

---

## **Parte 2 – A ponte – Aplicando o Método FSM do IFPUG**

---

---

Página intencionalmente deixada em branco.

---

## Parte 2 A Transição - Aplicando o Método de Medição de Tamanho Funcional do IFPUG

---

**Introdução** A Parte 1 fornece o processo de análise de ponto de função para medir funcionalidade de software de acordo com o método do IFPUG bem como regras detalhadas para identificar e medir as funções de dados e de transações.

A Parte 2 fornece uma visão geral do método do IFPUG, conjuntamente com orientações para aplicar as regras de determinação do tipo da contagem, estabelecimento da fronteira da aplicação e medição de funções de dados e de transações.

**Conteúdo** A Parte 2 inclui os seguintes capítulos:

<b>Tópico</b>	<b>Página</b>
Relação entre IFPUG e ISO	1-2
Visão Geral do Método FSM do IFPUG	2-1
Obter Documentação Disponível	3-1
Determinar Tipo da Contagem	4-1
Determinar Escopo da Contagem e Fronteira da Aplicação	5-1
Medir Funções de Dados	6-1
Medir Funções Transação	7-1
Índice	i-1

## Relação entre IFPUG e ISO

### **Diretriz Estratégica do IFPUG**

O método análise de pontos de função do IFPUG é um padrão ISO e deve ser aderente à ISO/IEC 14143-1:2007. O método pode medir apenas “tamanho funcional” e não “tamanho não-funcional”. Isto não significa que o tamanho não funcional não possa ou não deva ser medido, apenas deve ser tratado como uma medida separada (“A Framework for Functional Sizing” [IFPUG, 2003]).

### **“A Framework for Functional Sizing”**

“A Framework for Functional Sizing” fornece a base para o entendimento do tamanho funcional e como ele se relaciona com os requisitos. O artigo explora diferentes tipos de requisitos; estes conceitos fornecem a base para a medição de software de acordo com a ISO/IEC 14143-1 e o CPM do IFPUG.

A essência de “A Framework for Functional Sizing” é que pode haver vários métodos de medição para diferentes propósitos. O tamanho funcional pode ser medido usando o método de medição funcional do IFPUG de análise de pontos de função, baseado nos requisitos funcionais do usuário. Outras medidas de tamanho podem ser usadas para medir, por exemplo, requisitos não funcionais.

Ambos resultam em medidas distintas de tamanho, representando diferentes dimensões do tamanho do software: IFPUG-PF para tamanho funcional e alguma outra para tamanho não funcional. Embora estes tamanhos não possam ser adicionados, pois representam dimensões distintas (como volume e temperatura de uma sala), ambos podem ser usados na estimativa de esforço para desenvolvimento de uma aplicação ou sistema.

“A Framework for Functional Sizing” fornece orientações para distinguir tamanho funcional de tamanho não funcional.

## **ISO/IEC 14143-1 - Definição de Requisitos do Usuário**

Em 1998 o primeiro padrão de Medição de Tamanho Funcional ISO/IEC foi publicado (ISO/IEC 14143-1:1998). Este padrão define Tamanho Funcional como “um tamanho do software obtido através da quantificação dos Requisitos Funcionais do Usuário”. Ele foi atualizado em 2007 e publicado como ISO/IEC 14143-1:2007.

A ISO/IEC 14143-1 define os conceitos fundamentais de Medição de Tamanho Funcional (FSM) e descreve os princípios gerais para a aplicação do método FSM. Ele NÃO fornece regras detalhadas sobre como:

- Selecionar um método específico
- Medir Tamanho Funcional de software usando um método específico
- Usar os resultados obtidos de método específico

A definição de FSM na ISO/IEC 14143-1 é aplicada para se determinar se um método de medição de software é um Método de Medição Funcional. Ele não impede o desenvolvimento de vários métodos, em vez disso, oferece o embasamento para a avaliação se um método específico é aderente ao FSM.

A ISO/IEC 14143-1 classifica os requisitos do usuário em dois subconjuntos:

- Requisitos Funcionais do Usuário
- Requisitos Não-Funcionais do Usuário

As definições da ISO/IEC 14143-1 estão listadas a seguir:

## ISO/IEC 14143-1 - Definições

<b>Tamanho Funcional</b>	Um tamanho de software obtido através da quantificação dos Requisitos Funcionais do Usuário.
<b>Requisito Funcional do Usuário</b>	<p>Um subconjunto dos requisitos do usuário que descrevem o que o software deve fazer, em termos de tarefas e serviços.</p> <p>Nota: Requisitos funcionais do usuário incluem, mas não estão limitados a:</p> <ul style="list-style-type: none"><li>• Transferência de dados (por exemplo: entrada de dados de cliente, envio de sinais de controle)</li><li>• Transformação de dados (por exemplo: calcular taxa de juros bancária, calcular temperatura média)</li><li>• Armazenamento de dados (por exemplo: armazenar dados de cliente, registrar a mudança de temperatura ao longo do tempo)</li><li>• Recuperação de dados (por exemplo: listar os empregados atuais, recuperar posição da aeronave)</li></ul>

**Requisito  
Não-  
Funcional do  
Usuário**

A ISO não oferece definição para Requisito Não-Funcional do Usuário, mas apresenta alguns exemplos em uma nota.

Exemplos de requisitos do usuário que são Requisitos Não-Funcionais do Usuário incluem, mas não estão limitados aos seguintes:

- Restrições de qualidade (por exemplo, usabilidade, confiabilidade, eficiência e portabilidade)
- Restrições Organizacionais (por exemplo, locais de operação, hardware alvo e aderência a padrões)
- Restrições Ambientais (por exemplo, interoperabilidade, segurança, privacidade e sigilo)
- Restrições de Implementação (por exemplo, linguagem de desenvolvimento, cronograma de entrega)

---

### Visão Geral do Método FSM do IFPUG

---

**Introdução** Este capítulo apresenta uma visão geral do Método de Medição de Tamanho Funcional (FSM) do IFPUG. Também apresenta um exemplo simples dos procedimentos de contagem de pontos de função.

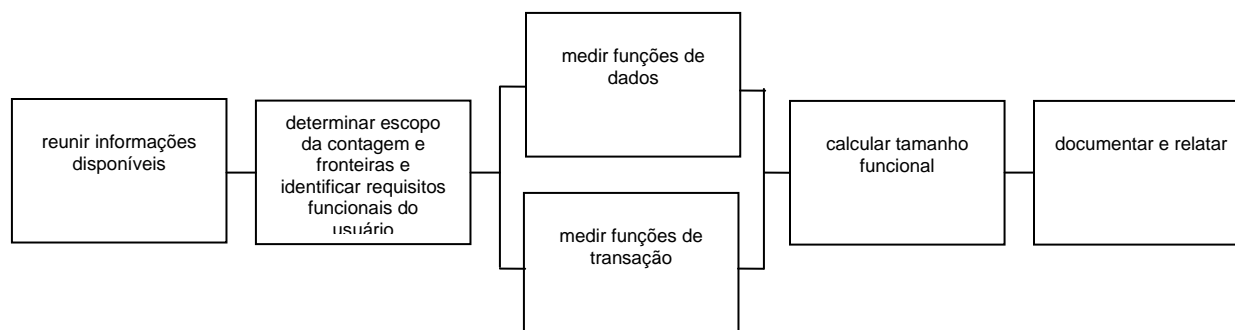
**Conteúdo** Este capítulo contempla as seguintes seções:

<b>Tópico</b>	<b>Página</b>
<b>Procedimento do Método de Medição de Tamanho Funcional</b>	<b>2-2</b>
Procedimento por Capítulo	2-2
Exemplo Simples de Contagem	2-3
Diagrama Simplificado	2-3
Obter Documentação Disponível	2-4
Determinar o Escopo da Contagem e Fronteira e Identificar os Requisitos Funcionais do Usuário	2-4
Medir Funções de Dados	2-5
Medir Funções de Transação	2-6
Calcular o Tamanho Funcional	2-7

# Procedimento do Método de Medição de Tamanho Funcional

Esta seção apresenta um procedimento de alto nível para o Método de Medição de Tamanho Funcional do IFPUG.

## Diagrama do Procedimento



## Procedimento por Capítulo

A tabela seguinte apresenta o procedimento de contagem de pontos de função, que é explicado nos capítulos restantes da Parte 2.

**Nota:** Um exemplo simples do procedimento de contagem é apresentado nas páginas seguintes deste capítulo.

Procedimento	Capítulo
Obter Documentação Disponível	3 Obter Documentação Disponível
Determinar o Escopo da Contagem e Fronteira e Identificar Requisitos Funcionais do Usuário	4 Determinar Tipo da Contagem
Determinar o Escopo da Contagem e Fronteira e Identificar Requisitos Funcionais do Usuário	5 Determinar o Escopo da Contagem e Fronteira e Identificar Requisitos Funcionais do Usuário
Medir Funções de Dados	6 Medir Funções de Dados
Medir Funções Transação	7 Medir Funções Transação

**Nota:** Não existem capítulos específicos para Calcular o Tamanho Funcional e Documentar e Relatar, pois estes procedimentos não necessitam de explicação adicional além da que foi fornecida na Parte 1.

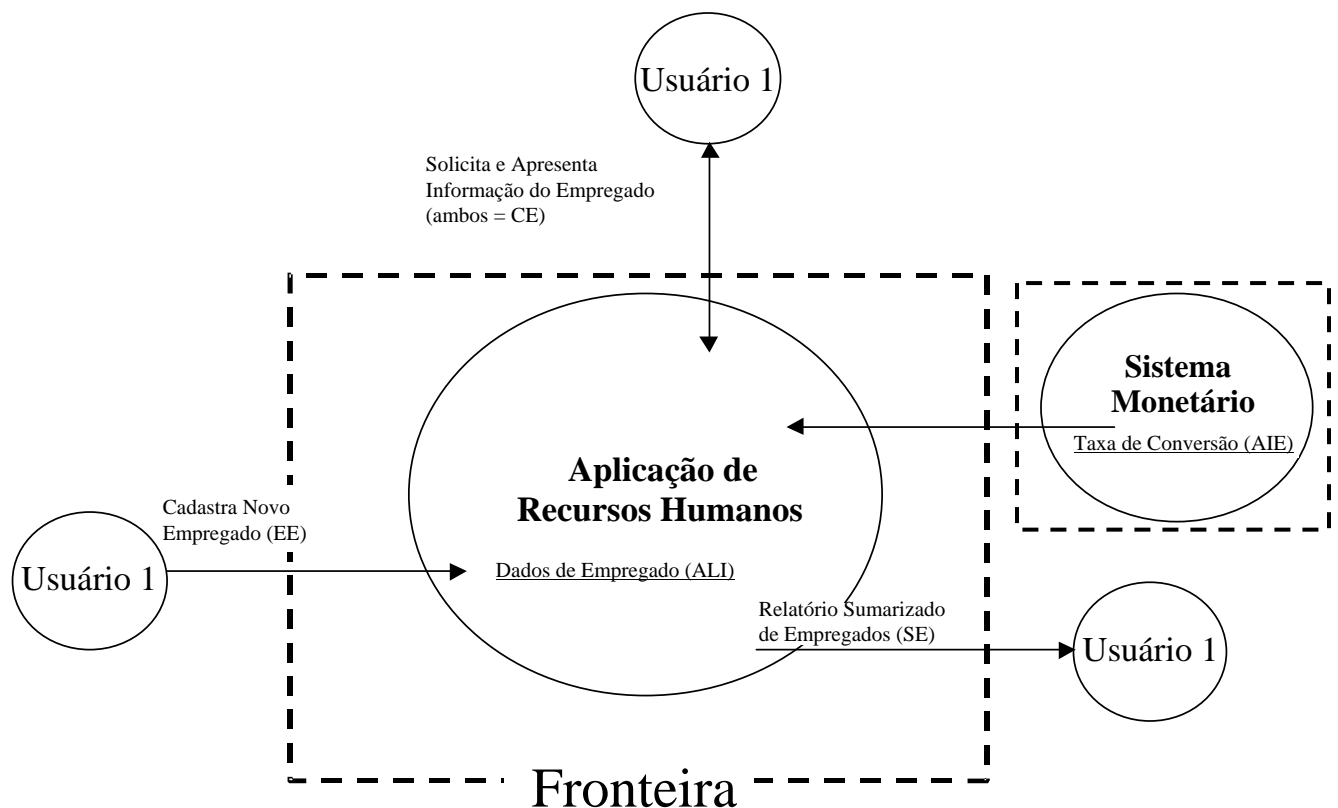


## Exemplo Simples de Contagem

Esta seção apresenta um exemplo simples do procedimento de contagem de pontos de função e os componentes do tamanho funcional.

### Diagrama Simplificado

O diagrama seguinte apresenta os componentes para o exemplo de contagem de uma Aplicação de Recursos Humanos. Use o diagrama como referência enquanto estiver lendo os parágrafos restantes deste capítulo.



