

Manual de Práticas de Contagem de Pontos de Função

Versão 4.3.1



**International Function Point
Users Group (IFPUG)**

Manual de Práticas de Contagem de Pontos de Função

Versão 4.3.1

Chairperson, Counting Practices Committee
Adri Timp
Equens SE, Netherlands
cpc@ifpug.org

Nota Importante Referente à Tradução Brasileira do CPM

A tradução brasileira do IFPUG Counting Practices Manual deve ser vista como um documento de apoio à utilização do manual original em inglês e não como uma fonte independente. Os tradutores, revisores e demais envolvidos no processo de tradução, especialmente o IFPUG, ***não assumem qualquer responsabilidade pela exatidão da presente tradução***, especialmente quanto à sua utilização em relações comerciais e contratuais de qualquer tipo, inclusive como fonte de informações para o exame de certificação CFPS do IFPUG. Em resumo: ao utilizá-la, o usuário desta tradução o faz por sua própria conta e risco, não existindo qualquer garantia, implícita ou explícita, quanto à adequabilidade desta tradução para qualquer finalidade específica. Quando a exatidão das informações for importante, deve ser utilizado o manual original em inglês – IFPUG Counting Practices Manual V. 4.3. Esta tradução foi concluída em 27/05/2010.

© 2010 IFPUG. Todos os direitos reservados. International Function Point Users Group, 2009. Os filiados ao IFPUG podem reproduzir partes deste documento em seus manuais internos de práticas de contagem. Se forem utilizadas partes deste documento, o texto a seguir deverá aparecer na página-título do documento derivado: "Este documento contém material extraído do Manual de Práticas de Contagem do IFPUG. Tal material é reproduzido neste documento com a permissão do IFPUG."

ISBN 978-0-9753783-5-9

Versão 4.3.1, Janeiro de 2010

Esta versão substitui a Versão 4.2.1, agora obsoleta.
As informações aqui contidas são modificadas periodicamente.

Comitê de Práticas de Contagem

Bonnie S. Brown, HP
Royce Edwards, Software Composition Technologies
E. Jay Fischer, JRF Consulting, Inc.
David Garmus, The David Consulting Group
Janet Russac, Software Measurement Expertise, Inc.
Adri Timp, Equens SE, Netherlands
Peter Thomas, Steria

Equipe de Tradução da Versão em Português

Coordenador : Márcio Silveira, HP
Tradutores e Revisores : Mauricio Aguiar, TI Métricas
Diana Baklizky, TI Métricas
Teresa C. S. Zenga Beraldo, Bradesco
Sandra Bica, HP
Sergio Brígido, HP
André Margoni, TI Métricas
Guilherme Simões, Fatto
Carlos Vazquez, Fatto

Para informações sobre cópias adicionais deste manual, entrar em contato com:

IFPUG
191 Clarksville Road
Princeton Junction, NJ 08550
U.S.A.
(609) 799-4900
E-mail: ifpug@ifpug.org
Web: <http://www.ifpug.org>

O método FSM do IFPUG apresentado neste manual tem sua origem no CPM 4.0, publicado em 1990. Muitas pessoas serviram como membros do CPC ao longo dos anos, aprimorando mutuamente sua visão e percepção em numerosas discussões complexas sobre a metodologia. Graças à contribuição dos membros atuais e anteriores, o método evoluiu até o padrão ISO FSM atual. O CPM é um acúmulo das contribuições de todos que serviram no CPC. O CPC também foi agraciado com o apoio e contribuição de nossos Diretores de Padrões de Contagem, aí incluídos Carol Dekkers, Bruce Rogora e Mary Bradley, assim como o esforço de Carol Dekkers na condução de ambos documentos pelo processo de aprovação da ISO. O IFPUG estende seu agradecimento a todos os membros atuais e anteriores do CPC, Diretores de Padrões de Contagem e Representantes na ISO.

Membros do CPC Atuais e Anteriores

Allan Albrecht, Fundador da APF
Kim Albee (CPM 4.0)
Maarten Barth (CPM 4.0)
Andy Belden (CPM 4.0)
Angela Benton (CPM 4.1)
Mary Bradley (CPM 4.0, CPM 4.1, Past Chair)
Bonnie Brown (CPM 4.2, CPM 4.3, Vice Chair)
Kevin Chinoy (CPM 4.1)
Jean-Marc Deshamais (CPM 4.0)
Rob Donnellan (CPM 4.0)
Ian Drummond (CPM 4.0)
Martin D'Souza (CPM 4.2, CPM 4.3)
Boyd Edmiston (CPM 4.0)
Royce Edwards (CPM 4.3)
Peter Fagg (CPM 4.1)
Jay Fischer (CPM 4.2, CPM 4.3)
Sean Furey (CPM 4.1)
Steve Galea (CPM 4.1)
Barbara Gardner (CPM 4.0)
David Garmus (CPM 4.0, CPM 4.1, CPM 4.2, CPM 4.3)
Jim Glorie (CPM 4.1, CPM 4.2)
Paul Goodman (CPM 4.0)
Phil Hain (CPM 4.0)
David Herron (CPM 4.0)
Steve Hone (CPM 4.2)
Bob Huckaby (Past Chair)
Valerie Marthaler (CPM 4.1, CPM 4.2, CPM 4.3, Past Chair)
Frank Mazzucco (CPM 4.0)
Pam Morris (CPM 4.0, CPM 4.1, CPM 4.2)
Jolijn Onvlee (CPM 4.0, CPM 4.1)
Bruce Paynter (CPM 4.2)
Dave Phillips (CPM 4.1)
Ben Porter (CPM 4.0, Past Chair)
Robin Ragland (CPM 4.0, CPM 4.1, Past Chair)
Roger Roy (CPM 4.0)
Eberhard Rudolph (CPM 4.0)
Grant Rule (CPM 4.1)

Membros Anteriores do CPC (cont.)

Bill Rumpf (CPM 4.0)

Janet Russac (CPM 4.3)

Michael Schooneveldt (CPM 4.0, CPM 4.1)

Linda Smith (CPM 4.0, Past Chair)

Jack Sprouls (CPM 4.0)

Denis St. Pierre (CPM 4.0, CPM 4.1)

Peter Thomas (CPM 4.3)

Koni Thompson (CPM 4.1, CPM 4.2)

Adri Timp (CPM 4.1, CPM 4.2, CPM 4.3, Chair)

Tony Tiongson (CPM 4.0)

Stephen Treble (CPM 4.1)

Eddy van Vliet (CPM 4.1, CPM 4.2, CPM 4.3)

Terry Vogt (CPM 4.2)

Gary Walker (CPM 4.0, CPM 4.1)

Ewa Wasylkowski (CPM 4.0, CPM 4.1)

Esta página foi deixada em branco intencionalmente.

Conteúdo

Prefácio	iii
Introdução ao Manual de Práticas de Contagem	vii

Parte 1 FSM

Prefácio	i
Introdução	iii
Capítulo 1	Escopo 1
Capítulo 2	Referências Normativas 1
Capítulo 3	Termos e Definições 1
Capítulo 4	Abreviaturas 8
Capítulo 5	Processo de Medição 8
Anexo A	A-1

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

Capítulo 1	Introdução 1-1
Capítulo 2	Visão Geral do Método FSM do IFPUG 2-1
Capítulo 3	Reunir Informações Disponíveis 3-1
Capítulo 4	Determinar Tipo de Contagem 4-1
Capítulo 5	Determinar Escopo da Contagem e Fronteiras da Aplicação 5-1
Capítulo 6	Medir Funções de Dados 6-1
Capítulo 7	Medir Funções de Transação 7-1
Índice	i-1

Parte 3 Práticas de Contagem

Introdução	1-1
Capítulo 1 Dados de Código	1-3
Capítulo 2 Arquivos Lógicos	2-1
Capítulo 3 Dados Compartilhados	3-1
Capítulo 4 Projetos de Melhoria e Atividades de Manutenção ..	4-1
Capítulo 5 Atividade de Conversão de Dados	5-1
Índice	i-1

Parte 4 Exemplos

Introdução	1-1
Capítulo 1 Exemplos de Contagem de Funções de Dados	1-3
Capítulo 2 Exemplos de Contagem de Funções de Transação ..	2-1
Índice	i-1

Parte 5 Apêndices e Glossário

Apêndice A Tabela de Cálculo de Tamanho Funcional	A-1
Apêndice B A Mudança em Relação à Versão Anterior	B-1
Apêndice C Tamanho Funcional Ajustado	C-1
Glossário	G-1

Prefácio

- Introdução** O uso de pontos de função como uma medida do tamanho funcional do software cresceu na última década: de poucas organizações interessadas, o mesmo passou para uma lista expressiva de organizações no mundo inteiro.
- IBM CIS & A Guidelines 313** Nos anos 70, Allan Albrecht da IBM definiu os conceitos que permitiram a medição do resultado de projetos de desenvolvimento de software. Estas definições foram ampliadas na *IBM CIS & A Guideline 313, AD/M Productivity Measurement and Estimate Validation*, datada de 1 de Novembro de 1984.
- Release 2.0** O crescimento do uso de pontos de função trouxe um grande aumento no número de aplicações medidas. Esta ampliação da aplicação desafiou a descrição original da medida e tornou necessária a criação de um guia para interpretação das regras originais em novos ambientes. Isto foi refletido na *Release 2.0* (abril de 1988) do *International Function Point Users Group (IFPUG) Function Point Counting Practices Manual*.
- Release 3.0** A *Release 3.0* (abril de 1990) do *IFPUG Function Point Counting Practices Manual* foi o principal marco na evolução da medição do tamanho funcional. Pela primeira vez, o Comitê de Práticas de Contagem do IFPUG trabalhou para transformar o que era uma coleção de muitas interpretações das regras em um documento verdadeiramente coerente, representando uma visão consensual das regras de contagem de ponto de função. Neste sentido, este foi o primeiro passo para o estabelecimento de padrões reais para a medição de pontos de função, os quais poderiam ser aplicados a diversas organizações.

Release 4.0 A *Release 4.0* (janeiro de 1994) foi o marco seguinte na evolução da medição do tamanho funcional. Essa *release* abordou o uso de pontos de função nas fases iniciais do desenvolvimento dos projetos, para estimar o tamanho do projeto utilizando disciplinas de engenharia da informação. O número rapidamente crescente de aplicações com janelas de interface gráfica do usuário (GUI) exigiu que fosse incluída a contagem de GUI nessa *release*. Devido a cada vez mais contagens estarem acontecendo em uma vasta variedade de situações, a *release* colocou ênfase na interpretação e prática das regras de contagem. Exemplos foram incluídos ao longo da documentação e estudos de casos complementaram o material. Finalmente, a *Release 4.0* continuou a esclarecer e aumentar a consistência da contagem de pontos de função.

Release 4.1 A *Release 4.1* (Janeiro de 1999) forneceu esclarecimentos para as regras existentes, regras novas ou corrigidas que abordam situações anteriormente não documentadas, bem como novas dicas e exemplos para auxiliar o entendimento. O Comitê de Práticas de Contagem do IFPUG revisou e processou solicitações dos filiados, seguindo o Processo de Revisão do Manual contido no Capítulo 1 deste manual. As revisões incluídas na 4.1 esclarecem:

- A identificação de usuário, processo elementar e informações de controle
- A diferença entre Saídas Externas (SEs) e Consultas Externas (CEs)
- A identificação de DERs – Dados Elementares Referenciados (tipos de dado elementares) e RLR – Registros Elementares Referenciados (tipos de registro elementares) para as funções de dados
- A identificação de DERs - Dados Elementares Referenciados (tipos de dado elementares) para as funções de transação

A *Release 4.1* continua o processo de esclarecimento e de melhoria da consistência da contagem de pontos de função. Finalmente, com exceção das 14 Características Gerais do Sistema, esta *release* foi elaborada para estar de acordo com o padrão ISO existente, se e quando algum guia de conformidade vier a ser tornar padrão.

Release 4.1.1 A *Release 4.1.1* (abril de 2000) corrigiu um pequeno número de erros, inclusive tipográficos.

Release 4.2 A *Release 4.2* (janeiro de 2004) não modifica qualquer regra publicada anteriormente, porém fornece esclarecimentos e melhores interpretações das regras existentes, que irão favorecer o aumento na consistência entre os contadores.

As regras e o processo da análise de pontos de função (APF) do IFPUG são concisos e fáceis de usar. Para refletir isto e tornar o Manual de Práticas de Contagem (CPM) cada vez mais atraente como um manual de referência, o Comitê de Práticas de Contagem (CPC) decidiu reestruturar o CPM 4.2 em quatro partes:

1. Processo e Regras
2. Práticas de Contagem
3. Exemplos
4. Apêndices

A fim de ajudar tempestivamente os praticantes da APF do mundo inteiro, o CPC publicou inicialmente os resultados de sua pesquisa como publicações separadas, anexas ao CPM:

- Contagem de Arquivos Lógicos (setembro de 2001)
- A APF em um Ambiente de Melhoria (abril de 2002)
- Contagem de Dados de Código (setembro de 2003)
- Contagem de Dados Compartilhados (setembro de 2003)

Estes documentos foram agora incorporados como capítulos na Parte 2 deste CPM.

O CPC acredita que o CPM 4.2, com estas diretrizes e exemplos adicionados, irá garantir resultados mais consistentes entre os Especialistas Certificados em Ponto de Função.

Release 4.2.1 A *Release 4.2.1* (janeiro de 2005) corrigiu um pequeno número de erros tipográficos.

Release 4.3 Release 4.3 (janeiro de 2010). As regras e o processo da análise de pontos de função (APF) do IFPUG são concisos e fáceis de usar. Para refletir isto e tornar o Manual de Práticas de Contagem (CPM) cada vez mais atraente como um manual de referência, o Comitê de Práticas de Contagem (CPC) decidiu reestruturar o CPM 4.3 de modo a compatibilizá-lo com os padrões de formatação da ISO.

Além disso, a *Release 4.3* contém pequenas modificações e fornece novos exemplos, bem como explicações e interpretações melhoradas para as regras existentes, que irão aumentar ainda mais a consistência entre contadores.

O CPC acredita que o CPM 4.3, com seu novo formato e exemplos, irá garantir resultados mais consistentes entre os Especialistas Certificados em Pontos de Função.

Release 4.3.1 Release 4.3.1 (janeiro de 2010). Corrigiu alguns erros de impressão e incorporou algumas revisões editoriais da ISO.

Releases Futuras Este documento destina-se a ser um documento vivo. Devemos ser capazes de identificar como contar novos ambientes conforme os mesmos são introduzidos. Precisamos ser capazes de fazer isto mantendo a validade das contagens já realizadas. Isto não será uma tarefa fácil, no entanto é essencial se quisermos ter a capacidade de medir nosso progresso na entrega de valor agregado aos usuários e às organizações que representam.

O Comitê de Prática de Contagens deseja agradecer a todos aqueles que ajudaram na pesquisa e na produção deste manual.

Adri Timp
Chairperson, Comitê de Práticas de Contagem

Introdução ao Manual de Práticas de Contagem

Introdução Esta introdução define os objetivos do manual e o processo de revisão. Também descreve publicações que estão relacionadas a este manual.

Conteúdo Este capítulo inclui as seguintes seções:

Tópico	Página
Objetivos do Manual de Práticas de Contagem	vii
Documentos Utilizados na <i>Release</i> 4.3	viii
Público-Alvo	vii
Organização do Manual de Práticas de Contagem	xi
Processo de Revisão Manual	xi
Frequência das Mudanças	xii
Processo de Mudança	xii
Documentação Associada do IFPUG	xv
Requisitos de Treinamento	xvii

Objetivos do Manual de Práticas de Contagem

Os principais objetivos do *IFPUG Counting Practices Manual, Release 4.3*, são

- Manter conformidade com a norma ISO/IEC 14143-1:2007 *Information technology – Software measurement – Functional size measurement – Definition of concepts*
- Prover uma descrição clara e detalhada da contagem de pontos de função
- Garantir que as contagens sejam consistentes com as práticas de contagem dos *affiliates* do IFPUG
- Fornecer um guia para permitir a contagem de pontos de função a partir dos entregáveis das metodologias e técnicas mais conhecidas
- Prover um entendimento comum para permitir que os fornecedores de ferramentas forneçam suporte automatizado para a contagem de pontos de função

Documentos Utilizados na *Release 4.3*

A seguinte documentação foi utilizada para desenvolver esta *release*:

- O método de APF do IFPUG está baseado no “*IBM CIS & A Guideline 313, AD/M Productivity Measurement and Estimate Validation*”, datado de 1 de Novembro de 1984. A metodologia de contagem de pontos de função descrita em 313 é geralmente referenciada como Albrecht 1984.
- A versão atual deste manual, CPM 4.3, está baseada principalmente no *IFPUG Function Point Counting Practices Manual, Release 4.2.1*.
- O CPM 4.3 foi projetado para manter conformidade com a ISO/IEC 14143-1:2007 *Information technology – Software measurement – Functional size measurement – Definition of concepts*.
- “*Framework for Functional Sizing*”; este documento do IFPUG explica que o tamanho do produto possui três dimensões: tamanho funcional, tamanho técnico e tamanho de qualidade. O método de APF do IFPUG fornece uma medida para o tamanho funcional.
- As questões não suficientemente cobertas nas fontes listadas acima foram decididas pelo Comitê de Práticas de Contagem do IFPUG, com base nas variações das práticas de contagem existentes e validadas através de estudos de impacto.

A partir de sua publicação, este manual deve ser considerado o padrão do IFPUG para a contagem de pontos de função. É imperativo que cada membro do IFPUG assuma um papel ativo para garantir a consistência nas contagens. A aderência dos filiados do IFPUG a este padrão em muito contribuirá para a consistência das contagens.

Público-Alvo

O padrão contido neste manual deve ser aplicado por qualquer pessoa que utilize a análise de pontos de função para medição do tamanho funcional. Este manual foi elaborado para ser utilizado tanto por iniciantes na contagem de pontos de função, assim como por aqueles com experiência intermediária ou avançada.

Organização do Manual de Práticas de Contagem

Há cinco partes principais no Manual de Práticas de Contagem (CPM):

- Parte 1: FSM-O método de medição de tamanho funcional do IFPUG
- Parte 2: A Transição – Aplicando o Método de Medição de Tamanho Funcional do IFPUG
- Parte 3: Práticas de Contagem
- Parte 4: Exemplos
- Parte 5: Apêndices e Glossário

Parte 1 – FSM

A Parte 1 é o método de medição de tamanho funcional do IFPUG, contendo as respectivas regras. Não é suficiente aprender somente as palavras e a gramática para falar um idioma como um nativo. Tais elementos fornecem apenas uma estrutura de referência. É necessária experiência com o idioma para entender como o mesmo é falado na prática, como devem ser aplicadas as regras gramaticais, quais expressões idiomáticas são comuns e assim por diante. O mesmo raciocínio é verdadeiro para a APF. É necessário o conhecimento do processo e das regras, conforme exposto na Parte 1, mas tal conhecimento não constitui condição suficiente para a correta aplicação da APF. É por isso que o CPM contém as partes abaixo.

Parte 2 – A Transição

A Parte 2 provê orientação para o dimensionamento de software segundo o método de medição funcional do IFPUG (FSM), incluído como Parte 1 deste Manual de Práticas de Contagem do IFPUG.

Parte 3 – Práticas de Contagem e Parte 4 - Exemplos

As Partes 3 e 4 fornecem exemplos detalhados para explicar os conceitos e regras das práticas de contagem. Cada exemplo deve ser considerado separadamente e por seus próprios méritos. Como a intenção de cada exemplo é ilustrar um cenário específico, podem existir variações entre os exemplos. Embora os exemplos constantes do manual lidem com assuntos similares, os mesmos não pretendem representar um único conjunto de requisitos do usuário.

Parte 5 – Apêndices e Glossário

A Parte 5 contém valiosas informações adicionais, tais como *templates* de cálculo prontos para uso, a transição do CPM 4.2 (e CPM 4.2.1) para o CPM 4.3, as Características Gerais dos Sistemas e o glossário.

A princípio, cada parte é independente das demais.

Processo de Revisão do Manual

Esta seção explica a frequência das mudanças no Manual de Práticas de Contagem e define o processo de mudança.

Frequência das Mudanças

No mês de janeiro de cada ano uma nova versão do Manual de Práticas de Contagem *pode* entrar em vigor. A mesma incluirá definições, regras ou práticas de contagem, novas ou alteradas, que tenham sido concluídas pelo Comitê de Práticas de Contagem (CPC) desde a versão anterior.

Processo de Mudança

As atividades seguintes esboçam o processo para a inclusão ou alteração de informações no Manual de Práticas de Contagem. Explicações sobre cada atividade seguem-se à tabela.

Passo	Ação
1	A questão é submetida ao CPC.
2	A questão é designada para pesquisa.
3	O CPC revisa e discute a questão.
4	O CPC apresenta uma solução proposta aos filiados do IFPUG.
5	Um estudo de impacto é iniciado caso a mudança proposta tenha algum impacto sobre as contagens existentes.
6	É tomada a decisão final.
7	Os filiados ao IFPUG são informados da decisão.
8	As mudanças são incluídas e entram em vigor na <i>release</i> seguinte do Manual de Práticas de Contagem.

Questão Submetida O leitor submete idéias, mudanças, ou questões ao Comitê de Práticas de Contagem, enviando um e-mail para ifpug@ifpug.org ou cpc@ifpug.org

Pesquisa Designada	Um membro do CPC recebe a responsabilidade de identificar todas as alternativas, a <i>racional</i> e o impacto potencial de cada alternativa se implementada. Por ocasião do levantamento de alternativas é efetuado um exame completo de todos os padrões de contagem e artigos históricos existentes. Adicionalmente, é realizado um esforço para determinar o que se acredita ser a <i>prática comum</i> .
Revisão do CPC	O CPC revisa e discute a <i>racional</i> para cada alternativa e seu impacto potencial. A revisão e discussão pode resultar em uma proposta de mudança, ou as mesmas podem levar o comitê a rejeitar a solicitação de mudança.
Solução Proposta	<p>Uma proposta de solução é submetida aos filiados ao IFPUG, sendo solicitados comentários por escrito.</p> <p>Um cópia das mudanças propostas é enviada aos contatos junto ao IFPUG das organizações filiadas. A proposta também pode ser anunciada e distribuída durante uma conferência do IFPUG. Esta última alternativa depende da época da reunião do comitê e não da programação da conferência.</p>
Estudo de Impacto Iniciado	<p>O CPC tem adotado uma postura conservadora quanto ao início de estudos de impacto. Se for possível que uma <i>prática comum</i> tenha que ser modificada, ou que várias organizações ou tipos de aplicação sejam impactados pela mudança, um estudo de impacto é iniciado.</p> <p>O sucesso do estudo de impacto é responsabilidade de cada filiado ao IFPUG. Se o CPC receber <i>feedback</i> escrito indicando que há pouco ou nenhum impacto, o estudo será descontinuado.</p>
Decisão Final Tomada	<p>O comitê toma uma decisão final utilizando resultados da pesquisa, comentários escritos dos filiados e o estudo de impacto.</p> <p>O comitê pode efetuar mais de uma iteração dos Passos 2 a 5 (da pesquisa ao estudo de impacto) antes de tomar uma decisão final. A decisão final pode resultar em uma mudança, ou o comitê pode decidir que não há razão para mudança.</p>
Decisão Comunicada	<p>A decisão final é comunicada por escrito aos filiados ao IFPUG, através dos contatos junto ao IFPUG existentes nas diversas organizações.</p> <p>Se algum resultado do estudo de impacto houver contribuído para a decisão, os resultados e uma recomendação sobre como minimizar o impacto da mudança também serão comunicados.</p>

**Data de
Vigência da
Decisão**

O Manual de Práticas de Contagem será atualizado a fim de incluir as decisões. A data de vigência das decisões será a data da próxima *release* do manual no mês de janeiro subsequente.

Documentação Relacionada do IFPUG

Este Manual de Práticas de Contagem é um módulo na documentação do IFPUG. Todos os documentos se complementam.

A tabela seguinte descreve as outras publicações

Documento	Descrição
Folheto do IFPUG (disponível)	<p>Esta publicação é uma introdução ao <i>International Function Point Users Group</i>. Inclui um breve histórico da organização, introduz a análise de pontos de função e define o objetivo do IFPUG. O folheto também inclui uma solicitação de filiação.</p> <p>Público-alvo: Esta publicação destina-se a qualquer pessoa que deseje ter uma visão geral do IFPUG, ou que deseje se associar.</p>
IFPUG: Estrutura Organizacional e Serviços (disponível)	<p>Esta publicação descreve os serviços do IFPUG, lista o quadro de diretores, comitês e organizações filiadas ao redor do mundo.</p> <p>Público-alvo: Esta publicação destina-se a qualquer pessoa que deseje informações sobre o IFPUG.</p>
Guia para a Medição de Software (Data de Publicação: agosto de 2004)	<p>Este manual fornece uma visão geral das métricas de software para as organizações que estejam trabalhando na criação ou melhoria de programas de medição de software. O manual aborda o gerenciamento de sistemas e de clientes, fornece justificativas de alto nível para a medição de software e examina os componentes dos programas de medição eficazes.</p> <p>Público-alvo: Este manual é direcionado a filiados ao IFPUG, Coordenadores de Pontos de Função, pessoas que preparam relatórios para a gerência e outros com conhecimento e que trabalham diretamente com pontos de função.</p>
Guia Rápido de Referência de Contagem (Data de Publicação: Janeiro de 2004)	<p>Este guia rápido de referência é um resumo das regras e procedimentos da contagem de pontos de função.</p> <p>Público-alvo: Este resumo é destinado a qualquer pessoa que esteja aplicando a análise de pontos de função.</p>
Guia Rápido de Referência de Contagem – Tamanho Ajustado (Data de Publicação: 2010)	<p>Este guia rápido de contagem é um resumo das Características Gerais dos Sistemas.</p> <p>Público-alvo: Este guia é destinado a qualquer pessoa que esteja utilizando as Características Gerais dos Sistemas, que são de uso opcional.</p>

Documento	Descrição
Estudos de Caso de Análise de Pontos de Função (Datas das Publicações: Estudo de Caso 1, Versão 3.0: Setembro de 2005 – CPM 4.2 Estudo de Caso 2, Versão 3.0: Março de 2006 – CPM 4.2 Estudo de Caso 3, Versão 2.0: Setembro de 2001 – CPM 4.1 Estudo de Caso 4, Versão 2.0: Setembro de 2005 – CPM 4.2)	Os estudos de caso ilustram as principais técnicas de contagem que constituem o Manual de Práticas de Contagem de Pontos de Função. Os casos ilustram contagens de pontos de função para uma aplicação exemplo. Incluem a contagem que ocorre ao final da fase de análise do desenvolvimento de software e depois da construção do sistema. Público-alvo: Os estudos de caso destinam-se a pessoas iniciantes na análise de pontos de função, bem como àquelas com experiência intermediária e avançada.
Glossário do IFPUG (Disponível com o CPM e com o Guia para a Medição de Software)	Este é um glossário completo, que define os termos usados pelas publicações do IFPUG. Público-alvo: O glossário é recomendado para qualquer pessoa que receba algum outro documento do IFPUG, ou qualquer pessoa que precise das definições dos termos do IFPUG.
“A Framework for Functional Sizing”, IFPUG, Setembro de 2003	Este documento explica que o tamanho do produto contém três dimensões: tamanho funcional, tamanho técnico e tamanho de qualidade. O método de APF do IFPUG fornece uma medida para o tamanho funcional.
“IT Measurement: Practical Advice from the Experts”, Addison-Wesley, Abril de 2002	Este livro é uma excelente compilação de artigos escritos por especialistas no campo da Tecnologia da Informação. Foi compilado pelo IFPUG para incluir o pensamento recente quanto à aplicação de métricas de software na prática.

Requisitos de Treinamento

As avaliações de usabilidade desta publicação verificaram que apenas a leitura do Manual de Práticas de Contagem não constitui treinamento suficiente para a aplicação da contagem de pontos de função no nível ótimo. O treinamento é recomendado, especialmente para os iniciantes na contagem de pontos de função.

Nota: No treinamento em pontos de função, esteja certo de que você seja treinado utilizando materiais certificados pelo IFPUG. Ligue para o Escritório Executivo do IFPUG em 0XX 1 609-799-4900 para obter uma lista de instrutores com cursos certificados.

Além das informações específicas sobre pontos de função, este manual inclui a utilização de termos da análise e projeto estruturados, tais como sistemas de negócio e entidade. O glossário inclui definições desses termos, mas o Manual de Práticas de Contagem não inclui explicações detalhadas das técnicas de análise e projeto estruturados. Dessa forma, nem todo o material será aplicável ou útil se você não tiver sido treinado nas técnicas de análise e projeto estruturados.

Esta página foi deixada em branco intencionalmente.

Parte 1 - FSM

Página intencionalmente deixada em branco.

Conteúdo

Prefácio	ii
Introdução.....	iii
1 Escopo	1
1.1 Propósito	1
1.2 Conformidade.....	1
1.3 Aplicabilidade.....	1
1.4 Público	1
2 Referências normativas.....	1
3 Termos e definições	1
4 Abreviaturas	8
5 Processo de Medição	9
5.1 Visão Geral	9
5.2 Reunir a documentação disponível	9
5.3 Determinar o escopo e fronteira da contagem, identificando os Requisitos Funcionais do Usuário	10
5.4 Medir funções de dados.....	11
5.4.1 Visão Geral	11
5.4.2 Identificar e agrupar todos os dados lógicos em funções de dados	11
5.4.3 Classificar cada função de dados como um ALI ou AIE.....	12
5.4.4 Contar os DERs para cada função de dados	12
5.4.5 Contar os RLRs para cada função de dados	13
5.4.6 Determinar a complexidade funcional de cada função de dados.....	14
5.4.7 Determinar o tamanho funcional de cada função de dados.....	14
5.5 Medir funções de transação	14
5.5.1 Visão geral	14
5.5.2 Identificar processos elementares.....	15
5.5.3 Classificar cada processo elementar como uma função de transação	17
5.5.4 Contar ALRs para cada função de transação	19
5.5.5 Contar DERs para cada função de transação.....	19
5.5.6 Determinar a complexidade funcional para cada função de transação.....	20
5.5.7 Determinar o tamanho funcional de cada função de transação	20
5.6 Medir a funcionalidade de conversão.....	21
5.7 Medir a funcionalidade correspondente a melhorias.....	21
5.8 Calcular o tamanho funcional.....	21
5.9 Documentar a contagem de pontos de função.....	23
5.10 Reportar o resultado da contagem de pontos de função	23
5.10.1	23
5.10.2	24
Anexo A (informativo) Tabelas consolidadas de complexidade e tamanho funcional	25

Prefácio

A ISO (Organização Internacional para a Padronização) e a IEC (Comissão Eletrotécnica Internacional) formam o sistema especializado para a padronização internacional. Os órgãos dos países membros da ISO ou da IEC participam do desenvolvimento de Padrões Internacionais por intermédio de comitês técnicos, estabelecidos pela respectiva organização, para tratar de áreas específicas relacionadas a atividades técnicas. Os comitês técnicos da ISO e IEC colaboram em áreas de interesse mútuo. Outras organizações governamentais ou não-governamentais também participam do trabalho, em conjunto com a ISO e com a IEC.

Os Padrões Internacionais são escritos em conformidade com as regras fornecidas na Parte 2 das Diretivas da ISO/IEC.

Na área da tecnologia da informação, a ISO e a IEC estabeleceram um comitê técnico conjunto, o ISO/IEC JTC 1. Os esboços de Padrões Internacionais adotados pelo comitê técnico conjunto circulam entre os órgãos nacionais para fins de votação. A publicação de um esboço como um Padrão Internacional requer a aprovação de pelo menos 75% dos órgãos nacionais votantes.

Chamamos a atenção para a possibilidade de alguns elementos deste Padrão Internacional estarem sujeitos a patentes e/ou respectivos direitos. Nem a ISO, nem a IEC deverão ser responsáveis pela identificação de nenhuma dessas patentes e/ou direitos.

O Padrão Internacional ISO/IEC 20926 foi preparado pelo Joint Technical Committee ISO/IEC JTC 1, *Information Technology*, Subcommittee SC 7, *Software and Systems Engineering*.

Esta segunda edição cancela e substitui a primeira edição (20926: 2004) em sua totalidade, a qual foi tecnicamente revisada.

Introdução

A utilização de pontos de função como medida do tamanho funcional do software tem crescido desde os meados da década de 70, de umas poucas organizações interessadas até uma impressionante listas de organizações no mundo inteiro. Allan Albrecht foi o primeiro a publicar um método denominado análise de pontos de função para dimensionar funcionalmente o software. O crescimento da utilização de pontos de função tem ampliado a aplicação e utilização da medida. Desde sua fundação em 1986, o International Function Point Users Group (IFPUG) tem aprimorado continuamente o método original de Albrecht para o dimensionamento funcional de software. Este Padrão Internacional é a *release* mais recente do método do IFPUG, continuamente aprimorado, que promove a interpretação consistente da medição funcional de software em conformidade com a ISO/IEC 14143-1:2007. O método de medição de tamanho funcional do IFPUG é conhecido como análise de pontos de função e suas unidades de tamanho funcional são chamadas Pontos de Função.

As organizações podem aplicar este Padrão Internacional para medir o tamanho de um produto de software a fim de:

- dar suporte à análise de qualidade e produtividade;
- estimar o custo e recursos requeridos para o desenvolvimento, melhoria e manutenção do software;
- fornecer um fator de normalização para a comparação de software;
- determinar o tamanho de um pacote de aplicação adquirido, por meio do dimensionamento funcional de todas as funções incluídas no mesmo;
- ajudar os usuários a determinar o benefício provido por um pacote de aplicação para a sua organização, por meio do dimensionamento funcional das funções que correspondam especificamente aos seus requisitos.

A análise de pontos de função mede o software quantificando as tarefas e serviços (isto é, funcionalidade) que o software fornece ao usuário, primordialmente com base no projeto lógico. Os objetivos da análise de pontos de função são medir:

- a funcionalidade implementada no software, que o usuário solicita e recebe;
- a funcionalidade impactada pelo desenvolvimento, melhoria e manutenção de software, independentemente da tecnologia utilizada na implementação.

O processo de análise de pontos de função é:

- suficientemente simples para minimizar o custo adicional introduzido pelo processo de medição;
- uma medida consistente entre diversos projetos e organizações.

A fim de aplicar eficazmente este Padrão Internacional, as pessoas podem ser treinadas formalmente no método, utilizando materiais de treinamento certificados pelo IFPUG.

Este documento é um componente das publicações do IFPUG. Recomenda-se que o mesmo seja lido juntamente com as outras publicações do IFPUG. Estas últimas fornecem orientações para a aplicação das regras descritas neste Padrão Internacional, bem como informações relacionadas, destinadas a auxiliar o entendimento da utilização e aplicabilidade do tamanho funcional resultante. As publicações de suporte do IFPUG incluem:

- A versão atual do Manual de Práticas de Contagem do IFPUG, a qual incorpora este Padrão Internacional, complementado com práticas de contagem e exemplos que suportam a sua implementação;
- O documento “Framework for Functional Sizing”, de 2005, o qual discute a contribuição do tamanho funcional e não-funcional ao tamanho total do produto de software; o método FSM do IFPUG é um método para medição do tamanho funcional;
- Site do IFPUG em www.ifpug.org.

1 Escopo

1.1 Propósito

Este Padrão Internacional especifica o conjunto de definições, regras e passos para a aplicação do método de medição de tamanho funcional (FSM) do IFPUG.

1.2 Conformidade

Este Padrão Internacional mantém conformidade com todos os quesitos obrigatórios da ISO/IEC 14143-1:2007.

1.3 Aplicabilidade

Este Padrão Internacional pode ser aplicado a todos os domínios funcionais.

NOTA: O IFPUG continua a publicar artigos ("white papers") fornecendo orientações para a utilização do método em ambientes e domínios em evolução.

Este Padrão Internacional é totalmente conversível em relação a edições anteriores dos métodos de dimensionamento de tamanho do IFPUG.

Os analistas de pontos de função do IFPUG identificaram diferentes taxas de entrega (horas para entregar um ponto de função) relacionadas à construção de aplicações em diferentes domínios funcionais, calibradas para diversos tamanhos de projeto e complexidade do software.

1.4 Público

Este Padrão Internacional pode ser aplicado por qualquer pessoa que requeira uma medição de tamanho funcional. As pessoas experientes com o método encontrarão neste documento uma referência útil.

2 Referências normativas

Os seguintes documentos referenciados são indispensáveis para a aplicação deste documento. Nas referências datadas aplica-se somente a edição citada. Nas referências não datadas, aplica-se a última edição do documento referenciado (incluindo quaisquer alterações).

ISO/IEC 14143-1:2007, *Information technology — Software measurement — Functional size measurement — Part 1: Definition of concepts*

3 Termos e definições

3.1

Análise de Pontos de Função

APF

método para a medição de tamanho funcional definido neste Padrão Internacional

3.2

aplicação

coleção coesa de procedimentos automatizados e dados, suportando um objetivo de negócio; consiste de um ou mais componentes, módulos ou subsistemas

EXEMPLOS contas a pagar, contas a receber, folha de pagamento, compras, produção de loja, controle de linha de montagem, radar de busca aérea, acompanhamento de alvo, acionamento de armas, programação de aeronaves e reservas de passagens.

3.3

Arquivo de Interface Externa

AIE

grupo de dados ou informações de controle logicamente relacionados, reconhecido pelo usuário, referenciado pela aplicação medida, porém mantido dentro da fronteira de outra aplicação

NOTA Um Arquivo de Interface Externa é um tipo de componente funcional básico.

3.4

Arquivo Lógico Interno

ALI

grupo de dados ou informações de controle logicamente relacionados, reconhecido pelo usuário, mantido dentro da fronteira da aplicação medida

NOTA Um Arquivo Lógico Interno é um tipo de componente funcional básico.

3.5

Arquivo Lógico Referenciado (tipo de arquivo referenciado)

ALR

função de dados lida e/ou mantida por uma função transacional

3.6

arranjar

atividade correspondente ao sequenciamento de atributos em uma transação funcional

3.7

autocontido

nenhum passo anterior ou subsequente é necessário para iniciar ou concluir o(s) Requisito(s) Funcional(is) do Usuário

EXEMPLO O Requisito Funcional do Usuário estabelece que um empregado deve ser incluído e atualizado. Poderiam existir várias partes que comporiam o conjunto completo de informações do empregado. Isto pode ser representado por telas físicas, janelas ou abas distintas, tais como

- Identificação do empregado,
- Localização do empregado,
- Informações de dependentes,
- Informações de salário e
- Instrução.

Para incluir um empregado, uma ou mais abas devem ser preenchidas, dependendo das regras de negócio. O processo de inclusão não estará autocontido até que todas as informações obrigatórias tenham sido digitadas e recebidas pelo sistema.

Para atualizar um empregado, uma ou mais abas podem ser atualizadas a qualquer momento, mas todas elas constituem passos do processo que satisfaz o Requisito Funcional do Usuário, de atualização do empregado.

Incluir, alterar ou excluir informações de cada aba individual não constituem processos elementares distintos, mas sim passos de processo envolvidos na atualização de um empregado. Embora seja possível entrar com informações adicionais no registro de empregado, o conjunto total de informações é considerado parte do único processo elementar: atualizar empregado.

Incluir Empregado e Atualizar Empregado seriam, cada um, um processo autocontido.

3.8 **classificação**

atividade correspondente ao sequenciamento de linhas ou registros em uma função transacional.

3.9 **complexidade funcional**

grau de complexidade específico atribuído a uma função, utilizando as regras definidas neste padrão internacional

3.10 **componente funcional básico** **CFB**

unidade elementar dos Requisitos Funcionais do Usuário definida por e utilizada por um método FSM para fins de medição

EXEMPLOS Um Requisito Funcional do Usuário poderia ser “Manter Clientes”, o qual poderia consistir dos seguintes CFBs: “Incluir um novo cliente”, “Reportar Compras do Cliente” e “Alterar Detalhes do Cliente”. Outro exemplo poderia incluir uma coleção de dados do negócio logicamente relacionados, mantida pelo software em estudo, tal como “Detalhes do Cliente”. Há muitos outros exemplos.

[ISO/IEC 14143-1:2007, definition 3.1]

3.11 **Consulta Externa** **CE**

processo elementar que envia dados ou informações de controle para fora da fronteira

NOTA Uma consulta Externa é um tipo de componente funcional básico.

3.12 **contagem de pontos de função**

atividade constituída pela aplicação das regras constantes deste Padrão Internacional à medição do tamanho funcional de uma aplicação ou projeto

NOTA Há três tipos de contagens de pontos de função: aplicação, projeto de desenvolvimento e projeto de melhoria.

3.13 **contagem de pontos de função da aplicação**

atividade correspondente à aplicação deste Padrão Internacional à medição do tamanho funcional de uma aplicação

3.14 **contagem de pontos de função do projeto de melhoria**

a atividade de aplicar este Padrão Internacional à medição do tamanho funcional de um projeto de melhoria

3.15 **contagem de pontos de função do projeto de desenvolvimento**

a atividade de aplicar este Padrão Internacional à medição do tamanho funcional de um projeto de desenvolvimento

3.16**Dado Elementar Referenciado (tipo de dado elementar)****DER**

atributo único, reconhecido pelo usuário e não repetido

3.17**dados derivados**

dados criados como resultado de processamento que envolva passos que não sejam a recuperação direta e validação de informações de funções de dados, ou que envolva passos adicionais em relação essas atividades

3.18**entidade dependente**

entidade por si só não significativa ou importante para o negócio sem a presença de outras entidades, de modo que

- uma ocorrência da entidade X deve estar ligada a uma ocorrência da entidade Y, e
- a exclusão de uma ocorrência da entidade Y resulta na exclusão de todas as ocorrências relacionadas da entidade X.

3.19**entidade independente**

entidade por si só significativa ou importante para o negócio sem a presença de outras entidades

3.20**Entrada Externa****EE**

processo elementar que processa dados ou informações de controle enviados de fora da fronteira

NOTA Uma Entrada Externa é um tipo de componente funcional básico.

3.21**escopo da contagem**

conjunto de Requisitos Funcionais do Usuário a serem incluídos na contagem de pontos de função

3.22**estado consistente**

ponto no qual o processamento foi totalmente executado; o Requisito Funcional do Usuário foi satisfeito e nada mais há a fazer

EXEMPLO 1 O Requisito Funcional do Usuário é imprimir um cheque e marcar como paga a conta correspondente. Se somente parte do Requisito Funcional do Usuário for satisfeita (por exemplo, apenas imprimir o cheque, ou apenas marcar a conta como paga) a aplicação não estará em um estado consistente. A impressão de um cheque sem marcar a conta como paga causa uma inconsistência na aplicação, assim como marcar a conta como paga sem imprimir o cheque.

EXEMPLO 2 O Requisito Funcional do Usuário é ter um processo *batch* que recebe um arquivo de entrada para atualizar um depósito de dados, produz um relatório de controle de produção e retorna um relatório de erros para a aplicação chamadora. O processo não ficará em um estado consistente a não ser que todas essas partes sejam concluídas.

EXEMPLO 3 O Requisito Funcional do Usuário é transferir um empregado para uma nova tarefa e validar o seu nível de autorização de segurança. Para executar isto, uma solicitação em tempo real é enviada à aplicação de segurança (a qual mantém as autorizações de segurança governamentais e não a segurança da aplicação) e uma resposta deve ser recebida antes que a transferência possa ser concluída. Todos os passos são necessários para criar um estado consistente. A interação com a aplicação de segurança não é uma ação ou passo independente. Não acontece por si só e, por outro lado, a transação para transferir um empregado não fica em um estado consistente sem ela.

3.23**fronteira**

interface conceitual entre o software em estudo e seus usuários

[ISO/IEC 14143-1:2007, definition 3.3]

NOTA As versões anteriores deste Padrão Internacional utilizaram o termo fronteira da aplicação.

3.24**função de dados**

funcionalidade fornecida ao usuário para atender requisitos internos ou externos de armazenamento de dados

NOTA Uma função de dados é um Arquivo Lógico Interno ou um Arquivo de Interface Externa.

3.25**função transacional (ou função de transação)**

processo elementar que fornece funcionalidade ao usuário para processar dados

NOTA Uma função transacional é uma Entrada Externa, Saída Externa ou Consulta Externa.

3.26**funcionalidade de conversão**

funções transacionais ou de dados, fornecidas para converter dados e/ou prover outros requisitos de conversão especificados pelo usuário

NOTA As funcionalidades de conversão existem somente durante o desenvolvimento ou melhoria de uma aplicação.

3.27**informações de controle**

dados que influenciam um processo elementar especificando o quê, quando ou como os dados devem ser processados

3.28**intenção primária**

intenção que é a primeira em importância

3.29**lógica de processamento**

quaisquer requisitos especificamente solicitados pelo usuário para executar um processo elementar, tais como: validações, algoritmos ou cálculos, bem como a leitura ou manutenção de uma função de dados

3.30**manter**

habilidade de incluir, alterar ou excluir dados por meio de um processo elementar

3.31**manutenção adaptativa**

modificação de um produto de software, executada após a entrega, a fim de manter o mesmo utilizável em um ambiente alterado ou em alteração

NOTA A manutenção adaptativa provê as melhorias necessárias para acomodar mudanças no ambiente no qual um produto de software tem que operar. Tais mudanças são aquelas que precisam ser realizadas para manter o produto atualizado em relação ao ambiente em alteração. Por exemplo, o sistema operacional precisa sofrer um “upgrade” e algumas mudanças precisam ser efetuadas para acomodar o novo sistema operacional.

[ISO/IEC 14764:2006 Software engineering – Software life cycle processes – Maintenance, definition 3.1]

3.32**manutenção corretiva**

modificação reativa de um produto de software, executada após a entrega para corrigir problemas descobertos

NOTA A modificação repara o produto de software para satisfazer requisitos.

[ISO/IEC 14764:2006 Software engineering – Software life cycle processes – Maintenance, definition 3.2]

3.33**manutenção perfectiva**

modificação de um produto de software após a entrega, a fim de detetar e corrigir defeitos latentes no mesmo antes que se manifestem como falhas

NOTA 1: A manutenção perfectiva fornece melhorias para os usuários, aprimoramento da documentação dos programas e recodificação dos mesmos para melhorar a performance, manutenibilidade e outros atributos do software.

NOTA 2: Contrastar com: manutenção adaptativa; manutenção corretiva.

[ISO/IEC 14764:2006 Software engineering – Software life cycle processes – Maintenance, definition 3.7]

3.34**Ponto de Função****PF**

unidade de medida de tamanho funcional conforme definida neste Padrão Internacional

3.35**processo elementar**

menor unidade de atividade significativa para o usuário

3.36**projeto de desenvolvimento**

projeto para desenvolver e derivar a primeira *release* de uma aplicação de software

3.37**projeto de melhoria**

projeto para desenvolver e entregar manutenção adaptativa

NOTA Um projeto de melhoria também pode desenvolver e entregar manutenção corretiva e perfectiva, porém estes não contribuem para o tamanho funcional do projeto de melhoria.

3.38**propósito da contagem**

razão para execução da contagem de pontos de função

NOTA Ver 5.3 a).

3.39**reconhecido pelo usuário**

requisitos para processos e/ou dados que são acordados com, assim como entendidos pelo(s) usuário(s) e desenvolvedor(es) de software

3.40**Registro Lógico Referenciado (tipo de registro elementar)****RLR**

subgrupo de dados elementares referenciados (tipos de dado elementares), reconhecido pelo usuário, contido em uma função de dados

3.41**Requisitos Funcionais do Usuário**

subconjunto dos requisitos do usuário especificando o que o software deverá fazer em termos de tarefas e serviços

NOTA Os Requisitos Funcionais do Usuário incluem, mas não estão limitados a:

- transferência de dados (por exemplo: Receber dados de entrada de cliente, Enviar sinal de controle);
- transformação de dados (por exemplo: Calcular juros bancários, Derivar temperatura média);
- armazenamento de dados (por exemplo: Armazenar pedido de cliente, Registrar temperatura ambiente ao longo do tempo);
- recuperação de dados (por exemplo: Listar empregados atuais, Recuperar posição de aeronave).

Requisitos do usuário que não constituem Requisitos 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 para operação, hardware-alvo e conformidade com os padrões);
- restrições ambientais (por exemplo: interoperabilidade, segurança e privacidade);
- restrições de implementação (por exemplo: linguagem de desenvolvimento, prazo para entrega).

[ISO/IEC 14143-1:2007, definition 3.8]

3.42**Saída Externa****SE**

processo elementar que envia dados ou informações de controle para fora da fronteira e inclui lógica de processamento adicional além daquela de uma Consulta Externa

NOTA Uma Saída Externa é um tipo de componente funcional básico.

3.43**significativo**

reconhecido pelo usuário e satisfazendo um Requisito Funcional do Usuário

3.44**tamanho funcional**

tamanho do software derivado por meio da quantificação dos Requisitos Funcionais do Usuário

[ISO/IEC 14143-1:2007, definition 3.6]

3.45**tamanho funcional da aplicação**

medida da funcionalidade que uma aplicação fornece ao usuário, determinada pela contagem de pontos de função da aplicação

3.46**tamanho funcional do projeto de melhoria**

medida da funcionalidade incluída, alterada ou excluída por ocasião da conclusão de um projeto de melhoria, conforme medida pela contagem de pontos de função do projeto de melhoria

NOTA O tamanho funcional de um projeto de melhoria pode incluir o tamanho da funcionalidade de conversão

3.47

tamanho funcional do projeto de desenvolvimento

medida da funcionalidade oferecida aos usuários pela primeira *release* do software, conforme medida pela contagem de pontos de função do projeto de desenvolvimento

NOTA O tamanho funcional de um projeto de desenvolvimento inclui o tamanho da funcionalidade de conversão.

3.48

tipo de entidade associativo

tipo de entidade contendo atributos que descrevem um relacionamento muitos-para-muitos entre dois outros tipos de entidade

3.49

tipo de entidade atributivo

tipo de entidade que descreve um ou mais atributos de outro tipo de entidade

3.50

tipo de função

cinco tipos de componentes funcionais básicos identificados neste Padrão Internacional

NOTA Os cinco tipos de função são: Entrada Externa, Saída Externa, Consulta Externa, Arquivo Lógico Interno e Arquivo de Interface Externa.

