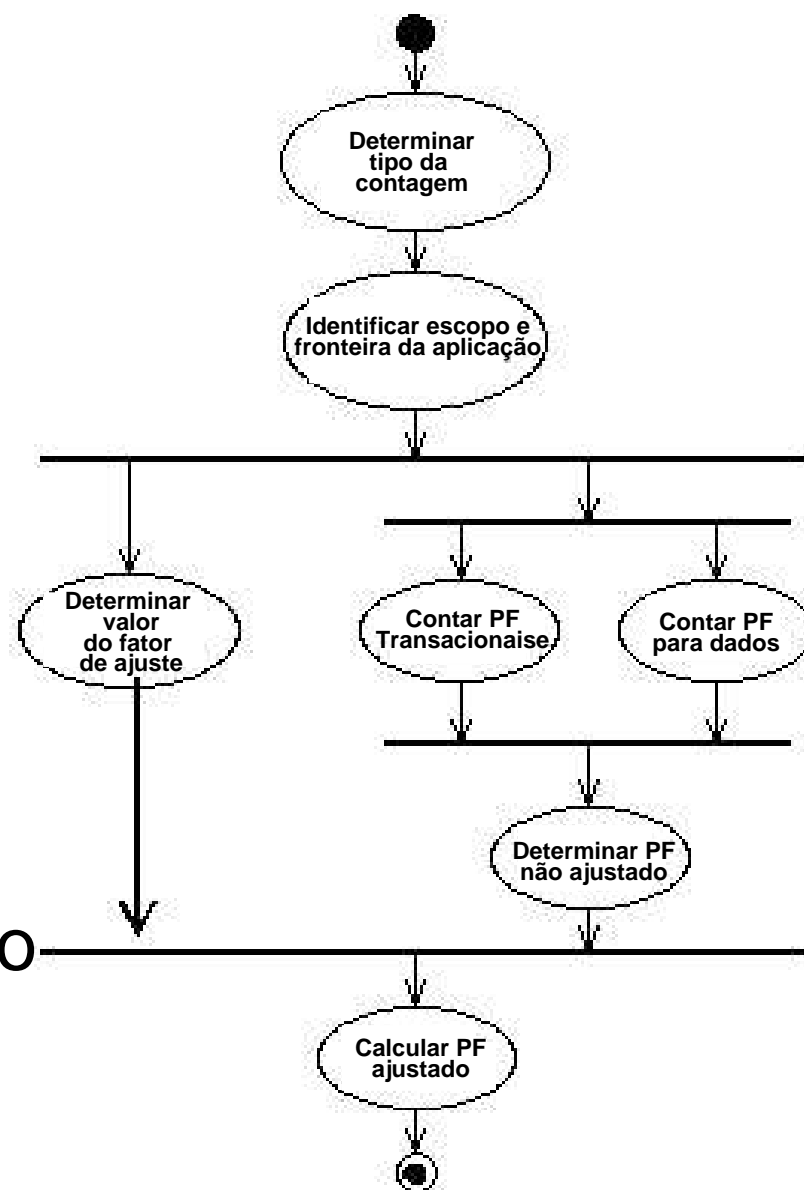
- ➡ Permite avaliar com segurança e precisão, com 10% de margem para sistemas existentes e 15-25% de margem para sistemas planejados:
- ➡ o valor do sistema para o usuário
 - ➡ o tamanho, custo e tempo de desenvolvimento
 - ➡ produtividade e qualidade de programadores
 - ➡ esforço de manutenção, modificação e customização
 - ➡ possibilidade de desenvolvimento
 - ➡ ganho com o uso de 4GL

Análise de Pontos de Função

- ➡ Quantifica a funcionalidade
- ➡ Caracteriza a complexidade de desenvolvimento das partes
- ➡ Pode ser usada para estimativas
- ➡ Aplicável a sistemas de informação

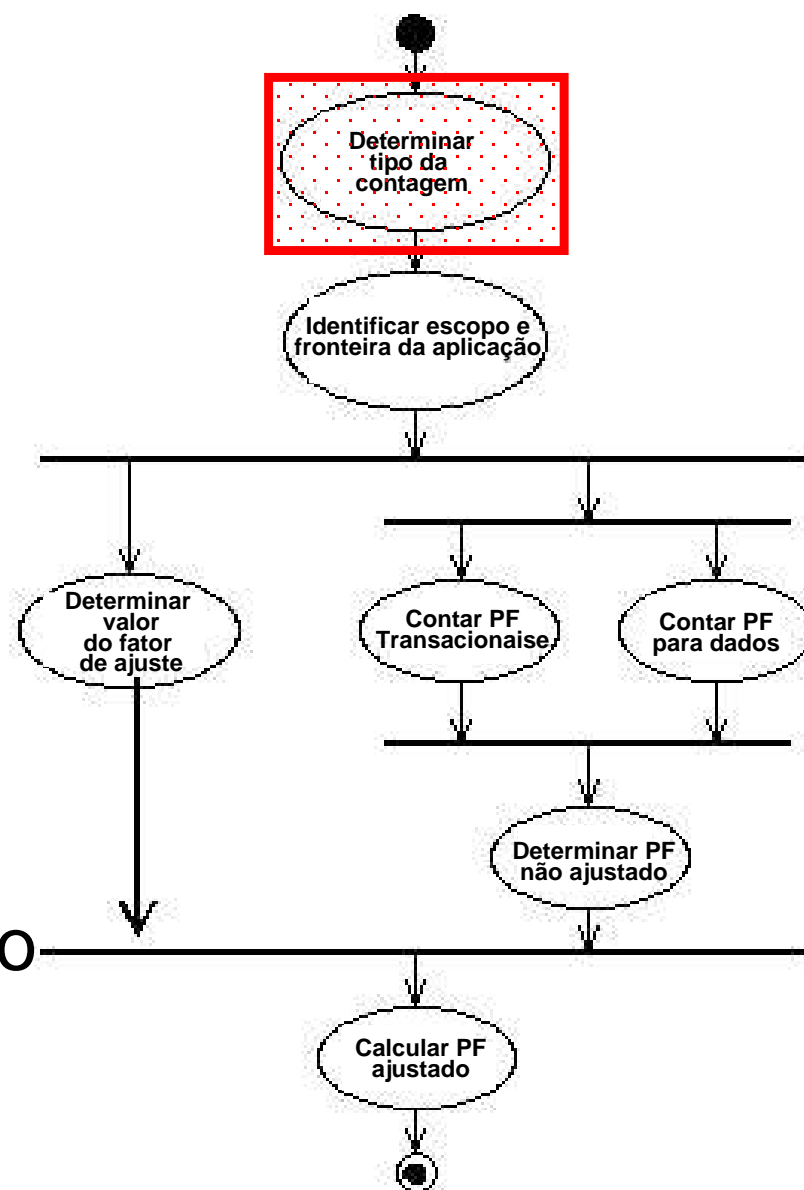
Análise de Pontos de Função

- ➡ Determinar Tipo da Contagem
- ➡ Identificar o Escopo de Contagem e a Fronteira da Aplicação
- ➡ Contar PFs Transacionais
- ➡ Contar PFs para Dados
- ➡ Determinar PFs Não-Ajustado
- ➡ Determinar Fator de Ajuste
- ➡ Calcular PFs Ajustado