## Obter Documentação Disponível

O primeiro passo do procedimento de contagem de pontos de função é obter a documentação disponível, de acordo com 5.2 da Parte 1, para sustentar a medição funcional de tamanho. Ela deve descrever a funcionalidade entregue pelo software ou a funcionalidade que é impactada pelo projeto de software que está sendo medido.

Uma documentação adequada pode incluir requisitos, modelos de dados/objetos, diagramas de classe, diagramas de fluxo de dados, casos de uso, descrições procedurais, layout de relatórios e telas, manuais de usuário e outros artefatos do desenvolvimento de software. Se não há documentação suficientemente disponível, deve-se buscar o acesso aos especialistas no negócio para cobrir as lacunas da documentação.

O capítulo 3 discute a documentação disponível durante o ciclo de vida de uma aplicação.

## Determinar o Escopo da Contagem e Fronteira e Identificar os Requisitos Funcionais do Usuário

Uma medição de tamanho funcional é feita para responder a uma questão de negócio. De acordo com 5.3 da Parte 1, é a questão de negócio que determina o propósito da contagem.

De acordo com o seu propósito, as contagens de pontos de função podem ser identificadas da seguinte maneira:

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

O Capítulo 4 fornece orientações para terminar o tipo da contagem de pontos de função.

O escopo da contagem define o conjunto dos Requisitos Funcionais do Usuário que serão incluídos na contagem de pontos de função.

A fronteira é uma interface conceitual entre o software em análise e seus usuários.

O diagrama simplificado anterior apresenta a fronteira da aplicação entre a Aplicação de Recursos Humanos (que está sendo medida) e o Sistema Monetário (externo). Também apresenta a fronteira da aplicação da Aplicação de Recursos Humanos e seus usuários.

O Capítulo 5 discute mais sobre escopo da contagem e fronteira da aplicação.

## Medir Funções de Dados

Uma função de dados representa a funcionalidade fornecida ao usuário para atender suas necessidades internas e externas de armazenamento de dados. Uma função de dados pode ser um arquivo lógico interno ou um arquivo de interface externa.

- Um arquivo lógico interno (ALI) é um grupo de dados ou informações de controle, reconhecido pelo usuário e mantido dentro da fronteira da aplicação sendo medida. A principal intenção de um ALI é armazenar dados mantidos por um ou mais processos elementares da aplicação sendo medida.

O diagrama simplificado anterior apresenta um grupo de dados relacionado a empregado mantido dentro da Aplicação de Recursos Humanos como um exemplo de Arquivo Lógico Interno.

- Um arquivo de interface externa (AIE) é um grupo de dados ou informações de controle, reconhecido pelo usuário, e que é apenas referenciado pela aplicação sendo medida, mas que são mantidos dentro da fronteira de outra aplicação. A principal intenção de um AIE é armazenar dados referenciados por um ou mais processos elementares da aplicação sendo medida. Isto significa que um AIE contado para uma aplicação deve ser um ALI em alguma outra aplicação.

O diagrama simplificado anterior mostra informação de taxa de conversão mantida pelo Sistema Monetário e que é referenciado pela Aplicação de Recursos Humanos como um exemplo de Arquivo de Interface Externa.

O Capítulo 6 da Parte 2 discute o uso das regras para medir funções de dados.

A Parte 3 contém orientações adicionais para medir funções de dados.

A Parte 4 contém exemplos que ilustram o uso das regras das funções de dados.

## Medir Funções de Transação

Uma função de transação é um processo elementar que fornece funcionalidade ao usuário para processamento de dados. Uma função de transação pode ser uma entrada externa, saída externa ou consulta externa.

- Uma entrada externa (EE) é um processo elementar que processa dados ou informações de controle recebidos de fora da fronteira da aplicação. A intenção primária de uma EE é manter um ou mais ALIs e/ou alterar o comportamento do sistema.

O diagrama simplificado anterior mostra o processo de cadastrar um novo empregado na Aplicação de Recursos Humanos como um exemplo de uma Entrada Externa.

- Uma saída externa (SE) é um processo elementar que envia dados ou informações de controle para fora da fronteira da aplicação e inclui processamento adicional além daquele existente em uma consulta externa. A intenção primária de uma saída externa é apresentar dados ao usuário através de lógica de processamento que não seja apenas recuperação de dados ou informação de controle. A lógica de processamento deve contar ao menos uma fórmula matemática ou cálculo, e/ou criar dados, e/ou manter um ou mais ALIs, e/ou alterar o comportamento do sistema.

O diagrama simplificado anterior mostra o processo de gerar um relatório que sumariza todos os empregados da Aplicação de Recursos Humanos como um exemplo de Saída Externa.

- Uma consulta externa (CE) é um processo elementar que envia dados ou informações de controle para fora da fronteira da aplicação. A intenção primária de uma consulta externa é apresentar dados ao usuário através de recuperação de dados ou informação de controle. A lógica de processamento não contém fórmula matemática, nem cálculo, nem cria dados derivados. Nenhum ALI é mantido durante o processamento, nem o comportamento do sistema é alterado.

O diagrama simplificado anterior mostra o processo de solicitar dados de empregado como um exemplo de Consulta Externa.

O Capítulo 7 da Parte 2 discute o uso das regras para medir funções de transação.

A Parte 3 contém orientações adicionais para medir funções de transação.

A Parte 4 contém exemplos que ilustram o uso das regras para medir funções de transação.

## Calcular o Tamanho Funcional

O tamanho funcional representa o tamanho do software obtido pela quantificação dos requisitos funcionais do usuário.

A funcionalidade específica da aplicação do usuário é avaliada em termos do que é entregue, não *como* é entregue. Apenas componentes solicitados e definidos pelo usuário são contados.

O tamanho funcional é obtido através da medição das funções de dados e de transação. Estas funções são detalhadas em tipo de funções descritas nos parágrafos seguintes.

Alguns indivíduos podem usar o valor do fator de ajuste (VAF), que considera 14 características gerais de sistema (CGSs). Para orientação no uso do VAF e das CGSs, consulte o Apêndice C.

Esta página foi deixada em branco intencionalmente.

---

### Obter Documentação Disponível

---

**Introdução** Este capítulo apresenta a documentação comumente disponível durante o ciclo de vida de uma aplicação e a importância do papel do usuário.

**Conteúdo** Este capítulo inclui:

<b>Tópico</b>	<b>Página</b>
<b>Visão do Usuário</b>	<b>3-2</b>
<b>Documentação Disponível Durante o Ciclo de Vida de uma Aplicação</b>	<b>3-3</b>
Fase: Requisitos Iniciais do Usuário	3-4
Fase: Requisitos Técnicos	3-5
Fase: Requisitos Funcionais Finais do Usuário	3-6
<b>Comparações entre as Fases do Ciclo de Vida</b>	<b>3-8</b>
<b>Documentação Útil do Projeto da Aplicação</b>	<b>3-9</b>
<b>Tamanho Funcional</b>	<b>3-10</b>

## Visão do Usuário

Um *usuário* é qualquer pessoa ou coisa que se comunica ou interage com o software a qualquer momento.

A visão do usuário é o Requisito Funcional do Usuário *como percebido* pelo usuário.

Requisitos Funcionais do Usuário são um subconjunto dos requisitos do usuário que descrevem o que o software deverá fazer em termos de tarefas e serviços.

A visão do usuário representa uma descrição formal das necessidades dos negócios do usuário, na linguagem do usuário. Os desenvolvedores traduzem a informação do usuário para informações em linguagem técnica a fim de prover uma solução.

A visão do usuário:

- É uma descrição das funções do negócio
- Pode ser feito por declaração verbal pelo usuário através de seu ponto de vista
- É aprovada pelo usuário
- Pode ser usada para medir o tamanho funcional
- Pode variar na forma física (ex., catálogo de transações, propostas, documento de requisitos, especificações externas, especificações detalhadas, manuais do usuário)

Uma medição de tamanho funcional é realizada utilizando a informação em uma linguagem que é comum para o usuário(s) e desenvolvedores.



## Documentação Disponível Durante o Ciclo de Vida de uma Aplicação

Os requisitos do usuário evoluem rapidamente nas fases iniciais de um projeto. Os usuários e desenvolvedores devem decidir, de comum acordo, quais funções deverão ser incluídas em uma aplicação. Estas decisões a respeito das funções de um projeto podem ser influenciadas por:

- Necessidades da organização
- Riscos (de negócios e técnicos) associados ao projeto
- Recursos disponíveis (ex. orçamento, pessoal) para o projeto
- Tecnologia disponível na organização
- Influência de outros usuários ou desenvolvedores através de comentários e sugestões.

No começo de um projeto é produzido o estudo de viabilidade. Ele é a especificação de nível mais alto e é normalmente muito curto; por exemplo:

- A organização precisa de uma aplicação para se adaptar a uma nova legislação sobre impostos
- A organização precisa de uma aplicação para administrar estoques de maneira mais eficiente
- A organização precisa de uma aplicação para administrar recursos humanos de maneira mais eficiente

Depois do estudo de viabilidade, o usuário desenvolve requisitos que se tornam mais precisos com o passar do tempo. Em algum momento, o usuário trocará idéias com os desenvolvedores para criar os requisitos detalhados. Os desenvolvedores de software podem antecipar seu trabalho de desenvolvimento e implementação dos requisitos com base no estudo de viabilidade. As conversas entre usuários e desenvolvedores de software levam a requisitos mais refinados. O processo de desenvolvimento varia de acordo com as diferentes organizações. Este manual irá considerar, para fins ilustrativos, um modelo com três categorias de documentos de requisitos:

- Requisitos Iniciais do Usuário
- Requisitos Técnicos Iniciais
- Requisitos Funcionais Finais

Assim como em outras metodologias de desenvolvimento, a Fase de Requisitos Funcionais Finais é fase mais precisa para a medição de tamanho funcional.

## Fase: Requisitos Iniciais do Usuário

Esta fase representa os requisitos dos usuários antes das sessões entre os usuários e os desenvolvedores de software. Pode ter uma ou mais das características abaixo:

- Incompleta

Por exemplo, Nos Requisitos Iniciais do Usuário podem faltar funções necessárias à integridade referencial

- Falta de funcionalidades "utilitárias"

Por exemplo, relatórios de validação essenciais ou consultas podem estar faltando

- Impossibilidade de implementação ou uso muito difícil

Por exemplo, um usuário pode pedir uma consulta on-line que requeira uma hora de processamento de CPU

- Muito genérica

Por exemplo, os requisitos podem não incluir a lista específica de campos de reconhecimento do usuário

- Não atender às necessidades para todos os usuários da aplicação

Por exemplo, os requisitos de um projeto específico podem variar de um usuário para outro, se não tiverem as mesmas necessidades funcionais

- Requisitos definidos sem considerar as fronteiras de aplicação

Por exemplo, fronteiras da aplicação futura e/ou atual podem não estar sendo consideradas

- Expressos em um contexto diferente ou em uma terminologia não compatível com a análise de pontos de função

Por exemplo: os Requisitos Iniciais do Usuário podem fazer referência ao aspecto físico ou manual do sistema.

### Exemplo

No departamento de RH de uma organização, um usuário expressa seu requisito assim:

“Sempre que eu estou trabalhando com um funcionário, quero poder ver as informações do funcionário informando o seu nome”.

Este requisito implica no desenvolvimento de uma tela de consulta e de um grupo de dados de funcionário.

Exemplo de Funcionalidades dos Requisitos Inicial do Usuário:

CE     consulta de um funcionário específico

ALI    grupo de dados de funcionário

## Fase: Requisitos Técnicos Iniciais

Esta segunda fase representa a visão dos desenvolvedores de software sobre os requisitos criados a partir do estudo de viabilidade. Um trabalho dos desenvolvedores de software, dentre outros, é organizar os requisitos dentro das aplicações existentes, se existirem. Os Requisitos Técnicos Iniciais podem incluir elementos necessários para a implementação, mas não são utilizados na medição de tamanho funcional (ex.: arquivos temporários, índices, etc.). Esta fase pode ter uma ou mais das características abaixo:

- Dependência tecnológica

Por exemplo: os arquivos físicos variam com base no ambiente de banco de dados.

- Terminologia não familiar com os usuários

Por exemplo, desenvolvedores de software podem se referir aos arquivos físicos ao invés de grupos lógicos de dados.

- Funcionalidades podem ser determinadas enfatizando restrições técnicas

Por exemplo: alguns desenvolvedores tendem a limitar o escopo dos requisitos pelo foco na capacidade computacional (CPU) disponível no momento na organização.

- Fronteiras são determinadas de acordo com a arquitetura técnica ao invés de processos do negócio

Por exemplo: pode haver requisitos técnicos separados para cliente e servidor, mas ambos devem ser considerados na mesma fronteira de aplicação quando se estiver medindo o tamanho funcional.