3.51

usuário

qualquer pessoa ou coisa que se comunique ou interaja com o software a qualquer tempo

EXEMPLOS Exemplos de coisas incluem, mas não estão limitados a: aplicações de software, animais, sensores ou outros hardwares.

[ISO/IEC 14143-1:2007, definition 3.11]

3.52

visão do usuário

Requisitos Funcionais do Usuário, conforme percebidos pelo usuário

NOTA Os desenvolvedores traduzem a visão do usuário para software, a fim de fornecer uma solução.

4 Abreviaturas

AIE	Arquivo de Interface Externa
ALI	Arquivo Lógico Interno
ALR	Arquivo Lógico Referenciado (tipo de arquivo referenciado)
APF	Análise de Ponto de Função
CE	Consulta Externa
CFB	Componente Funcional Básico
DER	Dado Elementar Referenciado (tipo de dado elementar)
EE	Entrada Externa
PF	Ponto de Função

RLR Registro Lógico Referenciado (tipo de registro elementar)

SE Saída Externa

5 Processo de Medição

5.1 Visão Geral

Para conduzir uma contagem de pontos de função devem ser executadas as seguintes atividades, a fim de identificar e classificar os componentes funcionais básicos (ALI, AIE, EE, SE, CE):

- a) reunir a documentação disponível em conformidade com 5.2,
- b) determinar o escopo e a fronteira da contagem, identificando os Requisitos Funcionais do Usuário em conformidade com 5.3,
- c) medir as funções de dados de acordo com 5.4, 5.6 e 5.7,

NOTA A funcionalidade de conversão (se aplicável) é medida conforme 5.6; a funcionalidade de melhoria (se aplicável) é medida conforme 5.7.

- d) medir as funções de transação conforme 5.5, 5.6 e 5.7,

NOTA A funcionalidade de conversão (se aplicável) é medida conforme 5.6; a funcionalidade de melhoria (se aplicável) é medida conforme 5.7.

- e) calcular o tamanho funcional de acordo com 5.8,
- f) documentar a contagem de pontos de função conforme 5.9, e
- g) reportar o resultado da contagem de pontos de função conforme 5.10.

NOTA A Figura 1 fornece uma visão geral, gráfica, do processo de contagem de pontos de função.

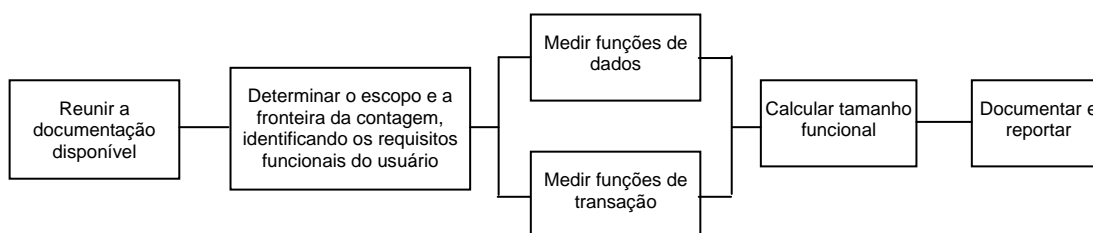


Figura 1 — Visão geral, gráfica, do processo de contagem de pontos de função

5.2 Reunir a documentação disponível

A documentação de suporte a uma contagem de pontos de função deve descrever a funcionalidade entregue pelo software ou a funcionalidade impactada pelo projeto de software medido.

Deve ser obtida documentação suficiente para conduzir a contagem de pontos de função, ou acesso a especialistas no assunto capazes de fornecer informações adicionais para suprir quaisquer falhas na documentação.

NOTA A documentação adequada pode incluir requisitos, modelos de dados/objetos, diagramas de classes, diagramas de fluxos de dados, casos de uso, descrições de procedimentos, formatos de relatórios, manuais de usuário e outros artefatos do desenvolvimento de software.

5.3 Determinar o escopo e fronteira da contagem, identificando os Requisitos Funcionais do Usuário

Para determinar o escopo e fronteira da contagem e identificar os Requisitos Funcionais do Usuário, devem ser executadas as seguintes atividades:

- a) identificar o propósito da contagem,

NOTA 1 Uma contagem de pontos de função é conduzida a fim de fornecer uma resposta para uma questão de negócio, sendo a questão de negócio que determina o propósito.

NOTA 2 O propósito da contagem determina o escopo da contagem.

EXEMPLO 1 O propósito da contagem poderia ser a determinação do tamanho de uma *release* de software específica.

EXEMPLO 2 O propósito da contagem poderia ser a determinação do tamanho de uma aplicação, como parte do esforço da organização para determinar o tamanho de seu *portfolio* de software.

- b) identificar o tipo de contagem, com base no propósito, como um dos seguintes:

- 1) uma contagem de pontos de função de projeto de desenvolvimento;
- 2) uma contagem de pontos de função de aplicação;
- 3) uma contagem de pontos de função de um projeto de melhoria,

- c) determinar o escopo da contagem, com base no propósito e tipo de contagem,

- d) determinar a fronteira de cada aplicação contida no escopo da contagem com base na visão do usuário e não em considerações técnicas

EXEMPLO 3 Se o propósito for estimar o custo de uma melhoria nas aplicações de Recursos Humanos (RH) e Benefícios, então:

- o tipo de contagem é uma contagem de projeto de melhoria;
 - o escopo deverá incluir as funções transacionais e de dados incluídas, alteradas ou excluídas referentes tanto à aplicação de RH quanto à de Benefícios, assim como quaisquer requisitos de conversão referentes a cada aplicação;
 - a visão do usuário é que RH e Benefícios são áreas funcionais diferentes, por esse motivo são aplicações separadas;
 - existe uma fronteira entre as aplicações de RH e de Benefícios, assim como entre cada aplicação e o usuário.
- e) os requisitos do usuário podem incluir uma mistura de requisitos funcionais e não-funcionais; identificar quais requisitos são funcionais e excluir os não-funcionais.

5.4 Medir funções de dados

5.4.1 Visão Geral

A funcionalidade de dados satisfaz os Requisitos Funcionais do Usuário referentes a armazenar e/ou referenciar dados. Toda a funcionalidade de dados dentro do escopo da contagem deve ser avaliada para identificar cada grupo lógico de dados.

Devem ser executadas as seguintes atividades para medir as funções de dados

- a) identificar e agrupar todos os dados lógicos em funções de dados conforme 5.4.2,
- b) classificar cada função de dados como um ALI ou AIE, de acordo com 5.4.3,
- c) contar os DERs para cada função de dados conforme 5.4.4,
- d) contar os RLRs para cada função de dados de acordo com 5.4.5,
- e) determinar a complexidade funcional de cada função de dados em conformidade com 5.4.6, e
- f) determinar o tamanho funcional de cada função de dados conforme 5.4.7.

5.4.2 Identificar e agrupar todos os dados lógicos em funções de dados

NOTA 1 As funções de dados são mais facilmente identificáveis utilizando-se um modelo lógico de dados; no entanto, isto não impede a utilização do processo de medição em ambientes onde forem utilizadas técnicas alternativas para a modelagem de dados ou objetos. A terminologia da modelagem de dados é utilizada para documentar as regras referentes às funções de dados, mas a mesma abordagem pode ser aplicada a outras técnicas.

Para identificar as funções de dados, devem ser executadas as seguintes atividades

- a) identificar todos os dados ou informações de controle logicamente relacionados e reconhecidos pelo usuário, dentro do escopo da contagem,
- b) excluir entidades que não sejam mantidas por nenhuma aplicação,
- c) agrupar entidades que sejam entidades dependentes,

NOTA 2 Entidades independentes são consideradas como grupos lógicos de dados distintos.

- d) excluir as entidades abaixo, denominadas dados de código:
 - 1) entidade de dados de substituição, que contém um código e um nome ou descrição explicativos;
 - 2) entidade de ocorrência única, que contém um ou mais atributos que raramente ou nunca mudam;
 - 3) entidade que contém dados basicamente estáticos, ou que muito raramente mudam;
 - 4) entidade de valores *default*, que contém valores para popular atributos;
 - 5) entidade de valores válidos, que contém valores disponíveis para seleção ou validação;
 - 6) entidade que contém uma faixa de dados para validação,

NOTA 3 As entidades acima referidas como dados de código podem conter outros atributos para fins de auditoria e para definir datas efetivas. A presença desses atributos não altera a natureza dessas entidades.

- e) excluir entidades que não contenham atributos requeridos pelo usuário, e
- f) remover entidades associativas que contenham atributos adicionais não requeridos pelo usuário e entidades associativas que contenham apenas chaves estrangeiras; agrupar os atributos referentes a chaves estrangeiras com as entidades primárias.

NOTA 4 Atributos referentes a chaves estrangeiras são dados requeridos pelo usuário para estabelecer um relacionamento com outra função de dados.

5.4.3 Classificar cada função de dados como um ALI ou AIE

Uma função de dados deve ser classificada como

- a) Arquivo Lógico Interno (ALI), se for mantida pela aplicação medida, ou
- b) Arquivo de Interface Externa (AIE) se for
 - referenciada, mas não mantida, pela aplicação medida, e
 - identificada como um ALI em uma ou mais outras aplicações.

5.4.4 Contar os DERs para cada função de dados

A fim de contar os DERs - Dados Elementares Referenciados (tipos de dado elementares) correspondentes a uma função de dados, devem ser executadas as seguintes atividades

- a) contar um DER para cada atributo único, reconhecido pelo usuário e não repetido, mantido na função de dados ou recuperado da mesma por meio da execução de todos os processos elementares pertinentes ao escopo da contagem,

NOTA 1 Por exemplo, dentro de um ALI ou AIE, contar um único DER para os 12 campos repetidos referentes aos valores do Orçamento Mensal. Contar um DER adicional para identificar o mês aplicável.

- b) quando duas ou mais aplicações mantiverem e/ou referenciarem a mesma função de dados, contar apenas os DERs utilizados pela aplicação medida,

NOTA 2 Os atributos não referenciados pela aplicação medida não são contados.

- c) contar um DER para cada atributo requerido pelo usuário para estabelecer um relacionamento com outra função de dados, e
- d) revisar os atributos relacionados a fim de determinar se os mesmos devem ser agrupados e contados como um único DER, ou como vários DERs. O agrupamento irá depender de como os processos elementares utilizam os atributos dentro da aplicação

EXEMPLO Os atributos (primeiro nome, nome do meio, sobrenome) são agrupados e contados como

- nome (primeiro nome, nome do meio, sobrenome) se esses atributos sempre forem utilizados juntos,
- primeiros nomes (primeiro nome e inicial do meio) e sobrenome se, além do que está acima, o sobrenome for utilizado independentemente, ou
- primeiro nome, inicial do meio e sobrenome, se os três puderem ser utilizados independentemente.

5.4.5 Contar os RLRs para cada função de dados

A fim de contar RLRs - Registros Lógicos Referenciados (tipos de arquivo referenciados) para uma função de dados, devem ser executadas as seguintes atividades

- a) contar um RLR para cada função de dados (i.e., por default, cada função de dados possui um subgrupo de DERs que é contado como um RLR),
- b) contar um RLR adicional para cada um dos seguintes subgrupos lógicos de DERs (dentro da função de dados) que contenham mais de um DER:
 - 1) entidade associativa com atributos não-chave;
 - 2) subtipo (sem ser o primeiro subtipo);
 - 3) entidade atributiva, em um relacionamento que não seja 1-1 obrigatório

NOTA 1 Um relacionamento 1-1 obrigatório é um relacionamento entre duas entidades onde cada uma se relaciona com uma e somente uma instância da entidade relacionada.

NOTA 2 Se um modelo de dados não estiver disponível, procurar grupos de dados repetidos para identificar RLRs.

EXEMPLO 1 Uma fatura possui um cabeçalho com informações do cliente e vários itens referentes às compras (por exemplo, número do item, descrição, preço, peso). O cabeçalho é contado como um RLR. Os itens constituem um grupo repetido e são contados como um RLR adicional.

EXEMPLO 2 Um único atributo repetido tal como vários números de conta para o mesmo cliente não constitui um grupo repetido, sendo contado como um único DER e não como um RLR.

EXEMPLO 3 Um grupo de DERs ocorrendo várias vezes, tal como ano, mês e valor orçado é um grupo repetido, mas é contado como três DERs e não como um RLR.

5.4.6 Determinar a complexidade funcional de cada função de dados

A complexidade funcional de cada função de dados deve ser determinada utilizando-se o número de DERs e RLRs, em conformidade com a Tabela 1.

Tabela 1 — Complexidade das funções de dados

		DERs		
		1 – 19	20 – 50	> 50
RLRs	1	Baixa	Baixa	Média
	2 – 5	Baixa	Média	Alta
	> 5	Média	Alta	Alta

5.4.7 Determinar o tamanho funcional de cada função de dados

O tamanho funcional de cada função de dados deve ser determinado utilizando-se o tipo e a complexidade funcional, de acordo com a Tabela 2.

Tabela 2 — Tamanho das funções de dados

		Tipo	
		ALI	AIE
Complexidade funcional	Baixa	7	5
	Média	10	7
	Alta	15	10

5.5 Medir funções de transação

5.5.1 Visão geral

A funcionalidade de transação satisfaz os Requisitos Funcionais do Usuário que processam dados. Toda a funcionalidade de transação dentro do escopo da contagem deve ser avaliada, a fim de identificar cada processo elementar único.

Para medir funções de transação, as seguintes atividades devem ser executadas

- identificar cada processo elementar requerido pelo usuário, em conformidade com 5.5.2,
- classificar cada função de transação como uma Entrada Externa (EE), Saída Externa (SE) ou Consulta Externa (CE), conforme 5.5.3,
- contar os ALRs - Arquivos Lógicos Referenciados (tipos de arquivo referenciados) para cada função de transação, de acordo com 5.5.4,

- d) contar os DERs - Dados Elementares Referenciados (tipos de dado elementares) para cada função de transação, conforme 5.5.5,
- e) determinar a complexidade funcional de cada função de transação de acordo com 5.5.6, e
- f) determinar o tamanho funcional de cada função de transação conforme 5.5.7.

5.5.2 Identificar processos elementares

5.5.2.1

Para identificar cada processo elementar devem ser executadas as seguintes atividades

- a) Compor e/ou decompor os Requisitos Funcionais do Usuário nas menores unidades de atividade que satisfaçam todos os itens abaixo:

- 1) é significativa para o usuário;
- 2) constitui uma transação completa;

NOTA Aos usuários de versões anteriores deste Padrão Internacional: o item 2 não é uma mudança, mas sim um refinamento para aumentar a especificidade a fim de promover interpretação consistente.

- 3) é autocontida;
- 4) deixa o negócio da aplicação contada em um estado consistente.

EXEMPLO 1 Um Requisito Funcional do Usuário pode estabelecer que deve ser fornecida uma função para Manter Informações de Empregado. Esse requisito é decomposto em unidades de trabalho menores tais como Incluir Empregado, Alterar Empregado e Consultar Empregado.

EXEMPLO 2 Os requisitos individuais podem estabelecer a necessidade de incluir diversos tipos de informações de empregado (por exemplo, endereço, salário e informações de dependentes), mas a menor unidade de atividade significativa para o usuário é Incluir Empregado.

- b) Identificar um processo elementar para cada unidade de atividade identificada, de modo a satisfazer todos os critérios em 5.5.2.1 a).

5.5.2.2

Para determinar processos elementares únicos devem ser executadas as seguintes atividades

- a) Ao comparar um possível processo elementar com um processo elementar já identificado, contar os dois processos elementares referidos como um único processo elementar se eles:

- 1) demandarem o mesmo conjunto de DERs;
- 2) demandarem o mesmo conjunto de ALRs;
- 3) demandarem o mesmo conjunto de lógicas de processamento para executar o processo elementar (consultar a lista abaixo em 5.5.2.2 b),

NOTA 1 As atividades referentes à medição das funções de transação são mostradas sequencialmente; no entanto, as mesmas são de fato iterativas. ALRs e DERs são identificados de acordo com 5.5.4 e 5.5.5, mas são necessários para comparar dois processos elementares semelhantes.

NOTA 2 Um processo elementar pode incluir variações pequenas em DERs ou ALRs, assim como múltiplas alternativas, variações ou ocorrências das lógicas de processamento abaixo.

EXEMPLO Quando é identificado um processo elementar para Incluir Empregado, o fato de um empregado poder ou não ter dependentes não faz com o que o processo elementar seja dividido em dois. O processo elementar ainda é Incluir Empregado, havendo variação na lógica de processamento e nos DERs para dar conta dos dependentes.

- b) Não dividir um processo elementar com várias formas de lógica de processamento em vários processos elementares.

NOTA 4 Um processo elementar que recebe e valida dados do usuário, lê e filtra dados de um ALI, classifica e apresenta os resultados ao usuário não pode ser dividido em vários processos elementares.

5.5.2.3

Várias formas de lógica de processamento para a execução de um processo elementar estão identificadas na Tabela 3:

Tabela 3 — Formas de lógica de processamento

Formas de lógica de processamento	
1. Validações são executadas	
EXEMPLO 1	Ao acrescentar um novo empregado a uma organização, o respectivo processo valida o DER referente ao tipo de empregado.
2. Fórmulas e cálculos matemáticos são executados	
EXEMPLO 2	Ao emitir relatório com todos os empregados de uma organização, o processo calcula o número total de empregados mensalistas, empregados horistas e o total geral.
3. Valores equivalentes são convertidos	
EXEMPLO 3	Com base em uma tabela, a idade do empregado é convertida para um grupo de faixas de idade.
4. Dados são filtrados e selecionados segundo critérios especificados para comparar vários conjuntos de dados	
EXEMPLO 4	A fim de gerar uma lista de empregados segundo as tarefas alocadas aos mesmos, um processo elementar utiliza o número da tarefa presente nas alocações para selecionar e listar os empregados alocados a cada tarefa.
5. Condições são analisadas para determinar as aplicáveis	
EXEMPLO 5	A lógica de processamento utilizada pelo processo elementar na inclusão de um empregado depende do mesmo ser mensalista ou horista. A entrada dos DERs (e a lógica de processamento resultante) com base em uma escolha distinta (mensalista ou horista) é parte de um processo elementar neste exemplo.
6. Um ou mais ALIs são atualizados	
EXEMPLO 6	Ao incluir um empregado, o processo elementar atualiza o ALI de empregado para manter os dados referentes a empregado.
7. Um ou mais ALIs ou AIEs são referenciados	
EXEMPLO 7	Ao incluir um empregado, o AIE de moeda é referenciado para obter a taxa de conversão para o dólar norte-americano, a fim de determinar o salário-hora dos empregados.
8. Dados ou informações de controle são recuperados	
EXEMPLO 8	Para permitir a visualização de uma lista de empregados, informações dos empregados são recuperadas de uma função de dados.

Formas de lógica de processamento	
9. Dados derivados são criados a partir de transformação dos dados existentes, com o objetivo de criar dados adicionais	<p>EXEMPLO 9 A fim de determinar (derivar) o número de matrícula de um paciente (por exemplo, SMIJO01), os seguintes dados são concatenados:</p> <ul style="list-style-type: none"> — as primeiras três letras do sobrenome do paciente (por exemplo, SMI para Smith); — as duas primeiras letras do primeiro nome do paciente (por exemplo, JO para John); — um número de sequência de dois dígitos (começando com 01).
10. É alterado o comportamento da aplicação	<p>EXEMPLO 10 O comportamento do processo elementar de pagamento dos empregados é alterado quando uma mudança é feita para efetuar o pagamento sexta-feira sim, sexta-feira não, ao invés de fazê-lo no dia 15 e no último dia do mês, resultando em 26 períodos de pagamento ao invés de 24.</p>
11. Preparar e apresentar informações fora da fronteira	<p>EXEMPLO 11 Uma lista de empregados é formatada e apresentada ao usuário.</p>
12. Existe a capacidade de receber dados e informações de controle que entram pela fronteira da aplicação	<p>EXEMPLO 12 Um usuário entra com informações para incluir um pedido de cliente na aplicação.</p>
13. Classificar ou arrumar um conjunto de dados. Esta forma de lógica de processamento não impacta a identificação do tipo ou contribui para a unicidade de um processo elementar; isto é, a orientação dos dados não constitui unicidade	<p>EXEMPLO 13 A lista de empregados é classificada em ordem alfabética ou de localização.</p> <p>EXEMPLO 14 Em uma tela de entrada de pedidos, as informações do cabeçalho do pedido são colocadas na parte superior da tela, enquanto os detalhes do pedido são colocados abaixo.</p> <p>NOTA 1 Versões anteriores deste Padrão Internacional (ISO/IEC 20926:2003) utilizaram erradamente os termos “reclassificar” e “rearrumar”; a utilização dos termos “classificar” e “arrumar” é uma correção e não uma alteração.</p>

5.5.3 Classificar cada processo elementar como uma função de transação

5.5.3.1

Para cada processo elementar

- a) a intenção primária deve ser classificada como uma das seguintes:
 - 1) alterar o comportamento da aplicação;
 - 2) manter um ou mais ALIs;
 - 3) apresentar informações ao usuário,
- b) as formas de lógica de processamento requeridas para executar o processo elementar devem ser identificadas na lista apresentada em 5.5.2.3.

5.5.3.2

Cada processo elementar deve ser classificado como

- a) uma EE, se o mesmo:
 - 1) incluir lógica de processamento para receber dados ou informações de controle que entrem pela fronteira da aplicação;
 - 2) tiver um das seguintes intenções primárias
 - i) manter um ou mais ALIs, ou
 - ii) alterar o comportamento da aplicação,
- b) uma SE, se o mesmo tiver a intenção primária de apresentar informações ao usuário e incluir pelo menos uma das seguintes formas de lógica de processamento:
 - 1) cálculos matemáticos são executados;
 - 2) um ou mais ALIs são atualizados;
 - 3) dados derivados são criados;
 - 4) o comportamento da aplicação é alterado,
- c) uma CE, se o mesmo tiver a intenção primária de apresentar informações ao usuário e:
 - 1) referenciar uma função de dados para recuperar dados ou informações de controle;
 - 2) não satisfizer os critérios para ser classificado como uma SE.

NOTA 1 A Tabela 4 apresenta um resumo do relacionamento entre a intenção primária e o tipo de função de transação.

Tabela 4 — Relacionamento entre a intenção primária e o tipo de função de transação

Função	Tipo de função de transação		
	EE	SE	CE
Alterar o comportamento da aplicação	IP	F	N/A
Manter um ou mais ALIs	IP	F	N/A
Apresentar informações ao usuário	F	IP	IP

legenda

IP a intenção primária do tipo de função de transação
 F uma função do tipo de função de transação que às vezes está presente, mas que não é a intenção primária
 N/A o tipo de função de transação não pode executar este tipo de função

NOTA 2 A Tabela 5 apresenta um resumo do relacionamento entre as formas de processamento lógico e o tipo de função de transação.

Tabela 5 — Relacionamento entre a lógica de processamento e o tipo de função de transação

Forma de lógica de processamento	Tipo de função de transação		
	EE	SE	CE
1. Validações são executadas	p	p	p
2. Cálculos matemáticos são executados	p	o*	n

Forma de lógica de processamento	Tipo de função de transação		
	EE	SE	CE
3. Valores equivalentes são convertidos	p	p	p
4. Dados são filtrados e selecionados utilizando-se critérios especificados para comparar vários conjuntos de dados	p	p	p
5. Condições são analisadas para determinar quais são aplicáveis	p	p	p
6. Pelo menos um ALI é atualizado	o*	o*	n
7. Pelo menos um ALI ou AIE é referenciado	p	p	o
8. Dados ou informações de controle são recuperados	p	p	o
9. Dados derivados são criados	p	o*	n
10. O comportamento da aplicação é alterado	o*	o*	n
11. Informações são preparadas e então apresentadas fora da fronteira	p	o	o
12. Dados ou informações de controle entrando pela fronteira da aplicação são recebidos	o	p	p
13. Um conjunto de dados é classificado ou arrumado	p	p	p

legenda

- o é obrigatório que o tipo de função de transação execute a forma de lógica de processamento
- o* é obrigatório que o tipo de função de transação execute pelo menos uma destas (o*) formas de lógica de processamento
- p o tipo de função de transação pode executar a forma de lógica de processamento, mas a mesma não é obrigatória
- n o tipo de função de transação não pode executar a forma de lógica de processamento

5.5.4 Contar ALRs para cada função de transação

Para cada função de transação, um ALR deve ser contado para cada função de dados única que for acessada (lida e/ou gravada) pela função de transação.

NOTA As atividades referentes à medição de funções de transação são apresentadas sequencialmente; contudo, na verdade as mesmas são iterativas. Para comparar dois processos elementares similares é necessária a identificação de ALRs, conforme dito em 5.5.2.2.

5.5.5 Contar DERs para cada função de transação

A fim de contar DERs para uma função de transação, as seguintes atividades devem ser executadas

- a) Revisar tudo o que atravesse (entre e/ou saia) a fronteira,
- b) Contar um DER para cada atributo único, reconhecido pelo usuário e não repetido que atravesse (entre e/ou saia) a fronteira durante o processamento da função de transação,

EXEMPLO 1 DERs que atravessam a fronteira incluem

- atributos que o usuário introduz por meio de uma tela, bem como aqueles apresentados em um relatório ou tela,
 - atributos que entram pela fronteira da aplicação e que são necessários para especificar quando, o quê e/ou como os dados devem ser recuperados ou gerados pelo processo elementar,
 - atributos fornecidos pelo usuário da função de transação, ou apresentados a ele, e
 - atributos em um arquivo eletrônico que entrem ou saiam pela fronteira.
- c) Contar apenas um DER por função de transação para a habilidade de enviar uma mensagem de resposta da aplicação, mesmo que existam várias mensagens,

EXEMPLO 2 Se forem apresentadas diversas mensagens de erro/confirmação ao usuário, somente um DER será contado.

- d) Contar apenas um DER por função de transação para a habilidade de iniciar ações, mesmo que existam diversas maneiras de fazer isso, e

EXEMPLO 3 Se o usuário puder iniciar a geração de um relatório clicando no botão OK ou pressionando uma tecla de função, apenas um DER será contado.

- e) Não contar os seguintes itens como DERs:

- constantes literais tais como títulos de relatórios, identificadores de telas ou painéis, cabeçalhos de colunas ou títulos de atributos;
- rótulos gerados pela aplicação tais como atributos referentes a data e hora;
- variáveis de paginação, números de página e informações de posicionamento, por exemplo, 'Linhas 37 a 54 de 211';
- auxílios à navegação tais como a habilidade de navegar em uma lista utilizando "anterior", "próximo", "primeiro", "último" e seus equivalentes gráficos;
- atributos gerados dentro da fronteira por uma função de transação e salvos em um ALI sem sair pela fronteira;
- atributos recuperados ou referenciados de um ALI ou AIE para participarem do processamento sem que saiam pela fronteira.

NOTA As atividades referentes à medição de funções de transação são apresentadas sequencialmente; contudo, na verdade as mesmas são iterativas. Para comparar dois processos elementares similares é necessária a identificação dos DERs, conforme dito em 5.5.2.2.

5.5.6 Determinar a complexidade funcional para cada função de transação

A complexidade funcional de cada função de transação será determinada utilizando-se o número de ALRs e DERs, em conformidade com a Tabela 6 ou 7.

Tabela 6 — Complexidade funcional das EE

		DERs		
		1 – 4	5 – 15	> 15
ALRs	0 – 1	Baixa	Baixa	Média
	2	Baixa	Média	Alta
	> 2	Média	Alta	Alta

Tabela 7 — Complexidade funcional das SE e CE

		DERs		
		1 – 5	6 – 19	> 19
ALRs	0 – 1	Baixa	Baixa	Média
	2 – 3	Baixa	Média	Alta
	> 3	Média	Alta	Alta
NOTA		Uma CE tem no mínimo 1 ALR.		

5.5.7 Determinar o tamanho funcional de cada função de transação

O tamanho funcional de cada função de transação será determinado utilizando-se o tipo e a complexidade funcional, de acordo com a Tabela 8.

Tabela 8 — Tamanho das funções de transação

		Tipo		
		EE	SE	CE
Complexidade Funcional	Baixa	3	4	3
	Média	4	5	4
	Alta	6	7	6

5.6 Medir a funcionalidade de conversão

O escopo da contagem de um projeto de desenvolvimento ou melhoria também pode incluir o tamanho funcional da funcionalidade de conversão requerida para o mesmo. As funções de transação de conversão e as funções de dados (que ainda não tenham sido contadas) deverão ser contadas em conformidade com 5.4 e 5.5.

NOTA O escopo da contagem determinará se a funcionalidade de conversão deverá ser contada.

5.7 Medir a funcionalidade correspondente a melhorias

Os projetos de melhoria podem envolver inclusões, alterações e exclusões da funcionalidade existente. A funcionalidade correspondente às melhorias deverá ser medida de acordo com o que segue:

- a) Não alterar a fronteira já estabelecida para a(s) aplicação(ões) modificada(s).
- b) Contar as funções de dados incluídas, alteradas ou excluídas de acordo com 5.4.
- c) Contar as funções de transação que forem incluídas, alteradas ou excluídas de acordo com 5.5, e
- d) O tamanho funcional da aplicação poderá ser atualizado para refletir:
 - 1) a funcionalidade incluída, que aumentará o tamanho funcional da aplicação;
 - 2) a funcionalidade alterada, que poderá aumentar, diminuir ou não ter efeito sobre o tamanho funcional da aplicação;
 - 3) a funcionalidade excluída, que diminuirá o tamanho funcional da aplicação.

NOTA 1 Uma alteração em uma função de dados pode envolver inclusão, alteração ou exclusão de DERs e/ou RLRs.

NOTA 2 Uma alteração em uma função de transação pode envolver inclusão, alteração ou exclusão de DERs e/ou ALRs e/ou alteração na lógica de processamento.

5.8 Calcular o tamanho funcional

O objetivo e escopo da contagem deverão ser considerados na seleção e utilização da fórmula apropriada para calcular o tamanho funcional.

O tamanho funcional de um projeto de desenvolvimento deverá ser calculado utilizando-se a Fórmula (1):

$$DFP = ADD + CFP \quad (1)$$

onde

DFP é a contagem de pontos de função do projeto de desenvolvimento;

ADD é o tamanho das funções a serem entregues ao usuário pelo projeto de desenvolvimento;

CFP é o tamanho da funcionalidade de conversão.

O tamanho funcional de uma aplicação, medido após o projeto de desenvolvimento, ou a qualquer tempo no ciclo de vida da aplicação deverá ser calculado utilizando-se a Fórmula (2):

$$AFP = ADD \quad (2)$$

onde

AFP é a contagem de pontos de função da aplicação;

ADD é o tamanho das funções a serem entregues ao usuário pelo projeto de desenvolvimento (excluído o tamanho de qualquer funcionalidade de conversão), ou a funcionalidade existente no momento da contagem da aplicação.

O tamanho funcional de um projeto de melhoria deverá ser calculado utilizando-se a Fórmula (3):

$$EFP = ADD + CHGA + CFP + DEL \quad (3)$$

onde

EFP é a contagem de pontos de função do projeto de melhoria;

ADD é o tamanho das funções incluídas pelo projeto de melhoria;

CHGA é o tamanho das funções alteradas pelo projeto de melhoria – conforme as mesmas estão / estarão após a implementação;

CFP é o tamanho da funcionalidade de conversão;

DEL é o tamanho das funções excluídas pelo projeto de melhoria.

O tamanho funcional de uma aplicação após um projeto de melhoria deverá ser calculado utilizando-se a Fórmula (4):

$$AFPA = (AFPB + ADD + CHGA) - (CHGB + DEL) \quad (4)$$

onde

AFPA é a contagem de pontos de função da aplicação após o projeto de melhoria;

AFPB é a contagem de pontos de função da aplicação antes do projeto de melhoria;

ADD é o tamanho das funções incluídas pelo projeto de melhoria;

CHGA é o tamanho das funções alteradas pelo projeto de melhoria – como estão / estarão após a implementação;

CHGB é o tamanho das funções alteradas pelo projeto de melhoria – como estão / estavam antes do início do projeto;

DEL é o tamanho das funções excluídas pelo projeto de melhoria.

5.9 Documentar a contagem de pontos de função

A contagem de pontos de função deve ser documentada como segue:

- o propósito e o tipo da contagem;
- o escopo da contagem e a fronteira da aplicação;
- a data da contagem;
- uma lista de todas as funções de dados e de transação, incluindo o respectivo tipo e complexidade, bem como o número de pontos de função atribuído a cada uma;
- o resultado da contagem (ver 5.10);
- quaisquer suposições feitas e questões resolvidas.

A documentação da contagem de pontos de função também pode incluir o seguinte::

- a identificação da documentação de origem na qual a contagem foi baseada;
- a identificação dos participantes, seus papéis e qualificações;
- para cada função de dados, o número de DERs e RLRs;
- para cada função de transação, o número de DERs e de ALRs;
- uma referência cruzada de todas as funções de dados para as funções de transação;
- uma referência cruzada de todas as funções de dados para as abstrações relacionadas na documentação de origem;
- uma referência cruzada de todas as funções de transação para as abstrações relacionadas na documentação de origem.

NOTA 1 Negociar o nível de documentação com o cliente e informar ao mesmo os custos e benefícios relacionados.

NOTA 2 Uma Contagem de Pontos de Função completamente documentada facilitará a rastreabilidade, usabilidade e manutenibilidade; contudo, um cliente pode estar interessado apenas no resultado final.

5.10 Reportar o resultado da contagem de pontos de função

NOTA A prática de reportar consistentemente os resultados das contagens de pontos de função permitirá que os leitores identifiquem o padrão com o qual as mesmas mantêm conformidade.

5.10.1 5.10.1

Os resultados que mantenham conformidade com este Padrão Internacional deverão ser reportados como segue:

S FP (IFPUG–IS)

onde

S é o resultado da contagem de pontos de função;

FP é a unidade de tamanho do método FSM do IFPUG;

IS é este Padrão Internacional (ISO/IEC 20926:200x).

EXEMPLO 250 FP (IFPUG-ISO/IEC 20926:200x)

5.10.2 5.10.2

Os resultados que mantiverem conformidade com uma customização local deste Padrão Internacional deverão ser reportados como:

S FP (IFPUG–IS–c)

onde

c representa um ou mais caracteres indicando que o resultado não mantém conformidade plena com este Padrão Internacional.

EXEMPLO 250 FP (IFPUG–ISO/IEC 20926:200x–a)

Anexo A (informativo)

Tabelas consolidadas de complexidade e tamanho funcional

Para maior conveniência, as tabelas de complexidade e tamanho funcional são repetidas nas Tabelas A.1 a A.5.

Tabela A.1 — Complexidade das funções de dados

		DERs		
		1 – 19	20 – 50	> 50
RLRs	1	Baixa	Baixa	Média
	2 – 5	Baixa	Média	Alta
	> 5	Média	Alta	Alta

Tabela A.2 — Tamanho das funções de dados

		Tipo	
		ALI	AIE
Complexidade funcional	Baixa	7	5
	Média	10	7
	Alta	15	10

Tabela A.3 — Complexidade funcional das EEs

		DERs		
		1 – 4	5 – 15	> 15
ALRs	0 – 1	Baixa	Baixa	Média
	2	Baixa	Média	Alta
	> 2	Média	Alta	Alta

Tabela A.4 — Complexidade funcional das SEs e CEs

		DERs		
		1 – 5	6 – 19	> 19
ALRs	0 – 1	Baixa	Baixa	Média
	2 – 3	Baixa	Média	Alta
	> 3	Média	Alta	Alta
NOTA		Uma CE tem no mínimo 1 ALR.		

Tabela A.5 — Tamanho das funções de transação

		Tipo		
		EE	SE	CE
Complexidade funcional	Baixa	3	4	3
	Média	4	5	4
	Alta	6	7	6

Esta página foi deixada em branco intencionalmente.

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:

Tópico	Página
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

Tamanho Funcional	Um tamanho de software obtido através da quantificação dos Requisitos Funcionais do Usuário.
Requisito Funcional do Usuário	<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">• Transferência de dados (por exemplo: entrada de dados de cliente, envio de sinais de controle)• Transformação de dados (por exemplo: calcular taxa de juros bancária, calcular temperatura média)• Armazenamento de dados (por exemplo: armazenar dados de cliente, registrar a mudança de temperatura ao longo do tempo)• Recuperação de dados (por exemplo: listar os empregados atuais, recuperar posição da aeronave)

**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.

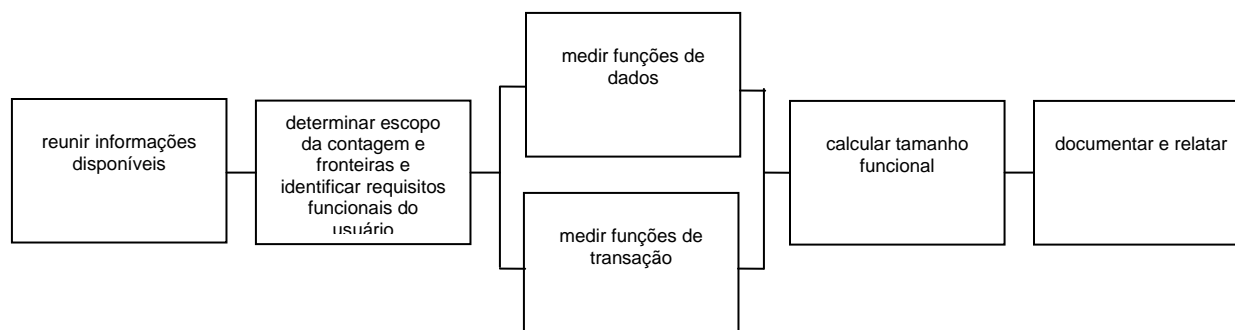
Conteúdo Este capítulo contempla as seguintes seções:

Tópico	Página
Procedimento do Método de Medição de Tamanho Funcional	2-2
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

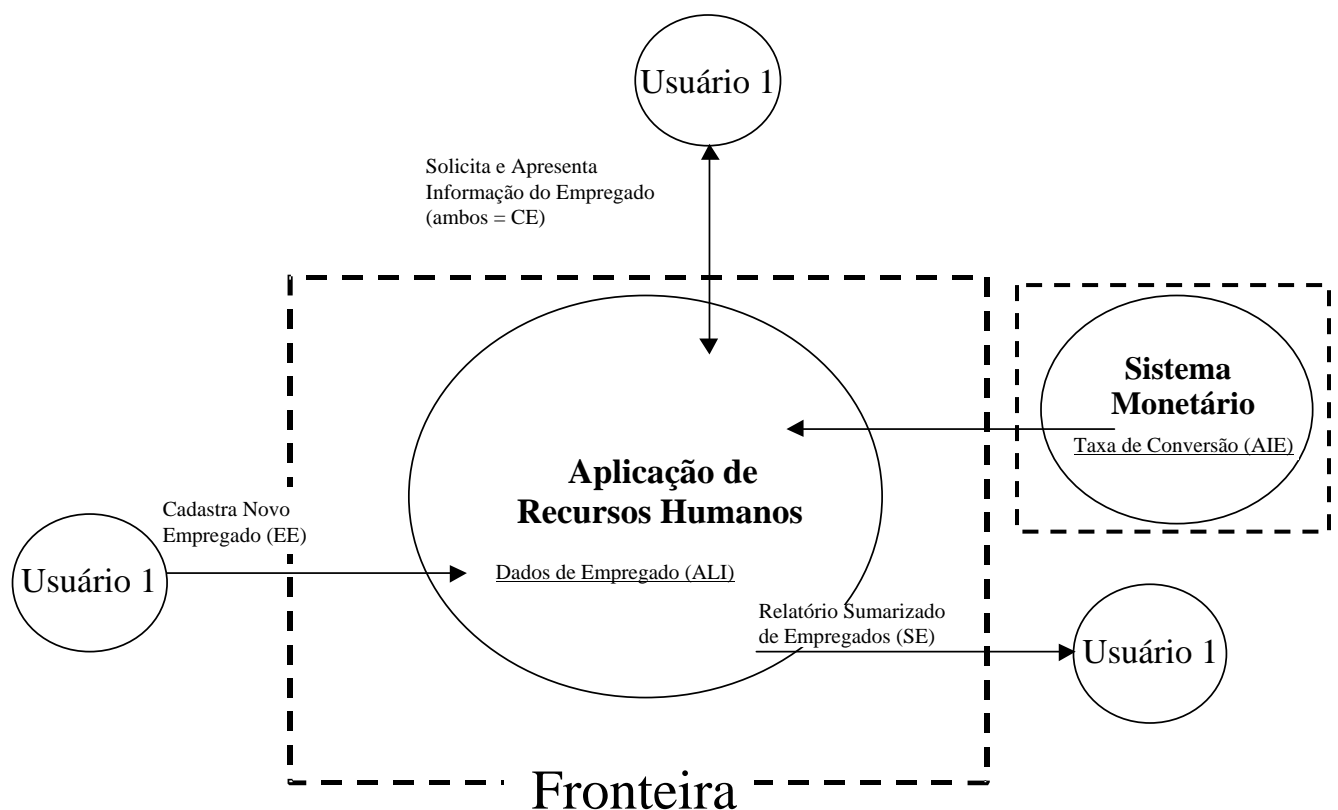
Nota: Não existem capítulos específicos para Calcular o Tamanho Funcional e Documentar e Relatar, pois estes procedimentos não necessitam de explicação adicional além da que foi fornecida na Parte 1.

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:

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

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
Proposta: usuários expressam necessidades e intenções	sim	não
Requisitos: desenvolvedores e usuário revisam e concordam quanto às necessidades e intenções do usuário	sim	sim
Projeto: os desenvolvedores podem incluir elementos para implementação que não são usados pela análise de pontos de função	sim	sim
Construção	sim	sim
Entrega	sim	sim
Manutenção	sim	sim
<p>Nota: 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 Sizing¹.

¹ 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:

Tópico	Página
Definições: Tipos de Contagem de Pontos de Função	4-2
Projeto de Desenvolvimento	4-2
Projeto de Melhoria	4-2
Aplicação	4-3
Diagrama dos Tipos de Contagem	4-4
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

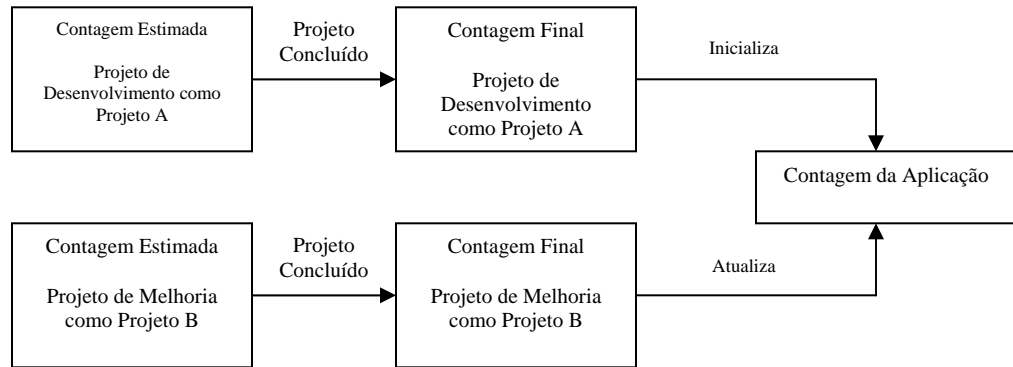
Uma aplicação é uma coleção coesa de procedimentos automatizados e dados apoiando um objetivo de negócio; isto consiste em um ou mais componentes, módulos, ou subsistemas.

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:

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

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

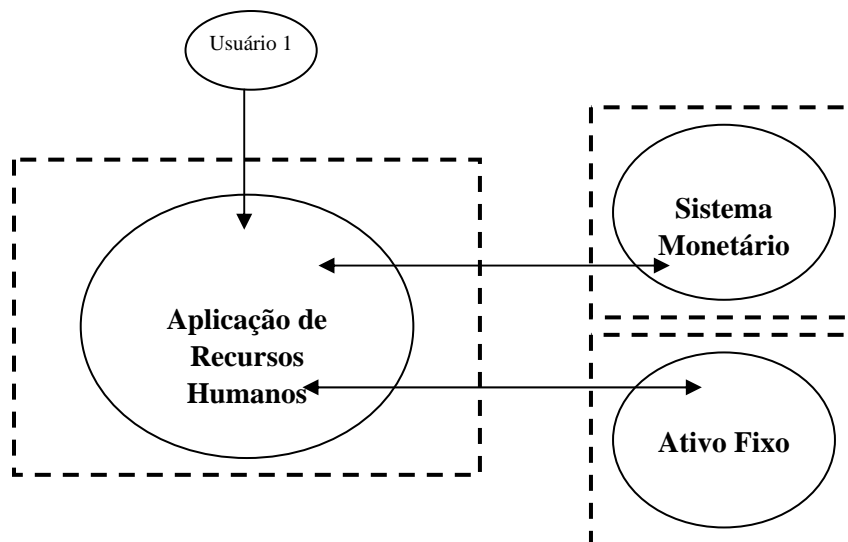
A fronteira é uma interface conceitual entre o software sob estudo e seus usuários.

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"> • O propósito da contagem • O escopo da contagem • A fronteira da aplicação • Quaisquer suposições relacionadas aos itens acima

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:

Tópico	Página
Definição e Intenção Primária: ALIs e AIEs	6-2
Arquivos Lógicos Internos	6-2
Arquivo de Interface Externa	6-2
Diferença entre ALIs e AIEs	6-2
Definições para Termos Utilizados	6-2
Procedimentos para Contagem de Funções de Dados	6-4
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
Dicas para Ajudar na Contagem	6-10

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.

Reconhecido pelo Usuário	<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>
Mantido	<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>
Processo Elementar	<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">• é significativo para o usuário• constitui uma transação completa• é auto contida e• deixa o negócio da aplicação contada em um estado consistente <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> <p><u>Por exemplo</u>, uma função de dados com 51 DERs e 2 RLRs se traduz uma complexidade funcional alta.</p>		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														
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:

Tópico	Página
Definição e Intenção Primária: EEs, SEs e CEs	7-3
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
Regras de Contagem de Função de Transação	7-9
Procedimentos de Contagem de Função de Transação	7-9
Identificar Cada Processo Elementar	7-10
Determinar Unicidade do Processo Elementar	7-11
Classificar Cada Processo Elementar	7-13
Classificar como uma EE	7-13
Classificar como uma SE	7-13
Classificar como uma CE	7-13

Continua na página seguinte

Tópico	Página
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
Dicas para Ajudar na Contagem EEs, SEs e CEs	7-21
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.

Processo Elementar	<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>
Informações de Controle	<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>
Mantido	<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>
Usuário	<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>
Significativo	<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.

Parte 3 – Práticas de Contagem

Página intencionalmente deixada em branco.

Parte 3 Práticas de Contagem

Introdução

A parte 3 fornece práticas de contagem detalhadas e exemplos aperfeiçoados para apoiar os praticantes no entendimento de:

- Dados de código, incluindo uma descrição dos vários tipos de dados, como distinguir dados de negócio e de referência de dados de código e como considerar os dados de código
- Arquivos lógicos, incluindo uma descrição de arquivos lógicos e o processo para agrupar corretamente dados em arquivos lógicos; para contar corretamente tipos de registro elementares; e para contar corretamente tipos de dados elementares
- Dados compartilhados, incluindo uma descrição de dados compartilhados e o processo para identificar quando dados compartilhados são funções do tipo dado (arquivo lógico interno, arquivo de interface externa) ou funções do tipo transação (entrada externa, saída externa, consulta externa)
- Projetos de melhoria e atividades de manutenção, incluindo descrições de melhoria e manutenção e aquelas atividades reconhecidas pelo processo de contagem de pontos de função

Conteúdo

A parte 3 inclui os seguintes capítulos:

Capítulo	Página
Dados de Código	1-3
Arquivos Lógicos	2-1
Dados Compartilhados	3-1
Projetos de Melhoria e Atividades de Manutenção	4-1
Atividades de Conversão de Dados	5-1

Esta página foi intencionalmente deixada em branco.

Dados de Código

Introdução Este capítulo usa o conceito de requisitos funcionais e não-funcionais de usuário (descritos na Parte 1 e em “A Framework for Functional Sizing” [IFPUG, 2003]) para identificar dados de código e determinar como devem ser considerados.

Este capítulo aborda especificamente um número de exemplos relevantes para dados de código. É reconhecido que pode haver outros exemplos de dados de código e esses podem ser abordados em futuras versões do CPM.

Conteúdo Este capítulo inclui as seguintes seções:

Tópico	Veja Página
Tipos de Entidades de Dado	1-4
Metodologia	1-9
Identificando Dados de Código	1-10
Considerando Dados de Código e Transações de Dados de Código	1-14
Bibliografia	1-15

Tipos de Entidades de Dado

Uma revisão dos dados da aplicação e de seu propósito fornece um entendimento das várias categorias de entidade de dado. Em geral, os analistas devem distinguir entre três categorias de entidades de dado:

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

As primeiras duas categorias de entidades usualmente são identificadas para satisfazer os Requisitos Funcionais do Usuário e dessa forma serão investigadas para contagem como arquivos lógicos (veja Parte 3, Capítulo 2).

A terceira categoria de dados, referenciada a seguir neste capítulo como “Dados de Códigos”, contudo, geralmente existe para satisfazer requisitos não-funcionais do usuário (para requisitos de qualidade, implementação física e/ou uma razão técnica) ao invés de um requisito funcional do usuário. As diferentes categorias de dados são delineadas abaixo para apoiar na identificação.

Dados de Negócio

Dados de Negócio também podem ser chamados Dados Essenciais do Usuário (“Core User Data”) ou Objetos de Negócio. Este tipo de dado reflete a informação necessária a ser armazenada e recuperada pela área funcional abordada pela aplicação. Dados de Negócio geralmente representam um percentual significativo das entidades identificadas. Possuem a maioria das seguintes características:

Lógicas

Características lógicas incluem:

- Obrigatória para a operação da área funcional do usuário
- Identificável pelo usuário (geralmente por um usuário do negócio)
- Capaz de ser mantida pelo usuário (geralmente um usuário do negócio)
- Armazena os Dados Principais do Usuário para apoiar transações de negócio
- Muito dinâmica – as operações normais de negócio fazem com que sejam regularmente referenciados e rotineiramente incluídos, alterados ou excluídos
- Capaz de ser reportada

Físicas

Características físicas incluem:

- Possui campos-chave e geralmente vários atributos
- Pode ter de zero a uma infinidade de registros

Exemplos

Exemplos de Dados de Negócio incluem:

- Arquivo de Cliente, Arquivo de Fatura, Arquivo de Empregado, Arquivo de Função
- O Arquivo de Função, no Sistema de Gerência de Funções, incluiria itens como:
 - Número da Função,
 - Nome da Função,
 - Nome da Divisão,
 - Data de Ativação da Função, etc.