Análise de Pontos de Função

- ➡ Determinar Tipo da Contagem
- ➡ Identificar o Escopo de Contagem e a Fronteira da Aplicação
- ➡ Contar PFs Transacionais
- ➡ Contar PFs para Dados
- ➡ Determinar PFs Não-Ajustado
- ➡ Determinar Fator de Ajuste
- ➡ Calcular PFs Ajustado



Iniciando a Contagem

- ➡ Estabelecer o objetivo da Contagem
- ➡ Identificar o Escopo da Contagem
- ➡ Identificar a Fronteira da Aplicação
- ➡ Documentar essas informações e as premissas que levaram a essas decisões

Propósito

- ➡ O procedimento de contagem se inicia com a determinação do propósito da contagem, isto é, a explicitação do motivo da contagem estar sendo realizada.
- ➡ Normalmente esse propósito estará relacionado a fornecer uma resposta a um problema de negócio existente, como a contratação de um serviço.

Contagem

➡ Estimativa

➡ Final

Determinar Tipo da Contagem

➡ A partir do propósito podemos determinar o tipo de contagem.

- ⇒ Contagem de Pontos de Função de um Projeto de Desenvolvimento
- ⇒ Contagem de Pontos de Função de um Projeto de Melhoria
- ⇒ Contagem de Pontos de Função de uma Aplicação

Projeto de Desenvolvimento

- ➡ Onde medimos as funcionalidades entregues ao usuário em uma versão onde o software é desenvolvido desde o início.
- ➡ Inclui as funcionalidades necessárias para a conversão de dados, mesmo que usadas uma única vez.

Projeto de Melhoria

- ➡ Onde medimos as modificações que alteram, adicionam ou removem funcionalidades em uma aplicação existente.
- ➡ Inclui as funcionalidades necessárias para a conversão de dados, mesmo que usadas uma única vez.

Aplicação

- ➡ Onde medimos uma aplicação já instalada.
- ➡ Também é chamada de "baseline" e é normalmente realizada no fim de um projeto de desenvolvimento e mantida atualizada nos projetos de melhoria.

Visão Geral da APF

➡ Determinar Tipo da Contagem

➡ **Identificar o Escopo de Contagem e a Fronteira da Aplicação**

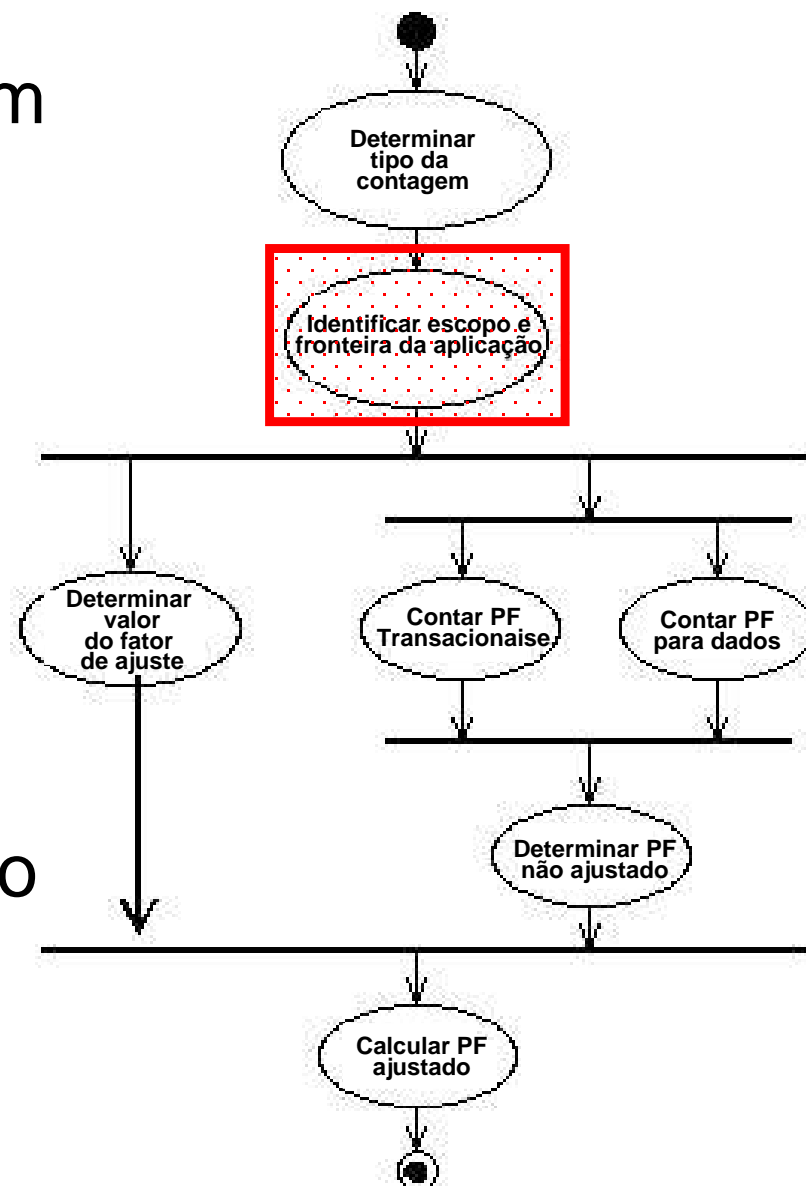
➡ Contar PFs Transacionais

➡ Contar PFs para Dados

➡ Determinar PFs Não-Ajustado

➡ Determinar Fator de Ajuste

➡ Calcular PFs Ajustado



Identificar o Escopo de Contagem

- ➡ O escopo da contagem define um conjunto ou um subconjunto do sistema, e permite dizer se uma funcionalidade deve ou não ser contada.

O que incluir no Escopo



Desenvolvimento:

- ▬ Tudo
 - ▬ Construído ou adaptado



Melhoria:

- ▬ Apenas o alterado
 - ▬ Inclusive o eliminado!