### Exemplo

Desenvolvedor: “Eu reconheço a necessidade de uma consulta de funcionários. Um índice é necessário para acelerar a busca de funcionários específicos”.

As funções dos Requisitos Técnicos Iniciais podem ser identificadas como:

CE      consulta de um funcionário específico

ALI     grupo de dados de funcionário

ALI\*   índice do arquivo de funcionário

\*Arquivos de índices não são incluídos na medição de tamanho funcional. Neste exemplo, o arquivo de índice foi incorretamente identificado como um ALI para ilustrar um erro potencial na contagem por parte dos desenvolvedores de software.

## Fase: Requisitos Funcionais Finais

Esta terceira fase dos requisitos origina-se de sessões conjuntas entre o(s) usuário(s) e o(s) desenvolvedor(es). As sessões conjuntas são necessárias para tornar os requisitos funcionais consistentes e completos para a aplicação. Esta fase é a versão final dos requisitos funcionais do usuário para a aplicação antes do início da fase de desenvolvimento e tem as seguintes características:

- Terminologia que pode ser entendida tanto pelos usuários quanto pelos desenvolvedores de software
- Descrições integradas de todos os requisitos do usuário, incluindo requisitos de todos os grupos de usuários existentes
- Todos os processos de negócio são completamente definidos, incluindo toda ação do usuário, campos entrando e saindo da fronteira de aplicação, fontes de dados são definidas por cada processo de negócio, e as validações que ocorram como parte de cada processo de negócio
- Cada processo e grupo de dados é aprovador por usuário e desenvolvedor
- A viabilidade e utilidade são aprovadas pelos desenvolvedores de software

### Exemplo

Usuário: “Sempre que eu estou trabalhando com um funcionário, quero poder ver as informações dos funcionários informando seu nome.”

Desenvolvedor: “Reconheço a necessidade de consulta de funcionários, mas muitos funcionários podem ter o mesmo nome. Não é possível especificar um funcionário individualmente através de seu nome; por esta razão, sugiro uma lista de funcionários on-line (nome, localização e número da previdência social), através da qual seja possível selecionar um funcionário. Será necessário um índice para acelerar a recuperação de um funcionário específico”.

Usuário: “Concordo que a lista de seleção de funcionários é necessária neste caso, e isto também pode ser usado para outros propósitos além da seleção de funcionário”.

Resultado desta conversa entre o usuário e o desenvolvedor:

- Incluir uma lista on-line de funcionários nos requisitos funcionais do usuário e no tamanho funcional
- Excluir o índice de funcionários da contagem de pontos de função já que esta é uma solução técnica

Funções do Exemplo de Requisitos Funcionais Finais:

CE     consulta a um específico funcionário  
CE     lista on-line de funcionários  
ALI    grupo de dados de funcionário

O documento de Requisitos Funcionais Finais é a versão final dos requisitos antes de iniciar a fase de desenvolvimento. Neste momento, deverá haver concordância quanto aos requisitos documentados estarem concluídos, formalizados e aprovados. A medição de tamanho funcional, assumindo que não haja nenhuma mudança adicional no escopo, deverá ser consistente com a medição na conclusão do desenvolvimento.

## Comparação das Fases do Ciclo de Vida

Antes de iniciar uma medição de tamanho funcional, determine a fase do ciclo de vida da aplicação e se você vai fazer uma aproximação, ou uma medição. Documente qualquer suposição.

Uma aproximação permite fazer suposições sobre funções desconhecidas e/ou suas complexidades, para determinar um tamanho funcional aproximado.

Uma medição inclui a identificação de todas as funções e suas complexidades, para efetuar uma análise de pontos de função.

Num primeiro estágio, os Requisitos Iniciais do Usuário podem ser o único documento disponível para a análise de pontos de função. Apesar das desvantagens, este tamanho pode ser muito útil para produzir uma estimativa antecipada. A utilização da análise de pontos de função para obter aproximações nas várias fases do ciclo de vida é apresentada a seguir:

Fase do Ciclo de Vida	O tamanho pode ser aproximado	O tamanho pode ser medido
<b>Proposta:</b> usuários expressam necessidades e intenções	sim	não
<b>Requisitos:</b> desenvolvedores e usuário revisam e concordam quanto às necessidades e intenções do usuário	sim	sim
<b>Projeto:</b> os desenvolvedores podem incluir elementos para implementação que não são usados pela análise de pontos de função	sim	sim
<b>Construção</b>	sim	sim
<b>Entrega</b>	sim	sim
<b>Manutenção</b>	sim	sim
<p><b>Nota:</b> Não foi assumido nenhum ciclo de vida específico. Se utilizar uma abordagem iterativa, você deve esperar uma aproximação do tamanho durante boa parte do ciclo de vida de desenvolvimento.</p> <p>Esteja certo de estar medindo somente requisitos novos ou refinados de acordo com as necessidades e intenções do usuário.</p>		

## Utilidade do Projeto/Documentação da Aplicação

Em geral, os itens a seguir são úteis quando se faz alguma medição de tamanho funcional:

- Documentos de requisitos
- Diagrama de entidades
- Modelos de objetos
- Modelos de dados
- Arquivo e esquemas banco de dados (com lógica, atributos necessários do usuário identificados)
- Acordos de Interface com descrições de entradas em lote/arquivos de transação e interfaces de/para outras aplicações
- Exemplos de relatórios, telas online e outras interfaces de usuário
- Demonstração da operação de aplicação
- Uma ou mais especialistas na aplicação (para a aplicação que está sendo medida)
- Um ou mais clientes/usuários de aplicação (passíveis de serem consultados durante o processo de medição)
- Guia de usuário, manual de treinamento e ajuda da aplicação
- Documentação do projeto do sistema
- Especificações funcionais
- Casos de uso

**Nota:** Esta lista acima não é exaustiva.

## Tamanho Funcional

O CPM do IFPUG foi transformado em padrão ISO para a medição de tamanho funcional, com a exclusão das Características Gerais do Sistema, que medem requisitos não funcionais (técnicos e de qualidade). Até certo ponto, esta transformação permitiu que o Comitê de Práticas de Contagem tratasse consistentemente itens como Dados de Código. A consideração mais importante em relação a estas questões é como os requisitos não funcionais afetam o tamanho. Como não são parte do tamanho funcional, não contribuem para o tamanho funcional. Entretanto, ainda são parte de todos os requisitos (funcional e não funcional) para o software, contribuindo então para o tamanho do mesmo.

Para uma discussão detalhada sobre tamanho funcional e como ele irá tanto conduzir como restringir a evolução da análise de pontos de função, consulte o documento do IFPUG Framework for Functional Sizing<sup>1</sup>.

---

<sup>1</sup> Framework for Functional Sizing, IFPUG CPC, Release 1.0, September 2003



---

### Determinar o Tipo de Contagem

---

**Introdução** Este capítulo inclui uma explicação detalhada dos tipos de contagem: projeto de desenvolvimento, projeto de melhoria, e aplicação.

**Conteúdo** O Capítulo inclui as seguintes seções:

<b>Tópico</b>	<b>Página</b>
<b>Definições: Tipos de Contagem de Pontos de Função</b>	<b>4-2</b>
Projeto de Desenvolvimento	<b>4-2</b>
Projeto de Melhoria	4-2
Aplicação	4-3
<b>Diagrama dos Tipos de Contagem</b>	<b>4-4</b>
Medidas Estimadas e Finais de Tamanho Funcional	4-4

## Definições: Tipos de Contagem de Pontos de Função

O tamanho funcional pode ser medido tanto para projetos quanto 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

O parágrafo seguinte define cada tipo de contagem de ponto de função.

**Nota:** Para aqueles indivíduos que aplicam um Fator de Ajuste (VAF), consulte no Apêndice C as fórmulas para calcular o VAF e a contagem de pontos de função ajustada.

### Projeto de Desenvolvimento

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

O tamanho funcional do projeto de desenvolvimento é uma medida de funcionalidade oferecida aos usuários com a primeira instalação do software, conforme medido pela contagem de pontos de função do projeto de desenvolvimento pela atividade de aplicação, o método de medição funcional (FSM) IFPUG.

### Projeto de Melhoria

Um projeto de melhoria é um projeto para desenvolver e entregar manutenção adaptativa. 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.

Orientações adicionais estão incluídas na Parte 3.

## Aplicação

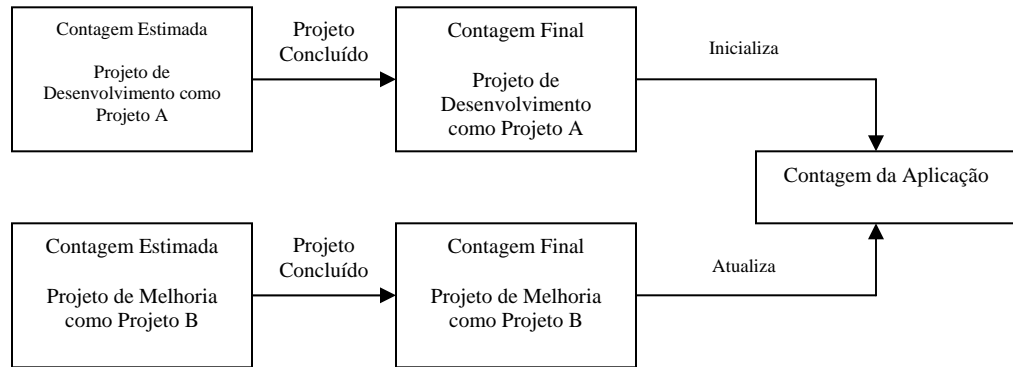
Uma aplicação é uma coleção coesa de procedimentos automatizados e dados apoiando um objetivo de negócio; isto consiste em um ou mais componentes, módulos, ou subsistemas.

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.

## Diagrama dos Tipos de Contagem

O diagrama a seguir ilustra os tipos de contagem de pontos de função e seus relacionamentos. (O Projeto A é completado primeiro, seguido do Projeto B.)



## Medidas Estimadas e Finais de Tamanho Funcional

É importante entender que as medidas de tamanho funcional são estimativas das funcionalidades entregues. Além disso, à medida que o escopo é esclarecido e as funções desenvolvidas, é bastante comum identificar funcionalidades adicionais que não foram especificadas nos requisitos originais. Este fenômeno é, algumas vezes, denominado como *scope creep*.

É essencial atualizar o tamanho funcional da aplicação na conclusão do projeto. Se a funcionalidade se alterar durante o desenvolvimento, o tamanho funcional ao final do ciclo de vida reflete precisamente toda funcionalidade entregue ao usuário.

---

# Determinar o Escopo da Contagem e Fronteira e Identificar Requisitos Funcionais de Usuários

---

**Introdução** Este capítulo fornece uma orientação na aplicação de regras, procedimentos e dicas para determinar a fronteira das aplicações e para estabelecer o escopo da contagem.

**Conteúdo** Este capítulo inclui as seguintes seções:

<b>Tópico</b>	<b>Página</b>
<b>Escopo da Contagem e Fronteira</b>	<b>5-2</b>
Propósito de Contagem	5-2
Escopo da Contagem	5-3
Fronteira	5-4
<b>Determine o Escopo da Contagem e Fronteira - Regras e Procedimentos</b>	<b>5-5</b>
Regras de Fronteira	5-5
Procedimentos do Escopo da Contagem e da Fronteira da Aplicação	5-5
<b>Dicas para Ajudar a Identificar o Escopo da Contagem e Fronteira</b>	<b>5-6</b>

## Escopo da Contagem e Fronteira

Esta seção define o escopo da contagem e a fronteira da(s) aplicação(ões) e explica como são influenciados pelo propósito da contagem.

### 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 deve 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 funcionalidade oferecida por cada um

## 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

O *escopo* de:

- Uma contagem de pontos de função de projeto de desenvolvimento inclui todas as funções impactadas (construídas ou customizadas) pelas atividades do projeto. Inclui ainda funções de conversão desenvolvidas como parte do projeto de desenvolvimento.
- Uma contagem de pontos de função de projeto de melhoria inclui todas as funções que estão sendo incluídas, alteradas e excluídas. Inclui ainda conversão de funções desenvolvidas como parte do projeto de melhoria. A fronteira da(s) aplicação(ões) impactadas permanecem as mesmas. A funcionalidade da(s) aplicação(ões) reflete o impacto das funções sendo adicionadas, modificadas ou excluídas.
- Uma contagem de pontos de função da aplicação pode incluir, dependendo do propósito (p.ex., fornecer um pacote como uma solução do software):
  - apenas as funções sendo usadas pelo usuário
  - todas as funções disponibilizadas

O escopo das duas contagens acima é diferente resultando em um tamanho funcional diferente medido para mesma aplicação.

Entretanto, o posicionamento da fronteira das aplicações permanece a mesmo e não é influenciado pela decisão de modificar o escopo. O posicionamento da fronteira é independente do escopo.