Dados de Referência

Este tipo de dado é armazenado para apoiar as regras de negócio na manutenção de dados de negócio; por exemplo, em uma aplicação de folha de pagamento ele seria o dado armazenado sobre as alíquotas de impostos do governo para cada faixa salarial e a data em que a mesma iniciou a sua vigência. Os Dados de Referência geralmente representam um pequeno percentual das entidades identificadas. Possuem a maioria das seguintes características:

Lógicas

Características lógicas incluem:

- Obrigatório para a operação da área funcional do usuário
- Identificável pelo usuário (geralmente um usuário de negócio)
- Geralmente capaz de ser mantido pelo usuário (Geralmente por um usuário administrativo)
- Geralmente estabelecido quando a aplicação é instalada pela primeira vez e mantido intermitentemente
- Armazena os dados para apoiar as principais atividades do usuário
- Menos dinâmico – ocasionalmente muda em resposta às mudanças no ambiente da área funcional, processos funcionais externos e/ou regras de negócio
- Transações processando Dados de Negócio costumam precisar de acesso a Dados de Referência

Físicas	Características físicas incluem: <ul style="list-style-type: none">• Possui campos-chave e poucos atributos• Geralmente pelo menos um registro ou um número limitado de registros
Exemplos	Exemplos de Dados de Referência incluem: <ul style="list-style-type: none">• Faixas Salariais, Taxas de Desconto, Alíquotas de Impostos, Configuração de Limites• Arquivo de Faixas Salariais – armazena informação sobre os valores pagos para cada tipo de função e a habilidade exigida para executar aquele tipo de função<ul style="list-style-type: none">• Tipo de Função• Situação, Valor Cobrado, Início de Vigência (1:n)• Descrição das Habilidades da Função (1:n)

Dados de Código

O usuário nem sempre especifica diretamente os Dados de Código, às vezes chamados Dados de Lista ou Dados de Tradução. Em outros casos são identificados pelo desenvolvedor em resposta a um ou mais requisitos não-funcionais do usuário. Os Dados de Código fornecem uma lista de valores válidos que um atributo descritivo pode ter. Normalmente os atributos de Dados de Código são Código, Descrição e/ou outros atributos ‘padrão’ descrevendo o código; por exemplo, abreviação padrão, data de início de vigência, data de expiração, dados de trilha de auditoria, etc.

Ao utilizar códigos em Dados de Negócio, é necessário ter meios de tradução para converter de código para algo mais reconhecível pelo usuário. De modo a satisfazer os requisitos não-funcionais do usuário, os desenvolvedores quase sempre criam uma ou mais tabelas contendo Dados de Código. Logicamente, o código e a sua descrição correspondente têm o mesmo significado. Sem uma descrição, o código nem sempre pode ser claramente entendido.

A diferença chave entre Dados de Código e Dados de Referência é:

- Com os Dados de Código, você pode substituir um pelo outro sem mudar o significado dos Dados de Negócio; por exemplo, Código do Aeroporto versus Nome do Aeroporto, Id da Cor versus Descrição da Cor.
- Com Dados de Referência, você não pode substituir (por exemplo, Código do Imposto pela Alíquota do Imposto).

Os Dados de Código possuem a maioria das seguintes características:

Lógicas

Características lógicas incluem:

- O dado é obrigatório para a área funcional, mas é opcionalmente armazenado como um arquivo de dados
- Não é geralmente identificado como parte dos requisitos funcionais do usuário; é geralmente identificado como parte da solução para atender requisitos não-funcionais do usuário
- É algumas vezes mantido pelo usuário (geralmente por uma pessoa de suporte ao usuário)
- Armazena dados para padronizar e facilitar atividades e transações de negócio
- É essencialmente estático – apenas muda em resposta a mudanças na forma como o negócio funciona
- As transações de negócio referenciam os Dados de Código para melhorar a facilidade da entrada de dados, melhorar a consistência dos dados, garantir a integridade dos dados, etc.
- Se reconhecido pelo usuário:
 - Algumas vezes é considerado como um grupo do mesmo tipo de dados
 - Pode ser mantido usando a mesma lógica de processamento

Físicas

Características físicas incluem:

- Consiste de campo-chave e geralmente apenas um ou dois atributos
- Normalmente possui um número estável de registros
- Pode representar 50% de todas as entidades em 3ª Forma Normal
- É às vezes desnormalizado e colocado em uma tabela física com outros Dados de Código
- Pode ser implementado sob diferentes formas (por exemplo, via aplicação distinta, dicionário de dados ou hard-coded dentro do software)

Exemplos

Exemplos de Dados de Código incluem:

- Estado
 - Código do Estado
 - Nome do Estado
- Tipo de Pagamento
 - Código do Tipo de Pagamento
 - Descrição do Pagamento

Origem dos Dados de Código

Historicamente, a motivação para os dados de código foi economizar espaço por meio do armazenamento de um código ao invés de uma longa descrição textual. Para facilidade de manutenção, esses códigos e descrições eram colocados em arquivos ou tabelas a fim de eliminar mudanças no software quando atualizações fossem necessárias.

Os Dados de Código são uma propriedade de um atributo descritivo chamado “Meta Dado”. Exemplos são valores válidos, descrições de códigos ou tabelas de tradução. Alguns Dados de Código são desenvolvidos para atender requisitos específicos do usuário e contém dados que estão dentro do domínio do usuário. Outros Dados de Código podem ser derivados a partir dos requisitos do usuário para restringir os valores permitidos. Os Dados de Código podem também ser criados em uma tentativa de reduzir requisitos de espaço em disco. Os requisitos podem também incluir a habilidade de manter Dados de Código. Todos esses são requisitos não-funcionais do usuário.

Os Dados de Código são uma implementação de requisitos não-funcionais do usuário. Como consequência, os Dados de Código podem influenciar o tamanho não-funcional do produto de software, mas *não* o tamanho *funcional* do mesmo [“A Framework for Functional Sizing”, IFPUG 2003”].

Metodologia

O impacto dos dados de código pertencerem à dimensão não-funcional é que nem os Dados de Código, nem as transações que os mantém devem ser contados.

Introdução

A seção “Identificando Dados de Código” abaixo fornece um processo passo-a-passo para a identificação do que é e do que não é dado de código. A seção é geralmente referenciada no passo “Identificando Arquivos Lógicos”, como definido na parte 3 – capítulo 2, onde os dados de código são desconsiderados. Conforme previamente declarado, os dados de código não são considerados parte do tamanho funcional. Isso tem várias consequências. Para sermos perfeitamente claros neste Guia de Implementação, resumimos as consequências abaixo.

Consequências

1	Não conte Dados de Código como um Arquivo Lógico Uma consequência de desconsiderar os Dados de Código é que os mesmos não podem ser considerados ALI ou AIE.
2	Não conte Dados de Código como um DER ou RLR Uma consequência de desconsiderar os Dados de Código é que os mesmos não podem ser considerados um RLR ou DER em um ALI ou AIE.
3	Não conte Dados de Código como um ALR Uma consequência de desconsiderar os Dados de Código é que os mesmos não podem ser considerados ALR ao avaliar a complexidade de uma função transacional (EE, SE, CE), porque não se trata de um arquivo lógico.
4	Não conte Funções Transacionais de Dados de Código Uma consequência dos Dados de Código serem uma parte de outra dimensão (a dimensão não-funcional em contraste à dimensão funcional) é que a manutenção de Dados de Código ou funções de relatório não são consideradas ao se medir o tamanho funcional da aplicação.

Identificando Dados de Código

Os tipos de Dados de Código resumidos em “O Que São Dados de Código” e “O Que Não São Dados de Código” podem ser usados como uma ajuda prática para determinar se uma entidade é ou não Dados de Código. Alguns critérios podem se sobrepor em parte. Assim que o critério de uma das subseções tiver sido satisfeito, a entidade deverá ser considerada como Dados de Código e não contada.

Os exemplos fornecidos não são uma lista exaustiva e podem não cobrir todos os casos possíveis. Na dúvida, avalie os tipos de entidade dentro do contexto de “Tipos de Entidades de Dado”.

O Que São Dados de Código

Introdução

Esses são vários tipos diferentes de Dados de Código, os quais se enquadram em três áreas gerais:

- Dados de Substituição fornecem um código e um nome explicativo ou descrição para um atributo de um objeto de negócio (Substituição é uma condição suficiente mas não necessária para ser considerado Dados de Código).
- Dados Constantes ou Estáticos que raramente mudam.
- Dados com Valores Válidos fornecem uma lista de valores disponíveis para um atributo de um ou mais tipos de objetos de negócio.

Tipos de Dados de Código

Substituição	Estáticos ou Constantes	Valores Válidos
Código + Descrição	Uma Ocorrência	Valores Válidos
	Dados Estáticos	Faixa de Valores Válidos
	Valores Default	

Quaisquer desses tipos de Dados de Código podem também incluir outros atributos, como data de início e fim de vigência para definir o período de tempo no qual o valor está disponível. Também podem incluir atributos de tipo auditoria tais como data de criação, criado por (id do usuário), data da última atualização, última atualização por (id do usuário). Ainda, uma diversidade de variações é possível (por exemplo, código + descrição resumida / completa). A presença desses atributos adicionais não afeta o processo de categorização, mas os atributos são considerados parte dos Dados de Código.

Substituição

Código + Descrição Este tipo de Dado de Código contém um código e um nome explicativo ou descrição. Este tipo de Dado de Código pode servir como um meio para tornar mais ágil a entrada de dados para usuários experientes, o nome / descrição explicativo para usuários menos experientes ou para listagens como em relatórios. Este tipo de Dado de Código também pode ser implementado para economizar espaço de armazenamento ou ser um resultado de normalização. Se for dado de substituição, é Dado de Código e não é contado.

Exemplos

- Estados: Código do Estado, Nome do Estado
- Cores: Código da Cor, Descrição da Cor

Variações

- Código, Idioma, Descrição (Para descrições em múltiplos idiomas)
- Código, Descrição Resumida, Descrição Completa, Abreviatura

Estáticos ou Constantes

Uma Ocorrência Este tipo de Dado de Código contém uma e apenas uma ocorrência independentemente da quantidade de atributos. Os Dados de Código tem apenas um registro de dados e os atributos são relativamente constantes; podem mudar, mas muito raramente.

Exemplos

- Uma entidade com dados sobre uma organização em particular; por exemplo, nome e endereço.
- Software COTS com o nome da companhia aérea, customizado pela organização usuária

Dados Estáticos Este tipo de Dado de Código contém dados que são basicamente estáticos. A quantidade de instâncias de dados estáticos pode mudar, mas muito raramente, e o conteúdo de uma instância raramente muda.

Exemplos

- Uma entidade elementos químicos: símbolo, número atômico, descrição
- As tabelas de pontos de função para valorar os tipos de função e os níveis de complexidade

Valores Default (template) Este tipo de Dado de Código contém valores default para (alguns atributos em) novas instâncias de um objeto de negócio.

Valores Válidos

Valores Válidos	<p>Este tipo de Dado de Código fornece uma lista de valores válidos para um atributo de um ou mais tipos de objetos de negócio. Este tipo de Dado de Código é implementado para satisfazer requisitos como reduzir erros e aumentar a facilidade de uso pelo usuário. Este tipo de Dado de Código é normalmente usado para listar valores disponíveis para seleção pelo usuário e/ou validar a entrada fornecida pelo mesmo. Este tipo de Dado de Código contém dados basicamente estáticos; se não forem, podem ser Dados de Referência ou Dados de Negócio.</p>
Exemplos	<ul style="list-style-type: none">• Nome do estado: Contém todos os valores válidos para o atributo nome do estado• Código do estado: Contém todos os valores válidos para o atributo código do estado• Cor: Contém todos os valores válidos para o atributo cor de um objeto de negócio
Faixa de Valores Válidos	<p>Este tipo de Dado de Código contém dados basicamente estáticos; se não forem podem ser Dados de Referência.</p>
Exemplos	<ul style="list-style-type: none">• Faixa de Números de Telefone Permissíveis: menor número de telefone, maior número de telefone.• Faixa de temperatura térmica.

O Que Não São Dados de Código

Esta seção descreve dados que não são considerados Dados de Código porque são Dados de Negócio ou Dados de Referência. A Parte 1 e a Parte 2 do CPM – Capítulo 2 (Arquivos Lógicos) contém as regras para esses tipos de entidades. Algumas vezes tabelas são chamadas ‘tabelas de código’ mas são na realidade Dados de Referência ou mesmo Dados de Negócio.

Exemplos

Exemplos de Dados de Negócio ou Dados de Referência que não devem ser considerados Dados de Código:

- Tipos de entidade com montantes financeiros, taxas de câmbio, e alíquotas de impostos, se não forem constantes. Esses dados não restringem valores válidos; ao invés disso, adicionam significado para um valor dentro de uma faixa em particular.
- Dados de controle: dados mantidos pelo usuário que contém regras de negócio dizendo à aplicação o que fazer ou como se comportar.
- Tabela de Taxa de Câmbio: Taxa de Câmbio contém a taxa de câmbio da moeda corrente do país na conversão para dólar. Não é possível substituir o código pela taxa de câmbio do país, o dado é essencialmente não estático, e os dados apoiam as atividades de negócio; portanto, isto é um exemplo de Dados de Referência.
- Faixa de percentual de tributação para um Sistema de Percentuais Progressivos: um percentual de tributação diferente é aplicável para diferentes faixas de receita. Contém um valor mínimo e máximo para cada percentual de tributação. Entretanto, não se pode substituir o valor pela entidade, e os dados apoiam atividades do negócio. O percentual de tributação não restringe a receita. Portanto, este é um exemplo de Dados de Referência.

Considerando Dados de Código e Transações de Dados de Código

Os Dados de Código como identificados em “O Que São Dados de Código” representam a implementação de requisitos não-funcionais do usuário ao invés da implementação de Requisitos Funcionais do Usuário.

Consequentemente, os Dados de Código e as transações que os mantém não contam para o tamanho funcional da aplicação.

Contudo, os dados que são parte de requisitos não-funcionais podem ser medidos usando uma medida distinta para dimensionar o tamanho não-funcional.

Neste momento não existe um método específico para contar Dados de Código, bem como suas funções de manutenção e relatório, a fim de produzir um tamanho para a dimensão não-funcional.

.

Bibliografia

As seguintes fontes foram consultadas ou citadas neste capítulo:

Definitions and Counting Guidelines for the Application of Function Point Analysis: A Practical Manual, Version 2.2. (NESMA, 2003).

ISBN: 90-76258-17-1.

Nota: Esse manual é também chamado Manual de Práticas de Contagem da NESMA. Descreve a metodologia padrão de APF e muitos aspectos relacionados à aplicação da mesma. Pode ser usado em conjunto com o manual do IFPUG. Para mais informações, visite o web site da NESMA www.nesma.org.

IFPUG “A Framework for Functional Sizing”, IFPUG, 2003.

ISO/IEC 14143-1:2007 Information technology – Software measurement Functional size measurement – Part 1: definition of concepts, ISO/IEC, 2007

Esta página foi intencionalmente deixada em branco.

Arquivos Lógicos

Introdução Este capítulo aplica as definições e regras referentes aos arquivos lógicos, usando um processo descritivo para a identificação e classificação das funções de dados.

Essas orientações ilustram a identificação de funções de dados a partir de modelos de dados normalizados. Uma visão geral de normalização de dados é fornecida para apoiar essa abordagem. Contudo, isso não exclui o uso dessas orientações em ambientes onde técnicas alternativas para a modelagem de dados ou objetos forem empregadas.

Conteúdo Este capítulo inclui as seguintes seções:

Tópico	Página
Resumo da Metodologia	2-2
Conceitos de Modelagem de Dados	2-3
Passo 1: Identificar Arquivos Lógicos	2-10
Passo 2: Classificar Arquivos Lógicos	2-26
Passo 3: Identifique Tipos de Dados Elementares	2-26
Passo 4: Identificar Tipos de Registro Elementares	2-38
Bibliografia	2-51

Resumo da Metodologia

Introdução

Um processo passo-a-passo para estabelecer um conjunto de arquivos lógicos (Arquivo Lógico Interno e Arquivo de Interface Externo) é usado, onde cada passo enxerga os dados em um nível de detalhe mais refinado. Os detalhes de cada passo são explicados nas seções subsequentes.

Passo	Ação
1	Identificar Arquivos Lógicos Para cada entidade lógica de dados, identifique como as entidades relacionadas são agrupadas em arquivos lógicos, os quais refletem a “visão do usuário”. Por exemplo, determine se as entidades de dados são um arquivo lógico independente por si só ou se entidades relacionadas devem ser agrupadas em um único arquivo lógico. Este passo é explicado na Seção "Passo 1: Identificar Arquivos Lógicos".
2	Classificar Arquivos Lógicos Cada arquivo lógico identificado é classificado como ALI ou AIE. Este passo é explicado na Seção “Passo 2: Classificar Arquivos Lógicos”.
3	Identificar Tipos de Dados Elementares Para cada arquivo lógico, identifique os elementos de dados usados pela aplicação medida. Este passo é explicado na Seção “Passo 3: Identifique Tipos de Dados Elementares”
4	Identificar Tipos de Registro Elementares Para cada arquivo lógico, identifique como os dados relacionados são agrupados em tipos de registro elementares, os quais refletem a “visão do usuário”. Este passo é explicado na Seção "Passo 4: Identifique Tipos de Registro Elementares".

O passo com o maior impacto no tamanho funcional é o Passo 1: Identificar Arquivos Lógicos, porque a identificação da quantidade correta de arquivos lógicos é crucial para alcançar a consistência entre contagens. Os passos restantes influenciam o tamanho funcional em um grau consideravelmente menor, porque não afetam a quantidade de arquivos lógicos, mas apenas o seu tipo e classificação de complexidade.

Conceitos de Modelagem de Dados

Introdução

Uma revisão das definições usadas no domínio da análise de dados (o qual inclui modelagem lógica e física de dados) pode fornecer uma base para o entendimento, assim como esclarecer o propósito das regras de práticas de contagem, conforme relacionadas à identificação de Arquivos Lógicos, Tipos de Registro Elementares e Tipos de Dados Elementares. Um profundo entendimento dos conceitos de modelagem de dados está implícito na Análise de Pontos de Função, ao medir Funções de Dados da maneira apropriada e correta. A seção “Termos da Modelagem de Dados” resume os termos da modelagem de dados.

Conceitos Chave em Modelagem de Dados

Uma revisão das definições usadas no domínio de análise de dados (o qual inclui modelagem de dados e SGBD) pode fornecer uma base para o entendimento, assim como esclarecer o propósito das regras de práticas de contagem na medida em que se relacionam especificamente a Arquivos Lógicos, Tipos de Registro Elementares (RLRs) e Tipos de Dado Elementares (DERs). Um entendimento de conceitos de dados é assumido na aplicação das orientações descritas no próximo capítulo para medir as Funções de Dados de maneira apropriada e correta.

Tipo de Entidade**Definições de Entidade (ou Tipo de Entidade)**

- ❑ Qualquer pessoa, local, coisa, evento ou conceito distinto sobre o qual informação é mantida (Thomas Bruce, 1992)
- ❑ Uma coisa que pode ser identificada de forma distinta (Peter Chen, 1976).
- ❑ Qualquer objeto distinto representado em uma base de dados (C.J. Date, 1986)
- ❑ Uma entidade de dados representa alguma “coisa” que será armazenada para referência futura. O termo entidade se refere à representação lógica dos dados (Clive Finkelstein, 1989)
- ❑ A palavra entidade significa qualquer coisa sobre a qual armazenamos informação (por exemplo, um cliente, fornecedor, ferramenta mecânica, empregado, poste, assento de companhia aérea, etc.). Para cada entidade, certos atributos são armazenados (James Martin, 1989)
- ❑ Uma entidade também pode representar o relacionamento entre duas ou mais entidades, chamadas entidades associativas (Michael Reingruber, 1994)
- ❑ Uma entidade pode representar um subconjunto de informações relevante para uma instância de uma entidade, chamada entidade subtipo (também conhecido como entidade secundária ou entidade categoria) (Michael Reingruber, 1994)

Para resumir, uma entidade:

- ❑ é um objeto de dados principal sobre o qual informações são coletadas
- ❑ é uma pessoa, local, coisa ou evento de informação
- ❑ pode ter uma instância (uma ocorrência)
- ❑ é uma coisa fundamental relevante para o usuário, sobre a qual uma coleção de fatos é mantida; uma associação entre entidades que contém atributos é por si só uma entidade
- ❑ envolve informações, uma representação de coisas similares que compartilham características ou propriedades
- ❑ é frequentemente representada graficamente em um modelo de dados por um retângulo, com o nome escrito dentro do mesmo

- Elemento de Dados** No mundo da modelagem de dados, o elemento básico é chamado *elemento de dados* ou *item de dados*. Ele é
- ❑ o componente fundamental
 - ❑ a partícula fundamental no universo do sistema de informações (Gary Schutt)
 - ❑ a menor unidade de dados com nome que tem significado no mundo real/do usuário (Graeme Simsion)

Atividades da Modelagem de Dados

A modelagem de dados aborda itens de dados, registros lógicos e arquivos. Um **Sistema de Arquivos** é composto de registros de itens de dados. **Itens de dados** são definidos como a menor unidade de dados com nome que tem significado para o mundo real. Um grupo de itens relacionados tratado como uma unidade é conhecido como um **registro**. Um **arquivo** é uma coleção de registros de um único tipo.

Na implementação física de dados por meio de bases de dados relacionais, são usados os seguintes termos: um item de dados é chamado “atributo” ou “coluna”, um registro é chamado “linha” ou “tupla” e um arquivo é chamado “tabela”. Esses termos não mudam o significado básico dos conceitos.

Mapeando Conceitos de Dados para a Terminologia de Pontos de Função

Podemos ainda mapear esses termos para a análise de pontos de função, conforme mostrado na seguinte matriz:

Conceito da Modelagem de Dados	Termo da Modelagem de Dados	Termo de Base de Dados Relacional	Termo da APF	Conceito da APF
Menor unidade de dado com nome que tem significado para o mundo real	Item de Dados	Atributo ou Coluna	Tipo de Dado Elementar (DER)	Um tipo de dado elementar (DER) é um campo único, não-repetido, reconhecido pelo usuário
Grupos de itens relacionados os quais são tratados como uma unidade	Registro	Linha ou Tupla	Tipo de Registro Elementar (RLR)	Um tipo de registro elementar (RLR) é um subgrupo de elementos de dados reconhecido pelo usuário e armazenado em um ALI ou AIE
Coleção de registros de um único tipo	Arquivo	Tabela	Arquivo Lógico (Arquivo Lógico Interno - ALI ou Arquivo de Interface Externa - AIE)	Arquivo refere-se a grupos de dados logicamente relacionados e não à implementação física desses grupos de dados

Uma vez que todos os dados tenham sido identificados, o analista de dados aplica várias regras de normalização para representar graficamente os dados em vários Diagramas de Entidade-Relacionamento. Um resumo das regras de normalização pode ser encontrado na Seção “Termos da Modelagem de Dados”.

Termos da Modelagem de Dados

Entidade (ou Tipo de Entidade)	<ul style="list-style-type: none">❑ Principais objetos de dados sobre os quais informações são coletadas❑ Pessoa, local, coisa ou evento de informação❑ Instância de entidade (uma ocorrência)❑ Representada graficamente por um retângulo, com o nome da entidade escrito em seu interior❑ Uma coisa fundamental relevante para o usuário, sobre a qual uma coleção de fatos é mantida. Uma associação entre entidades que contém atributos é por si só uma entidade
Tipo de Entidade Associativa	Um tipo de entidade que contém atributos que descrevem em mais detalhe um relacionamento de muitos para muitos entre dois outros tipos de entidades.
Tipo de Entidade Atributiva	Um tipo de entidade que descreve em mais detalhe uma ou mais características de outro tipo de entidade.
Entidade Subtipo	Uma subdivisão de tipo de entidade. Um subtipo herda todos os atributos e relacionamentos de seu tipo de entidade pai e pode ter atributos e relacionamentos adicionais próprios.
Relacionamentos	<p>Representam associações do mundo real entre uma ou mais entidades</p> <ul style="list-style-type: none">❑ Um-para-Um❑ Um-para-Vários❑ Vários-para-Vários❑ Representados por uma linha a qual conecta as entidades.❑ O nome do relacionamento é escrito ao lado da linha <p>Relacionamentos são definidos por como as entidades são conectadas:</p> <ul style="list-style-type: none">❑ Opcionais, apresentadas no texto com parêntesis 1:(N), (1):(N)❑ Obrigatórias, apresentadas no texto sem parêntesis 1:1, 1:N
Atributos	<ul style="list-style-type: none">❑ Uma característica de uma entidade. Atributos são geralmente análogos a Tipos de Dados Elementares (DERs).

Normalização Dados são normalizados pelo uso de 5 regras

1. Elimine Grupos Repetidos (1ª Forma Normal)
2. Elimine Dados Redundantes (2ª Forma Normal)
3. Elimine Colunas não dependentes da Chave (3ª Forma Normal)
4. Isole Relacionamentos Independentes múltiplos (nenhuma tabela pode conter dois ou mais relacionamentos 1:N ou N:M – 4ª Forma Normal)
5. Isole relacionamentos múltiplos semanticamente relacionados (restrições práticas podem justificar a separação de relacionamentos logicamente relacionados de muitos para muitos – 5ª Forma Normal)

Ao realizar a análise de pontos de função, é preferível analisar o modelo lógico em 3ª Forma Normal.

Ignore múltiplas entidades incluídas em função da tecnologia (em geral 5ª forma normal)

Conceitos de Entidade-Relacionamento

Uma vez que todos os dados necessários tenham sido identificados, o analista de dados aplica várias regras de normalização para representar graficamente os dados em vários Diagramas de Entidade-Relacionamento. A tabela a seguir pode ser aplicada para melhor entender o conceito de Tipo de Registro Elementar.

Conceito de Entidade-Relacionamento	Termo E-R	Termo na APF	Definição do IFPUG
Objeto de dados principal sobre o qual informações são coletadas (pessoa, local, coisa ou evento); um item de relevância fundamental para o usuário sobre o qual uma coleção de fatos é mantida	Entidade ou Tipo de Entidade	Arquivo Lógico Interno (ALI) ou Arquivo de Interface Externa (AIE)	O arquivo se refere a um grupo de dados logicamente relacionados e não à implementação física desses grupos de dados
Um tipo de entidade que contém atributos que descrevem complementarmente relacionamentos entre outras entidades	Tipo de Entidade Associativa	Tipo de Registro Elementar (RLR)	Subgrupo de elementos de dados reconhecido pelo usuário em um ALI ou AIE (opcional ou obrigatório)
Um tipo de entidade que descreve complementarmente uma ou mais características de outro tipo de entidade	Tipo de Entidade Atributiva	Tipo de Registro Elementar (RLR)	Subgrupo de elementos de dados reconhecido pelo usuário em um ALI ou AIE (opcional ou obrigatório)
Uma divisão de um tipo de entidade, a qual herda todos os atributos e relacionamentos de seu tipo de entidade pai; pode ter atributos e relacionamentos adicionais, únicos	Entidade Subtipo	Tipo de Registro Elementar (RLR)	Subgrupo de elementos de dados reconhecido pelo usuário em um ALI ou AIE (opcional ou obrigatório)

Passo 1: Identificar Arquivos Lógicos

Background

Na APF, um arquivo lógico é um grupo de dados conforme visto pelo usuário. Um arquivo lógico é composto de uma ou mais entidades de dados. Este capítulo fornece orientações sobre como agrupar as entidades candidatas identificadas em um ou mais arquivos lógicos.

O processo consiste dos seguintes passos, todos os quais são explicados em detalhe nos seguintes parágrafos desta seção:

Passo 1

Subpasso

1. Identifique todos os dados ou informações de controle reconhecidos pelo usuário logicamente relacionados no escopo da contagem
 2. Exclua as entidades não mantidas por qualquer aplicação.
 3. Agrupe em arquivos lógicos as entidades relacionadas que forem entidades dependentes
 4. Exclua aquelas entidades referenciadas como dados de código
 5. Exclua as entidades que não contenham atributos exigidos pelo usuário
 6. Remova as entidades associativas que contenham atributos adicionais não exigidos pelo usuário e entidades associativas que contenham apenas chaves estrangeiras; agrupe os atributos chave estrangeira com as entidades principais.
-

O passo mais difícil é o agrupamento de dados (subpasso 3). O agrupamento final de dados em arquivos lógicos é o resultado do efeito combinado de dois métodos de agrupamento:

- ☐ Método a) é orientado pelo processo, baseado nas transações de usuário na aplicação
- ☐ Método b) é orientado pelos dados, baseado nas regras de negócio

Contudo, como as transações do usuário também são (ou deveriam ser) baseadas em regras de negócio, cada método apoia o outro. Essa abordagem dupla pode revelar eventuais deficiências na especificação funcional e torna o processo de identificação de arquivos lógicos confiável e passível de repetição.

Subpasso 1.1

Identificar dados ou informações de controle logicamente relacionados reconhecidos pelo usuário dentro do escopo da contagem

Antes de tomar a decisão sobre quais entidades devem ser agrupadas em arquivos lógicos como um conjunto, deve-se determinar quais entidades candidatas devem ser consideradas para o agrupamento lógico das entidades (subpasso 1.3) e quais devem ser excluídas. Os passos a seguir ajudarão a identificar essas entidades de uma maneira passível de repetição.

Os princípios de orientação geral da Parte I – Medir Funções de Dados são claros: considere apenas entidades significativas e requeridas pelo usuário.

Preste atenção especial quando da identificação de arquivos lógicos a partir de um modelo (normalizado) de dados:

- ❑ Não assuma que todas as entidades são arquivos lógicos; por exemplo, arquivos de índice, entidades em um modelo de dados físico.
- ❑ Arquivos lógicos podem existir em uma perspectiva do usuário, mas em alguns casos podem não ser identificados no modelo (normalizado) de dados; por exemplo, arquivos históricos contendo dados agregados. Não esqueça de incluir esses arquivos lógicos no restante do processo.

Subpasso 1.2 Exclua entidades não mantidas por qualquer aplicação

Determine quais entidades não são mantidas por um processo elementar nesta ou em outra aplicação. Exclua essas entidades de considerações subsequentes, pois as mesmas não são contadas.

Subpasso 1.3 Agrupe em arquivos lógicos as entidades relacionadas que são entidades dependentes

Para cada entidade de dados restante, identifique como as entidades relacionadas devem ser agrupadas em arquivos lógicos, os quais refletem a “visão do usuário”; isto é, determine se as entidades de dados constituem por si mesmas um arquivo lógico independente ou se as entidades relacionadas devem ser agrupadas em um único arquivo lógico.

Identifique a visão do usuário (= visão de negócio) do agrupamento de dados investigando:

- a) Como os dados são acessados como um grupo por processos elementares dentro da fronteira da aplicação (Subpasso 1.3a, página 2-14)
- b) Os relacionamentos entre as entidades e a sua interdependência baseada nas regras de negócio (Subpasso 1.3b, da página 2-15 até a 2-24).

Subpasso 1.4 Excluir entidades referenciadas como dados de código

Filtrar dados de código. Dados de código são incluídos como resposta a um requisito não-funcional do usuário (requisitos de qualidade, implementação física e/ou razão técnica). Dados de código são explicados em detalhe na Parte 3 – Capítulo 1 “Dados de Código”.

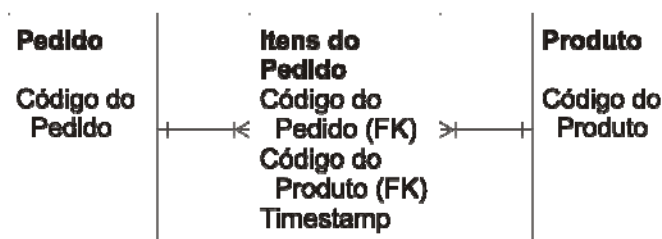
Subpasso 1.5 Excluir entidades que não contenham atributos requeridos pelo usuário

Determinar quais entidades não contém atributos reconhecidos e requeridos pelo usuário, contendo apenas atributos não funcionais. Exemplos de atributos não funcionais são aqueles que existem como um resultado de um projeto ou consideração de implementação; por exemplo, índice de arquivos criados por razões de performance, tais como índices alternados (ver Parte 1 – Medir Funções de Dados). Excluir tais entidades de considerações posteriores; as mesmas não são contadas como um arquivo lógico ou RLR.

Subpasso 1.6 Remover entidades associativas que contenham atributos adicionais não requeridos pelo usuário e entidades associativas que contenham apenas chaves estrangeiras; agrupar atributos de chaves estrangeiras com as entidades primárias

- 1.6.1** Determinar quais entidades são entidades associativas. Uma entidade associativa contém chaves estrangeiras de entidades conectadas juntamente com outros atributos. Note que duas situações podem surgir como resultado:
- Os atributos adicionais não-chave são um resultado do projeto ou consideração de implementação, ou existem para satisfazer um requisito técnico (não requeridos pelo usuário; ex.: um campo de data/hora com o propósito de recuperação de dados). Estes atributos técnicos não são contados como elementos de dados. Trate estas entidades como entidades key-to-key (veja abaixo).
 - Os atributos adicionais não-chave são necessários para satisfazer os requisitos funcionais do usuário e são requeridos pelo usuário. Estas entidades são avaliadas nas próximas sessões: Identificar Arquivos Lógicos (Passos 1.3a/1.3b).

Exemplo:



Timestamp é normalmente um atributo técnico não reconhecido pelo usuário. Neste caso, para a entidade Itens do Pedido, aplica-se a situação 1.3a. A entidade key-to-key é resolvida pela inclusão do Código do Produto como uma chave estrangeira no Pedido e do Código do Pedido no Produto (passo 1.6).

- 1.6.2** Determinar quais entidades são entidades key-to-key (intersecção); por ex., eles possuem apenas chaves como elementos de dados e não têm nenhum outro atributo não-chave.

Estas entidades normalmente representam a implementação de uma relação muitos para muitos (N:M) em um modelo de dados normalizado. Existem apenas por razões de modelagem de dados e projeto de banco de dados e não como resultado de um requisito do usuário.

Exclua estas entidades de outras considerações; elas não são contadas como arquivo lógico ou RLR. De acordo com as regras (Parte 1), o atributo que faz referência (chave estrangeira) é contado como um elemento de dado em ambas entidades conectadas pela entidade key-to-key. Veja também as diretrizes na Sessão “Passo 3: Identificar Tipos de Elementos de Dados”.

Verificação Final

Verificar se todas as entidades restantes são resultado de requisitos funcionais do usuário. Estas entidades e as relações e interdependências entre as mesmas serão abordados na próxima sessão: Identificar Arquivos Lógicos (subpasso 1.3a/1.3b).

Identificar Arquivos Lógicos Utilizando o Método de Processos Elementares (Subpasso 1.3a)

A visão de negócio do usuário sobre os dados é refletida em como as transações do usuário acessam os dados.

Reveja como os processos elementares dentro da fronteira da aplicação mantêm as entidades. Se várias entidades são sempre *criadas* juntas e *excluídas* juntas então esta é uma forte indicação de que as mesmas devem ser agrupadas dentro de um único arquivo lógico. Reveja também os processos elementares usados para extrair os dados, para determinar se o processo de extração acessa o mesmo grupo de entidades. Nota: as transações que modificam dados frequentemente têm como alvo apenas uma entidade no grupo; dessa forma, as transações de modificação não fornecem uma orientação tão eficaz para agrupamento de dados quanto as transações de inclusão e exclusão.

Exemplo

Um pedido de compra do cliente é um grupo único de dados a partir da perspectiva do negócio do usuário; ele é composto dos Dados Básicos do Pedido (cliente, endereço, data, etc.) e dos detalhes sobre cada item pedido. A partir da perspectiva do negócio, um pedido não pode ser criado sem pelo menos um item e se o pedido for excluído, tanto os dados básicos como todos os seus itens serão excluídos. Entretanto, os dados básicos e os itens podem ter transações de manutenção independentes; por ex., a alteração do status do pedido é uma função diferente da alteração dos itens do pedido. As funções de *inclusão* e *exclusão* indicam, a partir da perspectiva do usuário, que “pedido” é um arquivo lógico único que agrupa os dados básicos do pedido e os itens do pedido.

Utilize o subpasso 1.3a para validar os grupos de dados lógicos candidatos que foram identificados.

Identificar Arquivos Lógicos Utilizando o Método de (In)Dependência de Entidades (Subpasso 1.3b)

Introdução

O Método de (In)Dependência de Entidades, como definido e explicado nesta seção, fornece um método reproduzível para identificar corretamente Arquivos Lógicos (ALs) a partir de um modelo de dados. Nesta seção, o termo “entidade” refere-se a uma entidade em um modelo de dados normalizado (normalmente na terceira forma normal).

A Seção “Tipos de Relacionamentos” explica os diferentes tipos de relacionamentos e as diferenças entre os conceitos “relacionamento obrigatório/opcional” e “entidades dependentes /independentes”.

A Seção “(In)Dependência de Entidades Ilustrada para Todos os Tipos de Relacionamentos” explica o método em mais detalhes para cada tipo de relacionamento.

A Seção “Resumo: de Entidades para Arquivos Lógicos via (In)Dependência de Entidades” resume os tipos de relacionamentos e as condições para quando contar um AL.

O Método de Dependência de Entidades agrupa entidades pela avaliação dos relacionamentos e interdependências das entidades em comparação com as regras de negócio. Os princípios do guia são *entidades independentes* e *entidades dependentes*.

Entidades Independent

Entidade independente significa uma entidade que é significativa ou tem sentido para o negócio por si só, sem a presença de outras entidades.

Entidades Dependentes

Entidade dependente significa uma entidade que não é significativa ou não tem sentido para o negócio por si só, sem a presença de outras entidades, de modo que:

- uma ocorrência da entidade X deve estar ligada a uma ocorrência da entidade Y
- a eliminação de uma ocorrência da entidade Y resulta na eliminação de todas as ocorrências relacionadas da entidade X

Nota

Não confunda o conceito de *entidade independente/entidade dependente* com o conceito de *relacionamento opcional/obrigatório*. Os exemplos na Seção “(In)Dependência de Entidade Ilustrada para Todos os Tipos de Relacionamentos” mostram claramente que estes são conceitos diferentes.

Determinar Dependência

Para determinar se a entidade B é dependente ou independente da entidade A, é preciso determinar:

“B é significativa para o negócio independentemente da ocorrência de A ligada a ela?”

Um teste simples para determinar a situação (entidade dependente ou independente) é o seguinte. Mesmo que não haja requisito do usuário para a exclusão, (ainda assim) faça a pergunta:

“Suponha que nós quiséssemos excluir uma ocorrência “a” da entidade A; o que aconteceria à ocorrência “b” da entidade B ligada a “a”?”

Dependendo das regras do negócio, distinguimos duas situações essencialmente diferentes:

Situação 1

Se, de acordo com as regras de negócio, uma ocorrência de B não tem significado/importância independente para o usuário e pode também ser excluída, então aparentemente a ocorrência de B não tem significado para o usuário independentemente da ocorrência correspondente de A. A entidade B é considerada uma entidade dependente de A. As entidades A e B devem ser agrupadas juntas no mesmo arquivo lógico.

Situação 2

Se a ocorrência de B tem significado para o negócio mesmo independentemente da ocorrência correspondente de A, as regras de negócio não permitirão a exclusão da ocorrência de B. As entidades A e B serão consideradas arquivos lógicos separados.

Avaliar o modelo de dados de um sistema de informação por meio da avaliação de todos os pares de entidades ligadas resulta na identificação dos arquivos lógicos.

Na Seção “Entidades (In)Dependentes Ilustrada para Todos os Tipos de Relacionamento” este método é explicado em mais detalhes para diferentes tipos de relacionamentos.

A Seção “Resumo: de Entidades para Arquivos Lógicos via (In-) Dependência de Entidades ” resume como contar cada tipo de relacionamento na APF.

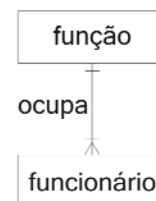
Tipos de Relacionamentos

Antes de assumir como conclusivos os princípios da (In)Dependência de Entidades para todos os tipos de relacionamento, deve-se entender claramente os diferentes tipos/naturezas dos relacionamentos. Esta seção explica os diferentes tipos, assim como os conceitos de “opcional” e “obrigatório”.

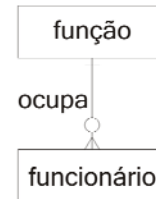
Exemplo Duas entidades, Função e Funcionário, por exemplo, podem ser conectadas entre si via um relacionamento; por ex.: “ocupa”.

Natureza do Relacionamento A natureza do relacionamento determina quantos funcionários podem trabalhar em uma função de acordo com o modelo de dados (0, 1 ou mais) e em quantas funções um funcionário pode trabalhar (0, 1 ou mais).

1 : N Assuma que as regras de negócio determinem que vários funcionários (no mínimo 1) podem ser utilizados em uma função, e que um funcionário tem que trabalhar em uma (e apenas uma) função. Neste caso dizemos que o relacionamento entre Função e Funcionário é 1:N



1 : (N) É mais provável que as regras de negócio determinem que uma função *pode* estar vaga, isto é, nenhum funcionário tenha sido alocado para a função. Neste caso o relacionamento entre Função e Funcionário é opcional e definido como 1:(N).

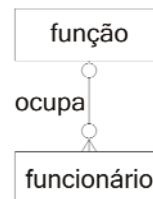


(1) : N Se as regras de negócio determinam que um funcionário pode existir sem uma função, mas uma função sempre tem um funcionário a ela alocado, definimos o relacionamento entre Função e Funcionário como (1):N



(1) : (N)

Na situação onde uma função pode estar vaga e um funcionário pode existir sem uma função, ambos os lados do relacionamento são opcionais. O relacionamento entre Função e Funcionário é definido como (1):(N).

**Conceito “Obrigatório/Opcional” versus “(In)Dependência de Entidades”.**

Para deixar clara a diferença entre os conceitos de “relacionamento obrigatório/opcional” e “dependência/independência de entidades”, assuma, como um exemplo, a seguinte extensão das regras de negócio “funcionário(s) não é(são) permitido(s) sem uma função” (relacionamento do tipo 1:N e 1:(N)).

Quando uma função se torna obsoleta, isto não significa que os funcionários não são mais significativos para o negócio. Um funcionário tem significado para o negócio independente da função relacionada. Funcionário é uma entidade independente de função. Devido ao relacionamento obrigatório com função, todos os funcionários têm que ser alocados a uma nova função, antes de a função poder ser excluída.

Então pode acontecer que uma ocorrência da entidade B (ex. Funcionário) possa ter um link obrigatório com uma ocorrência da entidade A (por ex. Função) no relacionamento A:B entre as entidades A e B, mas aquela entidade B é por si só significativa para o negócio. Neste caso, quando alguém quiser excluir uma ocorrência da entidade A, tem-se que antes reatribuir uma ocorrência ligada de B para outra ocorrência de A.

(In)Dependência de Entidade Ilustrada para Todos os Tipos de Relacionamento**(In)Dependência de Entidade em um Relacionamento (1):(N)****(1) : (N)**

Se um relacionamento entre duas entidades A e B é bilateralmente opcional, as entidades podem existir independentemente e (todas ocorrências de) A e B são significativas para o negócio independentemente da(s) ocorrência(s) relacionada(s) com a outra entidade.

Então, A e B são consideradas entidades independentes uma da outra. A APF conta as entidades A e B como dois arquivos lógicos separados, conforme indicado na tabela da Seção “Resumo: de Entidades para Arquivos Lógicos via (In-)Dependência de Entidades”.

(In)Dependência de Entidade em um Relacionamento 1:(N)

1 : (N) Em um relacionamento 1:(N) entre duas entidades A e B (veja figura 1), pode existir uma ocorrência da entidade A para nenhuma, uma ou muitas ocorrências da entidade B relacionadas. Por outro lado, cada ocorrência de B tem que ser associada a uma ocorrência de A.

Exemplo

No relacionamento 1:(N) entre Funcionário e Filho (ou Dependente) em uma Aplicação de RH, um Funcionário deve ter 0, 1 ou muitos Dependentes a ele relacionados, mas um Dependente tem que estar relacionado a um (e apenas um) Funcionário (veja figura 2).

Como B deve ser relacionado a um A, isto levanta a questão se B é dependente ou independente de A.

Para determinar se a entidade B é dependente ou independente de A, é necessário responder:

“B é significativo para o negócio independentemente do A a ela relacionado?”

Veja um teste simples para diferenciar a dependência e independência de entidades.

Mesmo que não existam requisitos do usuário para exclusão, faça a seguinte pergunta:

“Suponha que desejamos excluir uma ocorrência da entidade A; o que acontecerá com as ocorrências relacionadas da entidade B que têm um relacionamento obrigatório com uma ocorrência de A?”

As regras de negócio podem resultar em duas possibilidades:

Situação 1

Se a exclusão de A for permitida, todas as ocorrências de B relacionadas também deverão ser excluídas, pois o negócio não está mais interessado nas ocorrências de B. Por exemplo (veja figura 2): Uma aplicação de RH mantém informações sobre funcionários e seus dependentes. Assuma que as regras de negócio definiram que quando um Funcionário (A) deixa a companhia, não tem mais sentido para o negócio manter a informação sobre os dependentes (B).

Situação 2

A exclusão de A não é permitida enquanto ocorrências de B ainda estiverem a ela relacionadas, pois o negócio está ainda interessado nas ocorrências de B, mesmo além do contexto do A correspondente. Por exemplo (veja figura 3): Uma organização adota crianças e designa cada criança a um funcionário. O funcionário é a pessoa de contato entre a companhia e a criança. No caso de um funcionário deixar a companhia, as informações sobre a criança associada (do funcionário desligado) ainda são significativas para o negócio. Então, antes que se permita a exclusão do Funcionário (A), tem-se que primeiramente atribuir a Criança associada (B) a outro Funcionário (A) (pois a natureza deste relacionamento não permite uma Criança sem um relacionamento com Funcionário).

Na situação (1) dizemos que B é uma *entidade dependente* de A, e na situação (2) que B é uma *entidade independente* de A.

A APF conta as entidades A e B como um único arquivo lógico na situação (1) (*dependência*), enquanto na situação (2) A e B são arquivos lógicos separados (*independência*) conforme indicado na tabela da Seção “Resumo: de Entidades para Arquivos Lógicos via (In-)Dependência de Entidades”.

Ilustração do relacionamento 1:(N):

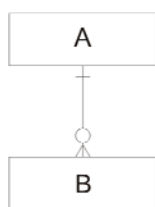


Fig. 1:

Cada entidade do tipo A pode referenciar 0, 1 ou muitas entidades do tipo B. Uma entidade do tipo B tem que referenciar exatamente uma entidade do tipo A.

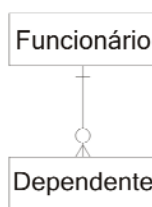


Fig. 2:

A aplicação de RH mantém informações sobre funcionários e seus dependentes.

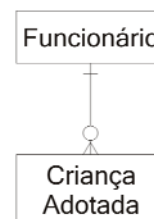


Fig. 3:

A aplicação de RH mantém informações sobre Funcionários e sobre as Crianças Adotadas que são designadas para um Funcionário.

As figuras 2 e 3 possuem modelos de dados similares, mas diferentes regras de negócio resultam em diferentes arquivos lógicos identificados.

(In)Dependência de Entidade em um Relacionamento (1):N

(1) : N Um relacionamento (1):N entre duas entidades A e B (veja figura 4) pode ser tratado de forma similar. Estes tipos de relacionamentos, entretanto, raramente aparecem na prática.

Em um relacionamento (1):N entre duas entidades A e B, cada A deve ser atribuído a 1 ou muitos Bs. Por outro lado, um B pode (mas não necessariamente) ser atribuído a uma ocorrência de A.

Exemplo

Em um relacionamento (1):N entre Comitê e Membro da Organização, um Comitê tem que ter membros (pelo menos 1). Um membro da organização pode (mas não necessariamente) servir em um Comitê (veja figuras 5 e 6).

Devido a uma ocorrência de A ter que estar relacionada a uma de B, levanta-se a questão se A é dependente ou independente de B.

Para determinar se a entidade A é dependente ou independente de B, precisa-se responder:

“A é significativa para o negócio independentemente da entidade B a ela relacionada?”

Veja a seguir um teste simples para diferenciar a dependência e independência de entidades.

Mesmo que não existam requisitos do usuário para exclusão, faça a pergunta:

“Assuma que temos uma ocorrência da entidade A à qual estão relacionadas uma ou mais ocorrências da entidade B. Suponha que desejamos excluir a última ocorrência relacionada à entidade B; o que aconteceria com esta ocorrência de A, que possui um relacionamento obrigatório com pelo menos uma ocorrência de B?”

As regras de negócio podem resultar em duas possibilidades:

Situação 1

Quando o último B é excluído, o A relacionado é também excluído pois o negócio não se interessa mais por ele. Por exemplo (veja a figura 5): Uma organização tem comitês aos quais membros são atribuídos.

A regra de negócio é que um comitê *deve* ter membros, mas nem todos os membros precisam participar de um comitê. Uma regra de negócio adicional é que a organização encerra um comitê assim que não haja mais membros participando do mesmo; pode-se dizer que os comitês são vistos como grupos de trabalho “ad hoc”.

Neste caso quando o último membro de um comitê sai do comitê, *não* tem sentido para o negócio manter informações sobre o comitê. Os dados do comitê são excluídos assim que o último membro deixa o comitê.

Situação 2

A exclusão do último B *não* é possível enquanto exista algum A ainda referenciado por ele, pois o negócio está ainda interessado neste específico A, mesmo além do contexto dos Bs que o referenciam. Por exemplo (veja figura 6), uma organização tem comitês aos quais membros são atribuídos.

A regra de negócio é que um comitê *deve* ter membros, mas nem todos os membros precisam participar de um comitê. Comitês são vistos como parte da estrutura organizacional. Eles têm significado para o negócio além dos membros que os servem.