Aplicação:

- ▬ Apenas o utilizado
- ▬ Tudo que existe

Identificar a Fronteira da Aplicação

- ➡ A fronteira da aplicação delimita o software medido, definindo sua interface com o mundo exterior.
- ➡ Ela servirá não só para considerarmos se uma função deve ou não ser contada, mas também para considerar se um arquivo lógico deve ser contado como interno ou externo a aplicação.

Visão Geral da APF

➡ Determinar Tipo da Contagem

➡ Identificar o Escopo de Contagem e a Fronteira da Aplicação

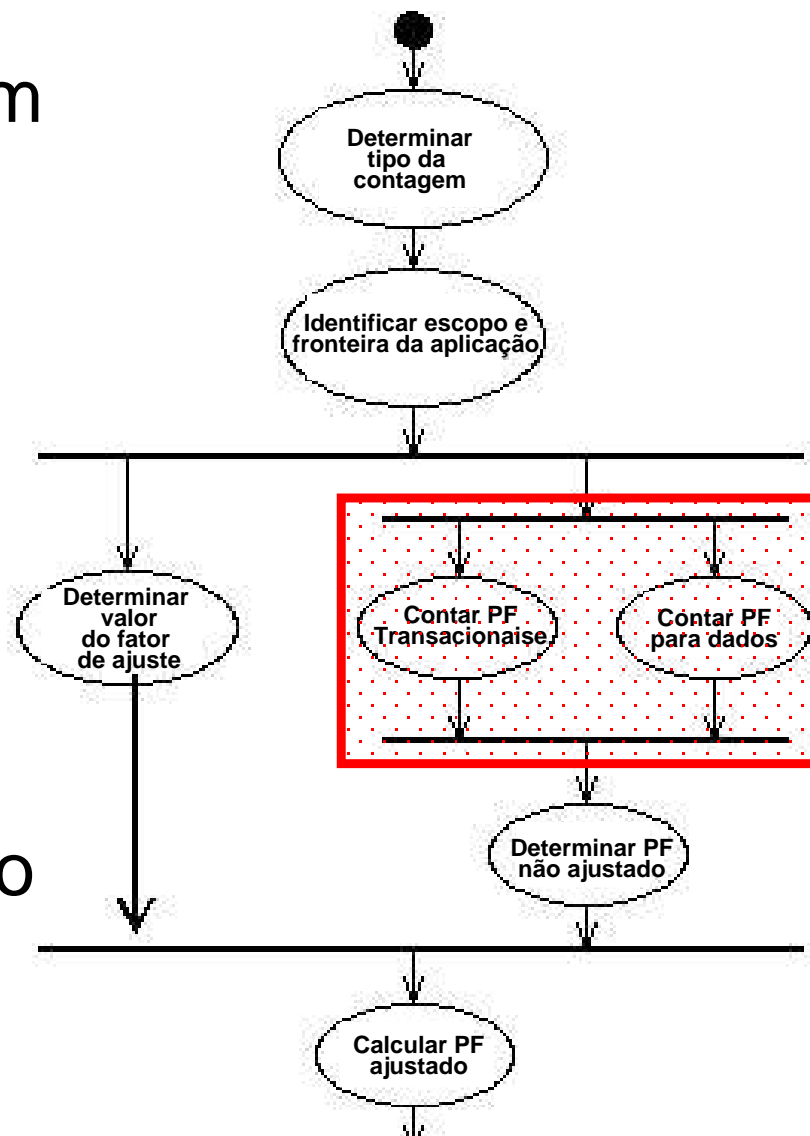
➡ Contar PFs Transacionais

➡ Contar PFs para Dados

➡ Determinar PFs Não-Ajustado

➡ Determinar Fator de Ajuste

➡ Calcular PFs Ajustado



Passos da Contagem

- ➡ Identificação das Funções
- ➡ Avaliação da Complexidade

Identificar Funções de Negócio

- ➡ Funções Transacionais
 - ▬ Processos Elementares
- ➡ Funções de Dados

Identificando Funções de Negócio

➡ Para identificar as funções de negócio devemos partir de algum documento que aponte as funções aprovadas e pelo usuário e úteis para o negócio.

- ☐ Não devem ser contadas funções necessárias por causa da tecnologia aplicada.
- ☐ Basicamente, só é cobrado o que o usuário pode ver e está disposto a pagar.
- ☐ Também é importante que as funções de negócio sejam cobradas como o usuário as percebem.
- ☐ Isto significa que não interessa se estamos usando um ou vinte arquivos para guardar uma informação, mas sim de quantas formas o usuário pode acessar essa informação.

Funções Contadas



Para serem contados as funções devem:

- Beneficiar claramente o usuário,
- Ser especificamente aprovado pelo usuário e
- Influenciar em algum grau mensurável o projeto, desenvolvimento, implementação e suporta à aplicação.

Ordem de Contagem

- ➡ Além disso, devemos identificar as funções seguindo certa ordem.
- ➡ A ordem é importante porque encontrar um tipo de função de negócio ajuda a encontrar as funções de outro tipo.

Determinar Tipo da Contagem

➡ Assim, em um sistema novo devemos usar a ordem:

— saídas, consultas, entradas, arquivos e interfaces.

➡ Por outro lado, em um sistema já existente devemos usar a ordem:

— arquivos, interfaces, saídas, consultas e entradas.