## Fronteira

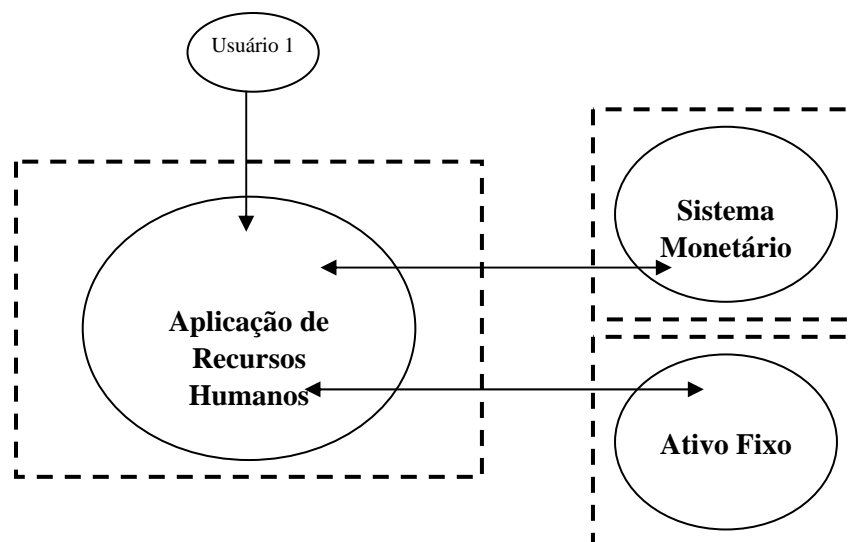
A fronteira é uma interface conceitual entre o software sob estudo e seus usuários.

A fronteira (também chamada de 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 (ALIs)
- 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

O posicionamento da fronteira entre o software sob análise e outra aplicação do software pode ser subjetivo. É comum haver dificuldade para delinear onde uma aplicação termina e a outra se inicia. Tente colocar a fronteira de uma perspectiva de negócio ao invés de se basear em uma consideração técnica ou física. É importante que a fronteira seja colocada com cuidado, de forma que todos os cruzamentos de dados da fronteira possam ser potencialmente incluídos no escopo da contagem

Por exemplo, o diagrama a seguir mostra fronteiras entre a aplicação de Recursos Humanos e as aplicações externas, Sistema Monetário e Ativo Fixo. O exemplo mostra ainda a fronteira entre o usuário humano (Usuário 1) e a aplicação de Recursos Humanos.





## Determine o Escopo da Contagem e Fronteira - Regras e Procedimentos

Esta seção define as regras e procedimentos que se aplicam quando se determina o escopo da contagem e da fronteira da(s) aplicação(ões).

O posicionamento da fronteira é importante porque impacta o resultado da medição de tamanho funcional. A fronteira auxilia na identificação de dados entrando na aplicação e que serão incluídos no escopo da contagem.

### Regras da Fronteira

As seguintes regras devem ser aplicadas para fronteiras:

- ❑ A fronteira é determinada com base na visão do usuário. O foco está no que o usuário pode entender e descrever.
- ❑ A fronteira entre aplicações relacionadas está baseada nas áreas funcionais separadas como pode ser visto pelo usuário, não em considerações técnicas.
- ❑ A fronteira inicial já estabelecida para a aplicação ou aplicações que estejam sendo modificadas não é influenciada pelo escopo da contagem.

**Nota:** Pode haver mais de uma aplicação incluída no escopo da contagem. Nesse caso, múltiplas fronteiras da aplicação deverão ser identificadas.

Quando a fronteira não está bem definida (como no início da análise), ela deverá ser posicionada da forma mais exata possível.

### Procedimentos do Escopo da Contagem e da Fronteira da Aplicação

Quando for executar uma medição de tamanho funcional, os passos abaixo devem ser seguidos:

Passo	Ação
1	Estabelecer o propósito da contagem
2	Identificar o escopo da contagem
3	Identificar a fronteira da aplicação
4	Documentar os seguintes itens: <ul style="list-style-type: none"> <li>• O propósito da contagem</li> <li>• O escopo da contagem</li> <li>• A fronteira da aplicação</li> <li>• Quaisquer suposições relacionadas aos itens acima</li> </ul>

## Dicas para Ajudar na Identificação do Escopo da Contagem e da Fronteira

### **Escopo da Contagem**

As seguintes dicas podem ajudar a identificar o escopo da contagem:

- Revisar o propósito da contagem de pontos de função, para ajudar a determinar o escopo da contagem.
- Na identificação do escopo da contagem de pontos de função da base instalada (p. ex., a funcionalidade suportada pelo grupo de manutenção), incluir todas as funções atualmente em produção e utilizadas pelos usuários.

### **Fronteira**

As seguintes dicas podem ajudar a identificar a fronteira da aplicação(ões):

- Utilize as especificações externas do sistema ou obtenha um fluxo do mesmo e desenhe a respectiva fronteira, destacando as partes internas e as externas à aplicação.
- Verifique como os grupos de dados estão sendo mantidos.
- Identifique as áreas funcionais, alocando certos tipos de objetos da análise (tais como entidades ou processos elementares) a uma área funcional.
- Observe dados de medição correlatos, tais como esforço, custo e defeitos. As fronteiras consideradas para os pontos de função e para os outros dados de medição devem ser as mesmas
- Entrevistar os especialistas no assunto para auxiliar na identificação da fronteira.

---

### Medir Funções de Dados

---

**Introdução** Este capítulo oferece um guia de aplicação das regras e procedimentos para medir funções de dados.

As funções de dados representam a funcionalidade oferecida ao usuário para satisfazer requisitos de dados internos e externos. Uma função de dado pode ser um arquivo lógico interno ou um arquivo de interface externo.

O termo *arquivo* aqui não significa arquivo físico ou tabela. Nesse caso, arquivo se refere a um grupo de dados logicamente relacionados e não à implementação física destes grupos de dados.

**Conteúdo** Este capítulo inclui as seguintes seções:

<b>Tópico</b>	<b>Página</b>
<b>Definição e Intenção Primária: ALIs e AIEs</b>	<b>6-2</b>
Arquivos Lógicos Internos	6-2
Arquivo de Interface Externa	6-2
Diferença entre ALIs e AIEs	6-2
Definições para Termos Utilizados	6-2
<b>Procedimentos para Contagem de Funções de Dados</b>	<b>6-4</b>
Regras de Identificação das Funções de Dados	6-4
Regras de Classificação de Funções de Dados	6-5
Definições e Regras de Complexidade e Contribuição	6-5
Definição de DER	6-5
Regras de DER	6-5
Definição RLR	6-7
Regras de RLR	6-7
Determinação da Complexidade e Contribuição	6-8
<b>Dicas para Ajudar na Contagem</b>	<b>6-10</b>

## Definição e Intenção Primária: ALIs e AIEs

Esta seção inclui a definição e a intenção primária de um arquivo lógico interno (ALI) e de um arquivo de interface externa (AIE). São definidos os termos utilizados nas definições, assim como exemplos foram incluídos ao longo desta seção.

### Arquivos Lógicos Internos

Um arquivo lógico interno (ALI) é um grupo de dados ou de informações de controle logicamente relacionados, reconhecido pelo usuário, mantido dentro da fronteira da aplicação que está sendo contada. A intenção primária de um ALI é armazenar dados mantidos através de um ou mais processos elementares da aplicação que está sendo contada.

### Arquivo de Interface Externa

Um arquivo de interface externa (AIE) é um grupo de dados ou de informações de controle logicamente relacionados, reconhecido pelo usuário, referenciado pela aplicação que está sendo contada, porém, mantido dentro da fronteira de uma outra aplicação. A intenção primária de um AIE é armazenar dados referenciados através de um ou mais processos elementares dentro da fronteira da aplicação que está sendo contada. Isto significa que um AIE contado para uma aplicação deve ser um ALI em outra aplicação.

### Diferença entre ALIs e AIEs

A diferença primária entre um arquivo lógico interno e um arquivo de interface externa é que um AIE **não** é mantido pela aplicação que está sendo contada, enquanto que um ALI é mantido pela aplicação que está sendo contada.

### Definições para Termos Utilizados

Os seguintes parágrafos ajudam na definição de ALIs e AIEs através da definição dos termos utilizados dentro das definições.

#### Informações de Controle

*Informações de Controle* são dados que influenciam um processo elementar. Especificam o que, quando ou como os dados serão processados.

Por exemplo, alguém do departamento da folha de pagamento estabelece ciclos de pagamentos para especificar quando os funcionários de cada local serão pagos. O ciclo de pagamento, ou cronograma, contém informações de periodicidade que determinam quando o processo elementar de pagamento de funcionários ocorrerá.

<b>Reconhecido pelo Usuário</b>	<p>O termo reconhecido pelo usuário refere-se a requisitos definidos para processos e/ou grupos de dados que foram acordados e entendidos tanto pelo(s) usuário(s) quanto pelos desenvolvedor(es) de software.</p> <p><u>Por exemplo</u>, usuários e desenvolvedores concordam que uma Aplicação de Recursos Humanos terá funcionalidade para manter e guardar informações do Funcionário na aplicação.</p>
<b>Mantido</b>	<p>O termo mantido refere-se à habilidade de incluir, modificar ou excluir dados a partir de um processo elementar.</p> <p><u>Exemplos</u> incluem, mas não estão limitados a, inclusão, modificação, exclusão, carga inicial, revisão, atualização, atribuição e criação.</p>
<b>Processo Elementar</b>	<p>Um processo elementar é a menor unidade de atividade que tem significado para o usuário.</p> <p>Deve-se compor e/ou decompor os Requisitos Funcionais do Usuário até a menor unidade de atividade, a qual satisfaz os itens a seguir:</p> <ul style="list-style-type: none"><li>• é significativo para o usuário</li><li>• constitui uma transação completa</li><li>• é auto contida e</li><li>• deixa o negócio da aplicação contada em um estado consistente</li></ul> <p><u>Por exemplo</u>, os requisitos do usuário para adicionar um funcionário inclui informações de salário e dependentes. Um funcionário não terá sido criado se não forem incluídas todas as respectivas informações. Incluir separadamente apenas parte das informações deixará o negócio de incluir um funcionário em um estado inconsistente. Se forem incluídos tanto o salário do empregado quanto as informações do(s) dependente(s), a unidade de atividade será concluída e o negócio será deixado em um estado consistente.</p>

## Procedimentos para Contagem de Funções de Dados

Esta seção oferece um guia na aplicação de regras para a contagem das funções de dado (arquivos lógicos internos e arquivos de interface externa).

Este resumo foi incluído para mostrar as regras no contexto dos procedimentos de contagem de ALI e de AIE.

**Nota:** Os procedimentos detalhados de contagem estão na página 6-4. Um guia adicional e exemplos estão na Parte 3 Arquivos Lógicos.

O procedimento para medir funções de dados inclui os seguintes passos:

Passo	Ação
1	Identificar as funções de dados.
2	Classificar cada função de dado como um ALI ou AIE.
3	Determinar a complexidade dos ALI ou AIE e sua contribuição para o tamanho funcional.

## Regras de Identificação das Funções de Dados

Uma função de dados representa a funcionalidade fornecida ao usuário para atender requisitos de armazenamento de dados internos e externos. Uma função de dados é um ALI ou um AIE.

**Nota:** Funções de dados são mais facilmente identificadas quando se utiliza um modelo lógico de dados; entretanto, isto não impede a medição em ambientes onde técnicas alternativas de modelagem de dados ou objetos são empregadas. A terminologia de modelagem de dados é utilizada para documentar as regras de funções de dados, porém a mesma abordagem pode ser aplicada com outras técnicas.

Para identificar funções de dados, as atividades a seguir devem ser cumpridas:

- ❑ Identificar no escopo da contagem todos os dados e informações de controle logicamente relacionados e reconhecidos pelo usuário.
- ❑ Excluir entidades que não são mantidas por nenhuma aplicação.
- ❑ Agrupe entidades relacionadas que são dependentes (Consulte Parte 3 Arquivos Lógicos)

**Nota:** Entidades independentes devem ser consideradas grupos lógicos de dados separados.

- ❑ Excluir as entidades classificadas como Dados de código (Consulte Parte 3 Dados de Código)

- ❑ Excluir entidades que não contém atributos necessários para o usuário.
- ❑ Remover entidades associativas que contém atributos adicionais não necessários para o usuário e entidades associativas que contém apenas chaves estrangeiras; agrupe atributos de chave estrangeira com as entidades primárias

**Nota:** Atributos chave estrangeira são dados necessários para o usuário estabelecer uma relação com outra função de dado.

## Regras de Classificação de Funções de Dados

Classificar como um ALI se o dado é mantido pela aplicação que está sendo medida.