Antes que o último membro de um específico comitê deixe o comitê, um novo membro tem que ser atribuído àquele comitê pois a natureza do relacionamento não permite um comitê sem membros.

Na situação (1), A é aparentemente *não* significativo para o negócio a menos que ele esteja relacionado a um ou mais Bs, enquanto na situação (2) ele *é* significativo.

Na situação (1) nós dizemos que A é uma *entidade dependente* de B e na situação (2) que A é uma *entidade independente* de B .

A APF conta as entidades A e B como um único arquivo lógico na situação (1) (*dependência*), enquanto na situação (2) A e B são arquivos lógicos separados (*independência*), como indicado na tabela da Seção “Resumo: de Entidades para Arquivos Lógicos via (In-)Dependência de Entidades”.

Ilustração do relacionamento (1):N:

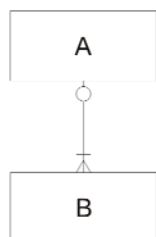


Fig. 4:
Cada entidade do tipo A tem que ser referenciada por 1 ou mais entidades do tipo B; uma entidade do tipo B pode, mas não necessariamente, referenciar uma entidade do tipo A.

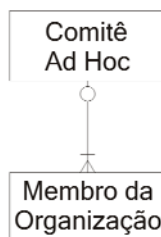


Fig. 5:
Membros de uma organização podem (mas não necessariamente) estar ativos em um comitê de trabalho. Um Comitê tem que ter (um ou mais) membros participando.

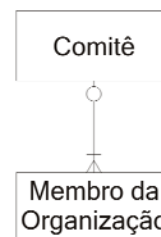


Fig. 6:
Membros de uma organização podem (mas não necessariamente) estar ativos em um comitê de trabalho. Um Comitê tem que ter (um ou mais) membros participando.

As figuras 5 e 6, tem modelo de dados similares, mas regras de negócio diferentes resultando em diferentes arquivos lógicos identificados.

(In)Dependência de Entidade em um Relacionamento 1:N

1 : N

Em um relacionamento 1:N entre duas entidades A e B, cada entidade B tem que ser atribuída a um e apenas um A, e a cada A tem que ser atribuído pelo menos a um B. Aplicam-se as mesmas regras de dependência e independência das entidades.



Situação 1

Se B *não* é significativo para o negócio independentemente do A a ele relacionado, então B é considerado uma entidade dependente de A.

Situação 2

Se B *é* significativo para o negócio independentemente do A a ele relacionado, então B é considerado uma entidade independente de A.

A APF conta as entidades A e B como um arquivo lógico na situação (1) (*dependência*), enquanto que na situação (2) A e B são arquivos lógicos separados (*independência*), como indicado na tabela da Seção “Resumo: de Entidades para Arquivos Lógicos via (In-)Dependência de Entidades”.

Resumo: De Entidades para Arquivos Lógicos via (In)Dependência de Entidades

Na tabela abaixo, A e B são duas entidades de um modelo de dados (normalizado) que devem ser contadas de acordo com esta Seção e que são interconectadas via um relacionamento. A tabela resume como as diversas situações são contadas.

Tipo de Relacionamento entre duas entidades, A e B	Quando esta Condição Existe	Então conte como Arquivo Lógico (AL)
(1) : (N)	(A e B são independentes)	2 ALs
1 : N	Se B é entidade dependente de A	1 AL
	Se B é entidade independente de A	2 ALs
1 : (N)	Se B é entidade dependente de A	1 AL
	Se B é entidade independente de A	2 ALs
(1) : N	Se A é entidade dependente de B	1 AL
	Se A é entidade independente de B	2 ALs
(1) : (1)	(A e B são independentes)	2 ALs
1 : 1	(A e B são dependentes)	1 AL
1 : (1)	Se B é entidade dependente de A	1 AL
	Se B é entidade independente de A	2 ALs
(N) : (M)	(A e B são independentes)	2 ALs
N : M	Se B é entidade dependente de A	1 AL
	Se B é entidade independente de A	2 ALs
N : (M)	Se B é entidade dependente de A	1 AL
	Se B é entidade independente de A	2 ALs

Legenda

AL = Arquivo lógico (ALI ou AIE)
 (..) = Lado opcional do relacionamento

Notas

1. Na dúvida, decida por entidades independentes.
2. Em algumas situações mais que duas entidades podem também formar um arquivo lógico.

Passo 2: Classificar Arquivos Lógicos

Os arquivos lógicos identificados precisam ser validados segundo as regras de contagem de ALI/AIE na Parte 1.

Classifique um arquivo lógico como um Arquivo Lógico Interno (ALI) se processos elementares dentro da fronteira da aplicação que está sendo contada, mantém (criam, alteram ou excluem) elementos de dados dentro do arquivo.

Classifique um arquivo lógico como um Arquivo de Interface Externa (AIE) se processos elementares dentro da fronteira da aplicação sendo contada apenas referenciam os elementos de dados dentro do arquivo, e o arquivo lógico é mantido por um processo elementar em outra aplicação.

Se um arquivo lógico identificado *não* é mantido por um processo elementar (dentro desta aplicação ou em outra), então o arquivo lógico não é contado de modo algum.

Passo 3: Identifique Tipos de Dados Elementares

O dado elementar é a menor unidade que tem significado para o usuário e representa um fato específico sobre um negócio, por exemplo:

Nome do Dado Elementar	Valor do Dado Elementar
Taxa	\$900
Data do Nascimento	15 Jan 1965
Nome	InfoMerge

Termos e Definições de Dados Elementares

Ao iniciar o estudo de um modelo de dados lógico, começamos considerando esses elementos de dados como atributos. Um atributo representa um fato específico sobre uma entidade ou um relacionamento. Na tabela abaixo as entidades são mostradas em MAIÚSCULO, os atributos em minúsculo:

Representação	Exemplo
ENTIDADE_atributo	CURSO_taxa
ENTIDADE.atributo	COMPANHIACLIENTE.nome

Atributos/elementos de dados podem ser encontrados em:

- ☐ Visões do usuário (relatórios, telas)
- ☐ Dicionários de dados (modelos do negócio, modelos de dados)
- ☐ Arquivos existentes (estrutura de registros em programas, layouts de arquivos)

Quando estiver revendo os dados, o analista de dados segue esta premissa básica: todos os elementos de dados reconhecidos pelo usuário devem ser tratados como um atributo, e dessa forma devem ser mostrados em relação a uma entidade específica.

O atributo pode ter as seguintes propriedades: nome (sinônimo), característica, propósito (uso), origem, valores válidos, valor (estrutura), unidade, e dependências. Iremos explorar estas propriedades antes de rever o mapeamento de DERs para atributos/elementos de dados da Análise de Pontos de Função do IFPUG.

Nome do Atributo	<p>Um nome único que resume as características apresentadas. Ele contém os seguintes componentes:</p> <p>origem (entidade/relacionamento) <i>seguido por um ponto</i></p> <p>descritivo (adjetivo do atributo) <i>seguido por um hífen</i></p> <p>classe (atributo base)</p> <p>Muitas tecnologias não aceitam espaços. Então vários nomes são concatenados com hífen.</p> <p>Exemplo:</p> <p>COMPANHIA_CLIENTE.endereço-entrega</p> <p>SEMINARIO_MATRICULA.efetividade-avaliação</p>
Característica	<p>A propriedade do ambiente sendo medido ou representado. O nome da entidade que tem a característica é sempre presente; por exemplo, Endereço-Entrega: endereço em que os materiais serão entregues para a COMPANHIA-CLIENTE.</p>
Propósito	<p>Fazer a pergunta “Como o atributo é utilizado pelo negócio” justifica o atributo.</p> <p>Exemplo: CURSO.data-qualificação</p> <p>Propósito: Usado nas seções de recapitulação de planejamento</p>
Dependências	<p>As situações onde outros valores de atributos no modelo influenciam ou restringem o valor deste atributo. Por exemplo, CURSO-grau final não pode existir antes do término do curso, mas tem que existir ao término do mesmo.</p>

**Atributos
Chave**

Fornecem o relacionamento entre uma entidade e outra. Existem diferentes tipos de chaves no modelo de dados, ex.: chaves primárias, chaves secundárias e chaves estrangeiras.

Uma **Chave Primária (PK)** é o identificador único de uma entidade.

Chaves Secundárias (SK) são atributos que fornecem acesso mais rápido às informações, como:

LIVROESCOLAR.preço (SK)

LIVROESCOLAR.nome-editora (SK).

Chaves Secundárias não fazem parte da informação do modelo de dados (modelo de dados lógico) mas são usados principalmente para auxiliar no acesso (implementação física).

Chaves Estrangeiras (FK) são atributos usados para representar relacionamentos de uma entidade com outra.

Atribuição

O último conceito de modelagem de dados que devemos considerar antes da análise dos DERs é Atribuição, que prescreve/descreve onde os atributos residem, dentro da entidade ou dentro do relacionamento. Existem algumas regras comuns de atribuição que são seguidas na modelagem de dados:

1. Um atributo é atribuído à sua melhor “origem” única, que é indicada na propriedade de característica no FORMULÁRIO DE DEFINIÇÃO DE ATRIBUTO.
2. O identificador único (chave primária) de uma entidade será atribuído também a cada relacionamento em que ele participa.

Existem algumas diretrizes adicionais a serem seguidas ao tentar colocar um atributo na entidade mais apropriada:

1. Se a definição do atributo referir-se a uma entidade, aloque o atributo àquela entidade
2. Se a definição do atributo referir-se a diversas entidades, então crie um relacionamento e aloque o atributo ao relacionamento ou à entidade à qual ele se aplicar.

Mapeando Elementos de Dados para Tipos de Elementos de Dados da APF

Agora que nós já revisamos os conceitos de elementos de dados e atributos na perspectiva da modelagem de dados, podemos relacionar estes conceitos às definições e regras de Pontos de Função do IFPUG:

Conceito de Modelagem de Dados	Termo da Modelagem de Dados	Termo de BD Relacional	Termo da APF	Conceito da APF
Menor unidade de dados definida que tem significado no mundo real	Item de Dados	Atributo ou Coluna	Tipo de Elemento de Dados (DER - Dado Elementar Referenciado)	Um tipo de elemento de dados (DER) é um campo reconhecido pelo usuário, único e não repetido
Grupos de itens relacionados que são tratados como uma unidade	Registro	Linha ou Tupla	Tipo de Registro Elementar (RLR - Registro Lógico Referenciado)	Um tipo de registro elementar (RLR) é um subgrupo de elementos de dados, reconhecido pelo usuário, dentro de um ALI ou AIE
Coleção de registros de um mesmo tipo	Arquivo	Tabela	Arquivo Lógico (Arquivo Lógico Interno – ALI ou Arquivo de Interface Externa – AIE)	Arquivo se refere a um grupo de dados relacionados logicamente e não à implementação física deste grupo de dados

Tipos de Elementos de Dados (DERs) são campos ou atributos, reconhecidos pelo usuário, únicos e não repetidos.

As seguintes regras se aplicam ao contar DERs em um arquivo lógico:

- Conte um DER para cada campo único, reconhecido pelo usuário e não repetido, mantido em/ou recuperado de uma função de dados através da execução de todos os processos elementares dentro de um

escopo de contagem

- Conte apenas aqueles DERs que estão sendo usados pela aplicação que está sendo contada quando duas ou mais aplicações mantêm e/ou referenciam a mesma função de dados
- Conte um DER para cada atributo requerido pelo usuário para estabelecer um relacionamento com outra função de dados
- Revise atributos relacionados para determinar se são agrupados e contados como um único DER ou se são contados como vários DERs; o agrupamento vai depender de como o processo elementar utiliza os atributos dentro da aplicação

Não conte atributos que existam puramente para satisfazer um requisito técnico e não foram especificados pelo usuário. Exemplos destes atributos não funcionais são atributos resultantes de considerações de projeto ou de implementação.

Exemplo: O campo PEDIDO_data é contado como um DER no Pedido já que ele precisa ser mantido para satisfazer um requisito do negócio do usuário. Entretanto, a marca (stamp) de data e hora de cada registro do pedido existe para satisfazer a integridade e a confiabilidade dos dados. A solução técnica para estes requisitos de qualidade foi copiar o banco de dados e disponibilizar para recuperação baseado nesta informação da marca (stamp). Consequentemente, a marca (stamp) de data e hora não deve ser contada como um DER.

Outras Situações

A seguir exemplos de contagem de tipo de elementos de dados (DERs).

Atributos

Atributos que são compostos de diversos elementos de dados relacionados são armazenados separadamente.

Devem os atributos serem contados como diversos elementos de dados ou como um único elemento de dados? As regras de DER na Parte 1 dizem para você “contar cada campo reconhecido pelo usuário”.

Como você determina se ele é reconhecido pelo usuário como uma coisa ou várias coisas? Reveja as transações dentro da aplicação para determinar se o atributo é tratado como um item ou mais de um.

Considere os seguintes itens na tomada de decisão:

- a) Se o atributo é sempre usado por inteiro, então ele é contado como um único elemento de dados (DER). Não devem existir situações em que um componente individual de um atributo é usado sem os outros. Baseado neste uso, o atributo é contado como um único elemento de dado.
- b) Se em algumas situações, apenas uma parte do atributo (ex. o sobrenome) é usada, então mais do que um elemento de dados deve ser contado. Olhe para o uso em componentes dentro da aplicação para determinar quantas partes reconhecidas existem. A opção não é necessariamente um ou todos. Baseado no que você está vendo, pode ser apropriado contar apenas dois DERs, ainda que existam na realidade cinco partes físicas.
- c) Olhe para a existência de requisitos de ordenação ou de edições e critérios de seleção. Se uma lista ou relatório é ordenado ou selecionado por um simples componente do atributo, isto sugere independência de componentes na visão do usuário.

Contando Nomes

Nome (primeiro nome, nome do meio, último nome)

Muitas aplicações precisam manter informações sobre os nomes das pessoas.

O nome deve ser contado como vários elementos de dados ou um elemento de dados único?

Reveja as transações dentro da aplicação para determinar se o nome é tratado como um item ou mais do que um item. Por exemplo: Nos Estudos de Caso 1,2 e 3, veja como o Nome do Funcionário é usado em várias funções de transação.

Essas funções sempre usam o nome inteiro ou algumas vezes apenas usam uma parte?

Nos Estudos de Caso, o Nome do Funcionário é sempre usado inteiramente. Não existem telas ou relatórios onde uma única parte do nome é usada sem as outras partes. Também não existem situações onde uma parte do nome é usada para ordenação, edição ou critério de seleção. Seu uso dentro da aplicação sugere que Nome do Funcionário é um elemento de dados único (DER).

Contando Endereços

Endereço (endereço, cidade, estado e CEP)

Muitas aplicações precisam manter informações sobre endereços. O endereço ser deve contado como vários elementos de dados ou um elemento de dados único?

Reveja as transações dentro da aplicação para determinar se o endereço é tratado como um item ou mais do que um item. Por exemplo: Nos Estudos de Caso 1, 2 e 3, veja como o Endereço de Localização é usado em várias funções de transação.

As transações sempre referenciam o endereço inteiro ou algumas vezes apenas usam uma parte?

Nos Estudos de Caso, o Endereço é sempre usado inteiramente. Não existem telas ou relatórios onde uma única parte do endereço seja usada sem as outras partes. Não existe situação onde uma parte do endereço seja usada para ordenação, edição ou critério de seleção. Seu uso dentro da aplicação sugere que Endereço da Localização é um elemento de dado único (DER).

Se existisse uma lista de locais que permitisse ao usuário listar todos os locais em uma cidade, estado ou CEP específico, mais de um elemento de dado deveria ser contado. Com base nas informações fornecidas, aparentemente cidade, estado e CEP são reconhecidos pelo usuário como partes independentes do endereço. Dessa forma, quatro DERs devem ser contados (Endereço, Cidade, Estado e CEP).

Contando Campos Repetitivos

Muitas vezes as aplicações mantêm múltiplas ocorrências de um elemento de dados. De acordo com as regras de DER na Parte 1, conte um campo repetitivo apenas uma vez.

Depois que os campos repetitivos forem reduzidos para um único DER, verifique se os requisitos do negócio estão ainda sendo satisfeitos. Considere os seguintes exemplos:

Exemplo 1: Código do Funcionário

Nos Estudos de Casos 1, 2 e 3, olhe para os requisitos de Manter Funcionário e para o relacionamento entre Funcionário e Dependente no Diagrama de Entidade e Relacionamento (DER). De acordo com os resultados dos Estudos de Casos, Funcionário é um AL que inclui Dependente. Os arquivos lógicos de Funcionário e Dependente conteriam cada um o Código do Funcionário.

Aplicando as regras de DER “repetitivos”, conte Código do Funcionário como um único DER para o ALI Funcionário. Agora determine se os requisitos do negócio ainda são satisfeitos.

O requisito era manter Dependente como uma parte da informação do Funcionário. Sim, os requisitos de negócio estão satisfeitos, portanto, um DER é contado.

Exemplo 2: Horas Trabalhada Diariamente

Um sistema de reportar horas tipicamente mantém o número de horas que uma pessoa trabalha a cada dia. Ao rever as estruturas de dados, as Horas Trabalhadas são salvas separadamente para cada dia da semana (Horas Trabalhadas de Segunda-feira, Horas Trabalhadas de Terça-feira, etc.).

Aplicando as regras de DER “repetitivos”, conte apenas Horas Trabalhadas. Determine se os requisitos do negócio ainda são satisfeitos. Se a aplicação apenas mantém informação sobre Horas Trabalhadas, está satisfeito o requisito do negócio para manter horas trabalhadas a cada dia?

Não, não está. A fim de satisfazer aquele requisito, conte Dia da Semana. A aplicação tem a habilidade de acompanhar separadamente cada hora trabalhada a cada dia (Horas Trabalhadas de Segunda-feira, Horas Trabalhadas de Terça-feira, etc.), portanto, dois DERs são contados.

Contando Campos de Status

As aplicações frequentemente mantêm informação do status atual dos dados (ex., Ativo, Inativo, Pendente, Aprovado, etc.). Este status é normalmente atualizado através das diversas transações dentro da aplicação. Estes campos de status podem ou não ser fisicamente visíveis ao usuário através das transações da aplicação. Considere os seguintes exemplos:

Exemplo 1: Status Inativo

Os Estudos de Caso 1, 2 e 3, incluem um indicador de Status. Quando um cargo ou um funcionário é excluído, os requisitos do usuário indicam que toda atribuição ao cargo excluído deve ser atualizada para o status de “inativo”.

Ao rever as telas de atribuição de cargo em toda a aplicação, o status nunca é mostrado ou atualizado diretamente em nenhuma tela. Apesar da falta de visibilidade, o Status ainda é contado como um DER para o cargo atribuído. O fato dele aparecer no Modelo de Dados Lógico e nos Requisitos do Usuário sugerem que ele é reconhecido pelo usuário no Arquivo Lógico, mas em nenhuma transação. Então, um DER é contado no ALI.

Exemplo 2: Status Não Contado

O usuário solicita a habilidade de excluir Funcionários. A equipe técnica não quer fazer a exclusão física dos registros; então, eles implementaram um “flag de status” em Funcionário. Quando o usuário exclui um Funcionário, o Status é marcado como “Inativo”. O usuário desconhece a existência do campo de Status em Funcionário. Então, o campo Status não deve ser contado como um DER.

Contando Datas do Sistema

As aplicações frequentemente retêm datas do sistema associadas com seus dados para refletir a versão dos dados. Datas do Sistema podem ter muitos nomes diferentes (última atualização, última aprovação, etc.) e são frequentemente acompanhadas pelo código do usuário (última atualização por, última aprovação por). Estas datas do sistema são tipicamente atualizadas através de diversas transações dentro da aplicação. Em muitos casos, a data do sistema é mantida por requisitos de negócio. O usuário necessita saber quando o dado foi alterado ou aprovado. Existem também casos onde a data do sistema está sendo mantida apenas por razões técnicas.

Considere os seguintes exemplos:

**Exemplo 1:
Data da
Efetivação** Os Estudos de Casos 1, 2 e 3, incluem uma referência à Data da Efetivação. Quando um cargo ou um funcionário é excluído, os requisitos do usuário indicam que qualquer atribuição de cargo associada deve ser atualizada para marcar a data da efetivação como sendo a data atual do sistema.

Ao rever as telas de atribuição de cargo em toda a aplicação, a data da efetivação nunca é mostrada ou atualizada diretamente em nenhuma tela. Apesar da falta de visibilidade, a Data da Efetivação é ainda contada como um DER para cargo. O fato dele aparecer no Modelo de Dados Lógico e os Requisitos do Usuário referenciarem-no pelo nome sugerem que o mesmo é reconhecido pelo usuário. Então, um DER é contado no Arquivo Lógico.

**Exemplo 2:
Data da
Recuperação** A ferramenta de Backup/Recuperação utilizada pela aplicação usa a data do sistema armazenada na tabela para recuperar o dado para um ponto particular no tempo. Neste caso, a *Data da Recuperação* não é reconhecida pelo usuário e não deve ser contada.

**Exemplo 3:
Data da
Auditação** A equipe técnica decide que deve registrar a data do sistema e o código do usuário sempre que o dado é alterado para resolver qualquer questão futura sobre quando ou por quem uma alteração foi feita. Neste caso, a *data do sistema* não é reconhecida pelo usuário e não deve ser contada.

Contando Chaves Estrangeiras

As aplicações frequentemente mantêm relacionamentos entre uma entidade e outra. Em alguns casos eles existem para satisfazer requisitos de validação de dados, mas em outros casos eles definem regras de negócio entre as duas entidades. O conceito de Atribuição na modelagem de dados é melhor ilustrado pela criação de chaves estrangeiras.

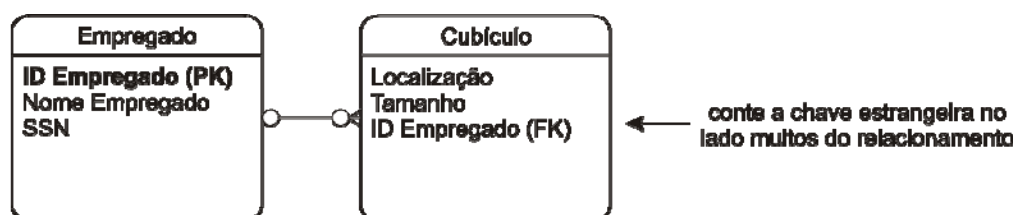
**Exemplo 1:
Local
(N) : (1)** Os Estudos de Casos 1, 2 e 3 incluem os requisitos do usuário para Inclusão e Atualização de Funcionários: “O *local* deve ser um local válido no Sistema de Ativo Fixo (SAF).”

O Diagrama de Entidades e Relacionamentos também reforça este requisito do negócio pela ilustração do relacionamento entre a entidade Funcionário e a entidade Local. De acordo com o diagrama, um Funcionário pode ter um Local, e um Local pode ter muitos Funcionários. Nesta situação, o Nome do Local deve ser incluído nos atributos das tabelas lógicas e físicas. Conte o atributo Nome do Local como um elemento de dados (DER) para Funcionário.

**Exemplo 2:
Cubículo** Considere uma variação do exemplo anterior. Um empregado pode ter um número ilimitado de cubículos. Um cubículo só pode ser ocupado por um

(1) : (N)

empregado por vez. O Código do Cubículo deve ser válido como identificado na tabela CUBÍCULO. O Diagrama de Entidades e Relacionamentos estaria garantindo o requisito do negócio pela ilustração do relacionamento entre a entidade CUBÍCULO e a entidade FUNCIONARIO. De acordo com o diagrama, um Empregado pode ter muitos cubículos, mas um Cubículo só pode ser ocupado por um Empregado. Nesta situação, o Código do Empregado é um atributo em CUBÍCULO.

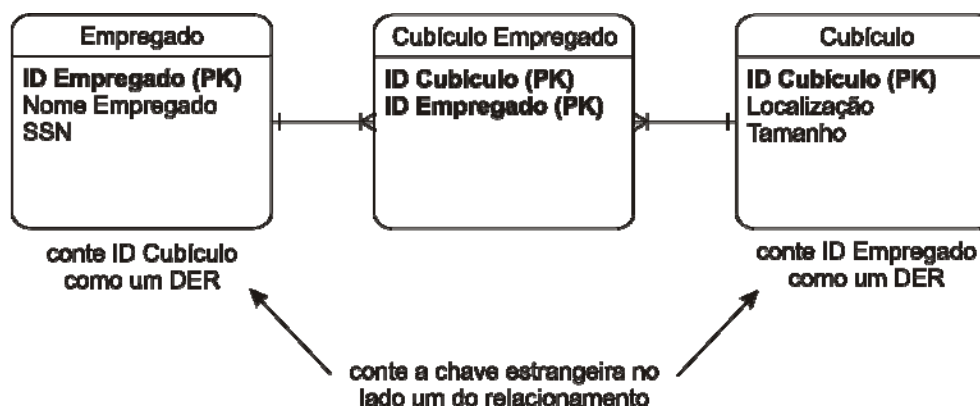


Nota: nem todos os atributos estão retratados nas entidades.

O relacionamento deveria ser refletido na tabela CUBÍCULO pela identificação do Empregado que ocupa o Cubículo. O código do Empregado é contado como um elemento de dado (DER) para Cubículo.

Exemplo 3 Cubículo (N) : (M)

Considere uma variação do exemplo anterior. Um empregado tem que ter pelo menos um cubículo, mas pode ter um número ilimitado de cubículos. Um cubículo pode ser ocupado por mais de um empregado por vez. O Código do cubículo deve ser válido como identificado no CUBÍCULO. O Diagrama de Entidades e Relacionamentos novamente garantiria o requisito de negócio pela ilustração do relacionamento entre Empregado e Cubículo. De acordo com este diagrama, um Empregado pode ter muitos cubículos, e um Cubículo pode ser ocupado por muitos Empregados.



Nota: nem todos os atributos estão retratados nas entidades.

O relacionamento é demonstrado na tabela Cubículo-Empregado, o qual conteria uma ocorrência de cada empregado no relacionamento com o

cubículo. Ele incluiria o Código do Empregado e o Código do Cubículo como chave primária. Como explicado nas Seções “Passo 1: Identificar Arquivos Lógicos” e “Passo 4: Identificar Tipos de Registros Elementares”, Cubículo-Empregado não é contado como um Arquivo Lógico nem como um RLR. Código do Cubículo é contado como um elemento de dados (DER) em Empregado porque estabelece um relacionamento com a entidade Empregado e Código Empregado é contado como um elemento de dados (DER) em Cubículo porque estabelece um relacionamento com a entidade Cubículo.

Observação

A identificação do número correto de DERs não influencia o número de arquivos lógicos, mas apenas sua complexidade. Enquanto, este passo influencia o tamanho funcional de forma limitada, o efeito é consideravelmente menor do que no Passo 1: Identificar Arquivos Lógicos.

Passo 4: Identifique Tipos de Registro Elementares

O tipo de registro elementar (RLR) representa a visão do usuário dos *subgrupos* de dados dentro de um arquivo lógico identificado, o qual foi discutido anteriormente na Seção “Passo 1: Identificar Arquivos Lógicos”.

Tipos de registros elementares correspondem tipicamente a entidades que foram agrupadas em arquivos lógicos como discutido na Seção “Passo 1: Identificar Arquivos Lógicos”. Eles precisam ser revistos cuidadosamente para garantir que o usuário os identifique como um subgrupo lógico, e deste modo, sejam contados como um Tipo de Registro Elementar (RLR).

Termos e Definições de Tipo de Registro Elementar

Este capítulo mostra um modelo de dados lógico na 3^a. Forma Normal e ignora diversas entidades criadas por razões técnicas; se você não tem um modelo de dados uma tentativa deve ser feita para (des)normalizar os dados.

As definições estão baseadas nos conceitos de modelo de dados descritos na Seção “Conceitos de Modelagem de Dados”.

Tipo de Entidade Associativa

Um tipo de entidade que contém atributos que descrevem em detalhe um relacionamento muitos-para-muitos entre dois outros tipos de entidades, também conhecida como entidade de intersecção.

Tipo de Entidade Atributiva

Um tipo de entidade que descreve em mais detalhes uma ou mais características de outro tipo de entidade.

Entidade Subtipo

Uma subdivisão de um tipo de entidade; herda todos os atributos e relacionamentos do seu tipo de entidade pai, e pode ter atributos e relacionamentos adicionais próprios.

Mapeando os Termos de Modelagem de Dados para a Terminologia de Pontos de Função

Os termos de Modelagem de Dados podem ser mapeados para a análise de pontos de função como mostrado na seguinte tabela:

Conceito de Modelagem de Dados	Termo de Modelagem de Dados	Termo de Banco de Dados Relacional	Termo da APF	Conceito de APF
Grupos de itens relacionados que são tratados como uma unidade	Registro	Linha ou Tupla	Tipo de Registro Elementar (RLR – Registro Lógico Referenciado)	Um <i>tipo de registro elementar</i> (RLR) é um subgrupo de elementos de dados, reconhecido pelo usuário dentro de um ALI ou AIE.
Coleção de registros de um mesmo tipo	Arquivo	Tabela	Arquivo Lógico (Arquivo lógico interno – ALI ou Arquivo de interface externa – AIE)	Arquivo se refere a um grupo de dados relacionados logicamente e não à implementação física deste grupo de dados.

A seguinte tabela pode ser aplicada para auxiliar o entendimento do conceito de Tipo de Registro Elementar.

Conceito de Entidades e Relacionamentos	Termo de Entidades e Relacionamentos	Termo da APF	Conceito de APF
Principais objetos de dados sobre os quais	Entidade ou Tipo de	Arquivo Lógico	Arquivo refere-se a um grupo de dados logicamente

informações são coletadas (pessoa, lugar, coisa ou evento); um item de fundamental importância para o usuário sobre os quais uma coleção de fatos é mantida.	Entidade		relacionados e não à implementação física deste grupos de dados; se não existirem outros subgrupos, o arquivo lógico é contado com um único Tipo de Registro Elementar (RLR)
Um tipo de entidade que contém atributos que ajudam a descrever um relacionamento entre outras entidades.	Tipo de Entidade Associativa	Pode ser um arquivo lógico ou um possível tipo de registro elementar (RLR); veja a Seção “Analisando Entidades Associativas para determinar RLRs” para mais considerações.	Subgrupo de elementos de dados reconhecidos pelo usuário dentro de um ALI ou AIE, pode ser opcional ou obrigatório.
Um tipo de entidade que ajuda a descrever uma ou mais características de outro tipo de entidade.	Tipo de Entidade Atributiva	Possivelmente um Tipo de Registro Elementar (RLR); veja a Seção “Analisando Entidades Atributivas para Determinar RLRs” para mais considerações.	Subgrupo de elementos de dados reconhecidos pelo usuário dentro de um ALI ou AIE, pode ser opcional ou obrigatório.
Uma divisão do tipo de entidade, que herda todos os atributos e relacionamentos de seu tipo de entidade pai, e pode ter atributos e relacionamentos adicionais únicos.	Entidade Subtipo	Possivelmente um Tipo de Registro Elementar (RLR); veja a Seção “Analisando Subtipos para Determinar RLRs” para mais considerações.	Subgrupo de elementos de dados reconhecidos pelo usuário dentro de um ALI ou AIE, pode ser opcional ou obrigatório.

A Análise de Pontos de Função considera as associativas, atributivas e subtipos como subgrupos de dados. Isto será explorado na discussão de como contar RLRs.

Analizando Entidades Associativas para Determinar RLRs

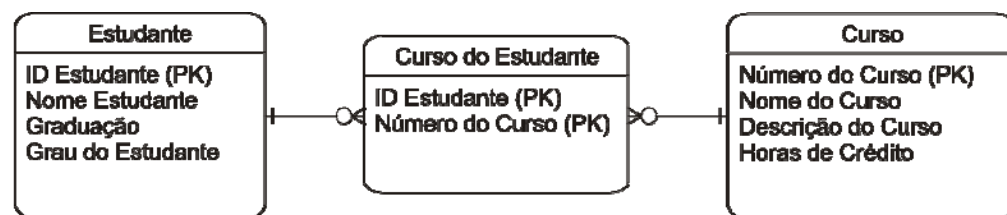
Uma entidade associativa é usada para associar duas ou mais entidades como uma maneira de definir um relacionamento muitos-para-muitos. Este tipo de entidade é frequentemente criada pelo modelador de dados para implementar algumas regras de negócio requeridas para relacionar duas entidades separadas.

Existem três possibilidades a considerar quando encontrar entidades associativas.

Situação 1

Entidade Associativa *não* é contada como um RLR

A secretaria acadêmica tem um requisito de gerenciar todos os estudantes registrados para um curso e saber os cursos que o estudante completou previamente. Curso é uma entidade e Estudante é uma entidade. O modelador de dados criou uma entidade associativa chamada Curso do Estudante como uma interseção entre as duas, e esta entidade associativa apenas contém as chaves de cada entidade.

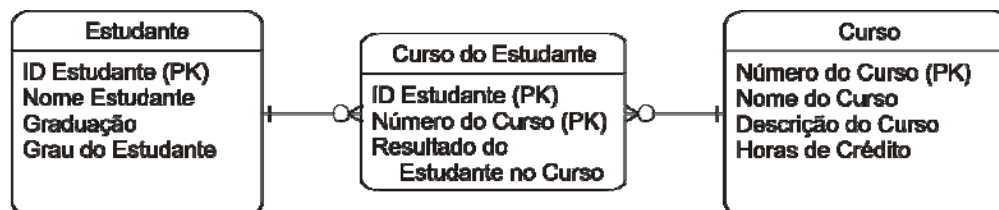


A entidade Curso do Estudante *não* é considerada um RLR nem deve ser contada como um arquivo lógico separado porque não contém nenhum elemento de dado adicional além das duas chaves primárias (PK) das entidades que fazem a interseção.

Estudante é um arquivo lógico com 1 RLR (Estudante) e Curso é um arquivo lógico com 1 RLR (Curso).

Situação 2 Entidade Associativa é contada como um RLR

A secretaria acadêmica tem um requisito de identificar todos os estudantes registrados para um curso. Além disso ela precisa da informação sobre os resultados do curso para o(s) estudante(s). O Curso é uma entidade e o Estudante é uma entidade. O modelador de dados criou uma entidade associativa chamada Curso do Estudante como uma interseção entre as duas, e esta entidade associativa contém as chaves de cada entidade bem como o resultado do curso do estudante.



Não existe regra de negócio que solicite que curso do estudante seja mantido independentemente; então Curso do Estudante não satisfaz as regras para ser contado como um arquivo lógico separado. Neste caso, a entidade Curso do Estudante é considerada um RLR, pois contém pelo menos um atributo reconhecido pelo usuário (*), além das duas chaves primárias das entidades que participam da intersecção.

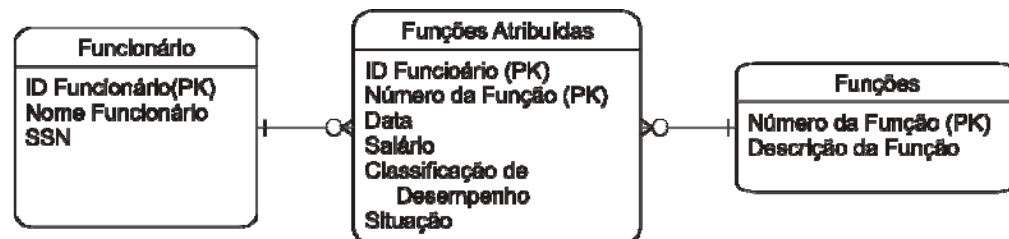
(*) Na Seção “Subpasso 1.5 Excluir entidades que não contém atributos requeridos pelo usuário”, atributos não chaves que são resultado de considerações de projeto ou de implementação, ou que satisfazem um requisito técnico, não são considerados elementos de dados.

A fim de ser contado como um RLR, um subgrupo deve conter um ou mais atributos além das chaves primárias. Estudante é um arquivo lógico com 2 RLRs (Estudante e Curso do Estudante) e Curso é um arquivo lógico com 2 RLRs (Curso e Curso do Estudante).

Se o requisito do negócio indicar que o relacionamento representado pela entidade associativa pertence a apenas um dos arquivos lógicos, o RLR será contado apenas naquele arquivo lógico.

Situação 3 Entidade Associativa contada como um *arquivo lógico*, com um único RLR

Um departamento de RH mantém informações sobre Funcionário, Funções e Funções Atribuídas. A entidade Funções Atribuídas é necessária, mesmo que um Funcionário não esteja mais associado com a Função ou que a Função não seja mais uma função disponível para alocação.



Nota: Nem todos os atributos são retratados como entidades.

Embora Funções Atribuídas seja uma entidade associativa, ela é mais que um mapeamento key-to-key entre duas entidades, e é mais que um RLR associado com um arquivo lógico. Se uma regra de negócio solicita que a informação de Funções Atribuídas deva ser retida independentemente, Funções Atribuídas é considerada um arquivo lógico como descrito na Seção “Identificar Arquivos Lógicos Utilizando o Método de (In)Dependência de Entidades (Subpasso 1.3b)”.

Se não existirem regras de negócio solicitando que as informações de Funções Atribuídas devam ser retidas independentemente, então esta entidade associativa é contada como na situação 2 acima.

Analizando Entidades Atributivas para Determinar RLRs

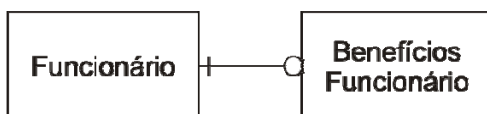
Uma entidade atributiva é um tipo de entidade que ajuda a descrever uma ou mais características de um outro tipo de entidade. Pela definição ela é uma extensão lógica de outra entidade; em Análise de Pontos de Função uma entidade atributiva representa um Tipo de Registro Elementar daquela entidade.

Uma entidade atributiva é contada como sendo um RLR do arquivo lógico que ela está definindo (Situação 1) ou como uma extensão do arquivo lógico (Situação 2).

Situação 1

Uma entidade atributiva *opcional*

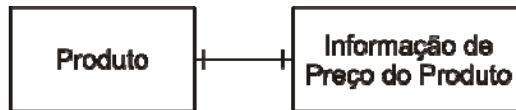
Um funcionário pode aderir a um plano de benefícios. Em nosso modelo de dados, Funcionário é uma entidade. Benefícios Funcionário é uma entidade atributiva de Funcionário e contém informações sobre os benefícios que o funcionário tem. Benefícios Funcionário não pode existir sem Funcionário, e é então logicamente relacionado.



Benefícios Funcionário é contado como um RLR pois é uma entidade atributiva opcional. Funcionário é um arquivo lógico com 2 RLRs, Funcionário e Benefícios de Funcionário.

Situação 2 Uma entidade atributiva *obrigatória*

Um sistema de vendas deve manter informações sobre cada produto e sobre seus respectivos preços. Produto é uma entidade. Informações de Preço do Produto é uma entidade atributiva relacionada a Preço, contendo: preço anterior, preço atual, preço futuro projetado e data efetiva do preço. Informações de Preço do Produto não existe sem Produto e é então logicamente relacionada.



Informações de Preço do Produto *não* é contado como um RLR. Produto é um arquivo lógico com 1 RLR, contendo Produto e Informações sobre Preço do Produto.

Analizando Subtipos para Determinar RLRs

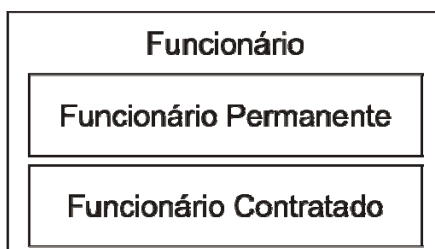
Uma entidade subtipo é uma subdivisão de tipo de entidade. Um subtipo herda todos os atributos e relacionamentos da entidade pai, e pode ter atributos e relacionamentos adicionais próprios. As regras de modelagem de dados determinam que uma entidade pode ter qualquer número de grupos de subtipos independentes associados a ela, os quais podem ser opcionais ou obrigatórios. Cada subtipo pode ter apenas um pai. Na modelagem de dados, embora o pai e o subtipo sejam representados como entidades diferentes, eles são logicamente parte da mesma entidade.

Ao analisar subtipos no modelo de dados, olhe para o relacionamento requeridos para a entidade “pai”, como mostrado nas seguintes situações.

Situação 1

Subtipo que é um subgrupo e então é contado como um RLR

Um funcionário tem que ser um funcionário permanente ou um funcionário contratado, mas não pode ser os dois. Os dados comuns do funcionário são pertinentes a todos os funcionários e são obrigatórios. Além disso, os dados comuns são herdados pelas entidades subtipo obrigatórias, permanente e contratado.



Na revisão destes dados a partir da perspectiva de pontos de função, dois subgrupos lógicos do arquivo lógico Funcionário são identificados:

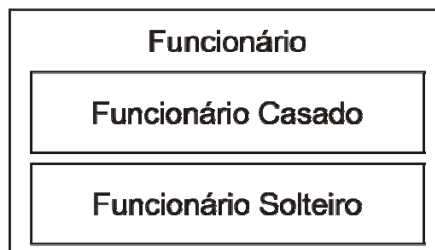
- ❑ Dados do funcionário permanente incluem informações do funcionário permanente como também as informações comuns do funcionário.
- ❑ Dados do funcionário contratado incluem informações do funcionário contratado como também as informações comuns do funcionário.

Nesta situação existe um arquivo lógico (Funcionário) com dois RLRs, funcionário permanente e funcionário contratado.

Situação 2 Subtipo que *não é* um subgrupo e então não é contado como um RLR

Se existem atributos únicos entre entidades subtipo, considere seriamente se um subgrupo separado realmente existe e desta forma constituiria um tipo de registro elementar (RLR). Um simples atributo opcional único não resultaria em um RLR diferente da perspectiva do usuário, mesmo se representado como uma entidade subtipo em um modelo de dados lógico.

O estado civil de um Funcionário pode ser casado ou solteiro. Se casado, o nome do cônjuge é armazenado. Embora possa ser representado como um subtipo em um modelo de dados, o nome do cônjuge é apenas um atributo opcional dentro do grupo lógico de dados do Funcionário.



Um atributo diferente neste caso não faz diferença significativa entre o funcionário casado e solteiro a partir da visão do negócio.

Dicas

Olhe para o modelo de dados cuidadosamente. Quando houver dúvida, pergunte ao usuário a intenção dos subtipos separados. O analista de dados cria o modelo de dados representando sua visão do mundo do usuário. Na prática, depende da visão do usuário/regras de negócio se estas entidades subtipo são importantes para o usuário e devem ser consideradas como RLRs.

Se existem transações separadas para incluir/alterar atributos únicos para estas entidades subtipo, isto seria uma indicação de que *deveríamos* ter RLRs separados para estes subtipos de entidades.

Outras Situações

Se você não tem um modelo de dados, procure grupos repetitivos de dados. Você pode encontrar algumas das seguintes situações. Aqui estão algumas dicas adicionais para contagem.

Grupos /Dados Repetitivos

Grupos repetitivos são múltiplas ocorrências dos mesmos dados, que podem ser repetidos diversas vezes dentro de um arquivo lógico.

Situação 1

Grupos repetitivos contados como RLR

O grupo de dados de Pedido consiste em Cabeçalho do Pedido e pode ter várias ocorrências de Item de Pedido. Item de Pedido contém mais do que um atributo único. Cabeçalho do Pedido e Item de Pedido representam dois subgrupos separados. Nós podemos contar dois RLRs para o arquivo lógico Pedido.

Situação 2

Dados repetitivos não contados como RLR

Um campo repetitivo (DER) não resultaria em um subgrupo separado ou RLR. Por exemplo, um Funcionário deve ter diversos números de contas de bancos. Isto *não* implicaria em dois RLRs para Funcionário (“todos os dados sem numero da conta do banco” e “números das contas dos bancos”).

Observação

Na dúvida, *não* conte um subgrupo de informações como um RLR.

A Identificação do número correto de RLRs não influencia o *número* de arquivos lógicos identificados, influencia apenas na complexidade do arquivo lógico. Embora este passo influencie no tamanho funcional, esta influência é em nível inferior a dos arquivos lógicos no Passo 1: “Identificar Arquivos Lógicos”.

Considerando Tipo de Dados Elementares e Tipo de Registros Elementares em Conjunto com Arquivos Lógicos via (In)Dependência de Entidades

Agora que os Tipos de Dados Elementares e Tipos de Registros Elementares foram discutidos, a tabela mostrada na Seção “Resumo: de Entidades para Arquivos Lógicos via (In)Dependência de Entidades” é expandida, incluindo DERs e RLRs.

Tipo de Relacionamento entre duas entidades, A e B	Quando esta Condição Existe	Então conte como Arquivos Lógicos com RLRs e DERs como abaixo:
(1) : (N)	(A e B são independentes)	2 ALs, 1 RLR e DERs para cada
1 : N	Se B é entidade dependente de A	1 AL, 2 RLRs, soma de DERs
	Se B é entidade independente de A	2 ALs, 1 RLR e DERs para cada
1 : (N)	Se B é entidade dependente de A	1 AL, 2 RLRs, soma de DERs
	Se B é entidade independente de A	2 ALs, 1 RLR, e DERs para cada
(1) : N	Se A é entidade dependente de B	1 AL, 2 RLRs, soma DERs
	Se A é entidade independente de B	2 ALs, 1 RLR, e DERs para cada
(1) : (1)	(A e B são independentes)	2 ALs, 1 RLR, e DERs para cada
1 : 1	(A e B são dependentes)	1 AL, 1 RLR, soma DERs
1 : (1)	Se B é entidade dependente de A	1 AL, 1 ou 2 RLRs, soma DERs
	Se B é entidade independente de A	2 ALs, 1 RLR, e DERs para cada
(N) : (M)	(A e B são independentes)	2 ALs, 1 RLR, e DERs para cada
N : M	Se B é entidade dependente de A	1 AL, 2 RLRs, soma DERs
	Se B é entidade independente de A	2 ALs, 1 RLR, e DERs para cada
N : (M)	Se B é entidade dependente de A	1 AL, 2 RLRs, soma DERs
	Se B é entidade independente de A	2 ALs, 1 RLR, e DERs para cada

Notas

- 1 RLR e DERs para cada significa: avaliar as duas entidades por conta própria.
- Soma DERs significa: contar todos os atributos únicos, não repetidos de entidades ligadas entre si.
- Contar a chave estrangeira do lado muitos do relacionamento.
- Em algumas situações mais de duas entidades podem formar um arquivo lógico; nesse caso mais de dois (2) RLRs devem ser contados.

Legenda

AL = Arquivo lógico (ALI ou AIE)
(..) = Lado opcional do relacionamento
RLR = Tipo de Registro Elementar
DER = Tipo de Dado Elementar

Bibliografia

As fontes foram consultadas ou citadas neste capítulo.

Booch, Grady, James Rumbaugh, Ivar Jacobson. The Unified Modeling Language User Guide. Reading: Addison-Wesley, 1994. ISBN: 0-2015-7168-4.

NESMA. Definitions and Counting Guidelines for the Application of Function Point Analysis: A Practical Manual, Version 2.2. (NESMA, 2003). ISBN: 978-90-76258-17-1.

Nota: Este manual é também chamado de NESMA Counting Practices Manual. Descreve o padrão da metodologia de APF, e muitos aspectos relacionados a aplicação de APF. Pode ser usado junto com o manual do IFPUG. Para maiores informações, acesse o site da NESMA www.nesma.org.

Garmus, David, David Herron. Function Point Analysis: Measurement Practices for Successful Software Projects. Boston: Addison-Wesley Information Technology Series, 2001. ISBN: 0-201-69944-3.

Martin, James, Carma McClure. Diagramming Techniques for Analyst and Programmers. Englewood Cliffs: Prentice-Hall, Inc., 1985. ISBN: 0-132-087944.

Modern Language Association of America. MLA Handbook for Writers of Research Papers, Fifth Edition. Boston: Addison Wesley, 1999.

Reingruber, Michael C. and William W. Gregory. The Data Modeling Handbook: A Best- Practice Approach to Building Quality Data Models. Canada: John Wiley & Sons, Wiley-QED Publication, 1994. ISBN: 0-471-05290-6.

Silverman, Len, W. H. Inmon, Kent Graziano. The Data Model Resource Book: A Library of Logical Data Models and Data Warehouse Design. Boston: Addison-Wesley, Inc. Out of Print: AISN: 0-471-15364-8.

Simsion, Graeme. Data Modeling Essentials: Analysis, Design, and Innovation. Boston: International Thomson Computer Press, 1994. ISBN: 1-850-932877-3.

Schuldt, Gary. "Information Modeling for Information Systems Analysts", A workshop at AT&T Bell Laboratories. Holmdel, N.J., May, 1992.

Teorey, Toby J. Database Modeling & Design: The Fundamental Principles, Second Edition. San Francisco: Morgan Kaufmann Publishers, Inc., 1994. ISBN: 1-558-60291-1.

Esta página foi deixada em branco intencionalmente.

Dados Compartilhados

Introdução Este capítulo fornece diretrizes adicionais para auxiliar na identificação de arquivos de interface externa (AIEs) e de arquivos lógicos internos (ALIs) bem como arquivos de transação quando dois ou mais sistemas interagem (isto é, diretrizes e esclarecimentos na contagem de dados que são compartilhados entre os sistemas).

Conteúdo Este capítulo inclui as seguintes seções:

Tópico	Página
Contagem de Dados Compartilhados Entre Aplicações	3- 2
Cenários de Contagem Cenários de Contagem – Grupo 1	3-7
Cenário 1: Leitura	3-7
Cenário 2: Cópia Estática de Imagem	3-9
Cenário 3: Cópia/Carga de Imagem – Sem Processamento Adicional	3-11
Cenário 4: Cópia/Carga de Imagem de uma Tabela Física – Sem Processamento Adicional	3-13
Cenário 5: Cópia e <i>merge</i>	3-15
Cenário 6: Screen Scraping	3-17
Cenários de Contagem Cenários da Contagem – Grupo 2	3-18
Cenário 7: Atualizando o Mesmo Dado Armazenado	3-18
Cenário 8: Dados de Transação Padrão	3-20
Resumo	3-22

Contagem de Dados Compartilhados Entre Sistemas

Aplicações que Compartilham Dados

As aplicações que compartilham dados com outras aplicações:

- Referenciam ou utilizam os dados para concluir uma transação que está sendo processada dentro do sistema que está recebendo ou acessando os dados, ou
- Mantêm arquivos lógicos internos dentro do sistema que está recebendo ou acessando os dados

Métodos de Compartilhamento de Dados

Os dados compartilhados, utilizados pelos processos elementares dentro de uma aplicação para manter dados em um arquivo lógico interno ou para apresentar dados ao usuário, podem ser transferidos via:

- Telas on-line (ex. *screen scraping*)
- Acesso direto aos arquivos de dados de outros sistemas
- Arquivos transferidos
- Recuperação direta on-line real-time das informações
- Aplicações *web*

Com o intuito de analisar corretamente estas implementações, os usuários precisam considerar a intenção primária e ter um entendimento comum dos termos que representam as diversas implementações técnicas.