Visão Geral da APF

➡ Determinar Tipo da Contagem

➡ Identificar o Escopo de Contagem e a Fronteira da Aplicação

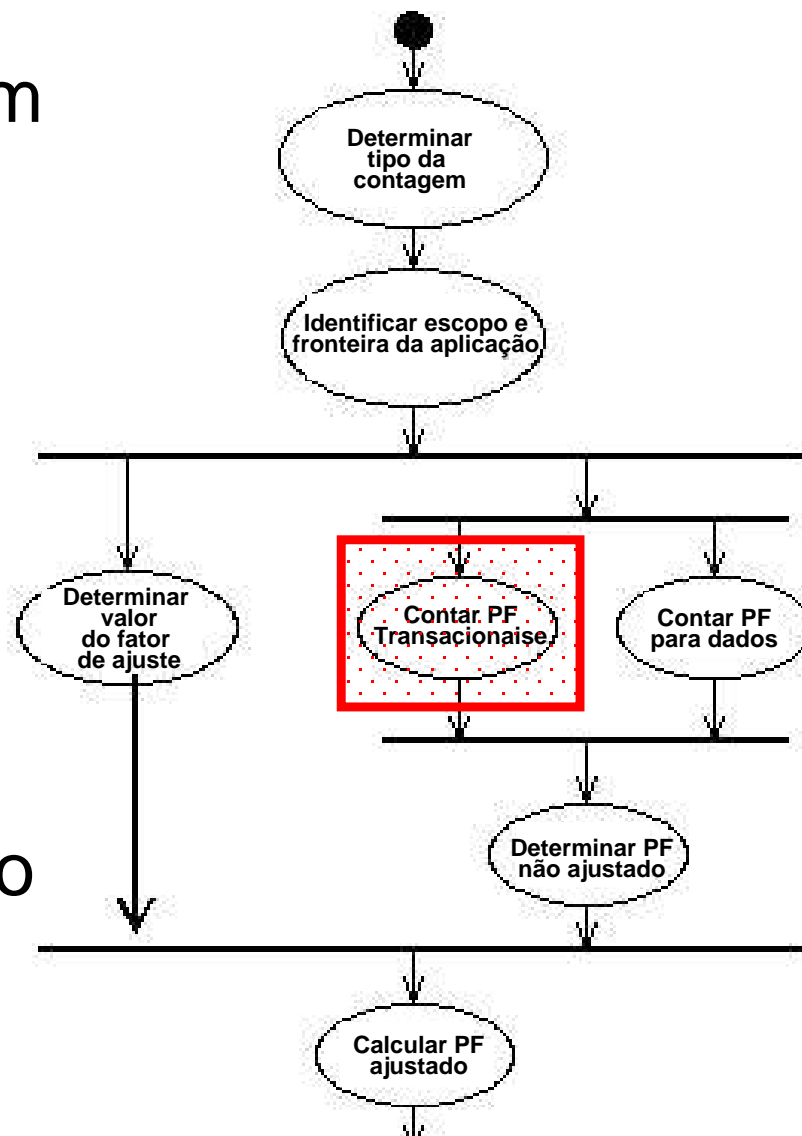
➡ **Contar PFs Transacionais**

➡ Contar PFs para Dados

➡ Determinar PFs Não-Ajustado

➡ Determinar Fator de Ajuste

➡ Calcular PFs Ajustado



Processo Elementar

- ➡ Um processo elementar é a menor atividade significativa para o usuário de forma indivisível.
- ➡ Isso significa que o usuário o vê como um processo único, completo e independente de outros, que se inicia com o sistema em um estado consistente e termina com o sistema em um estado consistente.

Funções Transacionais

 Saídas Externas

 Consultas Externas

 Entradas Externas

Saídas externas (SE ou EO)

- ➡ São informações de negócio que o usuário final pode receber, representando relatórios, telas e mensagens de erro como um todo e não em suas partes individuais;

Consultas externas (CE ou EQ)

- ➡ São saídas simples e imediatas, sem alteração na base, usualmente caracterizáveis por chaves simples de consulta.

Entradas externas (EE ou EI)

➡ São processos elementares que processam informações de negócio recebidas pelo sistema de fora da fronteira da aplicação e cuja finalidade principal é manter um Arquivo Lógico Interno .

Atenção

- ➡ Uma Saída Externa pode conter dados de entrada
 - Como, por exemplo, parâmetros de ordenação ou de busca para uma relatório
- ➡ Uma Entrada Externa pode contar saída
- ➡ Consultas são SE simplificadas

Identificando Saídas

➡ Deve ser contadas como saídas distintas cada formato utilizado.

- ▢ Basicamente, se for necessário fazer outro procedimento para produzir a informação, contamos como uma saída distinta.

➡ Também contamos cada tipo de lógica utilizada para fazer gerar a informação.

- ▢ Assim, se um relatório de vendas contém as vendas por vendedor e por loja, contaremos como duas saídas, pois são necessários procedimentos lógicos distintos para calcular cada um desses valores. Linhas de sumário e total, porém, não são contadas como novas saídas.

Identificando Saídas

- ➡ Uma saída externa pode ter uma "parte de entrada", para, por exemplo, selecionar os registros necessários em um relatório, usando alguns campos como filtro. Essa "parte entrada" não é contada a parte, já está considerada nessa contagem.

Identificando Consultas

- ➡ Consultas são, na prática, saídas simplificadas. Normalmente utilizadas para achar informações para modificá-las ou apagá-las. São sempre no monitor, não existe uma consulta em relatório de papel.
- ➡ Uma consulta não pode calcular nenhum valor. Em caso de cálculo de qualquer valor, temos uma saída.

Identificando Entradas

- ➡ Entradas permitem adicionar, modificar e apagar informações. Se uma tela permite estas 3 funções, são contadas 3 entradas. Normalmente as funções de modificar e apagar ainda exigem consultas correspondentes para achar a informação que será alterada.
- ➡ Um comando específico para o sistema executar algo é uma entrada.

Intenção Primária

Entradas

- Alterar comportamento do Sistema
- Manter Arquivos Lógicos Internos

Consultas

- Apresentar Informação para o Usuário

Saídas

- Apresentar Informação para o Usuário

Funcionalidade Admitida



Entradas

-  Apresentar Informação para o Usuário

Consultas

-  Nenhuma Outra!

Saídas

-  Alterar comportamento do Sistema
-  Manter Arquivos Lógicos Internos

Forma de Processamento Lógico	Tipo de Função Transacional		
	EE	SE	CE
Realiza validação dos dados	P	P	P
Realiza cálculos ou fórmulas matemáticas	P	O*	N
Converte valores equivalentes	P	P	P
Filtra dados e seleciona usando critérios específicos para comparar múltiplos conjuntos de dados	P	P	P
Analisa condições para determinar qual é aplicável	P	P	P
Altera ou inclui ao menos um ILF	O*	O*	N
Referencia ao menos um ILF ou EIF	P	P	O
Recupera dados ou informação de controle	P	P	O
Cria dados derivados	P	O*	N
Altera o comportamento do sistema	O*	O*	N
Prepara e apresenta informação fora das fronteiras do sistema	P	O	O
É capaz de aceitar dados ou informação de controle que entra pela fronteira da aplicação	O	P	P
Reordena ou reorganiza um conjunto de dados	P	P	P