Classificar como um AIE se:

- ❑ É referenciado, mas não mantido, pela aplicação que está sendo medida e
- ❑ É identificado como um ALI em uma ou mais aplicações

**Nota:** Se a função de dados satisfaz ambas as regras, classifique-a como ALI.

## Definições e Regras de Complexidade e Contribuição

O número de ALIs, AIEs, e suas respectivas complexidades funcionais determinam a contribuição das funções de dados para o tamanho funcional.

Atribua a cada ALI e AIE identificado uma complexidade funcional com base na quantidade de tipos de dados elementares (DERs) e de tipos de registros elementares (RLRs) associados ao ALI ou AIE.

Esta seção define DERs e RLRs e inclui regras para cada um.

### Definição de DER

Um *tipo de dado elementar* é um campo único, reconhecido pelo usuário e não repetido.

### Regras de DER

Para a contagem de Tipos de Dados Elementares (DERs) para uma função de dados, as regras a seguir devem ser aplicadas:

- ❑ Conte um DER para cada campo único, reconhecido pelo usuário e não repetido, mantido ou recuperado pela função de dados durante a execução de todos os processos elementares no escopo da contagem.

Por exemplo, o(s) resultado(s) do cálculo de um processo elementar, como o valor do imposto sobre uma venda, referente a um pedido de cliente mantido em um ALI é contado como um DER no ALI de pedido de cliente.

Por exemplo, acessar o preço de um item salvo em um arquivo de faturamento, ou campos como um time stamp, se requisitado pelo usuário, são contados como DERs.

Por exemplo, se um número de funcionário aparece duas vezes em um ALI ou AIE como: (1) chave do registro do funcionário e (2) chave estrangeira do registro do dependente, conte o DER apenas uma vez.

Por exemplo, dentro de um ALI ou AIE, conte um DER para os 12 campos Valor Mensal Orçado. Conte um DER adicional para identificar o mês aplicável.

- ❑ Conte apenas os DERs que estão sendo usados pela aplicação que está sendo medida quando duas ou mais aplicações estiverem sendo mantidas e/ou referenciando a mesma função de dados.

**Nota:** Atributos que não são referenciados pela aplicação que está sendo medida não são contados.

Por exemplo, a Aplicação A pode identificar e utilizar um endereço como: rua, cidade, estado e CEP. A Aplicação B pode ver o endereço como um bloco de dados sem considerar os componentes individuais. A Aplicação A contaria quatro DERs; a Aplicação B contaria um DER.

Por exemplo, a Aplicação X mantém e/ou referencia um ALI que contém CPF, Nome, Rua, Caixa Postal, Cidade, Estado e CEP. A Aplicação Z mantém e/ou referencia Nome, Cidade e Estado. A Aplicação X contaria sete DERs; a Aplicação Z contaria três DERs.

- ❑ Conte um DER para cada parte de dado requisitada pelo usuário para estabelecer um relacionamento com outra função de dado.

Por exemplo, na Aplicação de RH, as informações de um funcionário são mantidas dentro de um ALI. O nome da função do funcionário é incluído como parte das informações do funcionário. Este DER é contado porque é necessário para relacionar um funcionário a uma função existente na organização. Este tipo de dado elementar é conhecido como *chave estrangeira*.

Por exemplo, em uma aplicação orientada a objetos (OO), o usuário solicita uma associação entre classes de objetos, as quais foram identificadas como ALIs distintos. O Nome do local é um DER do ALI Local. O nome do local é requerido ao processar as informações do funcionário; conseqüentemente, também é contado como um DER dentro do ALI funcionário.

- ❑ Revisar os atributos relacionados para determinar se eles estão agrupados e contados como um simples DER ou se são contados como DERs múltiplos; o agrupamento dependerá de como os processos elementares usam os atributos dentro da aplicação



Por exemplo, um número de conta que é armazenado em vários campos é contado como um DER.

Por exemplo, uma imagem antes ou depois de um grupo de 10 campos mantidos para fins de auditoria deve ser contado como um DER para a imagem antes (todos os 10 campos) e um DER para a imagem depois (todos os 10 campos), totalizando 2 DERs.

### **Definição RLR**

Um *Tipo de Registro Elementar (RLR)* é um subgrupo de dados reconhecido pelo usuário dentro de uma função de dados.

### **Regras de RLR**

Para Contar Tipos de Registro Elementar (RLRs) para uma função de dados, as atividades a seguir devem ser executadas:

- ❑ Conte um RLR para cada função de dados (por padrão cada função de dado tem um subgrupo de DERs para ser contado como um RLR)
- ❑ Conte um RLR adicional para cada subgrupo de DER lógico adicional (com a função de dados) que contém mais de um DER:
  - entidade associativa com atributos não-chave
  - subtipo (outro além do primeiro subtipo) e
  - entidade atributiva, em um relacionamento que não seja obrigatório 1-1

**Nota:** Um relacionamento obrigatório 1-1 reflete a relação entre duas entidades onde cada uma é relacionada com uma, e apenas uma, instância de uma entidade relacionada.

Por exemplo, em uma Aplicação de Recursos Humanos, a informação para um funcionário é adicionado através da adição de informações gerais. Além das informações gerais, o funcionário é um assalariado ou horista.

O usuário determinou que um funcionário ou é assalariado ou é horista. Cada tipo de funcionário possui atributos próprios. Os dois tipos podem ter informações sobre dependentes. Neste exemplo, existem três subgrupos ou RLRs, como mostrado abaixo:

- Funcionário assalariado; incluindo informações gerais
- Funcionário horista; incluindo informações gerais
- Dependente do funcionário

**Nota:** Se não houver um modelo de dados, procure grupos de dados repetidos a fim de identificar RLRs.

**Nota:** Existem dois tipos de subgrupos: Opcional e Obrigatório.

*Subgrupos Opcionais* são aqueles que o usuário tem a opção de usar

um ou nenhum dos subgrupos durante o processo elementar que inclui ou cria uma instância do dado.

*Subgrupos Obrigatórios* são subgrupos onde o usuário deve usar pelo menos um durante um processo elementar que adiciona ou cria uma instância de dados.

## Determinação da Complexidade e Contribuição

A complexidade funcional de cada função de dados deve ser determinada utilizando os passos abaixo.

Passo	Ação																
1	Para identificar e contar os RLRs e DERs, use as regras de contagem de complexidade e contribuição que iniciam na página 6-5.																
2	A complexidade funcional de cada função de dados deve ser determinada utilizando o número de RLRs e DERs de acordo com esta matriz. <table><tr><th></th><th>1 a 19 DERs</th><th>20 a 50 DERs</th><th>51 ou mais DERs</th></tr><tr><td>1 RLR</td><td>Baixa</td><td>Baixa</td><td>Média</td></tr><tr><td>2 a 5 RLRs</td><td>Baixa</td><td>Média</td><td>Alta</td></tr><tr><td>6 ou mais RLRs</td><td>Média</td><td>Alta</td><td>Alta</td></tr></table>		1 a 19 DERs	20 a 50 DERs	51 ou mais DERs	1 RLR	Baixa	Baixa	Média	2 a 5 RLRs	Baixa	Média	Alta	6 ou mais RLRs	Média	Alta	Alta
	1 a 19 DERs	20 a 50 DERs	51 ou mais DERs														
1 RLR	Baixa	Baixa	Média														
2 a 5 RLRs	Baixa	Média	Alta														
6 ou mais RLRs	Média	Alta	Alta														

Por exemplo, uma função de dados com 51 DERs e 2 RLRs se traduz uma complexidade funcional alta.

- 3 O tamanho funcional de cada função de dados é determinado usando o tipo e a complexidade funcional de acordo com as tabelas abaixo.

Tabela de Contribuição de ALI: Use a tabela a seguir para atribuir um tamanho funcional para cada ALI.

Grau de Complexidade Funcional	Pontos de Função
Baixo	7
Médio	10
Alto	15

Tabela de Contribuição de AIE: Use a tabela a seguir para atribuir um tamanho funcional para cada AIE.

Grau de Complexidade Funcional	Pontos de Função
Baixo	5
Médio	7
Alto	10

Por exemplo, um alto grau de complexidade funcional para um AIE se transforma em 10 pontos de função.

*Continua na próxima página*

Passo	Ação
-------	------

**4** A contribuição dos ALIs e AIEs para o tamanho funcional podem ser totalizadas.

Por exemplo, a tabela a seguir mostra o cálculo para um ALI de complexidade alta, dois AIEs de complexidade média e um AIE de complexidade alta.

Tipo de Função	Complexidade Funcional			Total da Complexidade	Total do Tipo de Função
ALI	0	Baixa	X 7 =	0	15
	0	Média	X 10 =	0	
	1	Alta	X 15 =	15	
AIE	0	Baixa	X 5 =	0	24
	2	Média	X 7 =	14	
	1	Alta	X 10 =	10	

Neste exemplo, não existem ALIs de baixa ou alta complexidade e apenas um ALI de alta complexidade; portanto, o tamanho total para os ALIs é de 15 pontos. Para os AIEs, não existe nenhum de complexidade baixa, dois de complexidade média (14 pontos) e um de alta complexidade (10 pontos) totalizando 24 pontos para os AIEs.

O tamanho funcional engloba o total final para todos os tipos de funções. As contribuições para ALIs e AIEs podem ser acrescentadas à tabela que lista todos os tipos de funções. O Apêndice A inclui uma tabela que pode ser usada para registrar o total para todas as funções.

## Dicas para Ajudar na Contagem

As dicas a seguir podem ajudar a aplicar as regras de ALI e AIE.

Estas dicas *não são* regras e não devem ser usadas como regras.

- Os dados constituem um grupo lógico que suporta os requisitos específicos do usuário?
  - Uma aplicação pode usar um ALI ou AIE em diversos processos, mas o ALI ou AIE é contado apenas uma vez.
  - Um arquivo lógico não pode ser contado tanto como ALI e AIE para a mesma aplicação. Considere a intenção primária do grupo de dados. Se o grupo de dados satisfizer ambas as regras, conte-o apenas como um ALI.
  - Se o grupo de dados não foi contado como um ALI ou AIE por si só, conte seus atributos como DERs para o ALI ou AIE que inclui este grupo.
  - Não assuma que um arquivo físico, tabela ou classe de objeto equivale a um arquivo lógico quando observar dados lógicos na visão do usuário.
  - Apesar de algumas tecnologias de armazenamento como tabelas em um banco de dados relacional, arquivos seqüenciais ou classes de objetos estarem relacionadas a ALIs ou AIEs, não assuma que haverá sempre um relacionamento lógico-físico um-para-um.
  - Não assuma que um arquivo físico, tabela ou classe de objeto deva ser contado ou incluído como parte de um ALI ou AIE.
- Onde os dados são mantidos? Dentro ou fora da fronteira da aplicação?
  - Observe o fluxo de trabalho.
  - Na decomposição funcional do processo, identifique onde ocorrem as interfaces com o usuário e com outras aplicações.
  - Navegue através do diagrama de processos para conseguir dicas.
  - Contabilize os ALIs mantidos por mais de uma aplicação em cada aplicação no momento que a aplicação é medida.
  - Apenas os DERs utilizados por cada aplicação contada devem ser utilizados para medir o ALI/AIE.
- Os dados em um ALI são mantidos através de um processo elementar da aplicação?
  - Uma aplicação pode usar um ALI ou AIE várias vezes, mas o ALI ou AIE deve ser contado apenas uma vez.
  - Um processo elementar pode manter mais do que um ALI.
  - Navegue através do diagrama de processos para obter dicas.

---

### Medir Funções de Transações

---

**Introdução** Este capítulo fornece um guia para aplicação de regras para medir as funções de transação do tipo entrada externa (EE), saída externa (SE), e consulta externa (CE).

Uma função de transação é um processo elementar que oferece funcionalidade ao usuário para processar dados. Uma função de transação é uma entrada externa, saída externa, ou consulta externa.

Para exemplos que ilustran a aplicação dessas regras, vide Parte 4 – Exemplos.