Intenção Primária

O conceito de intenção primária é útil na identificação de arquivos lógicos internos e arquivos de interface externa com relação à utilização dos dados na aplicação sendo analisada. A intenção primária refere-se ao papel mais significativo ou importante que a função tem a intenção de realizar. A definição de intenção primária é “intenção que é o primeiro lugar em importância”. Portanto, é importante **determinar a intenção primária** na discussão de cada cenário. A implementação física não afeta a intenção primária e deste modo não deve influenciar a análise.

Definição & Intenção Primária (ALI/AIE)

Arquivo Lógico Interno:

Um Arquivo Lógico Interno (ALI) é um grupo de dados ou informações de controle logicamente relacionados, identificável pelo usuário, mantido dentro da fronteira da aplicação. A intenção primária de um ALI é armazenar dados mantidos através de um ou mais processos elementares da aplicação sendo contada.

Nota: O termo mantido é a capacidade de modificar dados através de um processo elementar. Exemplos incluem, mas não se limitam a, incluir, alterar, excluir, popular, revisar (corrigir), atualizar, assinalar e criar.

Arquivo de Interface Externa:

Um Arquivo de Interface Externa (AIE) é um grupo de dados logicamente relacionados ou informação de controle, reconhecido pelo usuário, referenciado pela aplicação sendo medida, mas que é mantido dentro da fronteira de outra aplicação. A intenção primária de um AIE é armazenar dados referenciados por um ou mais processos elementares dentro da fronteira da aplicação medida. Isto significa que um AIE contado por uma aplicação deve ser um ALI em outra aplicação.

Termos Comuns

Os seguintes termos comuns são utilizados neste documento para descrever técnicas de implementações físicas:

Termo	Utilizado no Documento
Cópia	Definição IEEE: (1) Ler os dados de uma origem, deixando a fonte de dados inalterada, e gravar o mesmo dado em outro lugar em uma forma física que pode ser diferente daquela utilizada na fonte. Por exemplo, copiar dados de um disco magnético para uma fita magnética. (2) O resultado de um processo de cópia como o descrito acima. Por exemplo, uma cópia de um arquivo de dados.
Arquivo	Definição IEEE: “...grupo de registros relacionados tratados como uma unidade. Por exemplo, um arquivo pode consistir de um grupo de registros de fatura.”
Imagem	Uma replicação exata de outro objeto, arquivo ou tabela normalmente criada através de um utilitário.
Carga	Definição IEEE: “... para copiar instruções do computador ou dados de um depósito externo para um depósito interno ...”
Merge	Vários arquivos com os mesmos elementos de dados consolidados em um único arquivo.
Refresh	Processo de recriação de um grupo de dados para atualização a partir da origem.

Organização dos Cenários da Contagem

A APF muitas vezes se baseia nas descrições de desenvolvedores das características físicas de uma aplicação. Este capítulo trata estas descrições físicas que um analista de pontos de função frequentemente encontra para auxiliar na correta interpretação de muitas destas interfaces de aplicação.

Este capítulo utiliza diversos cenários como ajuda na análise de situações de dados compartilhados.

Abordagem

Este capítulo usa as seguintes abordagens na discussão de dados compartilhados:

- **Descrição** - Uma expressão de alto nível do exemplo que está sendo discutido.
- **Cenário** - É apresentado um exemplo que geralmente descreve uma atividade física ou transação a respeito de arquivos sendo transferidos entre duas aplicações; ex., compartilhamento de dados.
- **Diagrama do Cenário** - O cenário é representado graficamente, como uma ajuda para mapear uma situação ou cenário similar. A seta nos diagramas reflete a direção do fluxo de dados, não a aplicação que inicia a interface.
- **Interpretação da Contagem** - Uma interpretação da contagem para o cenário é fornecida, que inclui uma discussão do exemplo e como o mesmo deve ser contado, bem como qualquer premissa com relação à intenção primária.
- **Diagrama da Solução** - A solução é representada graficamente.
- **Resumo da Contagem** - Os dados e funções de transação aplicáveis a cada aplicação são resumidos na tabela.
- **FAQs/Variações** - Se aplicável, algumas variações comuns do cenário podem ser incluídas na sequência da discussão.

Símbolos Utilizados no Diagrama de Solução

Os seguintes símbolos são utilizados no diagrama de solução:

☑ acima de um tipo de componente indica que o componente **é contado** para a aplicação;

☒ acima de um tipo de componente indica que o componente **não deve ser contado** para o cenário

⇐ o diagrama retrata a direção do fluxo de dados, não a aplicação que inicia a interface

Convenções de Nome dos Cenários Para manter a consistência, as seguintes convenções para nomenclatura foram utilizadas em todos os cenários:

Termo	Descrição
Sistema A	Sistema origem para os dados referenciados ou de transação.
Sistema B	Sistema que recebe os dados referenciados ou de transação.
Arquivo X	Um ALI contado no sistema A.
Arquivo X (Principal X)	Um AIE contado no Sistema B que é um subgrupo de dados do Arquivo X.
Arquivo Y	Um ALI contado no Sistema B.
Arquivo Z	Um arquivo de transferência de dados. Este arquivo é gerado pelo Sistema A e lido (processado) pelo Sistema B.
Fronteiras	As aplicações A e B representam duas aplicações <u>separadas</u> , assim, representam duas fronteiras separadas.

Resumo dos Cenários Os cenários a seguir não representam uma lista completa das diversas formas que os dados compartilhados são implementados, mas fornecem diretrizes para muitas situações encontradas. A compreensão destes exemplos facilitará o entendimento de cenários adicionais que podem ser encontrados.

Os cenários focam situações onde os dados solicitados para completar os processos elementares da Aplicação B são obtidos da Aplicação A. A Aplicação B é a aplicação que está sendo contada. Os cenários são divididos em dois grupos, cada um dos quais possui diversas implementações:

GRUPO 1: A intenção primária é a Aplicação B referenciar dados mantidos pela Aplicação A. Existem duas áreas que são tratadas: funcional e não-funcional.

Funcional Por Razões Funcionais (requisitos do sistema), os sistemas compartilham dados nos seguintes cenários

Número do Cenário	Cenário	Resumo da Descrição
1	LEITURA	A Aplicação B acessa fisicamente os dados da Aplicação A. Este exemplo está atualmente documentado no CPM.

Número do Cenário	Cenário	Resumo da Descrição
2	CÓPIA ESTÁTICA DE IMAGEM	A Aplicação A gera uma imagem de um depósito de dados, que reflete o estado atual dos dados em um certo tempo e permanece dentro desta fronteira.

Não-Funcional

Por Razões Não-Funcionais (performance, segurança, etc.), a Aplicação B deve usar os dados da Aplicação A e o faz da seguinte forma:

Número do Cenário	Cenário	Resumo da Descrição
3	CÓPIA/CARGA DE IMAGEM Sem Lógica de Processamento	A Aplicação A gera uma imagem sem lógica de processamento adicional e a envia para a Aplicação B; o Sistema B carrega a cópia sem lógica de processamento adicional.
4	CÓPIA/CARGA DE IMAGEM Subgrupo de um ALI	A Aplicação A gera uma cópia exata de um subgrupo (ex. RLR) sem lógica de processamento adicional e a envia para a Aplicação B. A Aplicação B carrega o RLR sem lógica de processamento adicional.
5	CÓPIA / MERGE “Refresh”	Os dados guardados em dois sistemas são copiados e mesclados para formar um arquivo que é carregado num terceiro sistema.
6	SCREEN SCRAPING	A Aplicação B acessa telas da Aplicação A para referenciar/obter dados para uso no processamento de uma transação.

GRUPO 2

A intenção primária é que a Aplicação B mantenha seus próprios dados através dos dados mantidos pela Aplicação A.

Número do Cenário	Cenário	Descrição do Cenário
7	MANTER DEPÓSITO DE DADOS COMUM	O mesmo depósito de dados é mantido por duas aplicações diferentes. Este exemplo está atualmente documentado no CPM.
8	DADOS PADRÃO DE TRANSAÇÃO	Dados de transação são fornecidos pela aplicação de origem.

Cenários de Contagem – Grupo 1

Em cada um dos cenários de 1 a 6, a intenção primária é uma aplicação referenciar dados mantidos por uma ou mais aplicações diferentes; isto pode ser implementado das seguintes maneiras:

Cenário 1: Leitura

Descrição A Aplicação B acessa fisicamente os dados da Aplicação A para executar uma consulta.

Cenário Uma transação processada pela Aplicação B precisa de informações de um depósito de dados mantido dentro da Aplicação A. A Aplicação B é responsável pelo acesso aos dados da Aplicação A, e a Aplicação B mantém o software para este acesso.

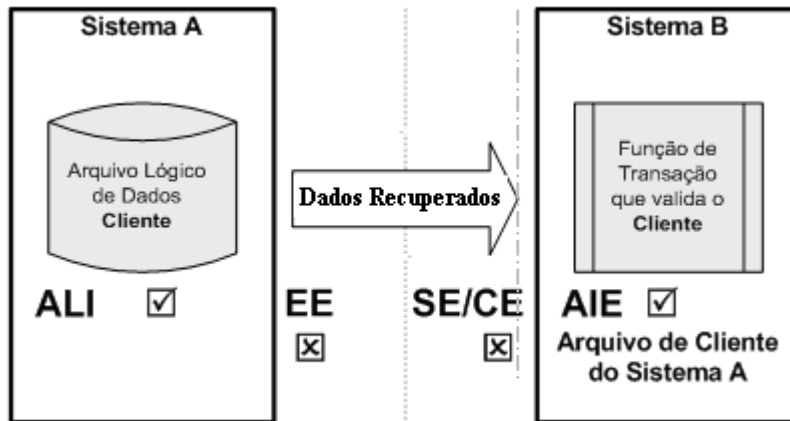
Diagrama



Interpretação de Contagem *Aplicação A:* Na perspectiva da Aplicação A, não existe requisito para enviar dados. Os dados estão disponíveis na Aplicação A. Nenhum crédito é dado para a Aplicação A para a transação executada pela Aplicação B, embora o arquivo de dados seja um ALI para a Aplicação A.

Aplicação B: Na perspectiva da Aplicação B, tanto logicamente quanto fisicamente, existe apenas um depósito de dados envolvido. A Aplicação B conta o depósito de dados que reside na Aplicação A como um AIE. A Aplicação B também conta aquele arquivo de dados como um ALR na transação.

Diagrama da Solução



Resumo da Contagem

	ALI	AIE	EE	SE/CE	Nota
Aplicação A	<input checked="" type="checkbox"/>				
Aplicação B		<input checked="" type="checkbox"/>			Cliente é também contado como um ALR na função de transação.

Cenário 2: Cópia Estática de Imagem

Descrição A Aplicação A gera uma cópia estática de um ALI, refletindo o estado atual dos dados daquele momento, e a cópia permanece em sua fronteira.

Cenário No setor bancário, as transações financeiras são conciliadas diariamente entre todas as instituições financeiras. As transações financeiras subsequentes dos clientes são validadas contra o respectivo saldo a partir desta conciliação. Para atender a este requisito do negócio, a Aplicação A periodicamente gera uma cópia estática dos dados do arquivo lógico Cliente para o Cliente principal (ou Cliente') para que outros sistemas possam referenciá-lo. Cliente' (ou Cliente principal) permanece dentro da fronteira da Aplicação A. Podem existir diferenças entre os dados atuais do Cliente e os dados do Cliente'. A Aplicação B utiliza Cliente'.

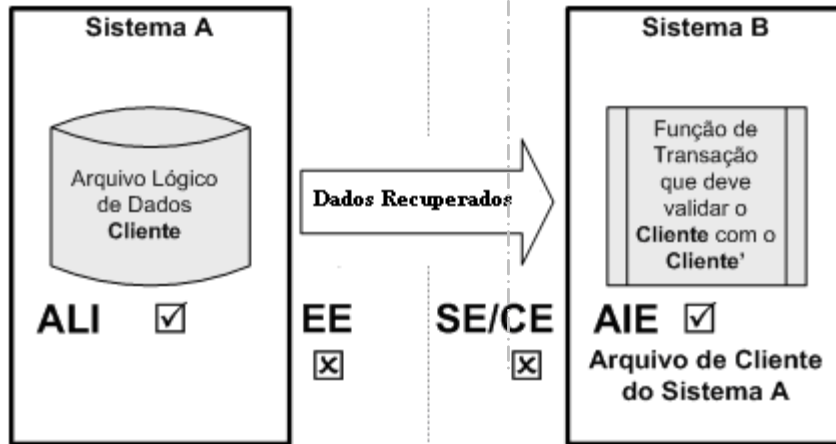
Diagrama de Solução



Interpretação da Contagem *Aplicação A:* Na perspectiva da Aplicação A, Cliente é um arquivo lógico interno para a Aplicação A. O Cliente' (principal) não é contado como um ALI separado, nem é tampouco contado como um RLR do Cliente. Cliente' é apenas uma fotografia de Cliente em um determinado tempo.

Aplicação B: Na perspectiva da Aplicação B, Cliente é um arquivo de interface externa para a Aplicação B e é também contado como um ALR para a transação da Aplicação B.

Diagrama da Solução



Resumo da Contagem

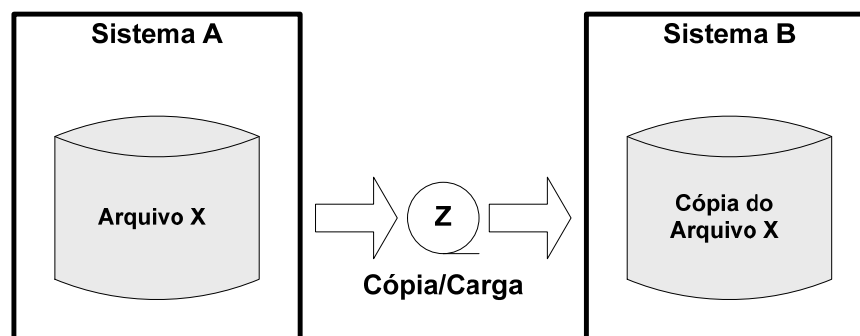
	ALI	AIE	EE	SE/CE	Nota
Aplicação A	✓				
Aplicação B		✓			Cliente é também contado como um ALR na função de transação.

Cenário 3: Cópia/Carga de Imagem - Nenhum Processamento Adicional

Descrição A Aplicação A gera uma cópia estática sem lógica de processamento adicional e a envia à Aplicação B; a Aplicação B carrega uma cópia sem lógica de processamento adicional

Cenário A Aplicação B requer a habilidade de acessar o arquivo X na Aplicação A apenas para validação e referência. A Aplicação B requer (ex. performance, etc) que a Aplicação A envie um arquivo completo para a Aplicação B. Os dados existentes armazenados na Aplicação B são atualizados a cada vez com a cópia.

Diagrama



Interpretação de Contagem Na perspectiva da Aplicação A, esta transferência de dados é uma solução técnica criada para satisfazer o requisito de negócio em que a Aplicação B deve ter acesso, com o propósito de recuperação de dados, ao Arquivo X da Aplicação A.

Logicamente os dados armazenados permanecem na Aplicação A. Neste caso, a cópia dos dados armazenados de uma aplicação para outra é a solução de um requisito não-funcional do usuário (por ex., os dados na Aplicação A não estão disponíveis quando são solicitados pela Aplicação B).

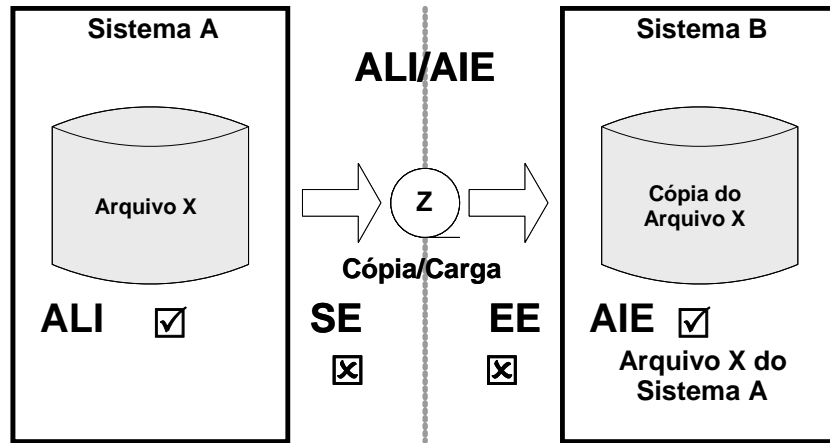
A *intenção primária* da perspectiva de B é referenciar os dados que estão logicamente em A. Uma indicação adicional é que o arquivo no Sistema B é "atualizado" a cada vez com a cópia. Da mesma forma, nenhuma lógica de processamento é executada nem na Aplicação A nem na Aplicação B.

Transações - As transações para transferência de dados: o download da aplicação A e a carga do arquivo pela Aplicação B são partes de uma solução técnica e não são contados em nenhuma das aplicações. Na prática, ao contar

a Aplicação A isoladamente, pode não ficar aparente para o Analista de Pontos de Função que esta solução exista para satisfazer um requisito não-funcional do usuário, e a mesma pode ser contada incorretamente como uma CE/SE. Nenhum dos dois sistemas conta o Arquivo Z como uma função de transação.

Arquivos - Existe apenas um arquivo lógico envolvido. A Aplicação A conta o Arquivo X como um ALI. A Aplicação B conta sua versão copiada do Arquivo X como um AIE. Nenhum dos dois sistemas conta o Arquivo Z como uma função de transação.

Diagrama da Solução



Resumo da Contagem

	ALI	AIE	EE	SE/CE	Nota
Aplicação A	<input checked="" type="checkbox"/>				
Aplicação B		<input checked="" type="checkbox"/>			

FAQs, Variações Adicionais

P? O que acontece se um arquivo lógico na Aplicação A é composto por diversas tabelas físicas e a Aplicação A fornece cópias individuais de mais de uma tabela para a Aplicação B?

R: O AIE é identificado da mesma maneira como no cenário acima; apenas os campos usados são contados como DERs.

P? O que acontece se você tem um armazenamento de dados particionado?

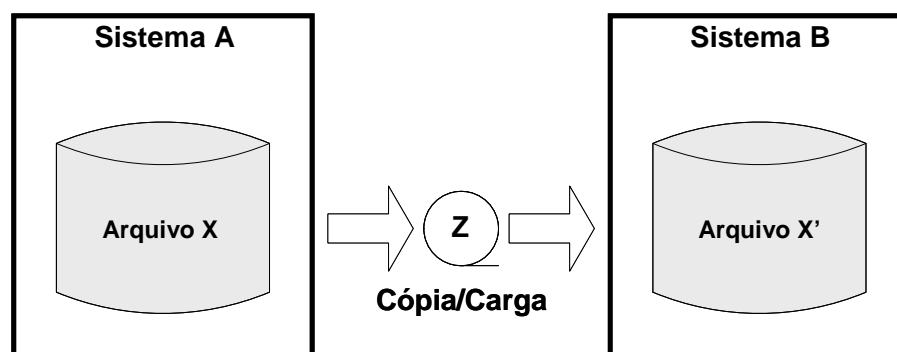
R: É particionado para melhor performance, mas ainda é logicamente um arquivo de dados; representa implementação física.

Cenário 4: Cópia/Carga de Imagem de uma Tabela Física – Nenhum Processamento Adicional

Descrição A Aplicação A gera uma cópia de uma tabela física dentro de um arquivo lógico da Aplicação A sem lógica de processamento adicional e a envia à Aplicação B. A Aplicação B carrega a tabela física sem qualquer lógica de processamento adicional.

Cenário A Aplicação B requer (p. ex., performance, etc.) a habilidade de acessar uma parte do arquivo X na Aplicação A apenas para validação e referência. A Aplicação A envia uma tabela física com o arquivo lógico para a Aplicação B. A visão existente daquela tabela física na Aplicação B é "atualizada" a cada vez com a cópia.

Diagrama



Interpretação de Contagem Uma vez que o dado é uma cópia da imagem dos dados da Aplicação A, a tabela do Arquivo X' é parte do arquivo lógico X da Aplicação A. A Aplicação B conta o Arquivo X' (com apenas os elementos de dados usados da tabela do Arquivo X') como um AIE.

Logicamente os dados armazenados permanecem na Aplicação A. Neste caso, a cópia dos dados armazenados de uma aplicação para outra é a solução de requisitos não-funcionais do usuário; p. ex., os dados na Aplicação A não estão disponíveis quando solicitados pela Aplicação B.

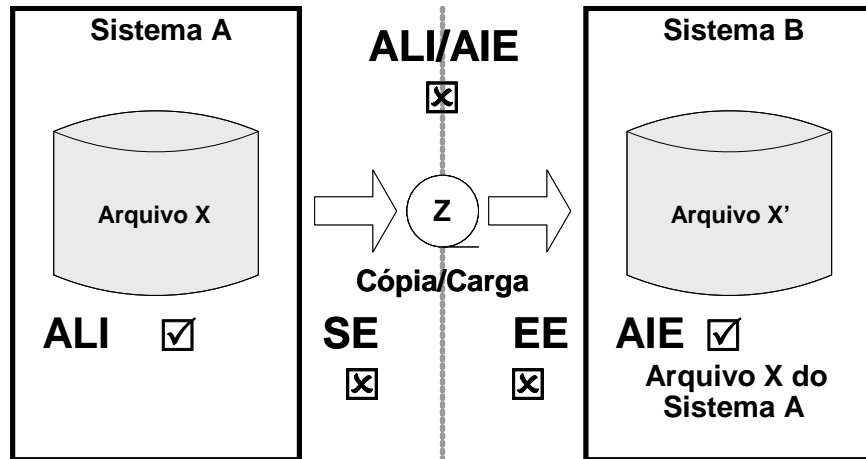
A *intenção primária* é a Aplicação B referenciar os dados que existem logicamente na Aplicação A.

Transações – Uma vez que não existem funções lógicas de transação na carga ou propagação da cópia, nenhuma transação é contada em nenhuma das aplicações para suportar a cópia e carga dos dados compartilhados. Então, a Aplicação A não conta a cópia para a Aplicação B como uma SE/CE, e a Aplicação B não a conta como uma EE. Uma indicação adicional seria que o arquivo na Aplicação B é "atualizado" a cada vez com a cópia.

Arquivos – Existe apenas um arquivo lógico envolvido. A Aplicação A conta o Arquivo X como um ALI. A Aplicação B conta a tabela copiada do

Arquivo X como um AIE.

Diagrama da Solução



Resumo da Contagem

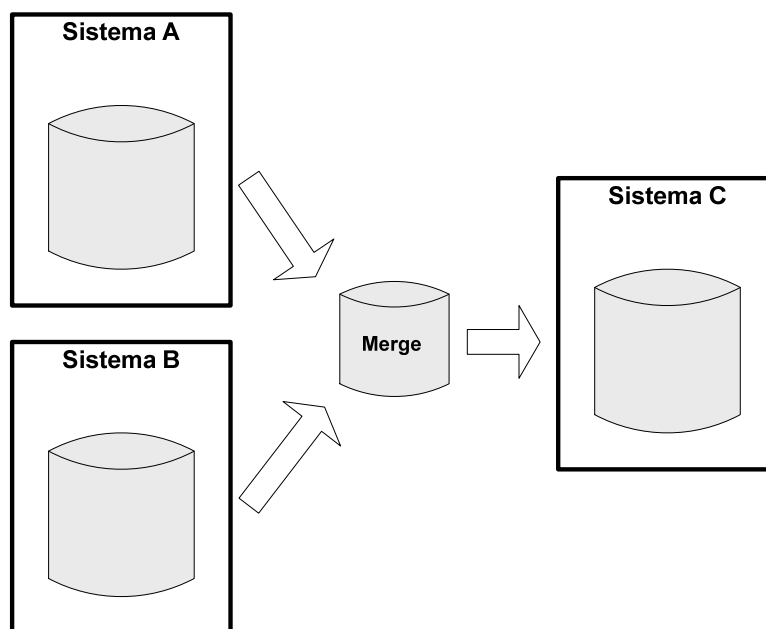
	ALI	AIE	EE	SE/CE	Nota
Aplicação A	<input checked="" type="checkbox"/>				
Aplicação B		<input checked="" type="checkbox"/>			

Cenário 5: Cópia/Merge

Descrição Dados armazenados em duas aplicações são copiados e fundidos para formar um arquivo que é carregado para dentro de uma terceira aplicação. Diversos arquivos com os mesmos elementos de dados são consolidados em apenas um arquivo.

Cenário Para evitar uma sobrecarga da Aplicação C para procurar dinamicamente os dados das Aplicações A e B, os dados são copiados da Aplicação A e da Aplicação B e fundidos em um novo depósito de dados na Aplicação C. O usuário solicitou que as informações das Aplicações A e B sejam atualizadas diariamente para validação ou apenas para referência. Utilitários de carga, *unload* e *merge* são utilizados. Não existe lógica de processamento envolvida. Isto é normalmente uma solução técnica onde duas aplicações têm diferentes instâncias dos mesmos dados lógicos requeridos por uma terceira aplicação.

Diagrama

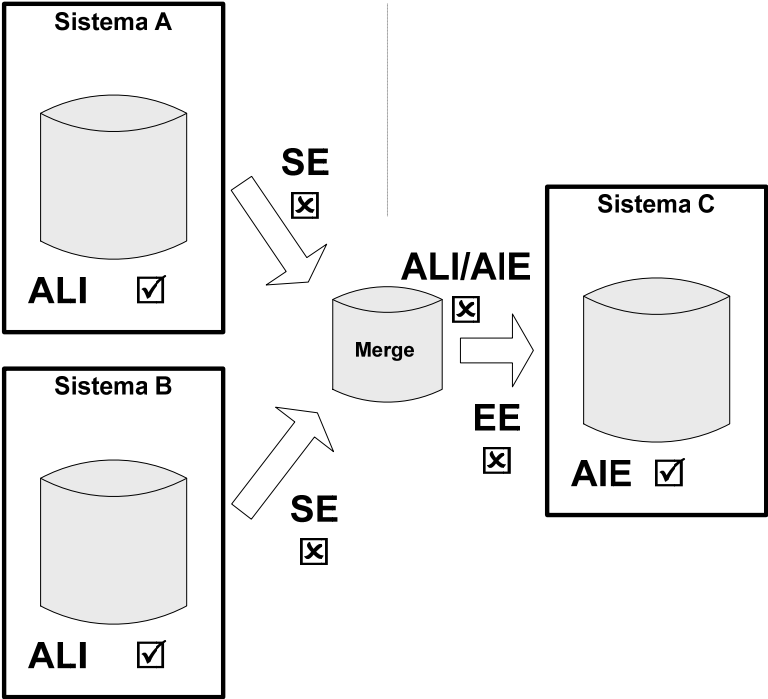


Interpretação da Contagem Logicamente, os depósitos de dados permanecem nas Aplicações A e B. A fusão de dados dentro de um único depósito de dados não é requisito suficiente para a criação um novo ALI para a Aplicação C. Desde que não exista lógica de processamento adicional, nenhuma transação é contada para nenhuma aplicação.

Os dados para a Aplicação C devem ser avaliados de acordo com o item 3.4 da Parte 1 – Medir Funções de Dados e Parte 3, Capítulo 2 – Arquivos Lógicos. Ainda que os dados venham de duas diferentes aplicações, os elementos de dados são exatamente os mesmos (veja definição para

“merge”). Dessa forma, um único arquivo lógico é identificado para a Aplicação C como um AIE.

Diagrama da Solução



Resumo da Contagem

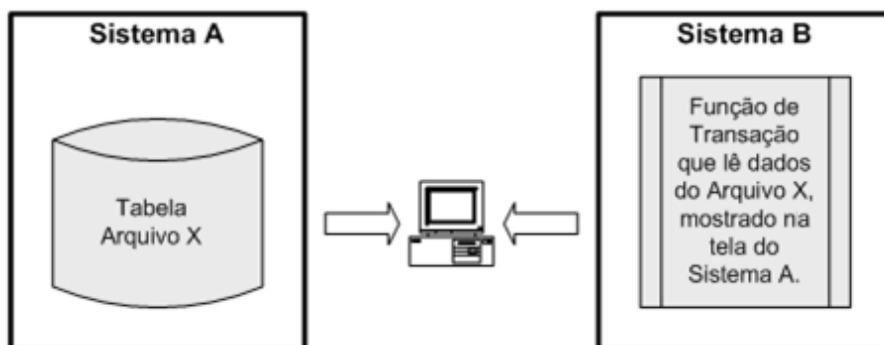
	ALI	AIE	EE	SE/CE	Nota
Aplicação A	<input checked="" type="checkbox"/>				
Aplicação B	<input checked="" type="checkbox"/>				
Aplicação C		<input checked="" type="checkbox"/>			Também contar um ALR na função de transação.

Cenário 6: *Screen Scraping*

Descrição Acesso a outras transações de telas da aplicação para referenciar/obter dados ou para atualizar aqueles dados da aplicação.

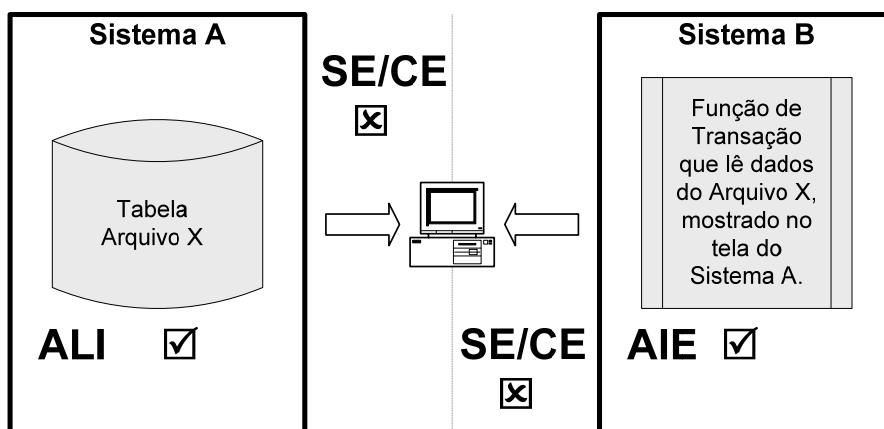
Cenário A Aplicação B "lê" o conteúdo de uma tela de consulta na Aplicação A e usa estes dados no processamento de uma função de transação.

Diagrama



Interpretação da Contagem Logicamente, a Aplicação B está lendo os dados da Aplicação A. A Aplicação A já contou os dados exibidos como um SE/CE (não contados aqui), enquanto a Aplicação B conta os dados como um AIE. Sob uma perspectiva transacional, a Aplicação A é passiva e não conta nada adicionalmente. Para a aplicação B, *screen scraping* é parte do processo elementar da transação e é contado como um ALR (AIE), uma vez que o dado foi originalmente recuperado a partir do ALI da Aplicação A.

Diagrama da Solução



Resumo da Contagem

	ALI	AIE	EE	SE/CE	Nota
Aplicação A	<input checked="" type="checkbox"/>				
Aplicação B		<input checked="" type="checkbox"/>			

Cenários da Contagem – Grupo 2

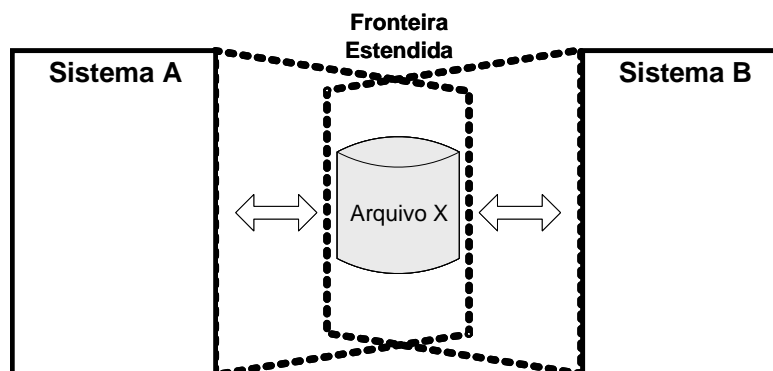
Em cada um dos cenários 7 e 8, a intenção primária é a Aplicação B manter seus próprios dados a partir de dados mantidos pela Aplicação A; isto pode ser implementado como segue:

Cenário 7: Atualizando o mesmo Depósito de Dados

Descrição O mesmo depósito de dados é mantido por duas diferentes aplicações.

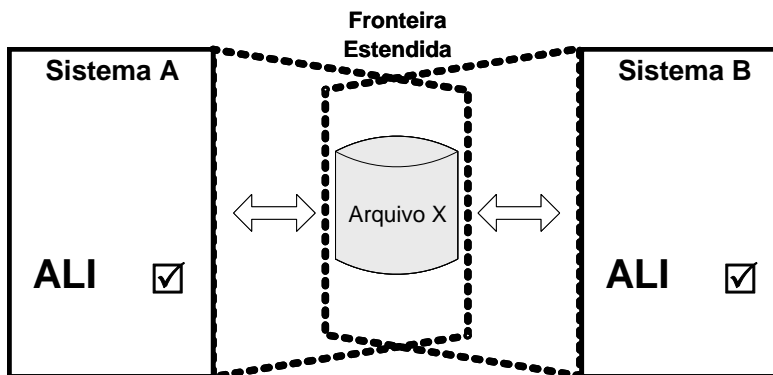
Cenário Tanto o Sistema A quanto o Sistema B mantêm o mesmo ALI. Cada um tem sua própria visão dos dados. Existem alguns elementos de dados comuns e outros são únicos para cada sistema.

Diagrama



Interpretação da Contagem Um ALI é contado para ambas as aplicações, pois cada um tem transações para mantê-lo. As Aplicações A e B mantêm dados no mesmo ALI. Cada aplicação conta apenas um RLR e DERs mantidos, utilizados ou referenciados por aquela Aplicação.

Diagrama da Solução



Resumo da Contagem

	ALI	AIE	EE	SE/CE	Nota
Aplicação A	<input checked="" type="checkbox"/>				
Aplicação B	<input checked="" type="checkbox"/>				

Cenário 8: Dados Padrão de Transação

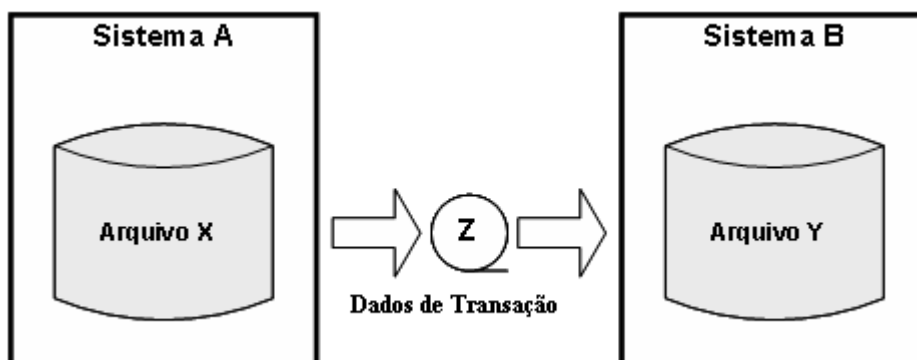
Descrição Dados de transação são fornecidos pela aplicação de origem.

Cenário A Aplicação A gera um arquivo de transação contendo modificações, o Arquivo Z, que é carregado na Aplicação B. Os registros são geralmente de mais de um tipo. A Aplicação B processa as transações de entrada de acordo com o tipo de transação dos registros do Arquivo Z, antes da atualização dos registros no Arquivo interno Y. Os DERs no Arquivo X da Aplicação A e no Arquivo X' da Aplicação B são diferentes. Por exemplo, o Arquivo X é o Catálogo Principal de Material enquanto o Arquivo Y é uma Lista de Produtos gerada localmente. O processamento inclui os seguintes tipos de transações:

- Inclusão
- Alteração
- Exclusão

Esta transferência de dados é um requisito de negócio do usuário. Tanto a Aplicação A quanto B possuem um requisito para acessar uma versão do Arquivo X, entretanto os DERs nos dois arquivos são diferentes. A Aplicação A envia apenas dados relacionados a alterações. A Aplicação B lê os registros no Arquivo Z e, baseado nos tipos de transação, inicia diferentes lógicas de processamento.

Diagrama



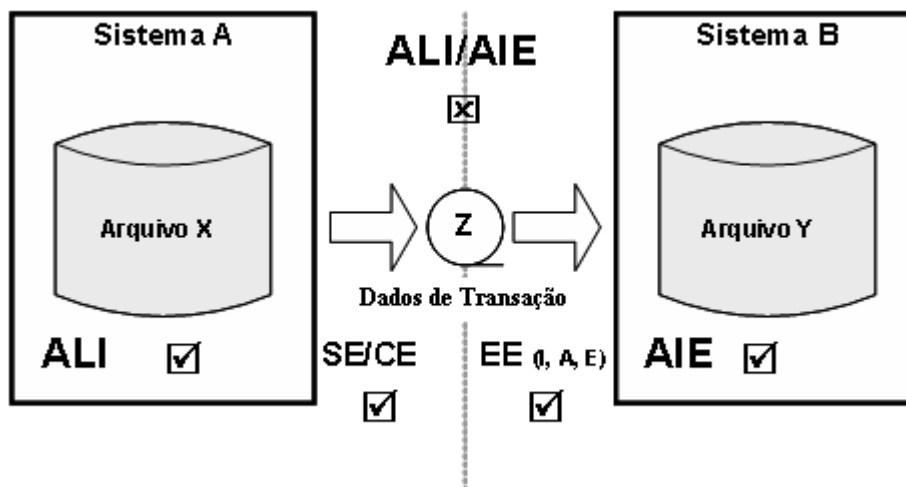
Interpretação da Contagem

Se cada registro gravado pela Aplicação A no Arquivo Z é processado da mesma forma, apenas uma CE/SE é contada. Apenas quando existirem diferentes lógicas de processamento envolvidas você poderá ter diversas funções de transação (ex., SE/CE) dentro de um único arquivo.

A Aplicação B conta EEs para cada função de manutenção única no Arquivo Y. O número de Tipos de Transações no arquivo de transação Z normalmente determina o número destas funções, mas isto não é necessariamente assim. Diferentes lógicas de processamento devem ser demonstradas.

Existem dois arquivos envolvidos. A Aplicação A conta o Arquivo X como um ALI. A Aplicação B conta o Arquivo Y como um ALI. Nenhum dos dois sistemas conta o Arquivo Z como um arquivo lógico.

Diagrama da Solução



Resumo da Contagem

	ALI	AIE	EE	SE/CE	Nota
Aplicação A	<input checked="" type="checkbox"/>			<input checked="" type="checkbox"/>	
Aplicação B	<input checked="" type="checkbox"/>		<input checked="" type="checkbox"/>		

Resumo

No decorrer deste capítulo, os cenários tiveram foco na utilização de dados dentro da aplicação que está sendo contada e a aplicação das regras de identificação de ALI e AIE relativas à ‘intenção primária’.

Este capítulo não ilustra todas as possíveis implementações de compartilhamento de dados entre aplicações. Entretanto, fornece exemplos suficientes que permitem a um Analista de Pontos de Função aplicar consistentemente as regras de identificação para ALIs e AIEs, focando na intenção primária para a utilização dos dados e fronteiras das aplicações envolvidas na contagem.

Projetos de Melhoria e Atividades de Manutenção

Introdução

Este capítulo fornece diretrizes adicionais na identificação e medição das mudanças funcionais em aplicações instaladas. Não é abordado o relacionamento entre o tamanho funcional da melhoria e o esforço necessário para implementar a melhoria. Para uma visão adicional para esse tópico, o leitor pode consultar a publicação da NESMA, “*Function Point Analysis for Software Enhancement*” [NESMA, 2001], que pode ser obtida através do site www.nesma.org.

Este capítulo também discute as diversas atividades de manutenção e suporte que podem ocorrer durante a vida útil de uma aplicação e qual medida de tamanho funcional fornece uma base útil para estimativas e reconciliação de custo.

Conteúdo

Este capítulo inclui as seguintes seções:

Tópico	Página
Medindo Projetos de Melhoria	4-2
Considerações e Dicas	4-11
Informações para Medição de Projetos de Melhoria	4-14
Procedimentos	4-15
Exemplo de Projetos de Melhoria	4-16
Considerações sobre Melhorias e Manutenções	4-22
Resumo	4-26

Medindo Projetos de Melhoria

O Tamanho Funcional do Projeto de Melhoria mede as modificações do projeto na aplicação instalada existente que adicionam, modificam ou excluem funções do usuário. Mudanças nas funcionalidades podem ocorrer a partir de novos requisitos, revisão de requisitos do usuário, mudanças legais/regulamentares ou novos usuários.

Escopo e Fronteira de um Projeto de Melhoria

O Tamanho Funcional do Projeto de Melhoria inclui todas as funções que estão sendo adicionadas, alteradas e excluídas. A(s) fronteira(s) da(s) aplicação(ões) impactada(s) permanece(m) a(s) mesma(s). As funcionalidades da(s) aplicação(ões) refletem o impacto das funções sendo adicionadas, alteradas ou excluídas.

Pode existir mais de uma aplicação incluída no escopo da contagem. Dessa forma diversas fronteiras deverão ser identificadas, resultando em um tamanho funcional do projeto de melhoria separado para cada aplicação afetada.

Se o tamanho total do projeto de melhoria é requerido, ele é calculado pela soma total das contagens de melhoria para todas as aplicações incluídas no escopo da contagem.

Medindo Funções de Dados em Projetos de Melhoria

A inclusão de novos arquivos lógicos internos ou arquivos de interface externa em um projeto de melhoria normalmente são facilmente identificados e medidos de acordo com as regras definidas na Parte 1. Capítulo 6 da Parte 2 (Medindo Funções de Dados) e Capítulo 2 da Parte 3 (Arquivos Lógicos) contem orientação adicional para medição de funções de dados bem como definições dos termos relacionados.

Porém, considerações podem ser levantadas, como a seguir:

- Se a mudança envolve apenas a inclusão de novos registros no arquivo lógico ou novos valores em um atributo existente dentro do arquivo lógico, não existe justificativa para contar a função de dado como sendo alterada.
- Se uma função de dado é alterada porque um atributo está sendo incluído e este atributo não é utilizado pela aplicação que está sendo medida, então não existe mudança naquela aplicação.

- Para que uma função de dado seja contada como uma função alterada, é obrigatório que a função seja estruturalmente alterada (ex.: inclusão ou remoção de atributos ou alteração de características de um atributo).

Nota: Um novo texto de ajuda é frequentemente adicionado à função de dados “Ajuda” que auxilia a nova transação. Uma vez que não há mudança na estrutura da função de dados “Ajuda”, não será contada como alterada.

- Se uma aplicação está solicitando o uso (para referenciar ou manter) um atributo existente que não era utilizado antes, então a função de dado relacionada é considerada alterada para aquela aplicação. Isto pode ocorrer sem que ocorra nenhuma mudança física no arquivo.
- Se novos atributos são incluídos em um ALI, procure por funções de transação novas ou modificadas que mantêm o atributo neste ALI para confirmar que a mudança ocorreu.
- Se um atributo é incluído em um ALI que é mantido por duas aplicações e se uma das aplicações mantém o novo atributo, mas a outra apenas referencia este atributo, então ambas as aplicações consideram o ALI alterado. Entretanto, a segunda aplicação não terá nenhuma função de transação nova ou alterada que mantenha este campo naquele ALI.
- Se uma aplicação não mantém nem referencia um atributo novo ou alterado, então esta aplicação não pode considerar esta função de dados como alterada.
- Se um arquivo físico é incluído por um projeto de melhoria, ele não resulta necessariamente em um novo arquivo lógico. Primeiramente precisamos determinar se o novo arquivo físico é uma mudança de um arquivo lógico existente com DERs adicionais e possivelmente um novo RLR, ou um novo arquivo lógico. Orientações adicionais em relação à medição de arquivos lógicos podem ser obtidas no Capítulo 6 da Parte 2 e no Capítulo 2 da Parte 3.

Medindo Funções de Transação em Projetos de Melhoria

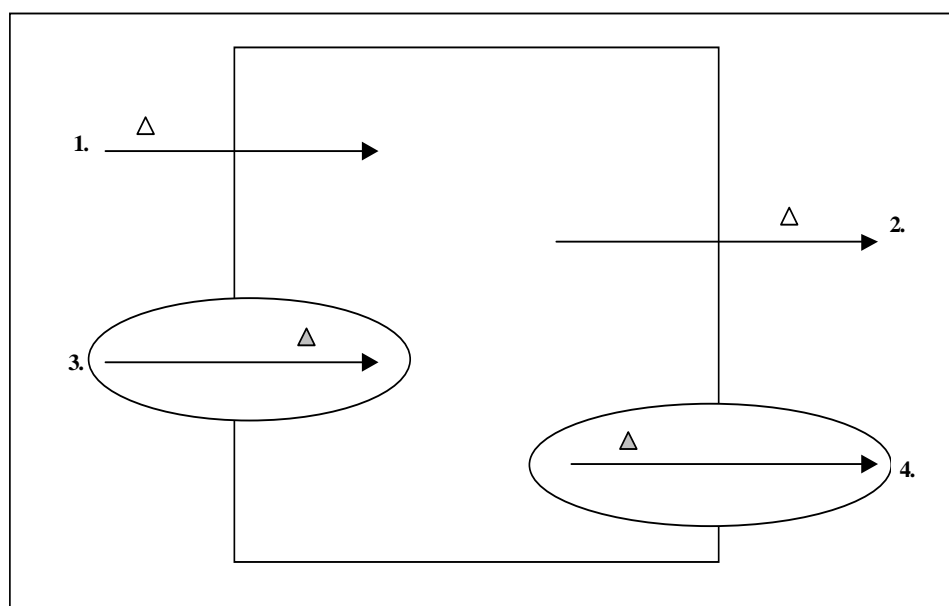
A inclusão de novas funções de transação geralmente são facilmente identificadas e medidas de acordo com as regras definidas na Parte 1. Orientações adicionais em relação à medição de funções de transação bem como definições de termos relacionados podem ser obtidas no Capítulo 7 da Parte 2 (Medindo Funções de Transação).

A identificação das funções de transação que foram alteradas pela inclusão ou exclusão de elementos de dados (DERs) é óbvia. Entretanto, não é óbvio quando os requisitos do usuário são para mudanças na lógica de processamento, como descrito na Parte 1. Quando a lógica de processamento sofreu alteração dentro da aplicação para satisfazer requisitos de negócio, o

processo elementar que incorpora aquela lógica deve ser identificado e contado como sendo alterado.

Uma simples mudança na lógica de processamento nem sempre afeta todas as transações relacionadas.

Por exemplo, quando uma alteração de edição ou validação é feita na lógica de processamento de entrada de dados e as transações existentes são de Inclusão, Alteração, Exclusão e Consulta Implícita, então apenas as transações de Inclusão e Alteração são contadas para a melhoria. A menos que exista uma alteração específica na lógica de exclusão (ex.: edição de integridade referencial) ou alteração na lógica de consulta (ex.: seleção ou recuperação), as transações de Exclusão e Consulta Implícita não serão alteradas.



No diagrama acima:

1. A transação 1 é modificada porque um DER adicional foi incluído e está cruzando a fronteira.
2. Igualmente, a transação 2 onde um DER adicional está cruzando a fronteira será contada como alterada.
3. Para a transação 3, uma rotina interna de validação da aplicação sofreu mudança. Uma vez que esta é uma mudança na lógica de processamento em um processo elementar, a transação 3 associada é contada como sendo alterada.
4. Para a transação 4 onde um critério de seleção ou um filtro sofreu mudança, a transação é contada como uma funcionalidade modificada.

Em alguns casos, uma mudança específica pode afetar como várias funções de transação são processadas. Sem considerar se as mudanças na lógica foram feitas fisicamente em uma rotina comum utilizada por várias transações, as

funções alteradas devem ser identificadas baseadas nos processos elementares que incorporam aquela lógica. Se vários processos elementares são afetados, então conte diversas funções de transação. Se apenas um único processo elementar foi afetado, conte uma função de transação como sendo alterada. Em todos os casos, os requisitos do usuário e a visão do negócio devem ser fatores determinantes.

Por exemplo, o requisito é para modificar a edição em “Novos Pedidos” para incluir uma validação para o saldo devedor do cliente. Se existirem diversas transações de “Pedido” diferentes (individual, comercial, administrativo, etc.), mas a edição alterada é apenas na transação de “Pedido Comercial”, então apenas esta transação deve ser contada como alterada. Mas se o requisito é para modificar todos os tipos de pedido, então cada transação separadamente deve ser contada como alterada.

Outra indicação de que uma função de transação deve ser contada seria a cobertura de casos de testes. Um único grupo de casos de testes indicaria que um único processo elementar foi alterado.

Freqüentemente a natureza de uma mudança é relatada pelo desenvolvedor, declarando que o único módulo que está sendo alterado é usado na produção de um grande número de relatórios ou extrações, ou no processamento de muitas transações de entrada. Todas as funções podem usar a rotina em comum, mas apenas um grupo dessas funções usa a edição que está sendo alterada nessa rotina em comum. Apenas as funções que incorporam a lógica de processamento alterada devem ser contadas como alteradas. O desafio reside na avaliação adequada do nível apropriado da mudança funcional. A ênfase deve ser nos requisitos de negócio, com o tamanho funcional do projeto de melhoria refletindo a intenção dos requisitos do usuário.

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. Estes requisitos podem incluir as seguintes ações:

1. Validações são efetuadas

Por exemplo, na inclusão de um novo funcionário em uma organização, o processo de funcionário valida o DER empregado.

- Se um requisito existe para executar uma validação diferente ou alterar a validação em uma função de transação existente, a transação, com o tamanho funcional alterado seria contada no projeto de melhoria.

2. Formulas matemáticas e cálculos são executados

Por exemplo, quando relacionar todos os funcionários de uma organização, o processo inclui o cálculo do número total de funcionários assalariados, funcionários horistas e de todos os funcionários.