Visão Geral da APF

➡ Determinar Tipo da Contagem

➡ Identificar o Escopo de Contagem e a Fronteira da Aplicação

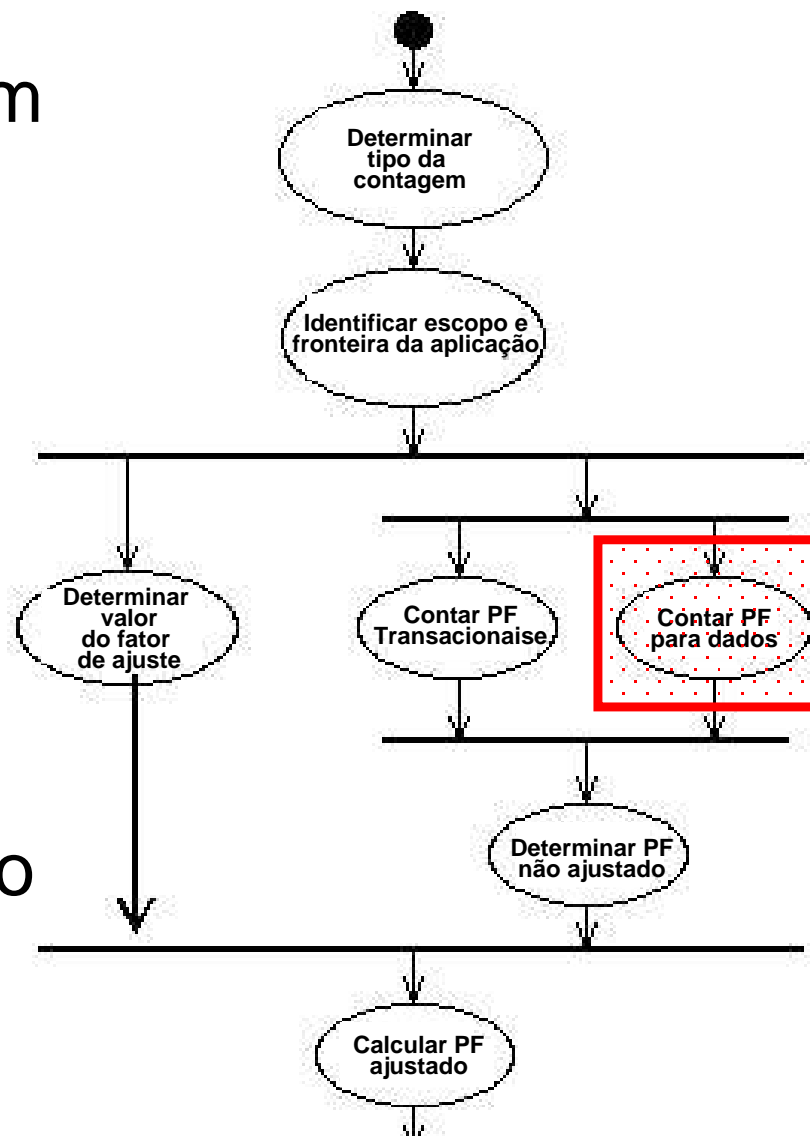
➡ Contar PFs Transacionais

➡ Contar PFs para Dados

➡ Determinar PFs Não-Ajustado

➡ Determinar Fator de Ajuste

➡ Calcular PFs Ajustado



Funções de Dados

- ➡ **Arquivos lógicos internos (ALI ou ILF)**
- ➡ **Arquivos de interface externos (AIE ou EIF)**

Arquivos lógicos internos (ALI ou ILF)

➡ que contém os dados permanentes, relevantes para o usuário e mantidos e utilizados pelo sistema. O sistema cria, altera e apaga esses dados.

Arquivos de interface externos (AIE ou EIF)

- ➡ que contém dados permanentes e relevantes para os usuários, guardados em algum lugar por outra aplicação, mas referenciados pela aplicação em questão.
- ➡ Outro sistema mantém esses dados.

Identificando Arquivos Internos

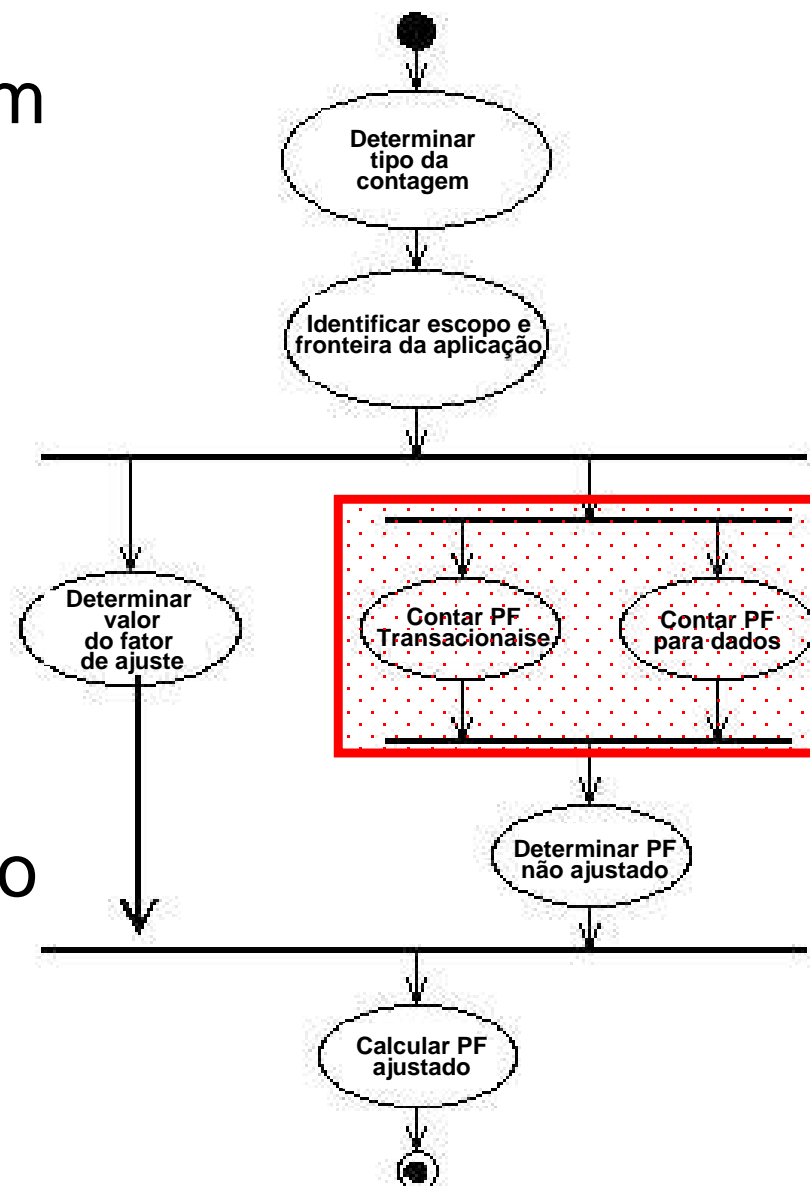
- ➡ Arquivos representam um grupamento lógico requerido pelo usuário. Podem incluir uma ou mais tabelas ou entidades.
- ➡ Esse é uma das partes mais difíceis da contagem de pontos de função, pois devemos separar o que o usuário pensa do modelo que criamos.
 - Nosso modelo muitas vezes usa vários grupos de dados, ou tabelas, ou entidades, para modelar algo que o usuário vê como um conceito único.
 - Mesmo na modelagem conceitual, a tendência do analista é incluir entidades que o usuário não "vê" naturalmente.