**Conteúdo** Este capítulo inclui as seguintes seções:

<b>Tópico</b>	<b>Página</b>
<b>Definição e Intenção Primária: EEs, SEs e CEs</b>	<b>7-3</b>
Entrada Externa	7-3
Consulta Externa	7-3
Saída Externa	7-3
Resumo das Funções Executadas pelas EEs, SEs e CEs.	7-4
Definições de Termos Utilizados	7-5
Resumo das Lógicas de Processamento Usadas pelas EEs, SEs e CEs.	7-8
<b>Regras de Contagem de Função de Transação</b>	<b>7-9</b>
Procedimentos de Contagem de Função de Transação	7-9
Identificar Cada Processo Elementar	7-10
Determinar Unicidade do Processo Elementar	7-11
<b>Classificar Cada Processo Elementar</b>	<b>7-13</b>
Classificar como uma EE	7-13
Classificar como uma SE	7-13
Classificar como uma CE	7-13

*Continua na página seguinte*

<b>Tópico</b>	<b>Página</b>
Regras e Definições de Complexidade e Contribuição	7-14
Definição de RLR	7-14
Regra de RLR	7-14
Definição de DER	7-14
Regras de DER	7-14
Diretrizes para Complexidade e Contribuição de EE	7-15
Orientações para contagem de RLR em uma EE	7-15
Orientações para contagem de DER em uma EE	7-15
Guia de Complexidade e Contribuição para SE/CE	7-17
Orientações para contagem de RLR para SEs	7-17
Orientações para contagem de RLR para CEs	7-17
Orientações compartilhadas de contagem de DER para SEs e CEs	7-17
Determinação de Complexidade e Contribuição	7-19
<b>Dicas para Ajudar na Contagem EEs, SEs e CEs</b>	<b>7-21</b>
Dicas Adicionais para Ajudar na Contagem de SEs e CEs	7-23

## Definição e Intenção Primária: EEs, SEs e CEs

Esta seção inclui a definição e intenção primária de uma entrada externa (EE), saída externa (SE) e uma consulta externa (CE). As definições dos termos utilizados neste manual, bem como os respectivos exemplos, foram incluídas no decorrer desta seção.

### Entrada Externa

Uma entrada externa (EE) é um processo elementar que processa dados ou informações de controle que vêm de fora da fronteira da aplicação. A intenção primária de uma EE é manter um ou mais ALIs e/ou alterar o comportamento do sistema.

### Consulta Externa

Uma consulta externa (CE) é um processo elementar que envia dados ou informações de controle para fora da fronteira da aplicação. A intenção primária de uma consulta externa é apresentar informações ao usuário através da recuperação de dados ou informações de controle. A lógica de processamento não contém fórmula matemática ou cálculo, e nem cria dados derivados. Nenhum ALI é mantido durante o processamento, nem o comportamento do sistema é alterado.

### Saída Externa

Uma saída externa (SE) é um processo elementar que envia dados ou informações de controle para fora da fronteira da aplicação e que inclui um processamento adicional ao de uma consulta externa. A intenção primária de uma SE é apresentar informações ao usuário através de lógica de processamento que não seja apenas a recuperação de dados ou informações de controle. A lógica de processamento deve conter pelo menos uma fórmula matemática ou cálculo, criar dados derivados, manter um ou mais ALIs ou alterar o comportamento do sistema.

## Resumo das Funções Realizadas pelas EEs, SEs e CEs

Como indicado na Parte 1, a principal diferença entre os tipos de função de transação é sua intenção primária. A tabela abaixo mostra resumidamente as funções que podem ser executadas por cada tipo de função de transação, especificando a intenção primária de cada uma.

Função:	Tipo de Função de Transição:		
	EE	SE	CE
Alterar o comportamento do sistema	IP	F	N/A
Manter um ou mais ILFs	IP	F	N/A
Apresentar a informação ao usuário	F	IP	IP

Legenda:

IP a intenção primária do tipo de função de transação

F é uma função do tipo de função de transação, mas não é a intenção primária e está presente algumas vezes

N/A a função não é permitida para o tipo de função de transação

A principal diferença entre EEs e SEs/CEs é a intenção primária.

Algumas das diferenças entre SEs e CEs são que uma SE pode alterar o comportamento do sistema, ou manter um ou mais ALIs enquanto executa a intenção primária de apresentar informações ao usuário. Outras diferenças são identificadas na seção abaixo, que resume as formas de lógica de processamento para cada função de transação.



## Definições de Termos Utilizados

Os parágrafos seguintes incluem as definições da Parte 1 necessárias para determinar EEs, SEs e CEs.

<b>Processo Elementar</b>	<p>Um <i>Processo Elementar</i> é a menor unidade de atividade que é significativa para o usuário.</p> <p><u>Por exemplo</u>, requisitos podem indicar a necessidade de adicionar diferentes tipos de informação do funcionário (ex. endereço, informações de salário e dependentes), mas a menor unidade de atividade significativa para o usuário é Adicionar Funcionário. Neste exemplo, adicionar um funcionário (sem adicionar endereço e informações de salário e dependente) não cumpre todos os critérios. Outros sistemas podem tratar a manutenção de salário e/ou informações do dependente independentemente do funcionário.</p>
<b>Informações de Controle</b>	<p><i>Informações de controle</i> são dados que influenciam um processo elementar da aplicação especificando o que, quando, ou como dados serão processados.</p> <p><u>Por exemplo</u>, alguém do departamento de folha de pagamento estabelece ciclos de pagamentos, para planejar quando os funcionários de cada local serão pagos. O ciclo, ou programa de pagamento contém informações de datas que irão afetar o momento da ocorrência do processo elementar de pagamento de funcionários.</p>
<b>Mantido</b>	<p>O termo <i>mantido</i> se refere a habilidade de incluir, alterar ou excluir dados através de um processo elementar.</p> <p><u>Os exemplos</u> incluem, mas não são limitados a inclusão, alteração, exclusão, carga inicial, revisão, atualização, atribuição e criação.</p>
<b>Usuário</b>	<p>Um <i>usuário</i> é qualquer pessoa ou coisa que se comunica ou interage com o software a qualquer momento.</p> <p><u>Os exemplos</u> incluem pessoas do departamento do RH que interagem com a aplicação para configurar funcionários, e a aplicação de Benefícios que interage com a aplicação de RH para receber informações sobre os dependentes dos funcionários.</p>
<b>Significativo</b>	<p>É reconhecido pelo usuário e satisfaz um Requisito Funcional do Usuário.</p>

**Lógica de Processamento**

*Lógica de Processamento* é definida como qualquer um dos requisitos especificamente solicitados pelo usuário para completar um processo elementar como validações, algoritmos ou cálculos e leitura ou manutenção de uma função de dados. Esses requisitos podem incluir as seguintes ações:

1. Validações são executadas  
Por exemplo, quando incluir um novo funcionário em uma organização, o processo de funcionário valida o tipo DER do funcionário.
2. Fórmulas matemáticas e cálculos são executados  
Por exemplo, ao produzir informações sobre todos os funcionários de uma organização, o processo inclui o cálculo do número total de funcionários assalariados, funcionários horistas e de todos os funcionários.
3. Valores equivalentes são convertidos  
Por exemplo, a idade do funcionário é convertida para um grupo de faixa etária usando uma tabela.
4. Dados são filtrados e selecionados através da utilização de critérios especificados para comparar vários grupos de dados  
Por exemplo, para gerar uma lista de funcionários por atribuição, um processo elementar compara o código da tarefa de uma atribuição para selecionar e listar os funcionários com esta atribuição.
5. Condições são analisadas para determinar quais são aplicáveis  
Por exemplo, a lógica de processamento empregada por um processo elementar na inclusão de um funcionário, vai depender do funcionário ser pago através de salário mensal ou horas trabalhadas. A entrada de DERs (e o resultado do processamento lógico) baseado em uma escolha diferente (assalariado ou horista) neste exemplo é parte de um processo elementar.
6. Um ou mais ALIs são atualizados  
Por exemplo, ao incluir um funcionário, o processo elementar atualiza o ALI “funcionário” para manter os dados do funcionário.
7. Um ou mais ALIs e AIEs são referenciados  
Por exemplo, ao incluir um funcionário, o AIE “moeda” é referenciado para usar a taxa de câmbio do dólar correta, para determinar o valor da hora do funcionário em dólares.
8. Dados ou informações de controle são recuperados  
Por exemplo, para ver uma lista de funcionários, as informações dos funcionários são recuperadas de uma função de dados.

9. Dados derivados são criados através da transformação de dados existentes, para criar dados adicionais

Por exemplo, para determinar (derivar) um número de registro do paciente (ex. BARJO01), os seguintes dados são concatenados:

- as 3 primeiras letras do sobrenome do funcionário (BAR de Barros)
- as 2 primeiras letras do nome do funcionário (JO de João)
- um número sequencial de dois dígitos (começando de 01)

10. O comportamento da aplicação é alterado

Por exemplo, o comportamento do processo elementar de pagamento de funcionários é alterado quando uma mudança é feita para pagá-los às sextas-feiras, a cada duas semanas, ao invés de pagá-los no 15º dia e no último dia do mês, resultando em 26 períodos de pagamento por ano, contra 24.

11. Preparar e apresentar informações fora da fronteira

Por exemplo, uma lista de funcionários apresentada ao usuário.

12. Existe a capacidade de receber dados ou informações de controle que entram pela fronteira da aplicação

Por exemplo, um usuário entra com várias informações para incluir uma ordem de compra na aplicação.

13. Dados são reclassificados ou rearrumados. Esta forma de processamento lógico não implica na identificação de tipo ou contribuição na unicidade de um processo elementar; ou seja, a orientação dos dados não constitui unicidade.

Por exemplo, uma lista de funcionários é classificada em ordem alfabética ou de localização.

Por exemplo, em um pedido na tela de entrada, a informação do cabeçalho é organizado no topo da tela, e os detalhes são colocados abaixo.

Um processo elementar pode incluir múltiplas alternativas ou ocorrências das ações acima.

Por exemplo, validações, filtros, reclassificações, etc.

## Resumo das Lógicas de Processamento Utilizadas pelas EEs, SEs e CEs

A tabela seguinte resume as formas de lógica de processamento que podem ser executadas pelas EEs, SEs e CEs. Para cada tipo de função de transação, certos tipos de lógica de processamento podem ser executadas para atender a intenção primária daquele tipo. As 13 ações, por si só, não permitem identificar processo elementares únicos.

Formas de lógica de processamento:	Tipo da Função de Transação		
	EE	SE	CE
1. Validações são efetuadas	p	p	p
2. Cálculos matemáticos são efetuados	p	d*	n
3. Valores equivalentes são convertidos	p	p	p
4. Dados são filtrados e selecionados por critérios específicos para comparar vários grupos de dados	p	p	p
5. Condições são analisadas para determinar quais se aplicam	p	p	p
6. Pelo menos um ALI é atualizado	d*	d*	n
7. Pelo menos um ALI ou AIE é referenciado	p	p	d
8. Dados ou informações de controle são recuperados	p	p	d
9. Dados derivados são criados	p	d*	n
10. O comportamento do sistema é alterado	d*	d*	n
11. Preparar e apresentar informações para fora da fronteira	p	d	d
12. Dados ou informações de controle entrando pela fronteira da aplicação são aceitos	d	p	p
13. Os dados são reclassificados ou reorganizados	p	p	p

Legenda:

- d** o tipo de função **deve** executar esta forma de lógica de processamento
- d\*** o tipo de função **deve** executar pelo menos uma destas formas de lógica de processamento
- p** o tipo de função **pode** executar esta forma de lógica de processamento, mas a mesma não é obrigatória
- n** o tipo de função **não pode** executar esta forma de lógica de processamento

## Regras de Contagem de Função de Transição

Esta seção oferece um guia para a aplicação de regras na contagem das EEs, SEs e CEs.

### Procedimentos de Contagem de Função de Transição

Os procedimentos de contagem de função de transação devem incluir os seguintes passos:

Passo	Ação
1	Identificar cada processo elementar.
2	Determinar o processo elementar único.
3	Classificar cada função de transição como Entrada Externa (EE), Saída Externa (SE) ou Consulta Externa (CE).
4	Determinar a complexidade funcional para cada função de transação e sua contribuição para o tamanho funcional.

As regras na Parte 1 são explicadas nos seguintes parágrafos.

## Identificar Cada Processo Elementar

Para identificar cada processo elementar, as atividades a seguir devem ser realizadas:

Compor e/ou decompor os Requisitos Funcionais do Usuário até a menor unidade de atividade que satisfaz todos os itens a seguir:

- ☐ é significativo para o usuário

Por exemplo, os requisitos funcionais do usuário requerem a habilidade de adicionar um novo funcionário na aplicação.

- ☐ constitui uma transação completa

Por exemplo, a definição do usuário inclui informações de salário e dependentes do funcionário. Se o número de dependentes é maior do que zero, ao adicionar um funcionário deve incluir informações do dependente. Neste exemplo, adicionar um funcionário (sem adicionar endereço e informações de salário e dependente) não cumpre todos os critérios. Outros sistemas podem tratar a manutenção de salário e/ou informações do dependente independentemente do funcionário.

- ☐ é auto-contido e

Por exemplo, o processo incluir não é auto-suficiente a menos que toda informação obrigatória seja informada e todos os passos do processamento são executados; ex. validações, cálculos, atualização de ALIs.

- ☐ deixa o negócio de aplicação sendo medida em um estado consistente

Por exemplo, os requisitos do usuário para adicionar um funcionário incluem a configuração de informações de salário e dependentes. Se toda informação do funcionário não é adicionada, um funcionário ainda não foi criado. Adicionar alguma informação sozinha deixa o negócio de adicionar um funcionário em um estado inconsistente. Se tanto o salário do funcionário e a informação do dependente são informados, a unidade de atividade é completada e o negócio é deixado em um estado consistente.