- Se um requisito de negócio existe para modificar um cálculo existente (ex.: antes a fórmula era $A + B = C$ e agora será $C = A * B$), a função que inclui esse cálculo seria contada com o tamanho funcional alterado no projeto de melhoria.
- Atualmente existe uma lista de funcionários que é contada como uma CE. O requisito do projeto de melhoria determina mostrar um resumo da contagem de todos os funcionários. A transação deve ser identificada como alterada e o tipo de função deve ser alterado de uma CE para uma SE no projeto de melhoria.

3. Valores equivalentes são convertidos

Por exemplo, a idade do empregado é convertido para uma faixa etária usando uma tabela.

- Se um requisito de negócio existe para alterar uma funcionalidade para incluir a habilidade de converter o salário do empregado em uma faixa salarial, a função seria contada com o tamanho funcional alterado no projeto de melhoria.

4. Dados são filtrados e selecionados pela utilização dos critérios especificados para comparar vários grupos de dados

Por exemplo, para gerar uma lista de funcionários por sua função, um processo elementar compara o número da função ao de uma função atribuída para o funcionário, para selecionar e relacionar os funcionários assinalados a estas funções.

- Se um requisito existe para modificar o critério de seleção ou incluir um critério de seleção adicional, excluindo alterando ou incluindo valores, para uma transação existente (uma lista de funcionários agora precisa mostrar uma lista de funcionários que tenham sido nomeados para um cargo em menos de um ano), a transação, com o tamanho funcional alterado seria contada no projeto de melhoria.
- Se um requisito existe apenas para alterar o(s) valor(es) de um critério existente, como selecionar um departamento diferente ou adicionar mais um departamento na lista de departamentos, então não tem contagem para a transação.
- Se o requisito é para alterar o critério de seleção de um único departamento para uma lista de departamentos, isso seria contato com uma mudança.

- Se um requisito existe para alterar a tela de pesquisa de funcionários para adicionar um filtro no local, esse filtro adicional não cria um novo processo elementar. Essa função é contada com o tamanho funcional alterado no projeto de melhoria.
5. Condições são analisadas para determinar quais são aplicáveis
- Por exemplo, a lógica de processamento é empregada por um processo elementar quando um funcionário é incluído dependerá se um funcionário é pago baseado em seu salário ou nas horas trabalhadas. A entrada dos DERs (e o resultado do processamento lógico) baseado em uma escolha diferente (salário ou horas trabalhadas) nesse exemplo é parte de um processo elementar.
- Se um requisito existe para modificar a condição ou incluir condições adicionais em uma transação existente, a transação, com o tamanho funcional alterado seria contada no projeto de melhoria.
6. Um ou mais ALIs são atualizados
- Por exemplo, quando incluir um funcionário, o processo elementar atualiza o ALI funcionário para manter os dados do funcionário.
- Se um requisito de negócio resulta na atualização de um ALI adicional ou diferentes DERs pela transação existente, a transação, com o tamanho funcional alterado seria contada no projeto de melhoria.
7. Um ou mais ALIs ou AIEs são referenciados
- Por exemplo, ao incluir um funcionário, o AIE "moeda" é referenciado para determinar o valor da hora do funcionário com a correta taxa de conversão para dólar.
- Se um requisito de negócio referencia novos ALIs, AIEs ou DERs em uma transação existente, a transação afetada seria contada com o tamanho funcional alterado no projeto de melhoria.
8. Dados ou informações de controle são recuperados
- Por exemplo, para visualizar uma lista de empregados, as informações do empregado são recuperadas de uma função de dados.
- Se um requisito de negócio estabelece a recuperação de informações adicionais em uma transação existente, a transação afetada seria contada com o tamanho funcional alterado no projeto de melhoria.

9. Dados derivados são criados pela transformação de dados existentes para criação de dados adicionais

Por exemplo, para determinar (derivar) um número de registro do paciente (ex. SILJO01), o seguinte dado é concatenado:

- as primeiras 3 letras do último nome do paciente (ex., SIL para Silva)
- as primeiras 2 letras do primeiro nome do paciente (ex., JO para João)
- um número sequencial de dois dígitos (começando de 01)
- Se o requisito de negócio resulta na mudança em como a transação deriva os dados, a transação afetada seria contada com o tamanho funcional alterado no projeto de melhoria.

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 pagamentos toda a sexta-feira ao invés de pagamentos realizados no 15o. dia e no último dia do mês; resultando em 26 períodos de pagamento por ano ao invés de 24.

- Se o requisito de negócio resulta na alteração do comportamento do sistema (ex.: no exemplo acima, a transação é alterada para que o parâmetro data de pagamento afete apenas os funcionários horistas e não a todos os funcionários), a transação afetada seria contada com o tamanho funcional alterado no projeto de melhoria.

11. Preparar e apresentar informações para fora da fronteira

Por exemplo, uma lista de funcionários é formatada e exibida para o usuário.

- Quando o requisito de negócio resulta na apresentação de DERs adicionais para fora da fronteira, a transação afetada seria contada com o tamanho funcional alterado no projeto de melhoria.
- Mudanças em literais, formatos, cores e outros elementos da apresentação física não são considerados mudanças na lógica de processamento e portanto não fazem parte do tamanho funcional do projeto de melhoria.
- Quando o requisito de negócio é enviar um arquivo de saída existente para uma aplicação diferente ou uma nova aplicação sem alterar de nenhuma forma o processamento lógico (ex.: critério de seleção, cálculos), não afeta o tamanho funcional do projeto de melhoria.
- Rearranjar dados na tela, relatório ou arquivo exibindo elementos de dados existentes em uma nova posição não é considerado alteração na

lógica de processamento e não é contada como uma melhoria.

- Alterações nas características (ex.: tamanho, tipo, precisão, etc.) de um atributo cruzando a fronteira que deve ter uma mudança para outra forma de lógica de processamento (ex.: validações, cálculos) a ser contada.
- Quando uma tela nova ou alterada requer uma ajuda adicional ou altera uma ajuda existente, a mudança na função Help não é contada porque é apenas inclusão ou atualização de texto ou valores.

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

Por exemplo, um usuário entra com informações para adicionar um pedido do cliente para a aplicação.

- Quando o requisito de negócio resulta em diferentes DERs que entram pela fronteira, a transação afetada seria contada com o tamanho funcional alterado no projeto de melhoria.
- Quando o requisito de negócio é para aceitar a entrada de um arquivo existente de uma aplicação adicional ou diferente sem mudanças em nenhuma lógica de processamento (ex.: validações, cálculos), não há impacto no tamanho funcional do projeto de melhoria.
- Mudanças nas características (ex.: tamanho, tipo, precisão, etc.) de um atributo cruzando a fronteira deve ter mudanças para outra forma de lógica de processamento (ex.: validações, cálculos) para ser contado.

13. Classificando ou organizando um grupo de dados. Essa forma de lógica de processamento não impacta a identificação do tipo ou contribuição da singularidade do processo elementar; ou seja, a organização dos dados não constitui uma singularidade.

Por exemplo, lista de empregados é classificada tanto por ordem alfabética quanto por ordem de local.

Por exemplo, em uma tela de entrada de pedido, o cabeçalho com informações do pedido é exibido na parte de cima da tela, e os detalhes do pedido são exibidos abaixo.

Nota: Alterações na sequência de classificação normalmente são contadas. Alterações na organização por si só não são normalmente contadas.

- Quando o requisito de negócio resulta na mudança da sequência de reclassificação existente (ex.: o usuário solicita a lista de funcionários acima referenciada em ordem de localização, ao invés de em ordem alfabética), a transação afetada seria contada com o tamanho funcional alterado no projeto de melhoria.

- Em uma tela de entrada de pedidos, o usuário solicita que as informações do cabeçalho do pedido sejam colocadas à esquerda das informações do detalhe do pedido ao invés de acima. Não há contagem para essa mudança.
- O usuário solicita que o atributo Sobrenome na tela de Contratação de Empregado seja exibido a esquerda da inicial do nome do meio e do primeiro nome. Não há contagem para esse reposicionamento. Se o requisito é também para preencher os dados conforme o ultimo sobrenome digitado, então há uma mudança na lógica de processamento e a função de Contratação de Usuário é contada.
- O usuário solicita um relatório adicional com os mesmos dados (lista de funcionários) classificados pela localização. Uma nova transação não deve ser contada, mas uma mudança na função existente seria incluída no tamanho funcional do projeto de melhoria.

Um processo elementar pode incluir diversas alternativas ou ocorrências das ações acima. Por exemplo: validações, filtros, reclassificações, etc.

Considerações e Dicas

Muitos projetos de melhoria envolvem mudanças apenas na lógica de processamento sem mudanças físicas de entradas, saídas ou arquivos (ex.: atributos incluídos ou excluídos). A seguir estão itens ou questões que podem ser discutidas com os desenvolvedores durante a medição de melhoria para aplicações instaladas:

Questões a Considerar num Projeto de Melhoria

As questões listadas abaixo podem ser usadas durante as entrevistas com os desenvolvedores ou especialistas de negócio durante as sessões de medição do tamanho funcional. Respostas positivas nas questões listadas abaixo indicam a possibilidade de mudanças na funcionalidade do usuário. Investigação adicional é necessária para determinar se e como elas afetariam o tamanho funcional.

Quais funcionalidades NOVAS foram criadas aos usuários na aplicação?

- Existe algum novo arquivo permanente do usuário, base de dados, tabelas, entidades ou objetos que foram desenvolvidos/criados neste projeto?
- A aplicação agora está recebendo e processando novas transações de entrada ou arquivos de entrada que não estavam antes em produção?
- Existe alguma nova tela sendo construída para o usuário?
- Existem novas interações (interfaces) com outras aplicações?
- A aplicação está gerando novas saídas, relatórios ou arquivos para outros sistemas?
- Existe algum tipo de registro novo que está sendo incluído em um arquivo de transação existente?
- Existe algum novo processo de consulta estabelecido pelo usuário?
- Existe algum processo batch sendo incluído?

Quais funcionalidades do usuário foram ALTERADAS/MODIFICADAS para atender aos requisitos do usuário?

- Existe algum arquivo permanente, base de dados ou tabelas existentes que tenham atributos ou colunas incluídas ou excluídas? Existem características de qualquer atributos ou coluna existente sendo modificados? Não deve ser contada a inclusão de novos valores ou novas linhas em atributos existentes de tabelas existentes.
- Existe alguma lógica de processamento sendo modificada, existente em telas de entradas ou em entradas recebidas via arquivos enviados pelos

usuários ou por outras aplicações ?

- Foi efetuada alguma alteração nas atuais telas de entrada ou em arquivos de transação vindos de outra aplicação (ex.: novos atributos, mudanças nos atributos dos atributos existentes) ?
- Existe algum relatório do usuário, arquivos de saída ou telas existentes sendo modificadas pela inclusão ou exclusão de atributos?
- Existe algum relatório do usuário, arquivos de saída ou telas existentes sendo modificadas pela inclusão ou exclusão de campos?
- Para as saídas existentes, existe alguma alteração na lógica de processamento da geração destes arquivos, relatórios ou telas?
- Existe alguma tela de consulta sendo modificada?
- Foi feita alguma alteração na edição, nos critérios de seleção ou nos filtros associados às telas de consulta?
- Existe algum processo batch sendo alterado?

Quais funcionalidades do usuário foram DELETADAS/EXCLUÍDAS devido a requisitos do usuário?

- Quais funcionalidades do usuário foram excluídas da produção?
- Existe algum arquivo permanente que não é mais necessário?
- A aplicação parou de aceitar algum arquivo de entrada de outro sistema ou excluiu alguma tela antes acessada pelo usuário?
- Existe algum relatório sendo removido porque o usuário não precisa mais dele?
- Existe algum arquivo de saída para outro sistema sendo eliminado?
- Existe alguma consulta do usuário sendo excluída?
- Existe algum processo batch sendo excluído?

Quais Funcionalidades de Conversão estão sendo fornecidas?

- Existe algum processamento de dados executado uma única vez a ser criado para limpar dados, organizar ou popular novos atributos como resultado de modificações na estrutura em arquivos permanentes?
- Existe algum requisito para popular algum novo arquivo permanente?
- Existe alguma solicitação do usuário para conversão de relatórios?

Quais outras mudanças estão sendo feitas na aplicação para este projeto?

- Esta pergunta pode incentivar o especialista de negócios ou o pessoal de desenvolvimento através do fornecimento de dicas adicionais para as mudanças que podem ser contadas.

Resumo das Considerações e Dicas

Estas diretrizes auxiliam na determinação das funcionalidades entregues pelo projeto e seu impacto no *baseline* da aplicação que está sendo medida. A combinação de novas funcionalidades incluídas, o efeito das mudanças feitas nas funcionalidades existentes e as funcionalidades excluídas devem ser utilizadas para medir o tamanho do Projeto de Melhoria, assim como para atualizar o *baseline* de Pontos de Função da aplicação..

Informações para Medição de Projetos de Melhoria

Além da documentação identificada na Parte 2 Capítulo 3 Coletar Documentação Disponível, os seguintes itens também devem ser fornecidos para um projeto de melhoria:

- Documentação da Medição do Tamanho Funcional da Aplicação existente
- Requisitos do usuário de qualquer alteração/modificação da aplicação existente
- Layout das telas, relatórios e/ou diagrama de fluxo de dados das funções batch existentes para todas as funções afetadas, mostrando como elas eram ANTES do projeto de melhoria
- Layouts revisados das telas, relatórios e/ou diagrama do fluxo de dados das funções batch de todas as funções afetadas, mostrando como elas ficarão APÓS o projeto de melhoria
- Documentação de todas novas funcionalidades (novos relatórios, saídas, entradas de dados ou entidades) que serão incluídas na aplicação como parte do projeto de melhoria
- Layouts das telas, relatórios e/ou diagrama do fluxo de dados das funções batch que serão EXCLUÍDAS da aplicação como resultado do projeto de melhoria.

É reconhecido que, em muitos casos, a documentação listada acima pode não estar disponível ou não ser aplicável. A fonte de informação mais importante é o conhecimento do especialista da aplicação e a documentação dos requisitos.

Se o baseline de uma aplicação não existe, um cuidado deve ser tomado no estabelecimento da fronteira e na identificação das funções de dados e de transação.

Procedimentos

Os passos a seguir são sugeridos para medir o tamanho funcional de um projeto de melhoria; no entanto, eles podem ser executados em qualquer ordem.

Passo	Ação
1	Reúna e revise a documentação disponível.
2	Converse com o especialista de negócios para discutir as mudanças planejadas/executadas.
3	Identifique e avalie as funcionalidades incluídas.
4	Identifique e avalie as funcionalidades alteradas: <ul style="list-style-type: none">• Determine as complexidades das funções antes da alteração (a partir da documentação da última medição do tamanho funcional, ou meça como existiam antes da mudança).• Determine as complexidades das funções depois da alteração.
5	Identifique e avalie as funcionalidades excluídas.
6	Identifique e avalie qualquer funcionalidade de conversão ou de execução uma única vez, solicitada para implementação desta melhoria.

Exemplo de Projetos de Melhoria

Exemplo

Esta seção mostra um exemplo de um projeto de melhoria. Os requisitos para o projeto de melhoria incluem as seguintes mudanças:

- O usuário precisa receber um relatório adicional sobre as funções que incluem totais.
- DERs adicionais são requeridos durante a inclusão de funções no modo batch e na correção de transações suspensas. Uma referência na segurança também é incluída na transação para inclusão de função.
- A função de Atribuição de Função deve verificar que tipo de função atribuída a um funcionário se iguala com a classificação do funcionário, que é mantido no sistema de Relações do Funcionário.
- O relatório de Funcionários por tempo na função deve ser alterado. Ao invés de exibir uma contagem de funcionários acima de 12 e 24 meses, o relatório deve agora mostrar apenas os funcionários assalariados.
- O usuário não precisa mais incluir uma função on-line; portanto esta funcionalidade deve ser ou foi excluída.

Funcionalidade da Aplicação

Os seguintes parágrafos explicam as funcionalidades da aplicação medidas para o exemplo de projeto de melhoria. As funcionalidades são descritas como incluídas, alteradas ou excluídas.

Funcionalidades Incluídas

A seguinte tabela mostra a complexidade funcional para as funcionalidades incluídas medidas quando o projeto foi entregue.

Nota 1: O fornecimento de um novo relatório é uma saída externa adicional.

Nota 2: Os dados do Sistema de Relações do Funcionário (RF) referenciado pela aplicação de RH é identificado como um AIE.

Funções de Dados	RLRs	DERs	Complexidade
Arquivo de Interface Externa			
Dados do Sistema RF	1	2	Baixa

Funções de Transação	ALRs	DERs	Complexidade
Saída Externa			
Relatório da Função	1	15	Baixa

Funcionalidades Alteradas

A seguinte tabela mostra a complexidade funcional para as funcionalidades alteradas e as funções ficarão depois que o projeto de melhoria estiver pronto.

Nota 1: A complexidade para a inclusão de uma função foi aumentada porque um novo tipo de arquivo é referenciado. A complexidade para a correção de transações suspensas permanece baixa.

Nota 2: Mesmo que um Tipo de Arquivo Referenciado adicional seja contado, não existe mudança na complexidade para a inclusão da atribuição da função pois ela já é Complexa.

Nota 3: Embora exista uma mudança no critério de seleção e na lógica de soma, não existe mudança na complexidade do relatório.

Funções de Transação	ALRs	DERs	Complexidade
Entrada Externa			
Inclusão das informações sobre funções (batch)	3	8	Alta
Correção de Transação Suspensa	1	8	Baixa
Inclusão da Atribuição da Função	4	7	Alta
Saída Externa			
Tempo dos Funcionários na Atribuição	3	7	Média

Funcionalidades Excluídas

A seguinte tabela mostra a complexidade funcional para as funcionalidades excluídas identificadas no final do projeto.

Funções de Transação	ALRs	DERs	Complexidade
Saída Externa			
Inclusão das informações sobre funções (tela)	1	7	Baixa

Contribuição da Aplicação ao Tamanho Funcional

Os parágrafos a seguir explicam a contribuição da funcionalidade de aplicação para o tamanho funcional total.

Funcionalidades Adicionadas

A tabela a seguir mostra a contribuição para o tamanho funcional para as funcionalidades adicionadas identificadas no final do projeto.

Tipo da Função		Complexidade Funcional		Totais por Complexidade	Totais por Tipo de Função
AIE	1	Baixa	X 5 =	5	
		Média	X 7 =		
		Alta	X 10 =		
					5
SE	1	Baixa	X 4 =	4	
		Média	X 5 =		
		Alta	X 7 =		
					4

Funcionalidades Alteradas

A tabela a seguir mostra a contribuição para o tamanho funcional das funcionalidades alteradas como passarão a existir após o projeto de melhoria ser concluído.

Tipo da Função		Complexidade Funcional		Totais por Complexidade	Totais por Tipo de Função
EE	1	Baixa	X 3 =	3	
		Média	X 4 =		
	2	Alta	X 6 =	12	
					15
SE		Baixa	X 4 =		
	1	Média	X 5 =	5	
		Alta	X 7 =		
					5

Funcionalidades Excluídas

A tabela a seguir mostra a contribuição para o tamanho funcional das funcionalidades excluídas.

Tipo da Função	Complexidade Funcional		Totais por Complexidade	Totais por Tipo de Função
EE	1	Baixa	X 3 =	3
		Média	X 4 =	
		Alta	X 6 =	
				3

Cálculo Final

Usando a complexidade e contribuições desse exemplo, o tamanho funcional do projeto de melhoria é exibido abaixo.

$$\text{EFP} = \text{ADD} + \text{CHGA} + \text{CFP} + \text{DEL}$$

Onde

- EFP é a contagem de pontos de função do projeto de melhoria
- ADD é o tamanho das funções que estão sendo adicionadas pelo projeto de melhoria
- CHGA é o tamanho das funções sendo alteradas pelo projeto de melhoria – como elas são / serão após a implementação
- CFP é o tamanho das funcionalidades de conversão
- DEL é o tamanho das funções sendo excluídas pelo projeto de melhoria

$$\text{EFP} = 9 + 20 + 0 + 3$$

$$\text{EFP} = 32$$

Tamanho Funcional Inicial da Aplicação

O tamanho funcional inicial da aplicação é exibido abaixo.

$$AFP = ADD$$

Onde

- AFP é a contagem de pontos de função da aplicação
- ADD é o tamanho das funções que serão entregues para o usuário pelo projeto de desenvolvimento (excluindo o tamanho de qualquer funcionalidade de conversão) ou a funcionalidade que sempre existiu quando a aplicação foi medida

$$AFP = 115$$

Nota: Apenas o tamanho das funcionalidades da aplicação instaladas para o usuário são incluídas no tamanho funcional inicial da aplicação.

Tamanho Funcional da Aplicação Após a Melhoria

A medição do tamanho funcional da aplicação para refletir as melhorias é exibida abaixo.

$$AFPA = (AFPB + ADD + CHGA) - (CHGB + DEL)$$

onde

- AFPA é a contagem de pontos de função após o projeto de melhoria
- AFPB é a contagem de pontos de função antes o projeto de melhoria
- ADD é o tamanho das funções sendo adicionadas pelo projeto de melhoria
- CHGA é o tamanho das funções sendo alteradas pelo projeto de melhoria – como elas são / serão após a implementação
- CHGB é o tamanho das funções sendo alteradas pelo projeto de melhoria – como elas são / eram antes da implementação
- DEL é o tamanho das funções sendo excluídas pelo projeto de melhoria

$$AFPA = (115 + 9 + 20) - (18 + 3)$$

$$AFPA = 123$$

Algumas pessoas podem utilizar um valor de fator de ajuste (VAF), que considera as 14 características gerais do sistema (GSCs). Para orientações para utilizar do VAF e dos GSCs, consulte o Apêndice C.

Considerações sobre Melhorias e Manutenções

Uma vez que uma aplicação foi desenvolvida e instalada, ela deve ser mantida (modificada) a fim de continuar satisfazendo às constantes necessidades de mudanças do negócio e do ambiente técnico. Esta manutenção inclui um grupo de atividades que são executadas durante esta fase do ciclo de vida da aplicação, algumas delas envolvem mudanças funcionais que se aplicam a APF.

Categorias de Manutenção

O IEEE (Institute of Electrical and Electronics Engineers Inc.) define três categorias de manutenção:

Manutenção Adaptativa: Manutenção para fazer com que o software continue sendo utilizável em um ambiente alterado.

Manutenção Corretiva: Manutenção para corrigir falhas no hardware ou software.

Manutenção Perfectiva: Manutenção para melhorar a performance, facilidade de manutenção ou outros atributos do software instalado.

Enquanto este capítulo fornece dicas e diretrizes para contagem de Pontos de Função para melhorias em aplicações existentes, não existem padrões na indústria para classificação consistente de atividades que se enquadram nas categorias acima. Esta seção fornece uma estrutura de trabalho baseada na experiência comum da indústria através da qual se pode avaliar a aplicabilidade de APF no suporte a aplicações instaladas.

A ISO (International Organization for Standardization) e IEC (International Electrotechnical Commission) definem três categorias de manutenção:

Manutenção Adaptativa

A modificação de um sistema, realizada após a entrega, para manter um software utilizável em um ambiente alterado ou em alteração. Manutenção adaptativa fornece as melhorias necessárias para adaptar as modificações no ambiente em que o software deve funcionar. Essas mudanças são aquelas que devem ser realizadas para regular com o ambiente em alteração. Por exemplo, o sistema operacional deve ser atualizado e algumas alterações podem ser feitas para adaptar o novo sistema operacional. (ISO/IEC 14764:2006)

Manutenção Corretiva

A modificação reativa de um software realizada depois da entrega para corrigir problemas descobertos. A modificação corrige o software para satisfazer requisitos. (ISO/IEC 14764:2006)

Manutenção Perfectiva

Manutenção de um software após a entrega para detectar e corrigir falhas ocultas no software antes que elas se manifestem como falhas. Manutenção perfectiva fornece melhorias para o usuário, melhoria da documentação do programa, e recodificação para melhorar a performance, manutenção ou outros atributos do software. Contraste com: manutenção adaptativa; manutenção corretiva. (ISO/IEC 14764:2006)

Medição do tamanho funcional quantifica o tamanho dos requisitos de negócio. Em um ambiente de melhoria, mede os efeitos dessas mudanças nos requisitos de negócio. Portanto, medição do tamanho funcional é aplicável a um subconjunto de manutenções adaptativas. Isso inclui as funcionalidades do software adicionadas, alteradas ou excluídas bem como as funcionalidades do software fornecidas para converter dados e atender outros requisitos de conversão (ex.: relatórios de conversão).

É impraticável fornecer uma lista completa e abrangente de atividades de suporte e desenvolvimento. De certo modo, as seguintes áreas são identificadas como sugestões em relação à aplicabilidade da APF. É fortemente recomendado que cada organização desenvolva seu próprio guia com atividades, definições e terminologias específicas para a organização.

Manutenção da Aplicação e Atividades de Suporte

Uma vez que as atividades de suporte e manutenção são assuntos para relatórios de inconsistência, diretrizes desenvolvidas localmente devem abordar estas áreas. A seguir estão algumas atividades mais comuns encontradas com sugestões de tratamento relativas à APF.

Solicitações de Manutenção

Independente da duração ou nível do esforço do trabalho solicitado, este é o tipo de atividade que determina como o trabalho é classificado. A APF não deve ser utilizada para medir trabalho de manutenção perfectiva ou corretiva. A manutenção corretiva deve ser contabilizada no projeto de desenvolvimento ou melhoria que introduziu o defeito. A manutenção perfectiva não deve ser contabilizada a nenhum projeto de desenvolvimento ou de melhoria.

Pode haver uma tendência em monitorar algumas funcionalidades de melhoria como trabalho de manutenção, mas este trabalho deve ser monitorado e reportado separadamente. A justificativa mais comum para inclusão é tanto para imediatismo ou para conveniência. Organizações freqüentemente fornecem uma forma mais rápida para pequenas solicitações de melhoria, normalmente 40 horas ou menos, a fim de reduzir o esforço de gerenciamento do projeto. Quando os requisitos de negócio são afetados, a APF deve ser aplicada para a medição dos resultados.

Se uma versão contém um misto de requisitos de manutenção adaptativa, corretiva e/ou perfectiva, deve ser tomado cuidado na separação do esforço de trabalho, uma vez que as últimas duas categorias não contribuem com Pontos de Função para o negócio. Embora a separação do esforço do trabalho possa ser relativamente fácil durante a fase de construção, dependendo do nível de detalhamento no controle do esforço, é normalmente mais difícil durante a maioria das fases finais de teste. Uma possível abordagem seria um rateio da versão inteira baseada no conteúdo proporcional.

Um projeto envolvendo apenas atualização de plataforma, linguagem, ou ambiente técnico, sem alteração nas funcionalidades do usuário, não deve ser submetido a uma medição do tamanho funcional da melhoria.

Atividade	Dentro do escopo da contagem de melhoria
Correção de erros em Produção (Break/Fix)	Não
Manutenção perfectiva ou preventiva	Não
Atualização de plataforma, nova release do software do sistema	Não
Projeto tanto com melhorias e manutenções corretivas	Parcialmente

Solicitações Eventuais (Ad Hoc)

Funcionalidades que são fornecidas ao usuário final na forma de relatórios de execução única ou eventual e extração de dados, são certamente contáveis. A decisão para contar deve ser feita baseada em se as funções serão mantidas e a necessidade de negócio que o tamanho funcional vai atender. Deve ser observado que esta discussão é limitada a relatórios ou extrações produzidas pela Área de Desenvolvimento e não cobre a geração de relatórios eventuais ou extrações criadas pelo Usuário. Deve ser observado que a metodologia utilizada para produzir um relatório eventual não é normalmente tão rigorosa quanto um projeto de melhoria. Portanto, um cuidado deve ser tomado ao comparar o custo relativo de um trabalho com aquele da atividade geral de melhoria.

Atividade	Dentro do escopo da contagem de melhoria
Relatórios executados uma vez	Convenção local
Atualização de tabelas	Não
Configuração de tarefa (Job)	Não
Correção de Dados	Não
Mudanças em massa de dados	Sim - como conversão se associado ao projeto.

Suporte ao Usuário Final

Qualquer esforço de trabalho extra-projeto relacionado a atividades classificadas como “não contáveis” devem ser contabilizados a uma classificação de trabalho que não seja de Novo Desenvolvimento ou de Melhoria. Para a Estimativa Preliminar ou Estudo de Viabilidade, o problema é que os requisitos do usuário ainda não estão bem definidos. Da mesma forma, um projeto neste estágio normalmente ainda não tem verba (e pode nunca ter). Na melhor das hipóteses, uma estimativa preliminar do tamanho em Pontos de Função pode ser determinada, mas nenhuma medida quantitativa deve ser aplicada neste ponto. Qualquer resultado obtido é apenas para propósito de orçamento e planejamento.

Geralmente atividades de suporte ao usuário extra-projeto, como uma resposta à pergunta “E se...?” e ajuda a usuários, não devem ser assuntos para APF.

Atividade	Dentro do escopo da contagem de melhoria
Estimativa Preliminar e Estudo de Viabilidade	Na melhor das hipóteses, uma estimativa de ordem de grandeza
Respondendo “E se...?”	Não
Suporte Geral ao Cliente Extra-Projeto	Não
Suporte Help-desk	Parcialmente

Conclusão

Este capítulo pretende apenas para fornecer diretrizes e não é para ser interpretado como um conjunto de regras absolutas. Exceções a estes cenários podem sempre ser encontradas. Cada caso deve ser avaliado baseado nos requisitos do usuário e situação do negócio. Adicionalmente, cada organização de TI tem suas atividades exclusivas e seus processos de desenvolvimento. Estes podem ser abordados em um Guia de Pontos de Função local, com direcionamento de todos os aspectos da APF, incluindo premissas da contagem e qualquer interpretação diferenciada das regras do CPM.

Resumo

Enquanto a maioria dos recursos de Desenvolvimento de Aplicações de TI estão habitualmente dedicados ao suporte ou melhoria das aplicações existentes, a maioria dos exemplos da Parte 3 estão direcionados a novos desenvolvimento. Este capítulo fornece diretrizes para o que contar e compreensão do processo de identificação em um ambiente de melhoria.

O primeiro passo é identificar corretamente as Funções de Dados e de Transação que devem ser incluídas, alteradas ou excluídas baseadas nos requisitos do usuário. Para modificações, a mais freqüente área observada são mudanças envolvendo apenas lógica de processamento interna. A Parte 1 inclui definições de lógica de processamento que são úteis na identificação de modificações nas funções de transação que devem ser contadas. Por último, cada Função de Dados e de Transação deve ser contada de acordo com as regras da Parte 1.

Também é importante separar as diferentes manutenções e atividades de suporte e seus esforços associados conforme discutido na Seção de Considerações sobre Manutenção e Melhoria.

Atividade de Conversão de Dados

- Introdução** Esta seção aborda a funcionalidade que será avaliada quando existem requisitos para migrar ou converter os dados em conjunto com um novo desenvolvimento ou projeto de melhoria ou para trocar uma aplicação para uma plataforma diferente. Parte 4 do CPM fornece outros exemplos de funções de dados e funções transacionais para conversão de dados.
- Conteúdo** Este capítulo irá discutir o seguinte como ilustrações de diferentes cenários de conversão:

Tópico	Página
Funcionalidade de Conversão	5-2
Cenário 1: Conversão de Dados em Projetos de Melhoria	5-3
Cenário 2: Conversão de Dados com AIEs Referenciados	5-3
Cenário 3: Atribuição de Valores Padrão	5-3
O Que Não É Funcionalidade de Conversão	5-4
Resumo	5-4

Funcionalidade de Conversão

Conversão de dados da aplicação é baseada na visão do usuário dos dados. Os usuários identificam os requisitos de dados com base em necessidades distintas, tais como Emprego, Contabilidade, Clientes ou Dados de Inventário. A visão do usuário destes dados abrange todos os atributos associados com o grupo de dados, tal como definido na aplicação. Este grupo de dados reconhecível pelo usuário e os dados associados atributos tornam-se a base para um grupo lógico de dados que cumpre uma exigência específica do usuário. Este é um arquivo lógico que exige que todos os seus atributos de dados devem ser mantidos como parte do todo (ligados e não independentes).

Atributos adicionais podem ser necessários por causa de exigências de negócios novas ou alteradas. Como parte da melhoria, pode ser necessário para converter e popular os atributos de dados adicionados como parte do projeto de melhoria. A visão do processo de conversão baseia-se na aplicação original, os arquivos lógicos que estão sendo convertidos e os requisitos de dados da nova aplicação.

O processo de conversão é executado contra todos os dados como visto pelo usuário para criar um arquivo lógico atualizado que cumpre os requisitos específicos do usuário para os novos / convertido dados da aplicação.

Aplicar as regras de identificação de PE padrão para identificar a funcionalidade de conversão. O processo elementar inclui todos os relatórios de exceção, os relatórios de erros, relatórios de conversão ou relatórios de controle necessários para garantir a integridade dos dados que estão sendo convertidos. Os ALIs da aplicação nova ou alterada, são populados com os dados convertidos e os requisitos de usuário determinam o que é exigido a partir da aplicação antiga para cumprir os requisitos funcionais do usuário do projeto.

Cenário 1: Conversão de Dados em Projetos de Melhoria

O projeto envolve a integrar em uma aplicação corporativa a função Habilidade em RH que uma divisão da empresa já tinha implementado como uma aplicação stand-alone. Há um requisito para capturar uma única vez todos os dados existentes de habilidade e preencher os dados existentes atributos em um ALI em um aplicativo de RH existente. Os dados de habilidades existentes a serem importados serão contados como uma EE. Há um relatório de controle e um relatório de erro que são gerados para garantir a integridade da migração. Este processo será executado como parte da implantação da nova funcionalidade. Há um processo elementar para a carga inicial dos novos atributos de dados em um ALI do sistema de RH, incluindo os relatórios de controle e de erro. O processo de conversão será contado como uma EE, que será incluído no Tamanho Funcional do Projeto de Melhoria, mas não será adicionado ao Tamanho Funcional da Aplicação porque o processo é executado uma única vez.

Cenário 2: Conversão de Dados com AIEs Referenciados

O usuário solicitou que um ALI (ou parte de um ALI) seja populado de um ALI de outra aplicação. Nesse exemplo, foi solicitado validar os dados com um outro ALI de uma terceira aplicação. Isso é especificado como um processo que será executado uma única vez e os dados referenciados na terceira aplicação não serão utilizados no futuro.

Os atributos a serem carregados servem como uma transação de entrada para popular o ALI que está recebendo e será contada como uma EE. Os dados referenciados na terceira aplicação para validação serão contados como um AIE e um ALR adicional. O ALI que está recebendo também será considerado como um ALR. Tanto a EE quanto o AIE devem ser incluídas na medição do projeto mas não devem ser adicionados ao tamanho funcional da aplicação.

Cenário 3: Atribuição de Valores Padrão

Um projeto de melhoria solicita a inclusão de um DER em um ALI existente. O novo DER será populado com um valor padrão específico. Embora o ALI e qualquer transação modificada forem contadas como alteradas, não é contada uma funcionalidade de conversão. Nenhum dado atravessa a fronteira para estabelecer o valor padrão.

O Que Não É Funcionalidade de Conversão

Esta seção descreve vários casos que não são considerados Conversões..

- Não conte atualizações de software devido à instalação de uma versão revista de pacotes de fornecedores como funcionalidade de conversão.
- Não conte a migração de uma aplicação para uma nova plataforma como uma funcionalidade de conversão.
- Não conte a conversão de dados realizada através de um utilitário de carga existente. Nenhuma funcionalidade foi desenvolvida para realizar a conversão.
- Mesmo que um AIE para a aplicação que está sendo medida é alterado, não pode haver qualquer funcionalidade de conversão. Apenas a aplicação que tem contada a função de dados como um ALI pode contar com a funcionalidade de conversão.

Resumo

Quando um ALI é adicionado ou modificado, existe a possibilidade que um processo de conversão possa ser solicitado para popular o novo ALI ou DER(s) em um ALI existente. Parte da análise é identificar o que está atravessando a fronteira da aplicação. No caso de novos desenvolvimentos, o(s) depósito(s) de dado(s) existente(s) ou ALI(s) do(s) sistema(s) sendo substituídos é considerado como cruzando a fronteira da aplicação. Quando uma melhoria envolve alterações em ALI e uma lógica de processamento é solicitada para popular o novo atributo (ex.: validações, comparações lógicas, etc.), o ALI existente pode ser considerado como cruzando a fronteira da aplicação com uma EE. Se um novo atributo em um ALI é populado somente com um valor padrão ou nulo, a conversão não deve ser contada porque nada atravessa a fronteira da aplicação.

Índice da Parte 3

A

Aproximação, 3-4
Arquivo, 2-4
 definição, 3-3
Arquivos de Índice, 2-9
Arquivos de Sistema, 2-4
Arquivos Lógicos
 Processo para estabelecimentos de agrupamentos,
 2-2
Atribuição, 2-23
Atributos, 2-6, 2-21
Atributos de Chave, 2-23
 Chave estrangeira, 2-23
 Chave primária, 2-23
 Chave secundária, 2-23
Atributos não-chave, 2-10
Atributo Técnico, 2-9, 2-24

B

Banco de dados relacional, 2-4

C

Campos repetidos, 2-27
Carga
 definição, 3-3
Cenário 4: Cópia/Carga de Imagem de uma Tabela
 Física – Nenhum Processamento Adicional
 Cenários, 3-13
Cenários
 Cópia e Merge, 3-15
 Cópia/Carga de Imagem - Nenhum Processamento
 Adicional, 3-11

Leitura, 3-7
Cenário 4, 3-13
Screen Scraping, 3-17
Transação de Dados Padrão, 3-20
Cópia Estática de Imagem, 3-9
Atualizando o Mesmo Depósito de Dados, 3-18

Conceito

Entidade (in-)dependente, 2-12, 2-15
 relacionamento Opcional / Obrigatório, 2-12, 2-15
Convenção para Nomenclatura dos Cenários, 3-5
Cópia
 definição, 3-3

D

Dados de Código
 definição, 1-6
 exemplos, 1-7
 identificando, 1-10
 características lógicas, 1-7
 origens, 1-8
 características físicas, 1-7
 o que é, 1-10
 o que não é, 1-13
Dado Elementar, 2-4, 2-21
 Exemplo de endereço, 2-26
 Exemplo de chave estrangeira, 2-29
 Exemplo de atributos simples/múltiplos, 2-25
 Exemplo de nomes, 2-26
 Exemplo de campos repetidos, 2-27
 Exemplo de campos de status, 2-28
 Exemplo de data do sistema, 2-29

reconhecido pelo usuário, 2-25

Dados de Negócio

definição, 1-4

definições, 1-4

exemplos, 1-5

características lógicas, 1-4

características físicas, 1-5

Dados referenciados

definição, 1-5

exemplos, 1-6

características lógicas, 1-5

características físicas, 1-6

Definições

manutenção adaptativa, 4-20

Entidade Associativa, 2-3

Tipo de Entidade Associativa, 2-5

Atributos, 2-6, 2-21

Tipo de Entidade Atributiva, 2-6

dados de código, 1-6

copia, 3-3

manutenção corretiva, 4-20

Dado Elementar, 2-4, 2-21

Tipo de Elemento de Dados, 2-24

Ítem de Dado, 2-4

Entidade, 2-3

Entidade Subtipo, 2-3, 2-6

Entidade Tipo, 2-3, 2-5

arquivo, 3-3

Arquivo, 2-4

Arquivos de Sistema, 2-4

imagem, 3-3

carga, 3-3

merge, 3-3

Normalização, 2-6

manutenção perfectiva, 4-20

intenção primária, 3-2

Registro, 2-4

dados referenciados, 1-5

refresh, 3-3

Relacionamento, 2-6

Diagrama da Solução

Cenário 1, 3-8

Cenário 2, 3-10

Cenário 3, 3-12

Cenário 4, 3-14

Cenário 5, 3-16

Cenário 6, 3-17

Cenário 7, 3-19

Cenário 8, 3-21

E

Entidade, 2-3

não mantida, 2-9

Entidade Associativa, 2-3, 2-10, 2-34

contada como arquivo lógico, 2-36

contada como RLR, 2-35

não contada, 2-34

Entidade Atributiva, 2-37

obrigatória, 2-37

opcional, 2-37

Entidade de interseção, 2-10

Entidade dependente, 2-12

Entidade independente, 2-12

Entidade Key-to-key, 2-10

Entidade Subtipo, 2-6

Entidade Subtipo, 2-3

Entidade Subtipo, 2-32

Entidade Subtipo, 2-38

Entidade Subtipo

contada como RLR, 2-38

Entidade Subtipo

não contada, 2-39

Entidade Tipo, 2-3, 2-5

Exemplos

funcionalidade adicionada, 4-14

funcionalidade alterada, 4-15

funcionalidade excluída, 4-15

F

Funcional, 3-5

Funcionalidade Adicionada

exemplo, 4-14

identificação, 4-10

Funcionalidade alterada

exemplo, 4-15

identificação, 4-10

Funcionalidade de Conversão

identificação, 4-11

Funcionalidade de Conversão, 4-2

Funcionalidade Excluída

exemplo, 4-15

identificação, 4-11

G

Grupos/dados repetidos

contados como RLR, 2-40

não contados como RLR, 2-40

I

Imagem

definição, 3-3

Importância pro negócio, 2-13

Intenção Primária

definição, 3-2

Definições, 3-2

Ítem de Dado, 2-4

L

Linha, 2-4

M

Manutenção Adaptativa

definição, 4-20

Manutenção Corretiva

definição, 4-20

Manutenção perfectiva

definição, 4-20

Merge

- definição, 3-3
- Modelos físico de dados, 2-9
- N**
- Nome do Atributo, 2-22
- Normalização, 2-6
- P**
- Processo de Classificação
 - Arquivos lógicos, 2-2
- Processo de Identificação
 - Tipo de Elemento de Dados, 2-2
 - Arquivos Lógicos, 2-2
 - usando o Método de Processo Elementar, 2-9
 - usando do Método de Entidade (In-)Dependente, 2-9
 - Tipos de Registros Elementares, 2-2
- Projeto de melhoria
 - medindo funções de transação, 4-3
 - lógica de processamento, 4-5
- Projeto de Melhoria
 - Conversão de Dados, 4-3
 - medindo funções de dados, 4-2
- R**
- Registros, 2-4
- Refresh
 - definição, 3-3
- Relacionamento, 2-6
 - Obrigatório 1-1, 1-N, 2-6, 2-14
 - origem, 2-14
 - Opcional 1-(N), (1)-(N), (1)-N, 2-6, 2-14
- Requisitos de Manutenção, 4-21
- Resumo dos cenários, 3-5
- S**
- Símbolos Usados no Diagrama da Solução, 3-4
- Solicitações Eventuais (ad hoc), 4-22
- Subgrupo, 2-31
- Suporte ao usuário final, 4-23
- T**
- Tabela, 2-4
- Técnico, 3-6
- Tipo de Dado
 - negócio, 1-4
 - código, 1-6
 - referência, 1-5
- Tipos de Dados de Código
 - estático ou constante, 1-11
 - substituição, 1-11
 - valores válidos, 1-12
- Tipo de Elemento de Dados
 - regras, 2-24
- Tipo de Entidade Associativa, 2-5, 2-32
- Tipo de Entidade Atributiva, 2-6, 2-32
- Tupla, 2-4
- V**
- Visão de negócio. *Veja Visão do Usuário*
- Visão do usuário, 2-9

Página intencionalmente deixada em branco

Parte 4 – Exemplos

Página intencionalmente deixada em branco.

Parte 4 Exemplos

Introdução

A Parte 4 apresenta exemplos de funções de dados e funções de transação para ilustrar as regras de contagem da Parte 1. Os exemplos fornecem:

- Uma descrição das funções de dados ou de transação
- A base para a medição
- As tabelas para serem usadas na aplicação das regras de contagem
- A identificação da complexidade funcional
- A contribuição para o tamanho funcional

Conteúdo

A Parte 4 inclui as seguintes seções:

Tópico	Página
Exemplos de Contagem de Funções de Dados	1-3
Exemplos de Contagem de ALI	1-7
Exemplos de Contagem de AIE	1-39
Exemplos de Contagem de Funções de Transação	2-1
Exemplos de Identificação de Processo Elementar	2-7
Exemplos de Contagem de EE	2-61
Exemplos de Contagem de SE	2-103
Exemplos de Contagem de CE	2-127

Página intencionalmente deixada em branco

Exemplos de Contagem de Funções de Dados

Introdução

Esta seção utiliza vários exemplos a fim de ilustrar procedimentos para a medição de funções de dados, cada um independentemente, válido por si só.

Nota: Cada exemplo mostra somente o requisito específico para a situação ilustrada, embora na prática devêssemos avaliar todos os requisitos e seu impacto funcional.

Esta seção utiliza uma aplicação de Recursos Humanos (RH) com uma aplicação de Segurança e outra aplicação de Distribuição de Correspondência para ilustrar procedimentos para identificar e medir funções de dados. Além desta seção, os exemplos estão nos Estudos de Caso, que são parte da documentação suplementar do IFPUG.

Nota: Os exemplos desta seção e no decorrer deste manual têm dois propósitos:

1. Ilustrar como as regras de contagem de pontos de função são aplicadas para um conjunto específico de requisitos do usuário.
2. Permitir a você praticar utilizando os procedimentos de contagem.

Cada contador deve:

- Analisar os requisitos específicos do usuário que são aplicados em cada projeto ou aplicação sendo medida, e
- Contar baseado naqueles requisitos.

Conteúdo

Esta seção explica a organização dos exemplos e inclui exemplos detalhados para contagem de ALIs e AIEs.

Tópico	Página
Exemplos de Contagem de Funções de Dados	1-3
Exemplos de Contagem de ALI	1-7
Exemplos de Contagem de AIE	1-39

Organização dos Exemplos de Contagem

Esta seção explica como os exemplos são apresentados.

Sumário da Organização

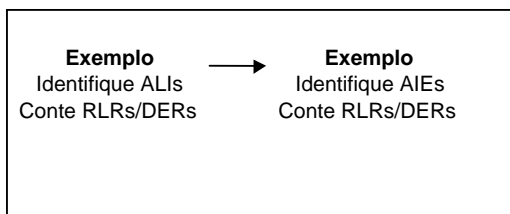
A seguinte lista sumariza a sequência da informação em exemplos detalhados.

Para cada exemplo:

1. As funções de dados são identificadas.
2. As funções de dados são classificadas como ALIs ou AIEs.
3. Os RLRs e DERs que contribuem para a complexidade funcional são identificados e contados.

Diagrama da Organização

O seguinte diagrama ilustra a organização dos exemplos.

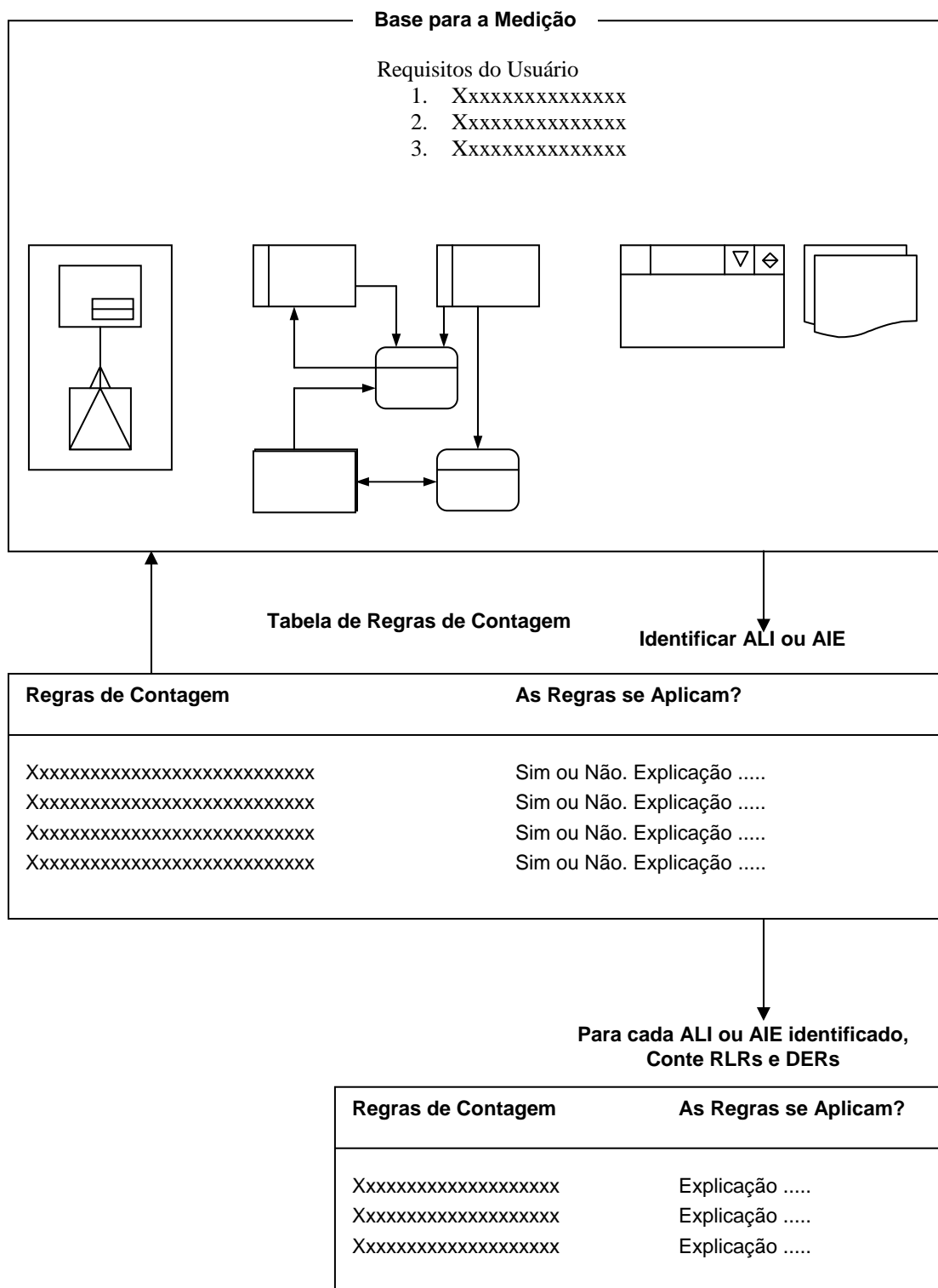


Conte para Cada Exemplo

Cada exemplo inclui os seguintes componentes:

1. Base para a medição
2. Tabela aplicando as regras de contagem

Diagrama dos Componentes O diagrama abaixo ilustra os componentes para cada exemplo e o fluxo de informação.