Identificando Arquivos Externos

- ➡ Arquivos Externos representam um grupamento lógico requerido pelo usuário. Podem incluir uma ou mais tabelas ou entidades.
- ➡ Arquivos Externos são mantidos por outras aplicações. Arquivos importados contam também como Entrada Externa, arquivos exportados contam também como Consulta Externa ou Saída Externa.

Visão Geral da APF

- ➔ Determinar Tipo da Contagem
- ➔ Identificar o Escopo de Contagem e a Fronteira da Aplicação
- ➔ Contar PFs Transacionais
- ➔ Contar PFs para Dados
- ➔ Determinar PFs Não-Ajustado
- ➔ Determinar Fator de Ajuste
- ➔ Calcular PFs Ajustado



RET

- ➡ Um Tipo de Elemento de Registro (RET, Record Element Type) é um subgrupo de elementos de dados dentro de um arquivo ou interface.
- ➡ Na prática, em um modelo de dados, um arquivo do usuário (ILF) é composto de um ou mais objetos do modelo.

RET

- ➡ Outra característica difícil de contar é que cada forma de acesso a um arquivo lógico conta novamente.
- ➡ Assim, por exemplo, se o usuário exige acessar um automóvel tanto por sua placa quanto por seu número do chassi, temos 2 arquivos lógicos para contar.
 - ➡ Exemplos: Uma nota fiscal é um arquivo lógico, com dois RETs: dados da nota, item de nota.

Identificando Itens de Dados (DETs)

➡ Itens de dados ou elementos de dados (DETs) campos únicos, reconhecidos pelos usuários, desconsiderando-se recursão e repetição. DETs também podem invocar ações. Exemplos de DETs são:

DETs em Entradas (EE)

- ➡ campos de entrada de dados
- ➡ mensagens de erro
- ➡ valores calculados que são guardados
- ➡ botões
- ➡ mensagens de confirmação
- ➡ campos repetidos contam apenas como um DET

DETs em Saídas

- ➡ Campos em relatório
- ➡ Valores calculados
- ➡ Mensagens de erro
- ➡ Cabeçalhos de coluna que são gerados dinamicamente em um relatório

DETs em Consultas

➡ Campos usados em filtros de procura

➡ O clique do mouse

Obs: DETs

- ➡ Em GUIs, botões onde só se pode fazer uma seleção entre muitas (normalmente *radio buttons*) devem ser contados como um DET apenas.
- ➡ Já *check boxes* são normalmente contadas uma a uma.
- ➡ Botões de comando devem ser contados como um elemento de dados levando em conta o fato de executarem uma função.

Arquivos

registro/ relacio- nament o	1- 19 Itens	20-50 Itens	+51 Itens
1	Simples (7)	Simples (7)	Médio (10)
2-5	Simples (7)	Médio (10)	Complexo (15)
+6	Médio (10)	Complexo (15)	Complexo (15)

Interfaces

registro/ relacio- nament o	1- 19 Itens	20-50 Itens	+51 Itens
1	Simples (5)	Simples (5)	Médio (7)
2-5	Simples (5)	Médio (7)	Complexo (10)
+6	Médio (7)	Complexo (10)	Complexo (10)

FTR

- ➡ File Type Referenced
- ➡ Arquivo Referenciado
 - ▬ ALI usado
 - ▬ AIE usado
- ➡ Contados quando alterados ou lidos
 - ▬ Mas só uma vez se ambas as ações acontecem

Saída

	1-5 Itens	6-19 Itens	+20 Itens
0-1 arquivos	Simples (4)	Simples (4)	Médio (5)
2-3 arquivos	Simples (4)	Médio (5)	Complexo (7)
+4 arquivos	Médio (5)	Complexo (7)	Complexo (7)

Entradas

	1- 4 Itens	5-15 Itens	+16 Itens
0-1 arquivos	Simples (3)	Simples (3)	Médio (4)
2 arquivos	Simples (3)	Médio (4)	Complexo (6)
+3 arquivos	Médio (4)	Complexo (6)	Complexo (6)

Consultas

	1-5 Itens	6-19 Itens	+20 Itens
0-1 arquivos	Simples (3)	Simples (3)	Médio (4)
2-3 arquivos	Simples (3)	Médio (4)	Complexo (6)
+4 arquivos	Médio (4)	Complexo (6)	Complexo (6)

Atenção

- ➡ É muito mais importante reconhecer todas as funções do que classificá-las corretamente
 - se você esquecer um arquivos, pode estar perdendo dezenas de pontos de função (entrada, saída, consulta, arquivo, interface)