Identifique um processo elementar para cada unidade de atividade identificada que agrupa todos os critérios acima.

## Determinar Processos Elementares Únicos

Para determinar processos elementares únicos, as atividades a seguir devem ser cumpridas

Quando comparado a um Processo Elementar (PE) já identificado, conte dois PEs similares como o mesmo Processo Elementar se eles:

- ❑ Requerem o mesmo conjunto de DERs e
- ❑ Requerem o mesmo conjunto de RLRs e
- ❑ Requerem o mesmo conjunto de lógicas de processamento para completar o processo elementar

**Nota:** Um processo elementar pode ter pequena variação em DERs ou RLRs assim como múltiplas alternativas, variações ou ocorrências de lógicas de processamento abaixo.

**Nota:** Quando os dois processos elementares são comparados e se determina que eles contém diferentes DERs, RLRs ou Processamento Lógico, eles são identificados como processos elementares separados se forem especificados como requisitos funcionais distintos pelo usuário.

**Nota:** O teste de unicidade acima deve ser utilizado para comparar dois PEs que já tenham sido identificados e não como justificativa para dividir um único PE em dois PEs como resultado de variações. Dividir um único PE em dois PEs baseado nas variações pode indicar que as regras para identificar um PE não tenha sido satisfeitas.

Por exemplo, quando um PE para Adicionar Funcionário requer DERs adicionais para tratar endereços de funcionários europeus e americanos (caixa postal/ CEP, país/estado, número de telefone e código da cidade). O PE não é dividido em dois PEs por conta da pequena diferença no endereço do funcionário. O PE é ainda Adicionar Funcionário, e há uma variação na lógica de processamento e DERs para contar as diferenças no endereço e número de telefone.

Por exemplo, quando um PE para Adicionar funcionário foi identificado, o mesmo não é dividido em dois PEs para contar o fato de que um funcionário pode ou não ter dependentes. O PE ainda é Adicionar Funcionário, e há variação no processo lógico e DERs para contar dependentes.

Por exemplo, quando o requisito funcional do usuário especificar a necessidade para dois relatórios semelhantes (tal como o Relatório 1 que contém Nome do Consumidor, Identidade do Consumidor, e Endereço e Relatório 2 que contém Nome do Consumidor, Identidade do Consumidor, Endereço e Telefone), os relatórios são identificados como PEs separados uma vez que o requisito funcional do usuário especifica a necessidade para diferentes DERs. Os relatórios não são combinados em um PE único apenas porque têm DERs semelhantes.

- ❑ Não divida um processo elementar com múltiplas formas de processamento lógico em múltiplos processos elementares. Se um processo elementar é subdividido inapropriadamente o mesmo não reúne os critérios (listados acima) de um processo elementar.



## Classificar Cada Processo Elementar

Classificar cada processo elementar como uma Entrada Externa (EE), Saída Externa (SE) ou uma Consulta Externa (CE) baseado na sua intenção primária.

A intenção primária de um processo elementar deve ser identificada pelos seguintes itens:

- ☐ alterando o comportamento da aplicação
- ☐ mantendo um ou mais ALIs
- ☐ apresentando informação ao usuário

As formas de lógica de processamento necessárias para completar o processo elementar deve ser identificado através da lista apresentada na página 7-8.

### Classificar como uma EE

Tem a intenção primária de:

- ☐ manter um ou mais ALIs ou
- ☐ alterar o comportamento da aplicação e

Incluir a lógica de processamento de aceitar dados ou informação de controle que entra na fronteira de aplicação

### Classificar como uma SE

Tem a intenção primária de apresentar a informação ao usuário, e

Incluir pelo menos uma das formas seguintes de lógica de processamento:

- ☐ cálculos matemáticos são realizados
- ☐ um ou mais ALIs são atualizados
- ☐ é criado dado derivado ou \*
- ☐ o comportamento da aplicação é alterado

**Nota:** \*Campos calculados são uma forma de dado derivado, apesar de que dados derivados podem ser também criados sem realizar o cálculo.

### Classificar como uma CE

Tem a intenção primária de apresentar informação ao usuário, e:

- ☐ referenciar uma função de dados para recuperar dados ou informações de controle e
- ☐ não satisfaz o critério de ser classificado como uma SE

## Regras e Definições de Complexidade e Contribuição

O número de EEs, SEs e CEs e suas complexidades funcionais determinam a contribuição das funções de transação para o tamanho funcional.

Atribua uma complexidade funcional a cada EE, SE e CE identificada, com base no número de tipos de arquivos referenciados (ALRs) e tipos de dados elementares (DERs).

Na aplicação de regras de RLR/DER para funções do tipo transação, é importante reconhecer as diferenças nas funções que cada tipo pode realizar de acordo com suas regras de classificação (ex. uma CE, pela regra não pode atualizar um RLR; consequentemente o guia para contar um RLR para uma atualização de ALI não se aplica). Para identificar as funções que um tipo de função de transação pode realizar, consultar a tabela na página 7-8.

### Definição de ALR

Um *tipo de arquivo referenciado* é uma função de dados lida e/ou mantida pela função de transação.

Um *tipo de arquivo referenciado* inclui:

- Um arquivo lógico interno lido ou mantido por uma função de transação ou
- Um arquivo de interface externa lido por uma função de transação

### Regra de ALR

Um ALR deve ser contado para cada função de dados que é acessado (lido e/ou escrito).

### Definição de DER

Um *tipo de dado elementar* é um campo único, reconhecido pelo usuário e não repetido.

### Regras de DER

Para contar DERs como uma função de transação, as atividades a seguir devem ser realizadas

- ❑ Revisar tudo que cruza (entra e/ou sai) da fronteira
- ❑ Conte um DER para cada campo único reconhecido pelo usuário, atributo não repetido que cruza (entra e/ou sai) a fronteira durante o processamento da função de transação
- ❑ Conte apenas um DER por função de transação para a habilidade de enviar uma mensagem de resposta de aplicação mesmo que sejam mensagens múltiplas
- ❑ Conte apenas um DER por função de transação para a habilidade de iniciar ação(ões) mesmo que haja múltiplos meios para realizá-la

- ❑ Não conte os itens a seguir como DERs:
  - ❑ literais como títulos de relatório, tela ou identificador do painel, títulos de coluna e títulos de atributos
  - ❑ selos gerados automaticamente pelo sistema como atributos de data e hora
  - ❑ variável de paginação, número de páginas e informação de posicionamento; ex. 'Linhas 37 a 54 de 211'
  - ❑ ajudas de navegação como a habilidade de navegar com uma lista utilizando "anterior", "próximo", "primeiro", "último" e seus gráficos equivalentes
  - ❑ atributos gerados dentro da fronteira por uma função de transação e armazenado em um ALI sem sair da fronteira
  - ❑ atributos obtidos ou referenciados de um ALI ou AIE para a participação em processamento sem sair da fronteira

**Regras de Complexidade e Contribuição de EE**

Esta seção define as regras de ALR e DER utilizadas para determinar a complexidade e contribuição das entradas externas.

**Regras de ALR para uma EE**

Reconhecendo que uma EE deve atualizar um ALI ou alterar o comportamento da aplicação, as seguintes regras são aplicáveis quando contar ALRs:

- ❑ Conte um ALR para cada ALI mantido
- ❑ Conte um ALR para cada ALI ou AIE lido
- ❑ Conte apenas um ALR para cada ALI que seja lido e mantido

**Diretrizes de DER para uma EE**

Reconhecendo que uma EE deve atualizar um ALI ou controlar o comportamento da aplicação, as seguintes regras são aplicáveis quando contar DERs:

- ❑ Revise tudo que cruza (entra e/ou sai) a fronteira
- ❑ Conte um DER para cada campo único reconhecido pelo usuário, atributo não repetido que cruza (entra e/ou sai) da fronteira durante o processamento da função de transação

Por exemplo, nome do trabalho e grade de pagamento são dois campos que o usuário informa quando inclui um trabalho.

- ❑ Conte apenas um DER por função de transação para a habilidade de enviar uma mensagem de resposta mesmo se forem várias mensagens

Por exemplo, se um usuário tenta incluir um funcionário existente na aplicação de Recursos Humanos, o sistema gera a respectiva mensagem de erro e o campo incorreto é marcado. Conte um DER que incluirá todas as respostas que indicam condições de erro, confirmam que o processamento está concluído, ou confirmam que o processamento deverá continuar.

- ❑ Conte apenas um DER por função de transação para a habilidade de iniciar ação(ões) mesmo que haja múltiplos meios para isto

Por exemplo, se o usuário pode iniciar a inclusão de um funcionário clicando no botão OK ou pressionando uma tecla PF, conte um DER para a habilidade de iniciar o processo.

- ❑ Não conte os itens a seguir como DERs:
  - ❑ literais como títulos de relatório, tela ou identificador do painel, títulos de coluna e títulos de atributos
  - ❑ selos gerados automaticamente pelo sistema como atributos de data e hora
  - ❑ variáveis de paginação, número de páginas e informação de posicionamento; ex. ‘Linhas 37 a 54 de 211’
  - ❑ ajudas de navegação como a habilidade de navegar com uma lista utilizando “anterior”, “próximo”, “primeiro”, “último” e seus equivalentes gráficos
  - ❑ atributos gerados dentro da fronteira pela função transacional e gravadas no ALI sem sair da fronteira

Por exemplo, a fim de manter o salário-hora em dólar para funcionários horistas que trabalhem em outros países com outras moedas, o salário-hora local é informado pelo usuário. Durante o processamento dos dados fornecidos para incluir um funcionário, uma taxa de câmbio é recuperada pelo sistema de moedas, para calcular o salário-hora em dólares. O salário-hora em dólar é mantido no ALI funcionário, como resultado da inclusão do funcionário. O salário-hora em dólar não poderia ser contado como um DER para a EE porque não entra pela fronteira, sendo ao invés disso calculado internamente (i.e., é um dado derivado).

- ❑ atributos obtidos ou referenciados de um ALI ou AIE para a participação no processamento sem sair da fronteira

Por exemplo, quando o pedido do cliente é incluído no sistema, o preço unitário é automaticamente recuperado para cada item pedido e gravado no registro da fatura. O preço unitário não poderia ser contado como um DER para a EE porque não atravessa a fronteira da aplicação quando o usuário inclui o pedido do cliente.

## **Regras de Complexidade e Contribuição de SE/CE**

Esta seção define as regras de ALR e DER usadas para determinar a complexidade e contribuição das saídas externas e consultas externas.

### **Regras Comuns para CEs**

Reconhecendo que uma CE não pode atualizar um ALI, o guia a seguir aplica quando contar RLRs para CEs:

- ❑ Conte um RLR para cada ALI ou AIE lido

### **Guia RLR para SEs**

Reconhecendo que uma SE pode atualizar um ALI, o guia adicional a seguir aplica quando conta RLRs para SEs:

- ❑ Conte um RLR para cada ALI ou AIE lido
- ❑ Conte um RLR para cada ALI mantido
- ❑ Conte apenas um RLR para cada ALI que é lido ou mantido

### **Regras Comuns de DER para SEs e CEs**

As seguintes regras são aplicáveis à contagem de DERs, tanto para SEs quanto para CEs:

- ❑ Revise tudo que cruza (entra e/ou sai) a fronteira
- ❑ Conte um DER para cada campo único, não repetido, reconhecido pelo usuário, que cruza (entra e/ou sai) a fronteira durante o processamento da função de transação

Por exemplo (SE/CE), para gerar uma lista de funcionários, o nome do funcionário é um campo que o usuário fornece para indicar quais funcionários devem ser listados.

Por exemplo (SE/CE), uma mensagem de texto pode ser uma única palavra, uma sentença ou uma frase – uma linha ou parágrafo incluído em um relatório como comentário explicativo conta como um único DER.

Por exemplo (SE/CE), um número de conta ou data fisicamente gravado em vários campos é contado como um DER quando requerido como um único pedaço de informação.

Por exemplo (SE/CE), um gráfico tipo pizza poderia ter uma legenda de categoria e um equivalente numérico na saída gráfica. Conte dois DERs – um para indicar a categoria e outro para o valor numérico.

- ❑ Conte apenas um DER por função de transação para a capacidade de enviar uma mensagem de resposta da aplicação mesmo que haja múltiplas mensagens

Por exemplo (SE/CE), se um usuário tenta solicitar uma listagem, mas não tem acesso à informação, conte um DER para a resposta do sistema.

- ❑ Conte um DER para a habilidade de especificar uma ação a ser executada por função de transação, mesmo que existam vários meios para isto.

Por exemplo (SE/CE), se o usuário pode iniciar a geração de um relatório clicando no botão OK ou pressionando a chave PF, conte um DER para a habilidade de iniciar o relatório.