Base para a Medição

A base para a medição inicia cada exemplo. Como mostrado no diagrama de componentes, a medição pode ser baseada nos seguintes componentes:

- Requisitos do usuário
- Modelo de dados e de processo
- Janelas, telas ou relatórios.

Nota: Nem todos os componentes no diagrama estão incluídos em todos os exemplos. Em alguns exemplos, apenas os requisitos são base para a medição. Outros exemplos incluem um modelo de dados ou processo, janelas, telas, e relatórios.

Tabela de Regras de Contagem

A análise para identificar funções é apresentada em uma tabela que lista as regras de contagem para o tipo de função. As regras são aplicadas aos componentes que formam a base para a medição. A análise é explicada na tabela na coluna “A Regra se Aplica?”

Nota: Se todas as regras se aplicam, o exemplo é contado como um ALI ou AIE.

A próxima tabela mostra as regras e a explicação para a complexidade para cada tipo de função identificado.

Exemplos de Contagem de ALI

Introdução Esta seção usa a aplicação de Recursos Humanos (RH) para ilustrar os procedimentos para identificar e medir funções de dados. Além desta seção, outros exemplos estão nos Estudos de Caso que são parte da documentação complementar do IFPUG.

Conteúdo Esta seção inclui os seguintes exemplos:

Tópico	Página
Resumo das Descrições dos Exemplos de Contagem de ALIs	1-8
Exemplo: Dados de Auditoria para Consultas e Relatórios	1-9
Exemplo: Definição de Relatório	1-15
Exemplo: Índice Alternativo	1-20
Exemplo: Dados Compartilhados por Aplicações	1-21
Exemplo: Diferentes Usuários/Diferentes Visões dos Dados	1-30

Resumo das Descrições dos Exemplos de Contagem de ALIs

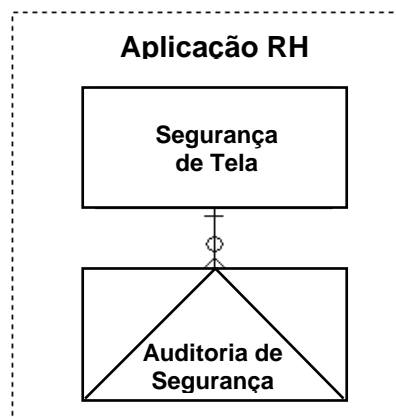
Os exemplos para ALIs são descritos na seguinte tabela:

Exemplo	Descrição Resumida	Página
Dados de Auditoria	Este exemplo mostra a análise e medição de dados que são mantidos para fins de auditoria.	1-9
Definição de Relatório	Este exemplo mostra a contagem de definições de relatórios definidos pelo usuário, mantidas dentro de uma aplicação.	1-15
Índice Alternativo	Este exemplo ilustra a análise dos requisitos do usuário para o exemplo de definição de relatório com foco nos requisitos para implementação física.	1-20
Dados Compartilhados por Aplicações	Este exemplo mostra a contagem de dados que são mantidos por mais de uma aplicação.	1-21
Diferentes Usuários/ Diferentes Visões dos Dados	Este exemplo mostra que duas aplicações podem contar o mesmo arquivo com diferentes DERs.	1-30

Exemplo: Dados de Auditoria para Consultas e Relatórios

- Requisitos do Usuário** Uma análise dos seguintes requisitos de segurança do usuário mostra uma necessidade para dados de auditoria:
1. Permitir ou recusar o acesso do usuário para cada tela da aplicação.
 2. Alterar o acesso do usuário para cada tela.
 3. Informar qualquer inclusão ou alteração de segurança de tela, utilizando os seguintes dados:
 - Identificação do usuário que está incluindo ou alterando a informação de segurança
 - O usuário da tela em que a segurança foi incluída ou alterada
 - O usuário e a imagem antes e depois de uma alteração feita na segurança da tela
 - A data e a hora que ocorreu a inclusão ou a alteração.
 4. Capturar dados de auditoria para monitorar e informar diariamente atividades da segurança. Este requisito foi determinado quando um projeto foi implementado para satisfazer os requisitos do usuário de segurança de telas.

MER



Legenda:




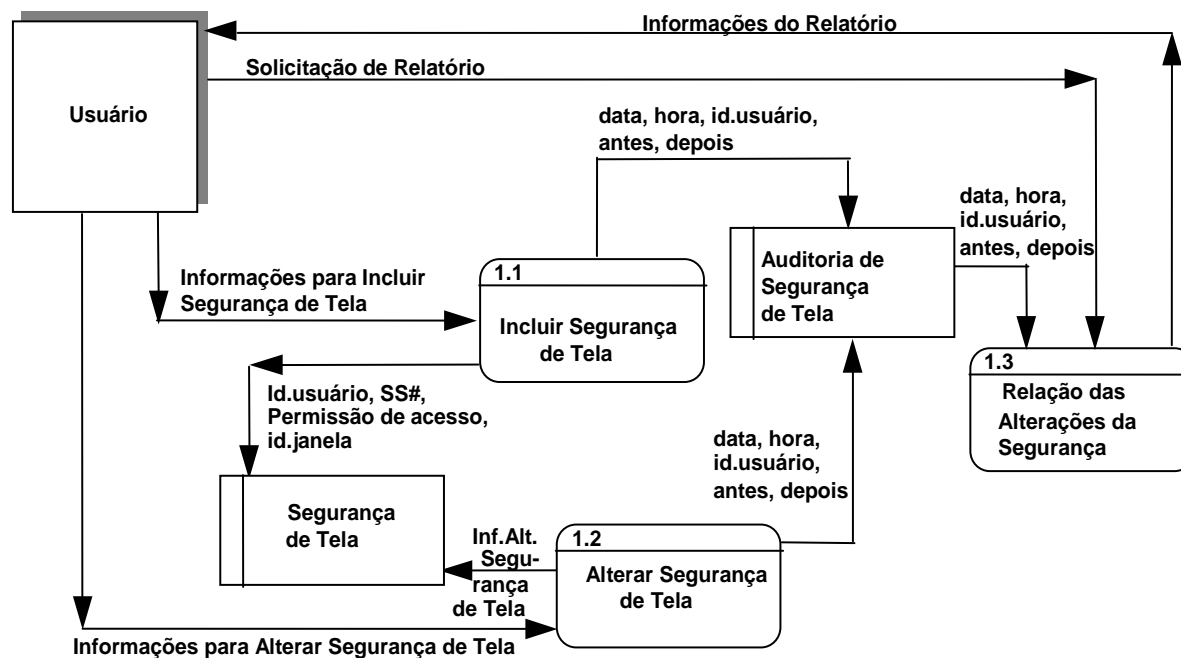
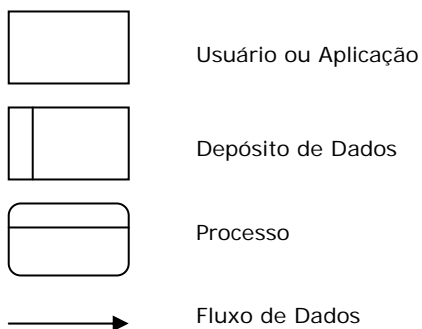
- | | |
|---|--|
|  | Entidade Tipo |
|  | Entidade Atributiva |
|  | Relacionamento Opcional Um-Para-Muitos |

Diagrama de Fluxo de Dados



Legenda:



Passo 1**Identificar as Funções de Dados**

Use as regras de Identificação de Funções de Dados para determinar se Auditoria de Segurança de Tela é uma função de dados. A tabela a seguir mostra a análise dos dados para Auditoria de Segurança de Tela

Regras de Identificação de Função de Dados	A regra se aplica?
1. Identifique todos os dados ou informações de controle logicamente relacionados e reconhecidos pelo usuário dentro do escopo da contagem.	Segurança de Tela e Auditoria de Segurança de Tela.
2. Exclua entidades que não são mantidas por qualquer aplicação.	Não existem entidades deste tipo.
3. Agrupe entidades relacionadas que são entidades dependentes.	Segurança de Tela e Auditoria de Segurança de Tela são relacionadas. Auditoria de Segurança de Tela é dependente de Segurança de Tela. Elas são agrupadas numa única função de dados.
4. Exclua as entidades referidas como Dados de Código.	Não existem entidades deste tipo.
5. Exclua entidades que não possuem atributos requeridos pelo usuário.	Não existem entidades deste tipo.
6. Remova as entidades associativas que contém atributos adicionais não requeridos pelo usuário e entidades associativas que contém somente chaves estrangeiras; agrupe os atributos chave estrangeira com as entidades primárias.	Não existem entidades deste tipo.

A Auditoria de Segurança de Tela não é contada como uma função de dados porque ela é dependente de Segurança de Tela. Auditoria de Segurança de Tela é parte da função de dados Segurança de Tela.

Passo 2 Classificar as Funções de Dados

A tabela a seguir mostra a análise para determinar se a informação de Segurança de Tela é classificada como um ALI.

Regras para Classificação da Função de Dados	A regra se aplica?
1. Classificar como um ALI se os dados são mantidos pela aplicação sendo medida	A função de dados Segurança de Tela é mantida dentro da aplicação.
2. Classificar como um AIE, se: <ul style="list-style-type: none"> • É referenciado, mas não mantido, pela aplicação sendo medida e • É identificado em um ALI em uma ou mais outras aplicações 	Classificada como um ALI; consequentemente, nenhum AIE é identificado.

Baseado na análise, a informação de Segurança de Tela é classificada como um ALI.

Passo 3 Contar os DERs

Para os DERs, observe cada atributo associado com o ALI Segurança de Tela e determine se as regras de contagem de DER se aplicam.

O ALI Segurança de Tela inclui:

- Id.usuario
- SS#
- Id.Janela
- Permissão de Acesso
- Data da Alteração
- Hora da Alteração
- Imagem antes
 - Id.usuario antes
 - Id.janela antes
 - Permissão de acesso antes
- Imagem depois
 - Id.usuario depois
 - Id.janela depois
 - Permissão de acesso depois

A análise dos DERs para o ALI Segurança de Tela é mostrada abaixo:

Regras de Contagem de DER para Função de Dados	A regra se aplica?
1. Conte um DER para cada atributo único reconhecido pelo usuário, não repetido mantido em ou recuperado de uma função de dados através da execução de todos os processos elementares dentro do escopo da contagem.	Id usuário, SS#, Id janela, Permissão de acesso, Data da Alteração e Hora da Alteração.
2. Conte somente aqueles DERs sendo usado pela aplicação sendo medida quando duas ou mais aplicações mantêm e/ou referenciam a mesma função de dados.	Não existem atributos deste tipo.
3. Conte um DER para cada atributo requerido pelo usuário para estabelecer um relacionamento com outra função de dados.	Não existem atributos deste tipo.
4. Revise os atributos relacionados para determinar se eles são agrupados e contados como um único DER ou se eles são contados como múltiplos DERs; o agrupamento dependerá de como os processos elementares usam os atributos dentro da aplicação.	Id usuário antes, Id janela antes e Permissão de acesso antes são agrupados e contados como Imagem Antes. O mesmo também é feito para os atributos Imagem Depois.

Passo 4

Contar os RLRs

Para os RLRs, identifique os subgrupos baseado nas regras de contagem de RLR.

Regras de Contagem de RLR	A regra se aplica?
1. Conte um RLR para cada função de dados (isto é, por default, cada função de dados tem um subgrupo de DERs para ser contado como um RLR).	Conte um RLR para o ALI Segurança de Tela.
2. Conte um RLR adicional para cada subgrupo lógico de DERs a seguir (dentro da função de dados) que contém mais do que um DER: <ul style="list-style-type: none"> Entidade associativa com atributos não-chave Subtipo (subtipo diferente do primeiro subtipo) e Entidade atributiva, em um relacionamento diferente de 1-1 mandatório. 	<p>Não existem entidades deste tipo.</p> <p>Não existem entidades deste tipo.</p> <p>Auditoria de Segurança de Tela é uma entidade atributiva num relacionamento 1-M opcional. Conte um RLR adicional para Auditoria de Segurança de Tela.</p>

O total de RLR e DER para Segurança de Tela é mostrado na tabela a seguir.

RLRs	DERs
<ul style="list-style-type: none"> • Segurança de Tela • Auditoria de Segurança de Tela 	<ul style="list-style-type: none"> • Id usuário • Nr. SS • Id tela • Permissão de Acesso • Data da Alteração • Hora da Alteração • Imagem Antes • Imagem Depois
Total 2 RLRs	Total 8 DERs

Passo 5 Determinar a Complexidade Funcional

2 RLRs e 8 DERs	Complexidade é Baixa
-----------------	----------------------

Passo 6 Determinar o Tamanho Funcional

Tamanho Funcional de 1 ALI Baixo	7 PF
----------------------------------	------

Exemplo: Definição de Relatório

- Requisitos do Usuário** O usuário requer a habilidade de executar as seguintes atividades:
1. Adicionar uma definição de relatório que inclui
 - Um identificador único do relatório
 - Um nome do relatório
 - Atributos utilizados no relatório
 - Cálculos para gerar o relatório.
 2. Reutilizar a definição do relatório a qualquer momento, modificando a definição se necessário.
 3. Visualizar e imprimir um relatório utilizando as definições do relatório.
 4. Consultar as definições de um relatório existente pelo nome do relatório ou pelo identificador do relatório.

Passo 1 Identificar as Funções de Dados

A partir dos requisitos do usuário, identificador do relatório, nome do relatório, atributos utilizados no relatório e cálculos, juntos, formam um agrupamento lógico de dados para uma definição de relatório porque eles são mantidos como um grupo.

A tabela a seguir mostra a análise para determinar se as informações de definição de relatório é uma função de dados. Veja os Estudos de Caso para saber como o restante dos requisitos podem ser contados.

Regras de Identificação de Função de Dados	A regra se aplica?
1. Identifique todos os dados ou informações de controle logicamente relacionados e reconhecidos pelo usuário dentro do escopo da contagem.	Definição de relatório.
2. Exclua entidades que não são mantidas por qualquer aplicação.	Não existem entidades deste tipo.
3. Agrupe entidades relacionadas que são entidades dependentes.	Não existem entidades deste tipo.
4. Exclua as entidades referidas como Dados de Código.	Informação de Definição de Relatório não é uma instância de dados de código. A entidade é usada para referência na geração de relatórios, consiste de mais do que código e descrição e é alterada sempre que necessário pelo usuário.
5. Excluir entidades que não possuem atributos requeridos pelo usuário.	Não existem entidades deste tipo.
6. Remova as entidades associativas que contém atributos adicionais não requeridos pelo usuário e entidades associativas que contém somente chaves estrangeiras; agrupe os atributos chave estrangeira com as entidades primárias.	Não existem entidades deste tipo.

Passo 2 Classificar as Funções de Dados

A tabela a seguir mostra a análise para determinar se a informação de Definição de Relatório é classificada como um ALI. Veja os Estudos de Caso para saber como o restante dos requisitos podem ser contados.

Regras para Classificação da Função de Dados	A regra se aplica?
1. Classificar como um ALI se os dados são mantidos pela aplicação sendo medida	A função de dados Definição de Relatório é mantida dentro da aplicação
2. Classificar como um AIE, se: <ul style="list-style-type: none"> • É referenciado, mas não mantido, pela aplicação sendo medida e • É identificado em um ALI em uma ou mais outras aplicações 	Classificada como um ALI; consequentemente, nenhum AIE é identificado.

Baseado na análise, a informação de Definição de Relatório é classificada como um ALI.

Passo 3 Contar os DERs

Para os DERs, observe cada atributo associado com o ALI Definição de Relatório e determine se as regras de contagem de DER se aplicam.

O ALI Definição de Relatório inclui:

- Identificador do Relatório
- Nome do Relatório
- Atributos
- Cálculos

A análise dos DERs para o ALI Definição de Relatório é mostrada abaixo:

Regra de Contagem de DER para Função de Dados	A regra se aplica?
1. Conte um DER para cada atributo único reconhecido pelo usuário, não repetido mantido em ou recuperado de uma função de dados através da execução de todos os processos elementares dentro do escopo da contagem.	Identificador do Relatório, Nome do Relatório, Atributos, Cálculos. Embora existam múltiplas ocorrências de ambos os Atributos e Cálculos, eles são contados cada um como somente um DER.
2. Conte somente aqueles DERs sendo usado pela aplicação sendo medida quando duas ou mais aplicações mantêm e/ou referenciam a mesma função de dados.	Não existem atributos deste tipo.
3. Conte um DER para cada atributo requerido pelo usuário para estabelecer um relacionamento com outra função de dados.	Não existem atributos deste tipo.
4. Revise os atributos relacionados para determinar se eles são agrupados e contados como um único DER ou se eles são contados como múltiplos DERs; o agrupamento dependerá de como os processos elementares usam os atributos dentro da aplicação.	Não existem atributos deste tipo.

Passo 4

Contar os RLRs

Para os RLRs, identifique os subgrupos baseado nas regras de contagem de RLR.

Regras de Contagem de RLR	A regra se aplica?
3. Conte um RLR para cada função de dados (isto é, por default, cada função de dados tem um subgrupo de DERs para ser contado como um RLR).	Conte um RLR para o ALI Definição de Relatório.
4. Conte um RLR adicional para cada subgrupo lógico de DERs a seguir (dentro da função de dados) que contém mais do que um DER: <ul style="list-style-type: none"> Entidade associativa com atributos não-chave Subtipo (subtipo diferente do primeiro subtipo) e Entidade atributiva, em um relacionamento diferente de 1-1 mandatório. 	<p>Não existem entidades deste tipo.</p> <p>Não existem entidades deste tipo.</p> <p>Não existem entidades deste tipo.</p>

O total de RLR e DER para Definição de Relatório é mostrado na tabela a seguir.

RLRs	DERs
<ul style="list-style-type: none"> • Grupo Definição de Relatório 	<ul style="list-style-type: none"> • Identificador do Relatório • Nome do Relatório • Atributos • Cálculos
Total 1 RLR	Total 4 DERs

Passo 5

Determinar a Complexidade Funcional

1 RLRs e 4 DERs	Complexidade é Baixa
-----------------	----------------------

Passo 6

Determinar o Tamanho Funcional

Tamanho Funcional de 1 ALI Baixo	7 PF
----------------------------------	------

Exemplo: Índice Alternativo

Requisitos do Usuário O usuário precisa consultar as definições de relatório utilizando o nome do relatório como chave para localizar a definição desejada. Para satisfazer o requisito do usuário, um índice alternativo é criado utilizando o nome do relatório como chave.

Passo 1 Identificar as Funções de Dados

A tabela a seguir mostra a análise resumida para determinar se o Índice Alternativo é um ALI.

Regras de Identificação de Função de Dados	A regra se aplica?
1. Identifique todos os dados ou informações de controle logicamente relacionados e reconhecidos pelo usuário dentro do escopo da contagem.	Não. A partir da perspectiva do usuário, esta função de filtro supre o usuário com atributos específicos das definições de relatórios criados que referenciam o ALI Definição de Relatório. Este filtro técnico, necessário para criar a lista de consulta, não se constitui em uma função de negócio.
2. Exclua entidades que não são mantidas por qualquer aplicação.	Não se aplica.
3. Agrupe entidades relacionadas que são entidades dependentes.	Não se aplica.
4. Exclua as entidades referidas como Dados de Código.	Não se aplica.
5. Exclua entidades que não possuem atributos requeridos pelo usuário.	Não se aplica.
6. Remova as entidades associativas que contém atributos adicionais não requeridos pelo usuário e entidades associativas que contém somente chaves estrangeiras; agrupe os atributos chave estrangeira com as entidades primárias.	Não se aplica.

Baseado na análise da tabela, o índice alternativo não é um grupo lógico, portanto, não é contado como um ALI.

Exemplo: Dados Compartilhados por Aplicações

Requisitos do Usuário O usuário de RH deseja a habilidade de manter informações de cada novo funcionário.

As informações que devem ser mantidas pelo usuário de RH incluem:

- Código do Funcionário
- Nome do Funcionário
- Endereço para correspondência do Funcionário
- Faixa Salarial do Funcionário
- Nome do Cargo do Funcionário

* Como resultado da criação de um novo registro de funcionário, a data elegível para aposentadoria do funcionário deve ser automaticamente calculada e salva com as outras informações do Funcionário.

O usuário da Segurança requer que um nível de segurança seja atribuído a cada novo funcionário. O departamento de Segurança faz uma discreta investigação logo que cada funcionário é contratado e atribui o apropriado nível de autorização de segurança organizacional.

Isto **não** é a aplicação de segurança que determina o acesso individual dos usuários dentro da aplicação.

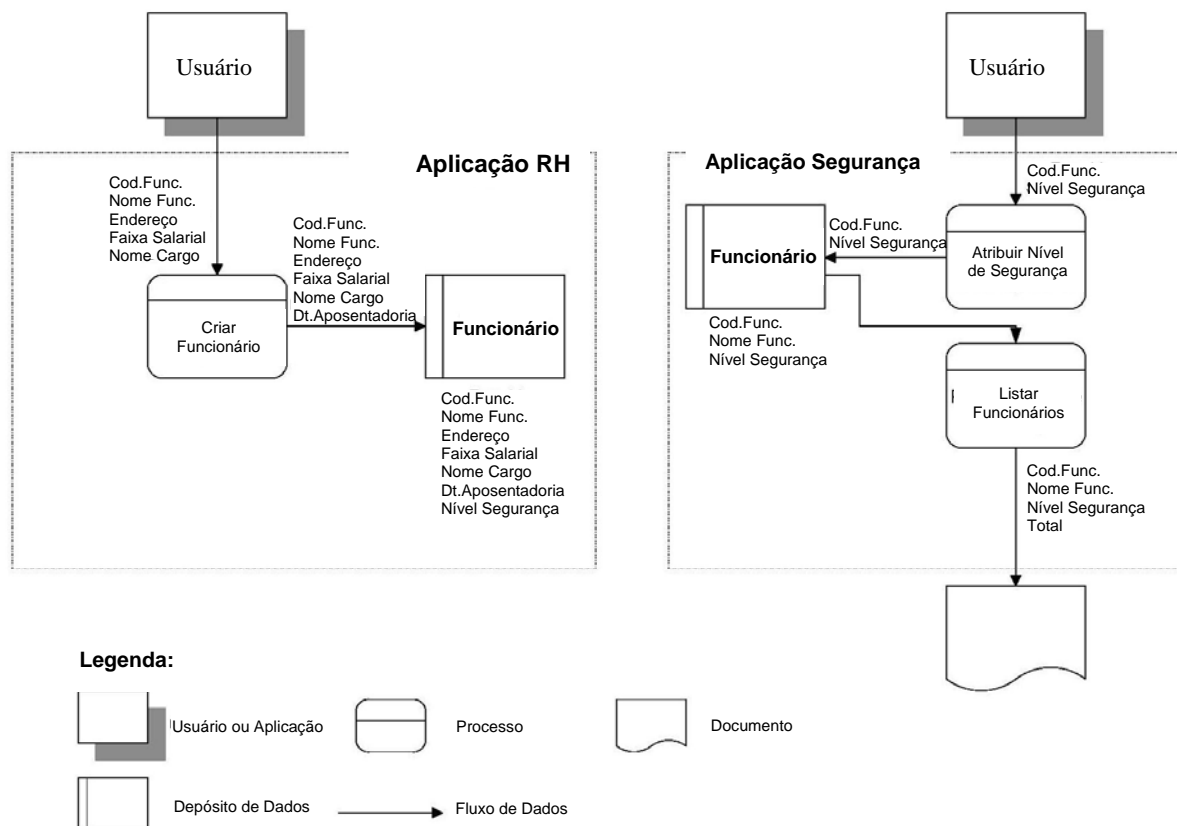
As informações que devem ser mantidas pelo usuário de Segurança incluem:

- Código do Empregado
- Nível de Autorização de Segurança Organizacional

O usuário de Segurança também requer um relatório listando as seguintes informações:

- Contagem dos Códigos do Funcionário
- Nome do Funcionário
- Código do Funcionário
- Nível de Autorização de Segurança Organizacional

Diagrama de Fluxo de Dados O diagrama a seguir mostra o fluxo de dados para este exemplo.



Passo 1**Identificar as Funções de Dados (para a aplicação de RH)**

Determine se as informações de Funcionário são uma função de dados para a aplicação de RH. A tabela a seguir mostra o resumo da análise.

Regras de Identificação de Função de Dados	A regra se aplica?
1. Identifique todos os dados ou informações de controle logicamente relacionados e reconhecidos pelo usuário dentro do escopo da contagem.	Funcionário.
2. Exclua entidades que não são mantidas por qualquer aplicação.	Não existem entidades deste tipo.
3. Agrupe entidades relacionadas que são entidades dependentes.	Não existem entidades deste tipo.
4. Exclua as entidades referidas como Dados de Código.	Não existem entidades deste tipo.
5. Exclua entidades que não possuem atributos reconhecidos pelo usuário.	Não existem entidades deste tipo.
6. Remova as entidades associativas que contém atributos adicionais não requeridos pelo usuário e entidades associativas que contém somente chaves estrangeiras; agrupe os atributos chave estrangeira com as entidades primárias.	Não existem entidades deste tipo.

Passo 2**Classificar as Funções de Dados (para a aplicação de RH)**

Determine se informações de Funcionário é classificada como um ALI para a aplicação de RH.

Regras para Classificação da Função de Dados	A regra se aplica?
1. Classificar como um ALI se os dados são mantidos pela aplicação sendo medida	A função de dados Funcionário é mantida dentro da aplicação de RH.
2. Classificar como um AIE, se: <ul style="list-style-type: none"> • É referenciado, mas não mantido, pela aplicação sendo medida e • É identificado em um ALI em uma ou mais outras aplicações 	Classificado como um ALI; consequentemente, nenhum AIEs são identificados.

A análise mostra que as informações do Funcionário são um ALI para a aplicação de RH.

Passo 3 Contar os DERs (para a aplicação de RH)

Para os DERs, observe cada atributo associado com o ALI Funcionário na aplicação de RH e determine se as regras de contagem de DER se aplicam.

A lista a seguir inclui os atributos para as informações do Funcionário:

- Código do Funcionário
- Nome do Funcionário
- Endereço de correspondência do Funcionário
- Faixa Salarial do Funcionário
- Cargo do Funcionário
- Data elegível para Aposentadoria
- Nível de Autorização de Segurança Organizacional

A análise dos DERs para o ALI Funcionário na aplicação de RH é mostrada abaixo:

Regra de Contagem de DER para Função de Dados	A regra se aplica?
1. Conte um DER para cada atributo único reconhecido pelo usuário, não repetido mantido em ou recuperado de uma função de dados através da execução de todos os processos elementares dentro do escopo da contagem.	Os seguintes atributos satisfazem esta regra: <ul style="list-style-type: none"> • Código do Funcionário • Nome do Funcionário • Endereço de Correspondência do Funcionário • Faixa Salarial do Funcionário • Cargo do Funcionário • Data elegível para aposentadoria
2. Conte somente aqueles DERs sendo usado pela aplicação sendo medida quando duas ou mais aplicações mantém e/ou referenciam a mesma função de dados.	Todos os atributos são usados dentro da aplicação de RH <i>exceto</i> o Nível de Autorização de Segurança Organizacional.
3. Conte um DER para cada atributo requerido pelo usuário para estabelecer um relacionamento com outra função de dados.	Não existem atributos deste tipo.
4. Revise os atributos relacionados para determinar se eles são agrupados e contados como um único DER ou se eles são contados como múltiplos DERs; o agrupamento dependerá de como os processos elementares usam os atributos dentro da aplicação.	Não existem atributos deste tipo.

Passo 4 Contar os RLRs (para aplicação de RH)

Para os RLRs, identifique os subgrupos baseado nas regras de contagem de RLR.

Regras de Contagem de RLR	A regra se aplica?
1. Conte um RLR para cada função de dados (isto é, por default, cada função de dados tem um subgrupo de DERs para ser contado como um RLR).	Conte um RLR para o ALI Funcionário.
2. Conte um RLR adicional para cada subgrupo lógico de DERs a seguir (dentro da função de dados) que contém mais do que um DER: <ul style="list-style-type: none"> Entidade associativa com atributos não-chave Subtipo (subtipo diferente do primeiro subtipo) e Entidade atributiva, em um relacionamento diferente de 1-1 mandatório. 	<p>Não existem entidades deste tipo.</p> <p>Não existem entidades deste tipo.</p> <p>Não existem entidades deste tipo.</p>

O total de RLR e DER para o ALI Funcionário na aplicação de RH é mostrado na tabela a seguir.

RLRs	DERs
<ul style="list-style-type: none"> Grupo de informações de Funcionário 	<ul style="list-style-type: none"> Código do Funcionário Nome do Funcionário Endereço de Correspondência do Funcionário Faixa Salarial do Funcionário Cargo do Funcionário Data elegível para aposentadoria
Total 1 RLR	Total 6 DERs

Passo 5 Determinar a Complexidade Funcional (para a aplicação de RH)

1 RLR e 6 DERs	Complexidade é Baixa
----------------	----------------------

Passo 6 Determinar o Tamanho Funcional (para a aplicação de RH)

Tamanho Funcional de 1 ALI Baixo	7 PF
----------------------------------	------

Passo 1**Identificar as Funções de Dados (para a aplicação Segurança)**

Determine se as informações de Funcionário são uma função de dados para a aplicação Segurança. A tabela a seguir mostra o resumo da análise.

Regras de Identificação de Função de Dados	A regra se aplica?
1. Identifique todos os dados ou informações de controle logicamente relacionados e reconhecidos pelo usuário dentro do escopo da contagem.	Funcionário.
2. Exclua entidades que não são mantidas por qualquer aplicação.	Não existem entidades deste tipo.
3. Agrupe entidades relacionadas que são entidades dependentes.	Não existem entidades deste tipo.
4. Exclua as entidades referidas como Dados de Código.	Não existem entidades deste tipo.
5. Exclua entidades que não possuem atributos requeridos pelo usuário.	Não existem entidades deste tipo.
6. Remova as entidades associativas que contém atributos adicionais não requeridos pelo usuário e entidades associativas que contém somente chaves estrangeiras; agrupe os atributos chave estrangeira com as entidades primárias.	Não existem entidades deste tipo.

Passo 2**Classificar as Funções de Dados (para a aplicação Segurança)**

Determine se informações de Funcionário é classificada como um ALI para a aplicação Segurança.

Regras para Classificação da Função de Dados	A regra se aplica?
1. Classificar como um ALI se os dados são mantidos pela aplicação sendo medida	A função de dados Funcionário é mantida dentro da aplicação Segurança.
2. Classificar como um AIE, se: <ul style="list-style-type: none"> • É referenciado, mas não mantido, pela aplicação sendo medida e • É identificado em um ALI em uma ou mais outras aplicações 	Classificado como um ALI; consequentemente, nenhum AIEs são identificados.

A análise mostra que as informações do Funcionário também são classificadas como um ALI para a aplicação Segurança.

Passo 3 **Contar os DERs (para a aplicação Segurança)**

Para os DERs, observe cada atributo associado com o ALI Funcionário na aplicação Segurança e determine se as regras de contagem de DER se aplicam.

O ALI Funcionário inclui:

- Código do Funcionário
- Nome do Funcionário
- Endereço de correspondência do Funcionário
- Faixa Salarial do Funcionário
- Cargo do Funcionário
- Data elegível para Aposentadoria
- Nível de Autorização de Segurança Organizacional

A análise dos DERs para o ALI Funcionário na aplicação Segurança é mostrada abaixo:

Regra de Contagem de DER para Função de Dados	A regra se aplica?
1. Conte um DER para cada atributo único reconhecido pelo usuário, não repetido mantido em ou recuperado de uma função de dados através da execução de todos os processos elementares dentro do escopo da contagem.	Os seguintes atributos satisfazem esta regra: <ul style="list-style-type: none"> • Código do Funcionário • Nome do Funcionário • Nível de Autorização de Segurança Organizacional
2. Conte somente aqueles DERs sendo usado pela aplicação sendo medida quando duas ou mais aplicações mantém e/ou referenciam a mesma função de dados.	Somente o Código do Funcionário, Nome do Funcionário e Nível de Autorização de Segurança Organizacional são usados pela aplicação Segurança.
3. Conte um DER para cada atributo requerido pelo usuário para estabelecer um relacionamento com outra função de dados.	Não existem atributos deste tipo.

4. Revise os atributos relacionados para determinar se eles são agrupados e contados como um único DER ou se eles são contados como múltiplos DERs; o agrupamento dependerá de como os processos elementares usam os atributos dentro da aplicação.	Não existem atributos deste tipo.
---	-----------------------------------

Passo 4 Contar os RLRs (para a aplicação Segurança)

Para RLRs, identifique os subgrupos baseado nas regras de contagem de RLR.

Regras de Contagem de RLR	A regra se aplica?
1. Conte um RLR para cada função de dados (isto é, por default, cada função de dados tem um subgrupo de DERs para ser contado como um RLR).	Conte um RLR para o ALI Funcionário.
2. Conte um RLR adicional para cada subgrupo lógico de DERs a seguir (dentro da função de dados) que contém mais do que um DER: <ul style="list-style-type: none"> Entidade associativa com atributos não-chave Subtipo (subtipo diferente do primeiro subtipo) e Entidade atributiva, em um relacionamento diferente de 1-1 mandatório. 	<p>Não existem entidades deste tipo.</p> <p>Não existem entidades deste tipo.</p> <p>Não existem entidades deste tipo.</p>

O total de RLR e DER para o ALI Funcionário na aplicação Segurança é mostrado na tabela a seguir.

RLRs	DERs
<ul style="list-style-type: none"> Grupo de informações de Funcionário 	<ul style="list-style-type: none"> Código do Funcionário Nome do Funcionário Nível de Autorização de Segurança Organizacional
Total 1 RLR	Total 3 DERs

Passo 5 Determinar a Complexidade Funcional (para a aplicação Segurança)

1 RLR e 3 DERs	Complexidade é Baixa
----------------	----------------------

Passo 6 Determinar o Tamanho Funcional (para a aplicação Segurança)

Tamanho Funcional de 1 ALI Baixo	7 PF
----------------------------------	------

Exemplo: Diferentes Usuários/Diferentes Visões dos Dados

Requisitos do Usuário As informações que devem ser mantidas pelo usuário do RH incluem:

- Código do Funcionário
- Nome do Funcionário
- Endereço de Correspondência do Funcionário

O Endereço de Correspondência do Funcionário mantido no depósito de dados Funcionário consiste de Andar, Código do Edifício, Rua, Cidade, Estado e CEP; entretanto, a aplicação de RH usa Endereço de Correspondência como um único atributo.

- Faixa Salarial do Funcionário
- Cargo do Funcionário
- Data elegível para aposentadoria *

* Como resultado da criação de um novo registro de funcionário, a previsão da Data da Aposentadoria do funcionário deve ser automaticamente calculada e salva com as outras informações do Funcionário.

O usuário de RH requer a habilidade de produzir etiquetas de endereço para cada funcionário.

O usuário da Distribuição de Correspondência requer a habilidade de manter o código do edifício para cada funcionário para refletir as mudanças nos códigos identificados.

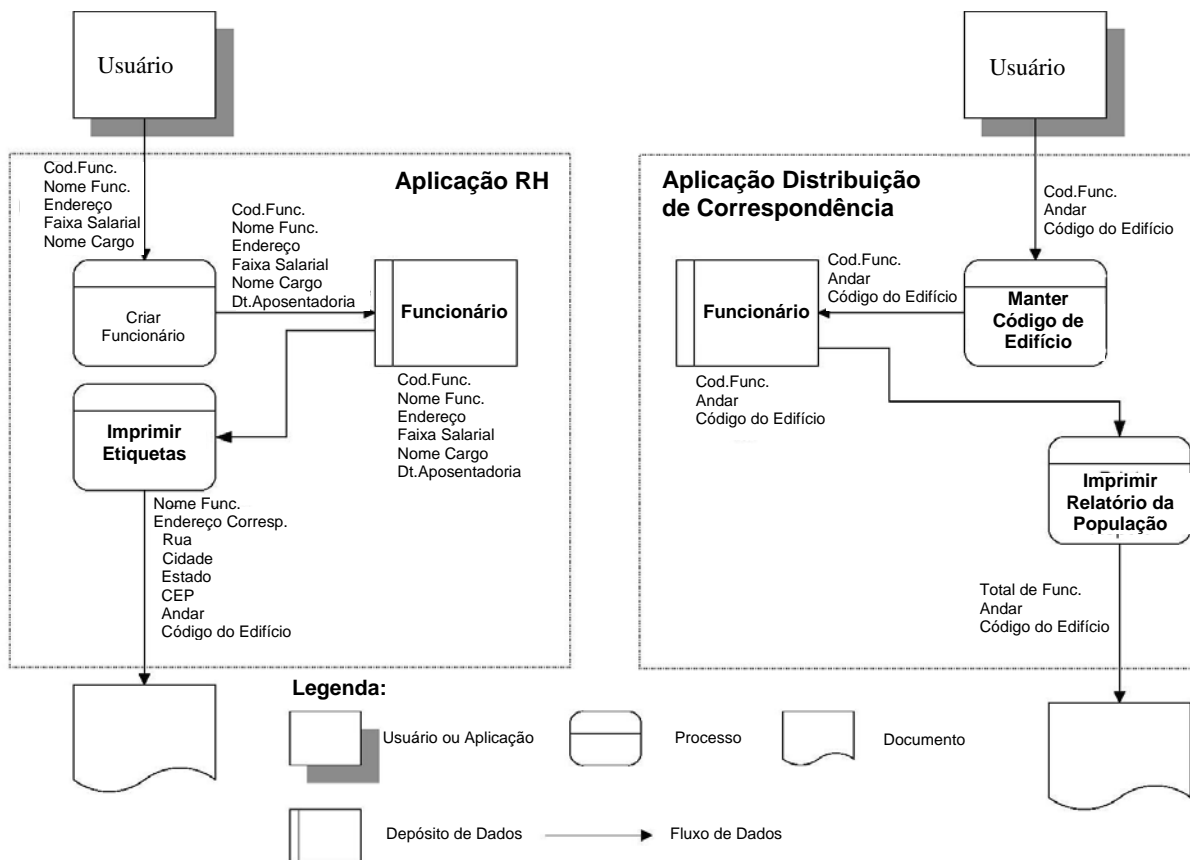
O usuário da Distribuição de Correspondência também requer a habilidade de avaliar a população em cada local para determinar qual o processo mais eficiente para entrega de correspondências internas. Um relatório é produzido com indicação do número de funcionários localizados em cada andar de cada edifício.

As informações que devem ser mantidas ou referenciadas pelo usuário da Distribuição de Correspondência incluem:

- Código do Funcionário
- Andar
- Código do Edifício

Outros atributos (por exemplo, Nível de Autorização de Segurança Organizacional) existem dentro da entidade Funcionário, mas eles não são referenciados ou mantidos pela aplicação de RH nem pela aplicação de Distribuição de Correspondência.

Diagrama de Fluxo de Dados O diagrama a seguir mostra o fluxo de dados para este exemplo.



Passo 1 Identificar as Funções de Dados (para a aplicação de RH)

Determine se as informações de Funcionário são uma função de dados para a aplicação RH. A tabela a seguir mostra o resumo da análise.

Regras de Identificação de Função de Dados	A regra se aplica?
1. Identifique todos os dados ou informações de controle logicamente relacionados e reconhecidos pelo usuário dentro do escopo da contagem.	Funcionário.
2. Exclua entidades que não são mantidas por qualquer aplicação.	Não existem entidades deste tipo.
3. Agrupe entidades relacionadas que são entidades dependentes.	Não existem entidades deste tipo.
4. Exclua as entidades referidas como Dados de Código.	Não existem entidades deste tipo.

Passo 2 Classificar as Funções de Dados (para a aplicação de RH)

Determine se informações de Funcionário é classificada como um ALI para a aplicação de RH.

Regras para Classificação da Função de Dados	A regra se aplica?
1. Classificar como um ALI se os dados são mantidos pela aplicação sendo medida	A função de dados Funcionário é mantida dentro da aplicação de RH.
2. Classificar como um AIE, se: <ul style="list-style-type: none"> • É referenciado, mas não mantido, pela aplicação sendo medida e • É identificado em um ALI em uma ou mais outras aplicações 	Classificado como um ALI; consequentemente, nenhum AIEs são identificados.

A análise mostra que as informações do Funcionário são um ALI para a aplicação de RH.

Passo 3 Contar os DERs (para a aplicação de RH)

Para os DERs, observe cada atributo associado com o ALI Funcionário na aplicação de RH e determine se as regras de contagem de DER se aplicam.

Informações do Funcionário incluem:

- Código do Funcionário
- Nome do Funcionário
- Endereço de correspondência do Funcionário (Andar, Código do Edifício, Rua, Cidade, Estado e CEP)
- Faixa Salarial do Funcionário
- Cargo do Funcionário
- Data elegível para Aposentadoria
- Nível de Autorização de Segurança Organizacional

A análise dos DERs para o ALI Funcionário na aplicação de RH é mostrada abaixo:

Regra de Contagem de DER para Função de Dados	A regra se aplica?
1. Conte um DER para cada atributo único reconhecido pelo usuário, não repetido mantido em ou recuperado de uma função de dados através da execução de todos os processos elementares dentro do escopo da contagem.	Os seguintes atributos satisfazem esta regra: <ul style="list-style-type: none"> • Código do Funcionário • Nome do Funcionário • Endereço de Correspondência do Funcionário • Faixa Salarial do Funcionário • Cargo do Funcionário • Data elegível para aposentadoria • Nível de Segurança Organizacional
2. Conte somente aqueles DERs sendo usado pela aplicação sendo medida quando duas ou mais aplicações mantêm e/ou referenciam a mesma função de dados.	Somente o Código do Funcionário, Nome do Funcionário, Endereço de Correspondência do Funcionário, Faixa Salarial do Funcionário, Cargo do Funcionário, e Data elegível para aposentadoria são usados pela aplicação de RH. O atributo Nível de Segurança Organizacional não é contado como DER, porque ele não é usado pela aplicação de RH.

3. Conte um DER para cada atributo requerido pelo usuário para estabelecer um relacionamento com outra função de dados.	Não existem atributos deste tipo.
4. Revise os atributos relacionados para determinar se eles são agrupados e contados como um único DER ou se eles são contados como múltiplos DERs; o agrupamento dependerá de como os processos elementares usam os atributos dentro da aplicação.	Endereço de correspondência do empregado é contado como um único DER.

Passo 4**Contar os RLRs (para a aplicação de RH)**

Para os RLRs, identifique os subgrupos baseado nas regras de contagem de RLR.

Regras de Contagem de RLR	A regra se aplica?
1. Conte um RLR para cada função de dados (isto é, por default, cada função de dados tem um subgrupo de DERs para ser contado como um RLR).	Conte um RLR para o ALI Funcionário.
2. Conte um RLR adicional para cada subgrupo lógico de DERs a seguir (dentro da função de dados) que contém mais do que um DER: <ul style="list-style-type: none"> Entidade associativa com atributos não-chave Subtipo (subtipo diferente do primeiro subtipo) e Entidade atributiva, em um relacionamento diferente de 1-1 mandatório. 	<p>Não existem entidades deste tipo.</p> <p>Não existem entidades deste tipo.</p> <p>Não existem entidades deste tipo.</p>

O total de RLR e DER para o ALI Funcionário na aplicação de RH é mostrado na tabela a seguir.

RLRs	DERs
<ul style="list-style-type: none"> Grupo de informações de Funcionário 	<ul style="list-style-type: none"> Código do Funcionário Nome do Funcionário Endereço de Correspondência do Funcionário Faixa Salarial do Funcionário Cargo do Funcionário Data elegível para aposentadoria
Total 1 RLR	Total 6 DERs

Passo 5 Determinar a Complexidade Funcional (para a aplicação de RH)

1 RLR e 6 DERs	Complexidade é Baixa
----------------	----------------------

Passo 6 Determinar o Tamanho Funcional (para a aplicação de RH)

Tamanho Funcional de 1 ALI Baixo	7 PF
----------------------------------	------

Passo 1 Identificar as Funções de Dados (para a aplicação de Distribuição de Correspondência)

Determine se as informações de Funcionário são uma função de dados para a aplicação Distribuição de Correspondência. A tabela a seguir mostra o resumo da análise.

Regras de Identificação de Função de Dados	A regra se aplica?
1. Identifique todos os dados ou informações de controle logicamente relacionados e reconhecidos pelo usuário dentro do escopo da contagem.	Funcionário.
2. Exclua entidades que não são mantidas por qualquer aplicação.	Não existem entidades deste tipo.
3. Agrupe entidades relacionadas que são entidades dependentes.	Não existem entidades deste tipo.
4. Exclua as entidades referidas como Dados de Código.	Não existem entidades deste tipo.
5. Exclua entidades que não possuem atributos requeridos pelo usuário.	Não existem entidades deste tipo.
6. Remova as entidades associativas que contém atributos adicionais não requeridos pelo usuário e entidades associativas que contém somente chaves estrangeiras; agrupe os atributos chave estrangeira com as entidades primárias.	Não existem entidades deste tipo.

Passo 2 **Classificar as Funções de Dados (para a aplicação Distribuição de Correspondência)**

Determine se informações de Funcionário é classificada como um ALI para a aplicação Distribuição de Correspondência.

Regras para Classificação da Função de Dados	A regra se aplica?
1. Classificar como um ALI se os dados são mantidos pela aplicação sendo medida	A função de dados Funcionário é mantida dentro da aplicação Distribuição de Correspondência.
2. Classificar como um AIE, se: <ul style="list-style-type: none"> • É referenciado, mas não mantido, pela aplicação sendo medida e • É identificado em um ALI em uma ou mais outras aplicações 	Classificado como um ALI; consequentemente, não existem AIEs identificados.

A análise mostra que as informações do Funcionário são um ALI para a aplicação Distribuição de Correspondência.

Passo 3 **Contar os DERs (para a aplicação Distribuição de Correspondência)**

Para os DERs, observe cada atributo associado com o ALI Funcionário na aplicação Distribuição de Correspondência e determine se as regras de contagem de DER se aplicam.

Informações do Funcionário incluem:

- Código do Funcionário
- Nome do Funcionário
- Endereço de correspondência do Funcionário (Andar, Código do Edifício, Rua, Cidade, Estado e CEP; na aplicação de Distribuição de Correspondência os atributos Andar e Código do Edifício são usados separadamente.)
- Faixa Salarial do Funcionário
- Cargo do Funcionário
- Data elegível para Aposentadoria
- Nível de Autorização de Segurança Organizacional

A análise dos DERs para as informações do Funcionário para a aplicação Distribuição de Correspondência é mostrada abaixo:

Regra de Contagem de DER para Função de Dados	A regra se aplica?
1. Conte um DER para cada atributo único reconhecido pelo usuário, não repetido mantido em ou recuperado de uma função de dados através da execução de todos os processos elementares dentro do escopo da contagem.	Os seguintes atributos satisfazem esta regra: <ul style="list-style-type: none"> • Código do Funcionário • Andar • Código do Edifício
2. Conte somente aqueles DERs sendo usado pela aplicação sendo medida quando duas ou mais aplicações mantém e/ou referenciam a mesma função de dados.	Somente o Código do Funcionário, Andar, e Código do Edifício são usados pela aplicação Distribuição de Correspondência.
3. Conte um DER para cada atributo requerido pelo usuário para estabelecer um relacionamento com outra função de dados.	Não existem atributos deste tipo.
4. Revise os atributos relacionados para determinar se eles são agrupados e contados como um único DER ou se eles são contados como múltiplos DERs; o agrupamento dependerá de como os processos elementares usam os atributos dentro da aplicação.	Não existem atributos deste tipo. Embora o Endereço de Correspondência do Funcionário tenha sido considerado um único atributo na aplicação de RH, são contados dois atributos separados (Andar e Código do Edifício) na aplicação de Distribuição de Correspondência.

Passo 4

Contar os RLRs (para a aplicação Distribuição de Correspondência)

Para os RLRs, identifique os subgrupos baseado nas regras de contagem de RLR.

Regras de Contagem de RLR	A regra se aplica?
1. Conte um RLR para cada função de dados (isto é, por default, cada função de dados tem um subgrupo de DERs para ser contado como um RLR).	Conte um RLR para o ALI Funcionário.
2. Conte um RLR adicional para cada subgrupo lógico de DERs a seguir (dentro da função de dados) que contém mais do que um DER: <ul style="list-style-type: none"> • Entidade associativa com atributos não-chave • Subtipo (subtipo diferente do primeiro subtipo) e • Entidade atributiva, em um relacionamento diferente de 1-1 mandatório. 	<p>Não existem entidades deste tipo.</p> <p>Não existem entidades deste tipo.</p> <p>Não existem entidades deste tipo.</p>

O total de RLR e DER para o ALI Funcionário na aplicação de Distribuição de Correspondência é mostrado na tabela a seguir.

RLRs	DERs
<ul style="list-style-type: none"> Grupo de informações do Funcionário 	<ul style="list-style-type: none"> Código do Funcionário Andar Código do Edifício
Total 1 RLR	Total 3 DERs

Passo 5 **Determinar a Complexidade Funcional (para a aplicação Distribuição de Correspondência)**

1 RLRs e 3 DERs	Complexidade é Baixa
-----------------	----------------------

Passo 6 **Determinar o Tamanho Funcional (para a aplicação Distribuição de Correspondência)**

Tamanho Funcional de 1 ALI Baixo	7 PF
----------------------------------	------

Exemplos de Contagem de AIE

Introdução Esta seção utiliza a aplicação de Recursos Humanos (RH) juntamente com a aplicação de Segurança e uma aplicação de Pensão para ilustrar procedimentos utilizados para medir funções de dados. Além desta seção, outros exemplos estão nos Estudos de Caso que são parte da documentação complementar do IFPUG.