Visão Geral da APF

➔ Determinar Tipo da Contagem

➔ Identificar o Escopo de Contagem e a Fronteira da Aplicação

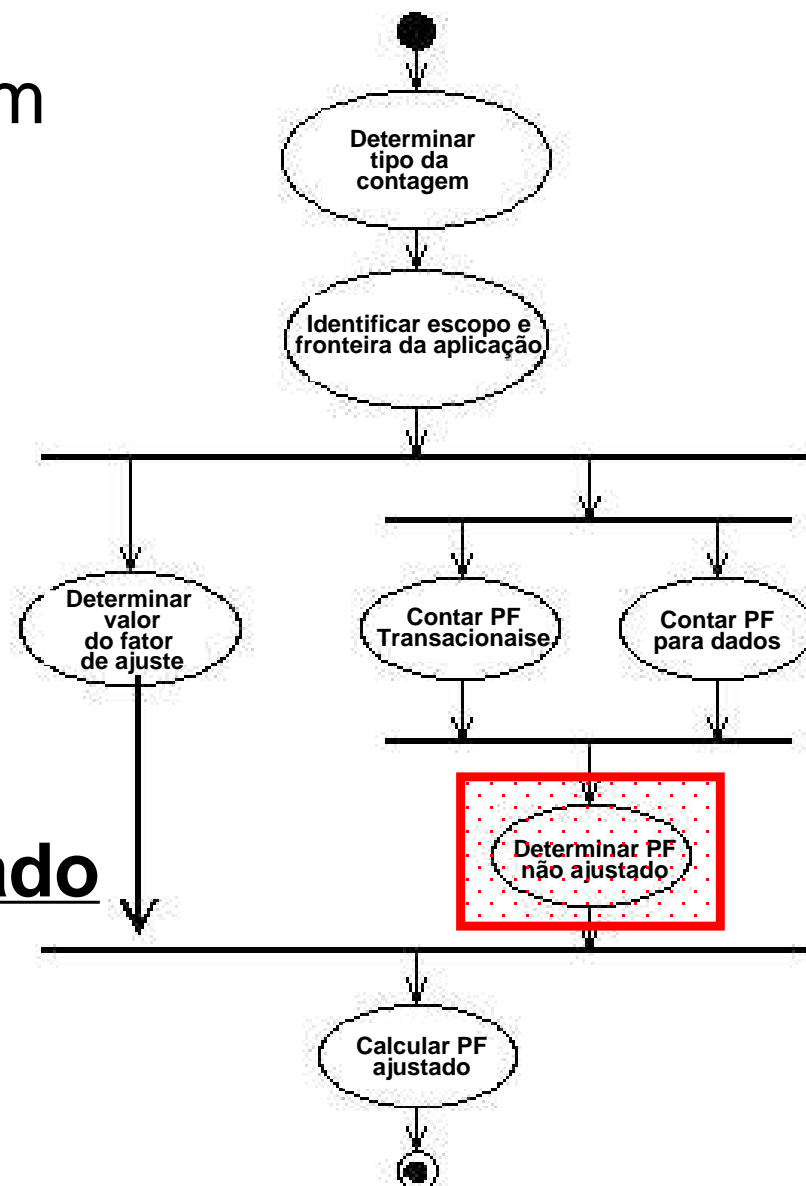
➔ Contar PFs Transacionais

➔ Contar PFs para Dados

➔ Determinar PFs Não-Ajustado

➔ Determinar Fator de Ajuste

➔ Calcular PFs Ajustado



Cálculo dos Pontos de Função

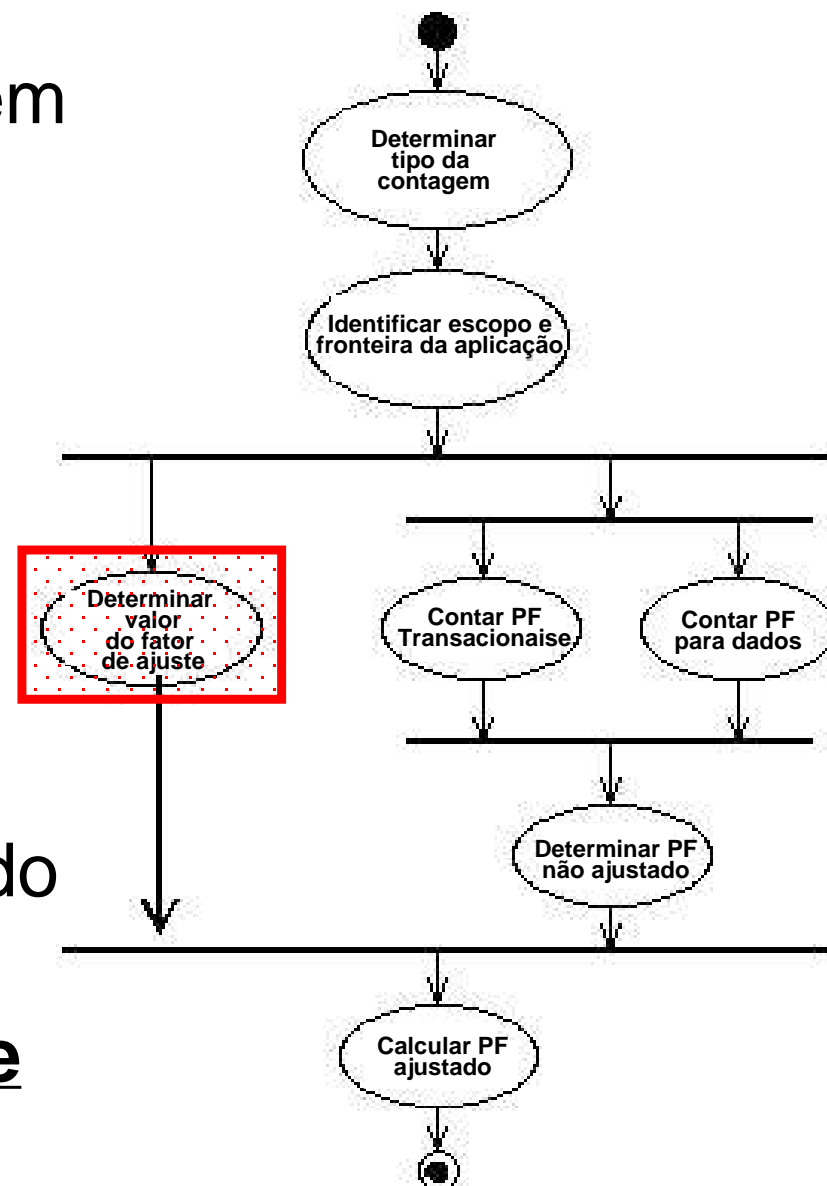
- ➡ Calculam-se os números de entradas, saídas, consultas, arquivos e interfaces do sistema
- ➡ Multiplica-se cada um desses números por um peso, de acordo com a complexidade do sistema e soma-se o resultado

Tabela Principal

		Complexidade			
Medida	Contagem Total	Simples	Médio	Complexo	Total
Entradas		×3	×4	×6	=
Saídas		×4	×5	×7	=
Consultas		×3	×4	×6	=
Arquivos		×7	×1 0	×15	=
Interfaces		×5	×7	×10	=
				Total	

Visão Geral da APF

- ➔ Determinar Tipo da Contagem
- ➔ Identificar o Escopo de Contagem e a Fronteira da Aplicação
- ➔ Contar PFs Transacionais
- ➔ Contar PFs para Dados
- ➔ Determinar PFs Não-Ajustado
- ➔ Determinar Fator de Ajuste
- ➔ Calcular PFs Ajustado



Perguntas de Ajuste

➡ São 14 as perguntas que devem ser feitas e ajudaram a determinar a quantidade de PF relativa a um sistema. Cada uma deve ser respondida com um número, de 0 a 5, indicando a importância da característica que se pergunta sobre o sistema, da seguinte forma:

Valores da Resposta

- ➡ 0 - Não tem influência
- ➡ 1 - Influência incidental
- ➡ 2 - Influência moderada
- ➡ 3 - Influência média
- ➡ 4 - Influência significativa
- ➡ 5 - Influência essencial em todo o sistema

Resposta

- ➡ Para cada pergunta, o padrão IFPUG determina tipos de respostas padronizadas que nos permitem dar a resposta (entre 0 e 5) mais facilmente, como é exemplificado no item 1 (Comunicação de Dados).
- ➡ Foge ao objetivo desse curso fornecer um detalhamento completo do padrão de contagem, que pode ser obtido junto ao IFPUG.

Pergunta 1 e suas Respostas

➡ Quantas facilidades de comunicação existem para facilitar a transferência ou troca de informação com a aplicação ou sistema?