- ❑ Não conte os itens a seguir como DERs:
  - ❑ literais como títulos de relatório, tela ou identificador do painel, títulos de coluna e títulos de atributos
  - ❑ Por exemplo (SE/CE), literais inclui títulos de relatório, tela ou identificador do painel, títulos de coluna e títulos de campo.
  - ❑ selos gerados automaticamente pelo sistema como atributos de data e hora

Por exemplo (SE/CE), campos de data e hora se são exibidos.

- ❑ variável de paginação, número de páginas e informação de posicionamento; ex. ‘Linhas 37 a 54 de 211’

Por exemplo (SE/CE), número de páginas aparecendo em um relatório.

- ❑ ajudas de navegação como a habilidade de navegar com uma lista utilizando “anterior”, “próximo”, “primeiro”, “último” e seus equivalentes gráficos

Por exemplo (SE/CE), botões anterior e próximo que permite o usuário navegar adiante e atrás de uma lista de registros.

- ❑ atributos gerados dentro da fronteira por uma função de transação e salvo em um ALI sem sair da fronteira

Por exemplo (SE), quando contracheque é impresso, o campo de estado do ALI funcionário é atualizado para indicar que o contracheque foi impresso. Não conte o campo de estado como um DER pois o mesmo não cruza a fronteira.

**Nota:** Uma CE pela regra não pode atualizar uma ALI, então esta regra não se aplica.

- ❑ atributos obtidos ou referenciados de um ALI ou AIE para a participação no processamento sem sair da fronteira

Por exemplo (SE/CE), quando um relatório de contas passadas é criado, a conta dos dados é referenciado para determinar se a conta é passada, mas isso não aparece no relatório. Não conte a conta de dados passados como uma DER pois o mesmo não cruza a fronteira.

## Determinação de Complexidade e Contribuição

A complexidade funcional de cada função de transação deve ser determinada utilizando os passos abaixo.

Passo	Ação
1	Identificar e contar os RLRs e DERs, as regras de contagem de complexidade e contribuição que se encontram na página 7-14 devem ser usadas.
2	A complexidade funcional de cada função de transação deve ser determinada usando o número de RLRs e DERs de acordo com as matrizes a seguir.

### Entrada Externas:

	1 a 4 DERs	5 a 15 DERs	16 ou mais DERs
0 a 1 RLRs	Baixa	Baixa	Média
2 RLRs	Baixa	Média	Alta
3 ou mais RLRs	Média	Alta	Alta

### Saída Externas e Consultas Externas:

	1 a 5 DERs	6 a 19 DERs	20 ou mais DERs
0 a 1 RLRs	Baixa	Baixa	Média
2 a 3 RLRs	Baixa	Média	Alta
4 ou mais RLRs	Média	Alta	Alta

Passo	Ação
-------	------

- 3 O tamanho funcional de cada função de transação deve ser determinada usando o tipo e a complexidade funcional de acordo com as tabelas abaixo.

**Entrada Externas e Consultas Externas:**

Classificação da Complexidade Funcional	Pontos de Função
Baixa	3
Média	4
Alta	6

**Saída externas:**

Classificação da Complexidade Funcional	Pontos de Função
Baixa	4
Média	5
Alta	7



## Dicas para Ajudar na Contagem de EEs, SEs, e CEs

As dicas seguintes podem ajudar na aplicação das regras de contagem de EEs, SEs e CEs e executar a medição de tamanho funcional.

As dicas *não são* regras e não devem ser usadas como regras.

- O dado é recebido de fora da fronteira da aplicação?
  - Observe o fluxo de dados.
  - Identifique onde ocorre a interface entre o usuário e outras aplicações na decomposição funcional do processo.
- O processo é a menor unidade de atividade na perspectiva do usuário?
  - Observe os diferentes formulários impressos ou on-line utilizados.
  - Revise os ALIs para identificar como o usuário agrupa as informações.
  - Identifique onde ocorre a interface com o usuário e outras aplicações na decomposição funcional do processo.
  - Observe o que acontece no sistema manual.
  - Note que uma entrada física, um arquivo de transação ou uma tela pode, quando visto logicamente, corresponder a um certo número de EEs, SEs ou CEs.
  - Note que duas ou mais entradas físicas, arquivos de transação ou telas (ex., abas em uma tela) podem corresponder a uma EE, SE ou CE se a lógica de processamento for idêntica.
  - Lembrar que dois ou mais relatórios físicos, telas ou arquivos de saída em lote podem corresponder a uma SE/CE se o processamento lógico for idêntico.
- O processo é autocontido e deixa o negócio em um estado consistente?
  - Revise outras entradas externas, saídas externas e consultas externas para entender como o usuário trabalha com a informação.
  - Analise o diagrama de processos para obter dicas.
  - Observe o que acontece no sistema manual.
  - Confira a consistência com outras decisões.
- Identifique a intenção primária do processo elementar antes de classificá-lo como uma EE, SE ou CE.
- A identificação do(s) processo(s) elementar(es) é baseada em um entendimento e interpretação comum dos requisitos entre o usuário e os desenvolvedores.

- Cada elemento de uma decomposição funcional pode não ser mapeado para um único processo elementar.
- A identificação do processo elementar requer a interpretação dos requisitos do usuário.
- O processamento lógico é único de outras EEs, SEs e CE?
  - Identificar entradas e saída batchs baseada no processamento lógico requerido.
  - Uma transação que ocorra em entrada física múltipla, arquivos de transação ou telas, mas o qual tem processamento lógico idêntico, tipicamente corresponde para uma função de transação (EE, SE, CE).
  - Lembre que a ordenação ou reorganização de um conjunto de dados não torna o processamento lógico único.
- Os atributos de dados são diferentes dos de outras EEs, SEs e CEs?
  - Se os atributos de dados parecem ser um subconjunto de atributos de dados de outra EE, SE e CE, certifique-se que dois processos elementares são requeridos pelo usuário — um para os atributos de dados principais e um para os subconjuntos.
- Conte apenas um ALR para cada ALI/AIE referenciado mesmo se os ALI/AIE tiverem vários RLRs.

## Dicas Adicionais para Ajudar na Contagem de SEs e CEs

- O processo é a menor unidade de atividade na perspectiva do usuário?
  - Uma SE ou CE pode ser disparada por um processo dentro da fronteira da aplicação.

Por exemplo, o usuário solicita que um relatório com todos os salários de funcionários alterados seja enviado para a área de orçamento a cada 8 horas, com base em um relógio interno.

Situação A. O relatório contém nome do funcionário, CPF e salário-hora, todos recuperados do arquivo de funcionário. Esta é a menor unidade de atividade na perspectiva do usuário, não contém fórmulas matemáticas ou cálculos, e nenhum ALI é mantido no processo. Esta é uma CE.

Situação B. O relatório contém nome do funcionário, CPF e salário-hora, todos recuperados do arquivo de funcionário. O relatório também inclui o percentual de mudança no salário do funcionário, calculado a partir dos dados do arquivo de funcionário. Esta é a menor unidade de atividade na perspectiva do usuário, e nenhum ALI é mantido no processo. No entanto, dado que o processo contém uma fórmula matemática, esta é uma SE.

- Dados derivados de uma SE não precisam ser mostrados na saída.

Por exemplo, todo mês, um relatório é gerado listando todos os funcionários a serem avaliados nos próximos 30 dias. Os registros são selecionados calculando-se a data da próxima avaliação com base na data da última avaliação do funcionário, que é um campo do arquivo de funcionários, e a data atual + 30 dias. Este seria contado como uma SE, e não como uma CE.

Esta página foi deixada em branco intencionalmente.

## Índice da Parte 2

### A

AIEs. ver Arquivos de interface externa  
 ALIs. Ver Arquivos Lógicos Internos  
 Análises de pontos de função  
     procedimentos por capítulo, 2-2  
 Aplicação  
     contagem de pontos de função, 4-2  
 Arquivo, 6-1  
 Arquivos de interface externa  
     definição, 6-2  
     dicas para ajudar na contagem, 6-10  
     diferenças de ALIs, 6-2  
     exemplo de identificável pelo usuário, 6-3  
     exemplo de informação de controle, 6-2  
     exemplo de mantido, 6-3  
     exemplo de processos elementares, 6-3  
     procedimentos de complexidade e contribuição, 6-8  
     regras de complexidade e contribuição, 6-5  
     regras de DER, 6-5  
     subgrupos obrigatórios para RLRs, 6-7  
     subgrupos opcionais para RLRs, 6-7  
     tabela de tradução, 6-8

### C

Ciclo de Vida, 3-1, 3-8  
 Classificação de Cada Processo Elementar, 7-13  
 Complexidade e contribuição  
     definição ALI/AIE, 6-5

Contagem estimada, 4-4  
 Contagem final, 4-4  
 Contagem de pontos de função não ajustadas  
     tipos de função transacional, 7-1

### D

Definições  
     arquivo de interface externa, 6-2  
     arquivos internos lógicos, 6-2  
     complexidade e contribuição, 6-5  
     informação de controle, 6-2  
     mantido, 6-3  
     processo elementar, 6-3  
     requisitos funcionais do usuário, 1-3  
     usuário identificável, 6-3  
 Definições e Regras de Complexidade e Contribuição, 7-14  
 DER  
     definição, 6-5  
 Desenvolvimento de Projeto  
     contagem de pontos de função, 4-2  
 Determinar Processos Elementares Únicos, 7-11  
 Diagrama  
     Tipos de contagem, 4-4  
 Dicas de contagem. Ver Dicas  
 Dicas  
     contagem de AIEs, 6-10  
     contagem de ALIs, 6-10  
     fronteira, 5-6

Diferenças entre ALIs e AIEs, 6-2

Documentação, 3-1, 3-3

Útil, 3-9

Documentação Recolhida Disponível, 3-1

## E

EEs, SEs e CEs

Definição de intenção primária, 7-3

Dicas, 7-21

Informação de Controle, 7-5

Processamento Lógico, 7-5

Processos Elementares, 7-5

EE/CE Guia de Complexidade e Contribuição, 7-17

Entrada externa

matriz de complexidade, 7-19

Exemplos

informação de controle, 6-2

processo elementar para ALIs/AIEs, 6-3

ALI/AIE subgrupos obrigatórios para RLRs, 6-7

ALI/AIE subgrupos opcional para RLRs, 6-7

mantido para ALIs/AIEs, 6-3

usuário identificável para ALIs/AIEs, 6-3

## F

Fronteiras

dicas, 5-6

regras, 5-5

## G

Guia de Complexidade e Contribuição de EE, 7-2, 7-15

## I

Identificar Cada Processo Elementar, 7-10

Informação de controle

exemplo ALI/AIE, 6-2

## M

Mantido

definição, 6-3

exemplo de ALI/AIE, 6-3

Matrizes. Ver Matrizes de complexidade

Matrizes de Complexidade

entradas externas, 7-19

arquivos de interface externa, 6-8

arquivos internos lógicos, 6-8

## N

Normas Internacionais

ISO/IEC 14143-12007, 1-2

## P

Procedimentos de Complexidade e Contribuição

arquivos de interface externa, 6-8

arquivos internos lógicos, 6-8

Procedimentos

por capítulo, 2-2

arquivos de interface externa de complexidade e contribuição, 6-8

arquivos internos lógicos de complexidade e contribuição, 6-8

passos, 2-2

Processos Elementares

exemplo de AIE, 6-3

exemplo de ALI, 6-3

Projeto de melhoria

contagem de ponto de função, 4-2

## R

Regras

fronteira, 5-5

complexidade e contribuição para ALIs/AIEs, 6-5

DERs para ALIs/AIEs, 6-5

ALI/AIE mandatory subgroups, 6-7

ALI/AIE optional subgroups, 6-7

Regras de Complexidade. Ver Regras de Complexidade e contribuição

Regras de Complexidade e contribuição

arquivos de interface externa, 6-5

arquivos lógicos internos, 6-5

Regras de DER

ALIs/AIEs, 6-5

Regras de RLR

subgrupos obrigatórios para ALIs/AIEs, 6-7

subgrupos opcionais para ALIs/AIEs, 6-7

RLR

definição, 6-7

Requisitos Funcionais do Usuário

definição, 1-3

## S

Scope creep, 4-4

Subgrupos obrigatórios, 6-7

Subgrupos opcionais, 6-7

## T

Tabela de intenção primária, 7-4

Tabela de processamento lógico, 7-8

Tabelas de tradução

ALIs/AIEs, 6-8

Tamanho Funcional, 3-10

Tipos de dados elementares. Ver DER

Tipos de elemento de registro. Ver RLR

Tipos de funções de dados

introdução, 6-1

visão geral, 2-5

Tipos de função transacional

visão geral, 2-6

Tipo de contagem

contagem estimada e final, 4-4

## U

Usuário identificável

definição, 6-3

exemplo de ALI/AIE, 6-3

## V

Visão do Usuário, 3-2

Esta página foi deixada em branco intencionalmente.