- ➡ Aplicação em batch ou computador isolado: 0
- ➡ Aplicação em batch com entrada ou (exclusivo) impressão remota: 1
- ➡ Aplicação em batch com entrada e impressão remota: 2
- ➡ A aplicação é um *front-end* que necessita de executar coleta de dados ou teleprocessamento para um sistema de fazer o processamento em batch ou de consultas: 3
- ➡ A aplicação é mais que um *front-end*, porém só executa um tipo de protocolo de teleprocessamento: 4
- ➡ A aplicação é mais que um *front-end* e executa vários protocolos de teleprocessamento: 5

Outras Perguntas

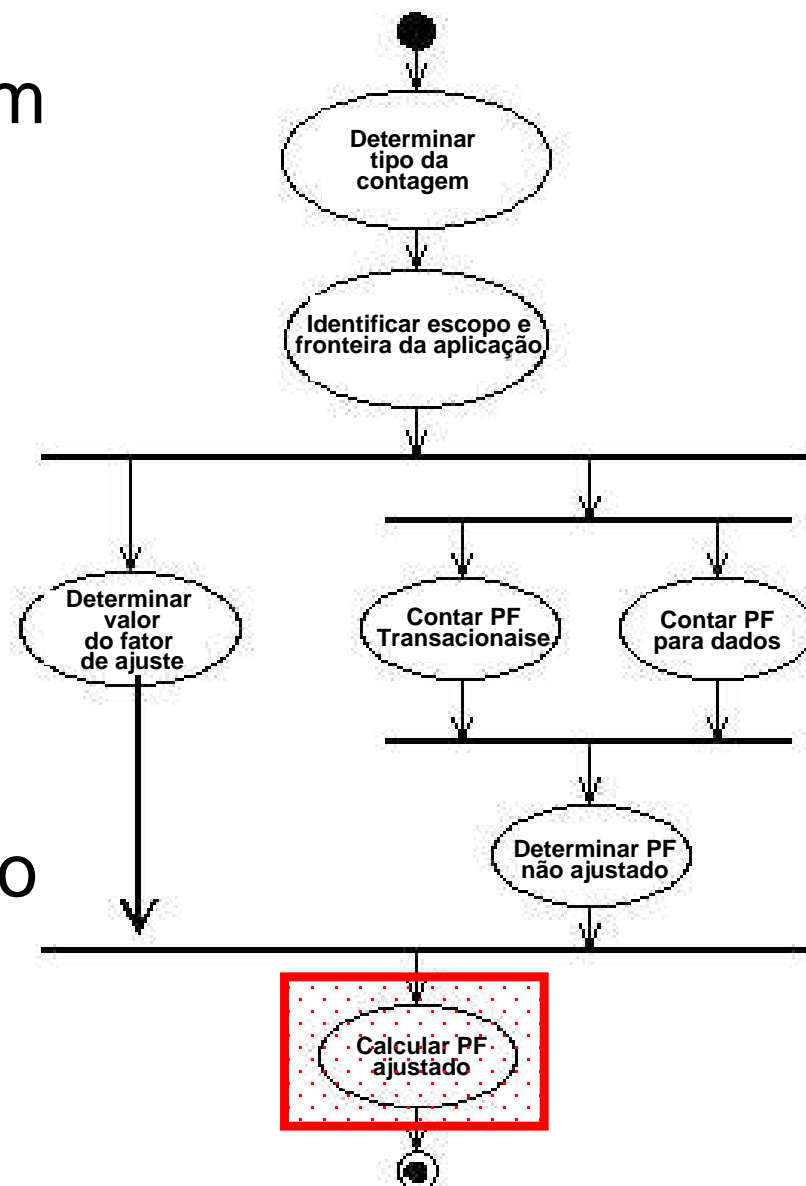
- ➡ Como será tratada a distribuição de dados e processamento?
- ➡ O usuário exige tempos de resposta ou throughput, ou seja, o desempenho é crítico?
- ➡ Quão fortemente é utilizada a plataforma (hardware) onde a aplicação vai ser executada?
- ➡ Qual a frequência das transações (diárias, semanais, altas o suficiente para exigir um estudo de desempenho)?
- ➡ Que percentagem das informações é inserida on-line?
 - ▬ Se mais de 30% das transações forem entradas de dados interativas, a nota é 5.
- ➡ A aplicação é projetada para eficiência para o usuário final?
- ➡ Quantas ILFs são alteradas por transações on-line?

Outras Perguntas

- ➡ A aplicação tem processamento lógico ou matemático extensivo?
- ➡ A aplicação é desenvolvida para atender um ou muitos tipos de usuários diferentes?
- ➡ Qual a dificuldade de conversão e instalação?
- ➡ Qual a eficiência e grau de automação de inicialização, backup e recuperação?
- ➡ A aplicação foi especialmente projetada, desenvolvida e suportada para funcionar em locais diferentes em diferentes organizações?
- ➡ A aplicação foi especialmente projetada, desenvolvida e suportada para facilitar mudanças?

Visão Geral da APF

- ➔ Determinar Tipo da Contagem
- ➔ Identificar o Escopo de Contagem e a Fronteira da Aplicação
- ➔ Contar PFs Transacionais
- ➔ Contar PFs para Dados
- ➔ Determinar PFs Não-Ajustado
- ➔ Determinar Fator de Ajuste
- ➔ **Calcular PFs Ajustado**



Cálculo dos Pfs Finais

➡ calcula-se o número de pontos de função com a equação:

➡
$$PF = PF \text{ Basicos} \times (0,65 + 0,01 \times \sum (\text{respostas as perguntas})))$$

Análise de Pontos de Função

➡ Permite dar tamanho ao software

➡ Mas, e o Esforço?

Produtividade por PF

➡ É possível manter, na empresa, um histórico de produtividade por pontos de função

- ➡ Possivelmente parametrizado
 - ➡ Tipo de projeto
 - ➡ Área de Aplicação
 - ➡ Linguagem de Programação

Linguagem	SLOC/FP			
	Média	Mediana	Mais Baixo	Mais Alto
Access	35	38	15	47
ASP	69	62	32	127
Assembler	172	157	86	320
C	148	104	9	704
C++	60	53	29	178
C#	59	59	51	66
Clipper	38	39	27	70
COBOL	73	77	8	400
Excel	47	46	31	63
J2EE	61	50	40	60
Java	60	59	14	97
Lotus Notes	21	22	15	25
Oracle	38	29	4	122
Oracle Dev 2K/FORMS	41/42	30	21/23	100
Powerbuilder	30	24	7	105
SQL	39	35	15	143
Visual Basic	50	42	14	276

Aula 32

Professor:

Geraldo Xexéo
UFRJ

Conteúdo:

**FIM: Análise de Pontos
de Função**