Conteúdo Esta seção inclui os seguintes exemplos:

Tópico	Page
Resumo das Descrições dos Exemplos de Contagem de AIEs	1-40
Exemplo: Referenciando dados de Outras Aplicações	1-41
Exemplo: Referenciando dados de Uma Outra Aplicação	1-45
Exemplo: Fornecendo Dados para Outras Aplicações	1-51
Exemplo: Aplicação de Help	1-53
Exemplo: Conversão de Dados	1-62
Exemplo: Arquivo de Entrada de Transação	1-64
Exemplo: Diferentes Usuários/Diferentes Visões do Usuário	1-66
Exemplo: Múltipla utilização de Dados	1-71

Resumo da Descrição dos Exemplos de AIEs

Os exemplos para AIEs são descritos na seguinte tabela:

Exemplo	Descrição Resumida	Página
Referenciando Dados de Outras Aplicações para gerar saída	Este exemplo identifica AIEs para uma aplicação que referencia dados mantidos por outra aplicação. Os dados são utilizados para gerar uma saída externa.	1-41
Referenciando Dados de Outra Aplicação para utilizar como parte de um processo de entrada	Este exemplo também mostra dados referenciados a partir de outra aplicação. Identifica AIEs para uma aplicação que referencia dados mantidos por outra aplicação para utilização em uma entrada externa..	1-45
Fornecendo Dados para Outras Aplicações	Este é outro exemplo de contagem de dados referenciados a partir de uma aplicação diferente.	1-51
Aplicação de Help	Este é um exemplo de contagem de uma facilidade de Help dentro da aplicação de RH.	1-53
Conversão de Dados	Este é um exemplo de contagem na conversão de uma nova aplicação.	1-62
Arquivo de Entrada de Transação	Este exemplo aplica as regras de contagem de AIE para um arquivo de entrada de transação processado para incluir cargos para a aplicação de Recursos Humanos.	1-64
Diferentes Usuários / Diferentes Visões do Usuário	Este exemplo mostra que a visão difere quando um AIE é utilizado por diversas aplicações.	1-66
Uso Múltiplo de Dados	Este exemplo ilustra várias utilizações para o mesmo dado.	1-71

Exemplo: Referenciando Dados de Outras Aplicações

Requisitos do Usuário O usuário deseja que o sistema de Recursos Humanos forneça a habilidade para:

1. Incluir, consultar e listar informações do Funcionário
2. Interface com o sistema de Ativo Fixo para recuperar informações de localização de cada edifício. A informação de localização inclui as informações de nome e descrição.

Passo 1 Identificar as Funções de Dados

A partir dos requisitos do usuário, existem dois grupos de informações:

- Informações do Funcionário
- Informações de Localização

A tabela a seguir mostra o resumo da análise para determinar se Informações do Funcionário é uma função de dados.

Regras de Identificação de Função de Dados	A regra se aplica?
1. Identifique todos os dados ou informações de controle logicamente relacionados e reconhecidos pelo usuário dentro do escopo da contagem.	Funcionário e Localização.
2. Exclua entidades que não são mantidas por qualquer aplicação.	Não existem entidades deste tipo.
3. Agrupe entidades relacionadas que são entidades dependentes.	Não existem entidades deste tipo. Funcionário e Localização são entidades independentes uma da outra.
4. Exclua as entidades referidas como Dados de Código.	Não existem entidades deste tipo.
5. Exclua entidades que não possuem atributos requeridos pelo usuário.	Não existem entidades deste tipo.
6. Remova as entidades associativas que contém atributos adicionais não requeridos pelo usuário e entidades associativas que contém somente chaves estrangeiras; agrupe os atributos chave estrangeira com as entidades primárias.	Não existem entidades deste tipo.

Baseado na análise, Funcionário e Localização são identificadas como função de dados.

Passo 2

Classificar as Funções de Dados (para informações do Funcionário)

Determinar se informações do Funcionário é classificada como um AIE para a aplicação RH.

Regras para Classificação da Função de Dados	A regra se aplica?
1. Classificar como um ALI se os dados são mantidos pela aplicação sendo medida	A função de dados Funcionário é mantida dentro da aplicação.
2. Classificar como um AIE, se: <ul style="list-style-type: none"> • É referenciado, mas não mantido, pela aplicação sendo medida e • É identificado em um ALI em uma ou mais outras aplicações 	Classificada como um ALI; consequentemente, nenhum AIE é identificado.

Baseado na análise, as informações de Funcionário não são externas à aplicação de RH. Elas são mantidas internamente; portanto, não é um AIE.

Passo 2

Classificar as Funções de Dados (para informações de Localização)

Determinar se as informações de Localização são classificadas como um AIE para a aplicação de RH.

Regras para Classificação da Função de Dados	A regra se aplica?
1. Classificar como um ALI se os dados são mantidos pela aplicação sendo medida	As informações de Localização não são mantidas na aplicação de RH.
2. Classificar como um AIE, se: <ul style="list-style-type: none"> • É referenciado, mas não mantido, pela aplicação sendo medida e • É identificado em um ALI em uma ou mais outras aplicações 	<p>A função de dados Localização é referenciada, mas não mantida, pela aplicação de RH para uso no relatório de funcionários.</p> <p>Inicialmente, não está claro se as informações de Localização são mantidas em outra aplicação. Depois de perguntar aos usuários, fomos informados que eles incluem a informação na aplicação de Ativo Fixo utilizando uma tela. Portanto, as informações de Localização são um ALI para a aplicação Ativo Fixo e um AIE para a aplicação de RH.</p>

Baseado na análise, as informações de Localização são classificadas como um AIE para a aplicação de RH.

Passo 3 Contar os DERs (para Localização)

Para os DERs, observe cada atributo associado com o AIE Localização e determine se as regras de contagem de DER se aplicam.

Os atributos a seguir são referenciados a partir do AIE Localização:

- Código do Edifício
- Nome do Edifício
- Descrição do Edifício
 - Linha 1
 - Linha 2
 - Linha 3
- Cidade
- Estado
- País

A tabela a seguir mostra a análise resumida da contagem de DER.

Regra de Contagem de DER para Função de Dados	A regra se aplica?
1. Conte um DER para cada atributo único reconhecido pelo usuário, não repetido mantido em ou recuperado de uma função de dados através da execução de todos os processos elementares dentro do escopo da contagem.	Código do Edifício, Nome do Edifício, Descrição do Edifício, Cidade, Estado e País. As linhas repetidas de Descrição do Edifício são contadas como um único DER.
2. Conte somente aqueles DERs sendo usado pela aplicação sendo medida quando duas ou mais aplicações mantêm e/ou referenciam a mesma função de dados.	Não existem atributos deste tipo.
3. Conte um DER para cada atributo requerido pelo usuário para estabelecer um relacionamento com outra função de dados.	Não existem atributos deste tipo.
4. Revise os atributos relacionados para determinar se eles são agrupados e contados como um único DER ou se eles são contados como múltiplos DERs; o agrupamento dependerá de como os processos elementares usam os atributos dentro da aplicação.	Não existem atributos deste tipo.

Passo 4 Contar os RLRs (para Localização)

Para os RLRs, identifique os subgrupos baseado nas regras de contagem de RLR.

Regras de Contagem de RLR	A regra se aplica?
1. Conte um RLR para cada função de dados (isto é, por default, cada função de dados tem um subgrupo de DERs para ser contado como um RLR).	Conte um RLR para o AIE Localização.
2. Conte um RLR adicional para cada subgrupo lógico de DERs a seguir (dentro da função de dados) que contém mais do que um DER: <ul style="list-style-type: none"> Entidade associativa com atributos não-chave Subtipo (subtipo diferente do primeiro subtipo) e Entidade atributiva, em um relacionamento diferente de 1-1 mandatório. 	<p>Não existem entidades deste tipo.</p> <p>Não existem entidades deste tipo.</p> <p>Não existem entidades deste tipo.</p>

O total de RLR e DER para o AIE Localização é mostrado na tabela a seguir.

RLRs	DERs
<ul style="list-style-type: none"> dados de Localização 	<ul style="list-style-type: none"> Código do Edifício Nome do Edifício Descrição do Edifício (linhas repetidas) Cidade Estado País
Total 1 RLR	Total 6 DERs

Passo 5 Determinar a Complexidade Funcional (para Localização)

1 RLR e 6 DERs	Complexidade é Baixa
----------------	----------------------

Passo 6 Determinar o Tamanho Funcional (para Localização)

Tamanho Funcional de 1 AIE Baixo	5 PF
----------------------------------	------

Exemplo: Referenciando Dados de uma Outra Aplicação

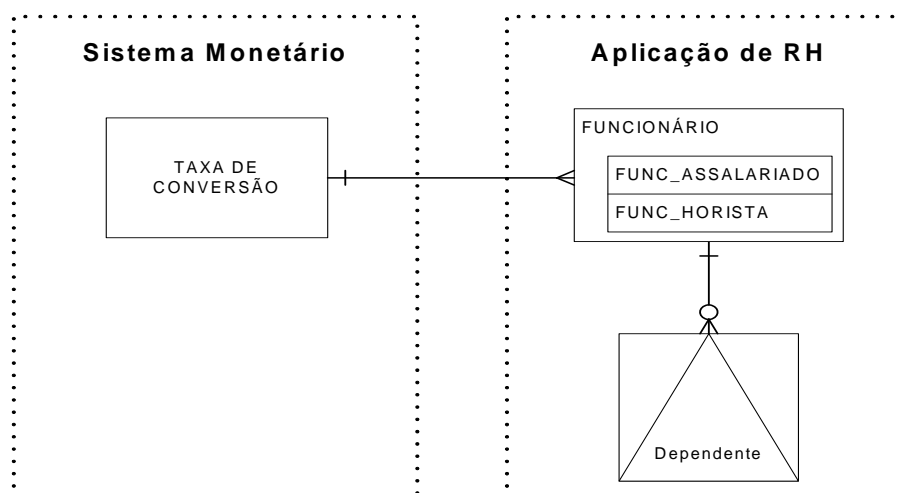
Requisitos do Usuário O usuário requer que a aplicação de Recursos Humanos forneça as seguintes habilidades:

- Todos os funcionários horistas devem ser pagos em dólares dos Estados Unidos.
- Quando o usuário incluir ou alterar informações do funcionário, a aplicação de Recursos Humanos deve acessar a o sistema Monetário para recuperar a taxa de conversão. Depois de recuperar a taxa de conversão, a aplicação de RH converte a taxa-hora padrão da localização do funcionário para a taxa-hora dos EUA utilizando o seguinte cálculo:

$$\frac{TaxaHoraPadrão}{TaxaConversão} = TaxaHoraDolarEUA$$

Modelo de Dados

O diagrama a seguir mostra os relacionamentos para este exemplo.



Legenda:

- Entidade Tipo
- Entidade Atributiva
- Entidade Subtipo
- Relacionamento Obrigatório de Um-para-muitos
- Relacionamento Opcional de Um-para-muitos

As informações de conversão de moeda incluem:

MOEDA

Taxa_Base_Para_Conversão_Moeda

País

Passo 1

Identificar as Funções de Dados

Para os requisitos, existem dois grupos de informações:

- Informações de Conversão de Moeda
- Informações de Funcionário

A tabela a seguir mostra o resumo da análise para determinar se Informações de Conversão de Moeda é uma função de dados.

Regras de Identificação de Função de Dados	A regra se aplica?
1. Identifique todos os dados ou informações de controle logicamente relacionados e reconhecidos pelo usuário dentro do escopo da contagem.	Conversão de Moeda, Funcionário e Dependente.
2. Excluir entidades que não são mantidas por qualquer aplicação.	Não existem entidades deste tipo.
3. Agrupe entidades relacionadas que são entidades dependentes.	A entidade Moeda de Conversão é independente das outras entidades. Dependente é uma entidade dependente da entidade Funcionário.
4. Exclua as entidades referidas como Dados de Código.	Embora Conversão de Moeda possa parecer uma instância de dados de código, Código do País e Taxa de Conversão não são substituíveis (isto é, não podem ser substituído um pelo outro). Informações de Conversão de Moeda também mudam regularmente, de modo que não satisfazem os critérios de serem essencialmente estáticos.
5. Exclua entidades que não possuem atributos requeridos pelo usuário.	Não existem entidades deste tipo.
6. Remova as entidades associativas que contém atributos adicionais não requeridos pelo usuário e entidades associativas que contém somente chaves estrangeiras; agrupe os atributos chave estrangeira com as entidades primárias.	Não existem entidades deste tipo.

Baseado na análise, Conversão de Moeda e Funcionário são identificadas como função de dados. Dependente não é uma função de dados própria, mas é parte da função de dados Funcionário.

Passo 2

Classificar as Funções de Dados (para Funcionário)

A tabela a seguir mostra a análise para determinar se informações do Funcionário é classificada como um AIE.

Regras para Classificação da Função de Dados	A regra se aplica?
1. Classificar como um ALI se os dados são mantidos pela aplicação sendo medida	Informações do Funcionário são mantidas pela aplicação de RH.
2. Classificar como um AIE, se: <ul style="list-style-type: none"> É referenciado, mas não mantido, pela aplicação sendo medida e É identificado em um ALI em uma ou mais outras aplicações 	Classificado como um ALI; consequentemente, nenhum AIEs são identificados.

Baseado na análise, as informações de Funcionário não são externas à aplicação de RH. Elas são mantidas internamente; portanto, não são um AIE.

Passo 2 Classificar as Funções de Dados (para Conversão de Moeda)

A tabela a seguir mostra a análise para determinar se informações de Conversão de Moeda são classificadas como AIE.

Regras para Classificação da Função de Dados	A regra se aplica?
1. Classificar como um ALI se os dados são mantidos pela aplicação sendo medida	Conversão de Moeda não é mantida pela aplicação de RH.
2. Classificar como um AIE, se: <ul style="list-style-type: none"> • É referenciado, mas não mantido, pela aplicação sendo medida e • É identificado em um ALI em uma ou mais outras aplicações 	<p>A função de dados Conversão de Moeda é referenciada pela aplicação de RH para uso no cálculo da remuneração do empregado.</p> <p>Embora Conversão de Moeda possa parecer uma instância de dados de código, Código do País e Taxa de Conversão não são substituíveis (isto é, não podem ser substituído um pelo outro). Informações de Conversão de Moeda também mudam regularmente, de modo que não satisfazem os critérios de serem essencialmente estáticos.</p>

Como a aplicação Sistema Monetário fornece a taxa de conversão para a aplicação de RH, o grupo de dados de conversão de moeda é um AIE para a aplicação de RH.

Passo 3 Contar os DERs (para Conversão de Moeda)

Para os DERs, observe cada atributo associado com o AIE Conversão de Moeda e determine se as regras de contagem de DER se aplicam. A tabela a seguir mostra o resumo da análise da contagem de DER.

Regra de Contagem de DER para Função de Dados	A regra se aplica?
1. Conte um DER para cada atributo único reconhecido pelo usuário, não repetido mantido em ou recuperado de uma função de dados através da execução de todos os processos elementares dentro do escopo da contagem.	Taxa de Conversão, Moeda.
2. Conte somente aqueles DERs sendo usado pela aplicação sendo medida quando duas ou mais aplicações mantém e/ou referenciam a mesma função de dados.	Todos os atributos são referenciados pela aplicação de RH.
3. Conte um DER para cada atributo requerido pelo usuário para estabelecer um relacionamento com outra função de dados.	Não existem atributos deste tipo.
4. Revise os atributos relacionados para determinar se eles são agrupados e contados como um único DER ou se eles são contados como múltiplos DERs; o agrupamento dependerá de como os processos elementares usam os atributos dentro da aplicação.	Não existem atributos deste tipo.

Passo 4 Contar os RLRs (para Conversão de Moeda)

Para os RLRs, identifique os subgrupos baseado nas regras de contagem de RLR.

Regras de Contagem de RLR	A regra se aplica?
1. Conte um RLR para cada função de dados (isto é, por default, cada função de dados tem um subgrupo de DERs para ser contado como um RLR).	Conte um RLR para o AIE Conversão de Moeda.
2. Conte um RLR adicional para cada subgrupo lógico de DERs a seguir (dentro da função de dados) que contém mais do que um DER: <ul style="list-style-type: none"> Entidade associativa com atributos não-chave Subtipo (subtipo diferente do primeiro subtipo) e Entidade atributiva, em um relacionamento diferente de 1-1 mandatório. 	<p>Não existem entidades deste tipo.</p> <p>Não existem entidades deste tipo.</p> <p>Não existem entidades deste tipo.</p>

O total de RLR e DER para o AIE informações de Conversão de Moeda é mostrado na tabela a seguir.

RLRs	DERs
<ul style="list-style-type: none"> Informações de Conversão 	<ul style="list-style-type: none"> Taxa de Conversão Moeda
Total 1 RLR	Total 2 DERs

Passo 5 Determinar a Complexidade Funcional (para Conversão de Moeda)

1 RLR e 2 DERs	Complexidade é Baixa
----------------	----------------------

Passo 6 Determinar o Tamanho Funcional (para Conversão de Moeda)

Tamanho Funcional de 1 AIE Baixo	5 PF
----------------------------------	------

Exemplo: Fornecendo Dados para Outras Aplicações

Requisitos do Usuário

O usuário tem os seguintes requisitos para o sistema Monetário:

- Manter taxa de conversão de outras moedas para o dólar americano.
- Fornecer uma interface para habilitar outras aplicações, como Recursos Humanos, a recuperar informação de conversão.

Passo 1

Identificar as Funções de Dados

Para este exemplo, determinar se informações de Conversão de Moeda é uma função de dados para a aplicação Sistema Monetário. A tabela a seguir mostra o resumo da análise.

Regras de Identificação de Função de Dados	A regra se aplica?
1. Identifique todos os dados ou informações de controle logicamente relacionados e reconhecidos pelo usuário dentro do escopo da contagem.	Conversão de Moeda.
2. Exclua entidades que não são mantidas por qualquer aplicação.	Não existem entidades deste tipo.
3. Agrupe entidades relacionadas que são entidades dependentes.	Não existem entidades deste tipo.
4. Exclua as entidades referidas como Dados de Código.	Embora Conversão de Moeda possa parecer uma instância de dados de código, Código do País e Taxa de Conversão não são substituíveis (isto é, não podem ser substituído um pelo outro). Informações de Conversão de Moeda também mudam regularmente, de modo que não satisfazem os critérios de serem essencialmente estáticos.
5. Exclua entidades que não possuem atributos requeridos pelo usuário.	Não existem entidades deste tipo.
6. Remova as entidades associativas que contém atributos adicionais não requeridos pelo usuário e entidades associativas que contém somente chaves estrangeiras; agrupe os atributos chave estrangeira com as entidades primárias.	Não existem entidades deste tipo.

Passo 2**Classificar as Funções de Dados (para Conversão de Moeda)**

A tabela a seguir mostra a análise para determinar se informações de Conversão de Moeda é classificada como um AIE.

Regras para Classificação da Função de Dados	A regra se aplica?
1. Classificar como um ALI se os dados são mantidos pela aplicação sendo medida	A aplicação Sistema Monetário mantém os dados de Conversão de Moeda através de transações a partir de um serviço online.
2. Classificar como um AIE, se: <ul style="list-style-type: none">• É referenciado, mas não mantido, pela aplicação sendo medida e• É identificado em um ALI em uma ou mais outras aplicações	Classificado como um ALI; consequentemente, nenhum AIEs são identificados.

As informações de Conversão de Moeda não são externas à aplicação Sistema Monetário; portanto, ela é contada como um ALI ao invés de um AIE para a aplicação Sistema Monetário. Veja o exemplo anterior neste capítulo para rever como a referência a Conversão de Moeda pode ser contada como um AIE.

Exemplo: Aplicação de Help

Requisitos do Usuário

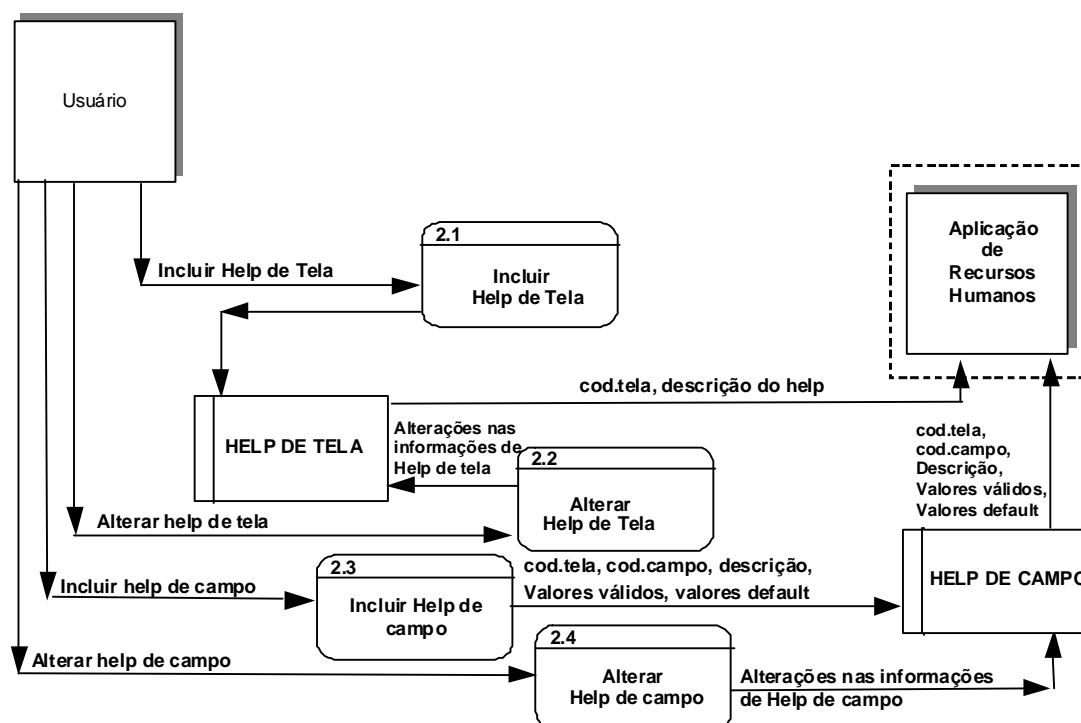
O usuário requer ao sistema de Help fornecer:

1. A habilidade de descrever a forma como cada tela é utilizada para realizar cada função de negócios disponível na mesma.
2. A habilidade de alterar o Help de tela.
3. A habilidade para estabelecer uma definição, valores default, e valores válidos para cada atributo na aplicação de Recursos Humanos.
4. A habilidade de alterar o Help de campo.
5. A habilidade para a aplicação de Recursos Humanos recuperar o Help de tela e de campo para apresentação.

O Help de tela e Help de campo são mantidos independentemente. Pode existir uma entrada em um tipo de help sem existir em outro.

Diagrama de Fluxo de Dados

O diagrama a seguir ilustra o fluxo de dados para este exemplo.



Legenda:

Usuário ou Aplicação



Processo



Depósito de Dados



Fluxo de Dados

Passo 1 Identificar as Funções de Dados

A partir dos requisitos para a aplicação de Recursos Humanos (RH), existem dois grupos de dados:

- Help de tela
- Help de campo

A tabela a seguir mostra o resumo da análise para determinar se Help de tela e Help de campo são funções de dados para a aplicação de RH.

Regras de Identificação de Função de Dados	A regra se aplica?
1. Identifique todos os dados ou informações de controle logicamente relacionados e reconhecidos pelo usuário dentro do escopo da contagem.	Help de tela, Help de campo.
2. Exclua entidades que não são mantidas por qualquer aplicação.	Não existem entidades deste tipo.
3. Agrupe entidades relacionadas que são entidades dependentes.	Help de tela e Help de campo são independentes uma da outra. Pode existir uma entrada numa entidade sem existir na outra.
4. Exclua as entidades referidas como Dados de Código.	Help de tela e Help de campo consiste de mais do que apenas atributos de código e descrição, não são usados para substituição e armazenam dados para suportar atividades essenciais do usuário; consequentemente, elas não são consideradas dados de código.
5. Exclua entidades que não possuem atributos requeridos pelo usuário.	Não existem entidades deste tipo.
6. Remova as entidades associativas que contém atributos adicionais não requeridos pelo usuário e entidades associativas que contém somente chaves estrangeiras; agrupe os atributos chave estrangeira com as entidades primárias.	Não existem entidades deste tipo.

Passo 2 **Classificar as Funções de Dados (para Help de tela)**

A tabela a seguir mostra a análise para determinar se informações de Help de tela é classificada como um AIE.

Regras para Classificação da Função de Dados	A regra se aplica?
1. Classificar como um ALI se os dados são mantidos pela aplicação sendo medida	Help de tela não é mantido pela aplicação de RH.
2. Classificar como um AIE, se: <ul style="list-style-type: none">• É referenciado, mas não mantido, pela aplicação sendo medida e• É identificado em um ALI em uma ou mais outras aplicações	<p>A aplicação de RH referencia, mas não mantém Help de tela.</p> <p>A aplicação Help identificou Help de tela como um ALI.</p>

As informações de Help de tela são um AIE na aplicação de RH porque as informações são referenciadas pela aplicação de RH. Help de tela é mantida na fronteira da aplicação Help, onde ela é contada como um ALI.

Passo 3 Contar os DERs (para Help de tela)

Para os DERs, observe cada atributo associado com o help de tela e use as regras de contagem de DER para contá-los. Os atributos para help de tela incluem:

- Identificador da Tela
- Descrição da Função de Negócio.

A tabela a seguir mostra a análise de DER para Help de tela.

Regra de Contagem de DER para Função de Dados	A regra se aplica?
1. Conte um DER para cada atributo único reconhecido pelo usuário, não repetido mantido em ou recuperado de uma função de dados através da execução de todos os processos elementares dentro do escopo da contagem.	Identificador de tela, Descrição da Função de Negócio.
2. Conte somente aqueles DERs sendo usado pela aplicação sendo medida quando duas ou mais aplicações mantêm e/ou referenciam a mesma função de dados.	Todos os atributos são referenciados pela aplicação de RH.
3. Conte um DER para cada atributo requerido pelo usuário para estabelecer um relacionamento com outra função de dados.	Não existem atributos deste tipo.
4. Revise os atributos relacionados para determinar se eles são agrupados e contados como um único DER ou se eles são contados como múltiplos DERs; o agrupamento dependerá de como os processos elementares usam os atributos dentro da aplicação.	Não existem atributos deste tipo.

Passo 4 Contar os RLRs (para Help de tela)

Para os RLRs, identifique os subgrupos baseado nas regras de contagem de RLR.

Regras de Contagem de RLR	A regra se aplica?
1. Conte um RLR para cada função de dados (isto é, por default, cada função de dados tem um subgrupo de DERs para ser contado como um RLR).	Conte um RLR para Help de tela.
2. Conte um RLR adicional para cada subgrupo lógico de DERs a seguir (dentro da função de dados) que contém mais do que um DER: <ul style="list-style-type: none"> Entidade associativa com atributos não-chave Subtipo (subtipo diferente do primeiro subtipo) e Entidade atributiva, em um relacionamento diferente de 1-1 mandatório. 	<p>Não existem entidades deste tipo para qualquer função de dados.</p> <p>Não existem entidades deste tipo para qualquer função de dados.</p> <p>Não existem entidades deste tipo para qualquer função de dados.</p>

O total de RLR e DER para o AIE Help de tela é mostrado na tabela a seguir.

RLRs	DERs
<ul style="list-style-type: none"> informações de Help de tela 	<ul style="list-style-type: none"> Identificador de tela Descrição da função de negócio
Total 1 RLR	Total 2 DERs

Passo 5 Determinar a Complexidade Funcional (para Help de tela)

1 RLR e 2 DERs	Complexidade é Baixa
----------------	----------------------

Passo 6 Determinar o Tamanho Funcional (para Help de tela)

Tamanho Funcional de 1 AIE Baixo	5 PF
----------------------------------	------

Passo 2 Classificar as Funções de Dados (para Help de Campo)

A tabela a seguir mostra o resumo dos resultados da análise para determinar se Help de campo é classificado como um AIE.

Regras para Classificação da Função de Dados	A regra se aplica?
1. Classificar como um ALI se os dados são mantidos pela aplicação sendo medida	Help de campo não é mantido pela aplicação de RH.
2. Classificar como um AIE, se: <ul style="list-style-type: none">• É referenciado, mas não mantido, pela aplicação sendo medida e• É identificado em um ALI em uma ou mais outras aplicações	<p>A aplicação de RH referencia, mas não mantém Help de campo.</p> <p>A aplicação Help identificou Help de campo como um ALI.</p>

Informações de Help de campo é um AIE na aplicação de RH porque as informações são recuperadas pela aplicação de RH. As informações de Help de campo são mantidas na aplicação Help onde ele é contado como um ALI.

Passo 3 Contar os DERs (para Help de Campo)

Para os DERs, observe cada atributo associado com o Help de campo e utilize as regras de contagem de DER para contá-los. A lista a seguir mostra os atributos para Help de campo:

- Código da tela
- Código do campo
- Descrição do campo
- Valores default
- Valores Válidos

A tabela a seguir mostra a análise de DER para o Help de campo.

Regra de Contagem de DER para Função de Dados	A regra se aplica?
1. Conte um DER para cada atributo único reconhecido pelo usuário, não repetido mantido em ou recuperado de uma função de dados através da execução de todos os processos elementares dentro do escopo da contagem.	Código da tela, código do campo, Descrição do Campo, Valores Default, Valores Válidos.
2. Conte somente aqueles DERs sendo usado pela aplicação sendo medida quando duas ou mais aplicações mantêm e/ou referenciam a mesma função de dados.	Todos os atributos são referenciados pela aplicação de RH.
3. Conte um DER para cada atributo requerido pelo usuário para estabelecer um relacionamento com outra função de dados.	Não existem atributos deste tipo.
4. Revise os atributos relacionados para determinar se eles são agrupados e contados como um único DER ou se eles são contados como múltiplos DERs; o agrupamento dependerá de como os processos elementares usam os atributos dentro da aplicação.	Não existem atributos deste tipo.

Passo 4 Contar os RLRs (para Help de Campo)

Para os RLRs, identifique os subgrupos baseado nas regras de contagem de RLR.

Regras de Contagem de RLR	A regra se aplica?
1. Conte um RLR para cada função de dados (isto é, por default, cada função de dados tem um subgrupo de DERs para ser contado como um RLR).	Conte um RLR para o Help de Campo.
2. Conte um RLR adicional para cada subgrupo lógico de DERs a seguir (dentro da função de dados) que contém mais do que um DER: <ul style="list-style-type: none"> Entidade associativa com atributos não-chave Subtipo (subtipo diferente do primeiro subtipo) e Entidade atributiva, em um relacionamento diferente de 1-1 mandatório. 	<p>Não existem entidades deste tipo para qualquer função de dados.</p> <p>Não existem entidades deste tipo para qualquer função de dados.</p> <p>Não existem entidades deste tipo para qualquer função de dados.</p>

O total de RLR e DER para o AIE Help de campo é mostrado na tabela a seguir.

RLRs	DERs
<ul style="list-style-type: none"> Informações de Help de Campo 	<ul style="list-style-type: none"> Código da tela Código do Campo Descrição do Campo Valores Default Valores Válidos
Total 1 RLR	Total 5 DERs

Passo 5 Determinar a Complexidade Funcional (para Help de Campo)

1 RLR e 5 DERs	Complexidade é Baixa
----------------	----------------------

Passo 6 Determinar o Tamanho Funcional (para Help de Campo)

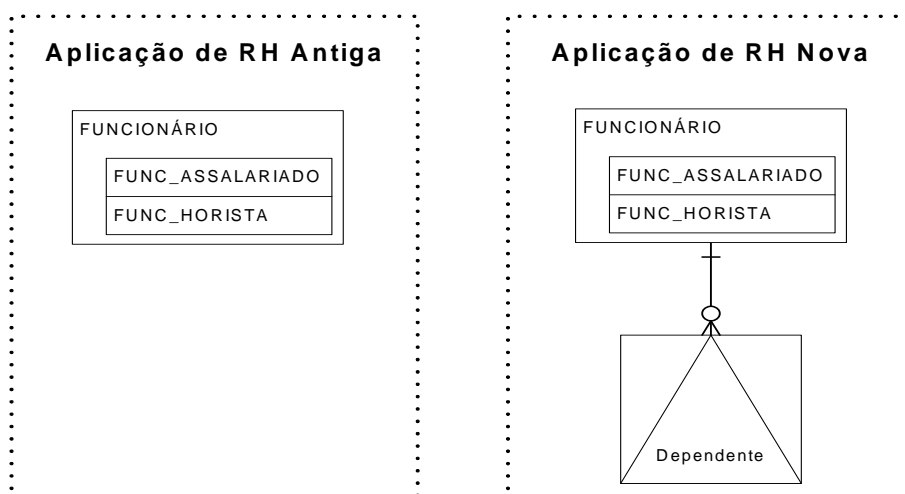
Tamanho Funcional de 1 AIE Baixo	5 PF
----------------------------------	------

Exemplo: Conversão de Dados






Requisitos do Usuário Uma organização adquiriu um novo pacote da aplicação de RH. A organização está requerendo converter seu arquivo de funcionários do sistema de RH existente para o sistema comprado.

O sistema antigo não fornecia a capacidade de manter informações do dependente do funcionário. A informação do dependente é inicializada quando os funcionários existentes são migrados para a nova aplicação

Modelo de Dados O diagrama a seguir mostra os dados para as duas aplicações.



Legenda:

-  Entidade Tipo
-  Entidade Atributiva
-  Entidade Subtipo
-  Relacionamento Obrigatório de Um-para-muitos
-  Relacionamento Opcional de Um-para-muitos

O arquivo Funcionário da aplicação de RH antiga é utilizado para incluir funcionários na nova aplicação de RH. O arquivo Funcionário da aplicação de RH antiga tem a intenção primária de manter (isto é, popular) o arquivo Funcionário na nova aplicação. O arquivo Funcionário da aplicação de RH antiga não satisfaz a intenção primária de um AIE que é armazenar dados referenciados através de um ou mais processos elementares.

Passo 1 Identificar as Funções de Dados

A partir dos requisitos do usuário, determinar se o antigo arquivo Funcionário é uma função de dados. A tabela a seguir mostra o resumo da análise.

Regras de Identificação de Função de Dados	A regra se aplica?
1. Identifique todos os dados ou informações de controle logicamente relacionados e reconhecidos pelo usuário dentro do escopo da contagem.	O antigo arquivo Funcionário não é um grupo lógico de dados na perspectiva do usuário.
2. Exclua entidades que não são mantidas por qualquer aplicação.	O antigo arquivo Funcionário é uma saída (isto é, extração) da aplicação anterior ao invés de um arquivo lógico que é mantido.
3. Agrupe entidades relacionadas que são entidades dependentes.	Não existem entidades deste tipo.
4. Exclua as entidades referidas como Dados de Código.	Não existem entidades deste tipo.
5. Exclua entidades que não possuem atributos requeridos pelo usuário.	Não existem entidades deste tipo.
6. Remova as entidades associativas que contém atributos adicionais não requeridos pelo usuário e entidades associativas que contém somente chaves estrangeiras; agrupe os atributos chave estrangeira com as entidades primárias.	Não existem entidades deste tipo.

O arquivo de informações de funcionários é um arquivo de transação da informação do funcionário que é migrado para o novo sistema. O processo de conversão utiliza o arquivo de transação para manter informação de funcionário após a entrada da nova aplicação de RH.

O antigo arquivo Funcionário não é um grupo lógico de dados na perspectiva do usuário da aplicação de RH nova. A intenção primária do antigo arquivo Funcionário é servir como uma entrada para a nova aplicação de RH, não armazenar dados utilizados como referência por um ou mais processos elementares da nova aplicação de RH, portanto ele não é um AIE. Consulte os exemplos de contagem de EE/SE/CE para ver como o antigo arquivo Funcionário pode ser contado como uma Entrada Externa.

Exemplo: Arquivo de Entrada de Transação

Requisitos do usuário O usuário solicita a habilidade para:

1. incluir, alterar, excluir, consultar e imprimir a informação da função online.
2. incluir e alterar informação da função no modo batch.

Layout do Registro O diagrama a seguir mostra o layout dos registros para este exemplo para inclusão e alteração de informação de função no modo batch.

	123456789101234567890123456789012345678901234567890123456789012345678901234567890
1	ADD 01 SRENG SENIOR ENGINEER INFORMATION SYSTEMS 05
2	ADD 02 SRENG 01 STARTS AT PAY GRADE 05
3	ADD 02 SRENG 02 OTHER PAY GRADES:06 AND 07
4	CHG 01 STENG 04
5	CHG 02 STENG 02 OTHER PAY GRADES:05 AND 06
6	
7	
9	
0	

Descrição dos Registros A tabela a seguir inclui descrições para cada tipo de registro

Registro	Posição	Descrição
01	1-3	Tipo de Transação
	4-5	Tipo de Registro
	6-10	Número da Função
	11-45	Nome da Função
	46-47	Faixa Salarial da Função
02	1-3	Tipo de Transação
	4-5	Tipo de Registro
	6-10	Número da Função
	11-12	Número da Linha da Descrição da Função
	13-41	Linhas de Descrição da Função

Passo 1 Identificar as Funções de Dados

A partir dos requisitos do usuário, determinar se o arquivo de transação é uma função de dados. A tabela a seguir mostra o resumo da análise.

Regras de Identificação de Função de Dados	A regra se aplica?
1. Identifique todos os dados ou informações de controle logicamente relacionados e reconhecidos pelo usuário dentro do escopo da contagem.	Sim. Dados são agrupados no arquivo de transações que entram pela fronteira da aplicação para manter o ALI Função.
2. Exclua entidades que não são mantidas por qualquer aplicação.	O arquivo de transação é excluído. As transações que entram pela fronteira da aplicação para manter o ALI Função compõem o processo elementar. Não existe processo elementar para atualizar o arquivo de transação.
3. Agrupe entidades relacionadas que são entidades dependentes.	Não existem entidades deste tipo.
4. Exclua as entidades referidas como Dados de Código.	Não existem entidades deste tipo.
5. Exclua entidades que não possuem atributos requeridos pelo usuário.	Não existem entidades deste tipo.
6. Remova as entidades associativas que contém atributos adicionais não requeridos pelo usuário e entidades associativas que contém somente chaves estrangeiras; agrupe os atributos chave estrangeira com as entidades primárias.	Não existem entidades deste tipo.

Não existem AIEs para este exemplo. Consulte os exemplos de contagem para EE/SE/CE para ver a explanação de como um arquivo de transação de entrada pode ser contado como uma Entrada Externa.

Exemplo: Diferentes Usuários/Diferentes Visões do Usuário

Requisitos do Usuário O usuário de RH deseja a habilidade para manter informação de cada novo funcionário.

A informação que deve ser mantida pelo usuário de RH inclui:

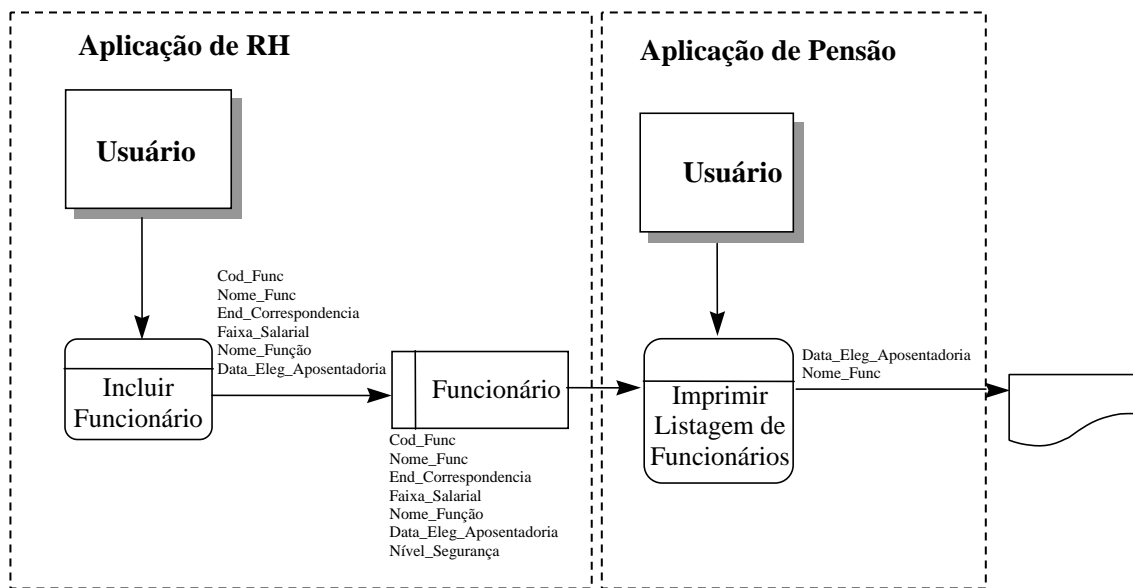
- ID do funcionário
- Nome do funcionário
- Endereço de correspondência do funcionário
- Faixa salarial do funcionário
- Título da função do funcionário
- Data elegível para aposentadoria *

* Como resultado da criação de um novo registro de funcionário, a data elegível para aposentadoria do funcionário deve ser automaticamente calculada e salva com as outras informações do Funcionário.

O usuário de Pensão solicita a habilidade para gerar uma lista de funcionários com sua data prevista de elegível para aposentadoria.

Outros atributos (por exemplo, Nivel de Segurança Organizacional) existem na entidade Funcionário, mas eles não são referenciados ou mantidos pela aplicação de RH ou aplicação de Distribuição de Correspondência.

Diagrama de Fluxo de Dados O diagrama a seguir mostra o fluxo de dados para este exemplo.



Passo 1**Identificar as Funções de Dados (para a aplicação Pensão)**

A tabela a seguir mostra o resumo da análise se a informação do funcionário é uma função de dados para a aplicação Pensão.

Regras de Identificação de Função de Dados	A regra se aplica?
1. Identifique todos os dados ou informações de controle logicamente relacionados e reconhecidos pelo usuário dentro do escopo da contagem.	Funcionário.
2. Exclua entidades que não são mantidas por qualquer aplicação.	Informação de Funcionário é mantida pela aplicação de RH. Ela não é excluída.
3. Agrupe entidades relacionadas que são entidades dependentes.	Não existem entidades deste tipo.
4. Exclua as entidades referidas como Dados de Código.	Não existem entidades deste tipo.
5. Exclua entidades que não possuem atributos requeridos pelo usuário.	Não existem entidades deste tipo.
6. Remova as entidades associativas que contém atributos adicionais não requeridos pelo usuário e entidades associativas que contém somente chaves estrangeiras; agrupe os atributos chave estrangeira com as entidades primárias.	Não existem entidades deste tipo.

Passo 2 Classificar as Funções de Dados (para a aplicação Pensão)

A tabela a seguir mostra o resumo do resultado da análise para determinar se Funcionário é classificado como uma AIE para a aplicação Pensão.

Regras para Classificação da Função de Dados	A regra se aplica?
1. Classificar como um ALI se os dados são mantidos pela aplicação sendo medida	A aplicação Pensão não mantém informação de Funcionário.
2. Classificar como um AIE, se: <ul style="list-style-type: none"> • É referenciado, mas não mantido, pela aplicação sendo medida e • É identificado em um ALI em uma ou mais outras aplicações 	<p>A aplicação Pensão referencia, mas não mantém os dados de Funcionário.</p> <p>A aplicação de RH identificou Funcionário como um ALI.</p>

As informações de Funcionário preenche todos os requisitos para um AIE para a aplicação Pensão.

Passo 3 Contar os DERs (para a aplicação Pensão)

Para os DERs, observe cada atributo associado com o AIE Funcionário para a aplicação Pensão. Para contar os DERs utilize as regras de contagem de DER.

Os atributos para informações de funcionário incluem:

- Código do Funcionário
- Nome do Funcionário
- Endereço de correspondência do Funcionário
- Faixa Salarial do Funcionário
- Cargo do Funcionário
- Data elegível para Aposentadoria

A tabela a seguir mostra a análise de DER para o AIE Funcionário para a aplicação Pensão.

Regra de Contagem de DER para Função de Dados	A regra se aplica?
<p>1. Conte um DER para cada atributo único reconhecido pelo usuário, não repetido mantido em ou recuperado de uma função de dados através da execução de todos os processos elementares dentro do escopo da contagem.</p>	<p>As informações de Funcionário incluem os seguintes atributos:</p> <ul style="list-style-type: none"> • Código do Funcionário • Nome do Funcionário • Endereço de correspondência do Funcionário • Faixa Salarial do Funcionário • Cargo do Funcionário • Data elegível para Aposentadoria • Nivel de Segurança Organizacional
<p>2. Conte somente aqueles DERs sendo usado pela aplicação sendo medida quando duas ou mais aplicações mantêm e/ou referenciam a mesma função de dados.</p>	<p>Somente o Nome do Funcionário e Data elegível para Aposentadoria são reconhecidos pelo usuário de Pensão. Todos os outros atributos não são contados como DERs para a aplicação Pensão.</p>
<p>3. Conte um DER para cada atributo requerido pelo usuário para estabelecer um relacionamento com outra função de dados.</p>	<p>Não existem atributos deste tipo.</p>
<p>4. Revise os atributos relacionados para determinar se eles são agrupados e contados como um único DER ou se eles são contados como múltiplos DERs; o agrupamento dependerá de como os processos elementares usam os atributos dentro da aplicação.</p>	<p>Não existem atributos deste tipo.</p>

Passo 4 Contar os RLRs (para a aplicação Pensão)

Para os RLRs, identifique os subgrupos baseado nas regras de contagem de RLR.

Regras de Contagem de RLR	A regra se aplica?
1. Conte um RLR para cada função de dados (isto é, por default, cada função de dados tem um subgrupo de DERs para ser contado como um RLR).	Conte um RLR para o AIE Funcionário.
2. Conte um RLR adicional para cada subgrupo lógico de DERs a seguir (dentro da função de dados) que contém mais do que um DER: <ul style="list-style-type: none"> Entidade associativa com atributos não-chave Subtipo (subtipo diferente do primeiro subtipo) e Entidade atributiva, em um relacionamento diferente de 1-1 mandatório. 	<p>Não existem entidades deste tipo.</p> <p>Não existem entidades deste tipo.</p> <p>Não existem entidades deste tipo.</p>

O total de RLR e DER para o AIE Funcionário na aplicação Pensão é mostrado na tabela a seguir.

RLRs	DERs
<ul style="list-style-type: none"> informações do Funcionário 	<ul style="list-style-type: none"> Nome do Funcionário Data elegível para Aposentadoria
Total 1 RLR	Total 2 DERs

Passo 5 Determinar a Complexidade Funcional

1 RLR e 2 DERs	Complexidade é Baixa
----------------	----------------------

Passo 6 Determinar o Tamanho Funcional

Tamanho Funcional de 1 AIE Baixo	5 PF
----------------------------------	------

Exemplo: Uso Múltiplo de Dados

Requisitos do Usuário

As informações de Funcionário são mantidas pela aplicação de RH.

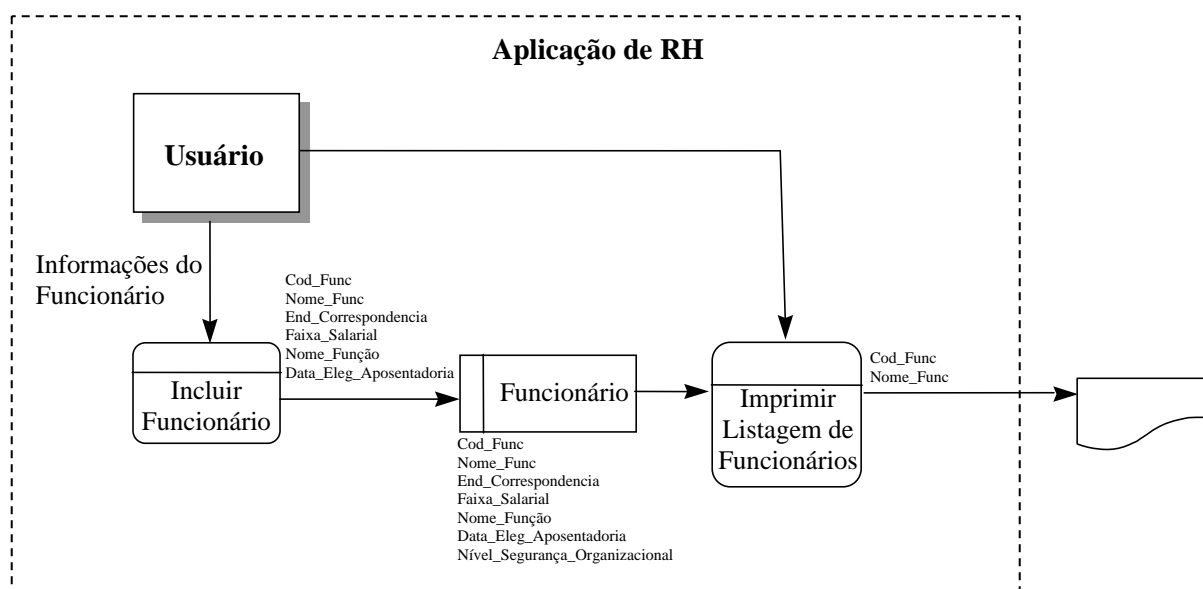
O usuário de RH requer a habilidade para gerar uma listagem de todos os funcionários.

As informações que devem ser apresentadas para cada funcionário incluem:

- Código do Funcionário
- Nome do Funcionário

Diagrama de Fluxo de Dados

O diagrama a seguir mostra o fluxo de dados para este exemplo.



Passo 1 Identificar as Funções de Dados

A tabela a seguir mostra o resumo da análise para determinar se as informações de funcionários são uma função de dados para a aplicação de RH.

Regras de Identificação de Função de Dados	A regra se aplica?
1. Identifique todos os dados ou informações de controle logicamente relacionados e reconhecidos pelo usuário dentro do escopo da contagem.	Funcionário.
2. Exclua entidades que não são mantidas por qualquer aplicação.	As informações de Funcionário são mantidas pela aplicação de RH. Ela não é excluída.
3. Agrupe entidades relacionadas que são entidades dependentes.	Não existem entidades deste tipo.
4. Exclua as entidades referidas como Dados de Código.	Não existem entidades deste tipo.
5. Exclua entidades que não possuem atributos requeridos pelo usuário.	Não existem entidades deste tipo.
6. Remova as entidades associativas que contém atributos adicionais não requeridos pelo usuário e entidades associativas que contém somente chaves estrangeiras; agrupe os atributos chave estrangeira com as entidades primárias.	Não existem entidades deste tipo.

Passo 2 Classificar as Funções de Dados

A tabela a seguir mostra o resumo da análise para determinar se as informações de funcionários que são utilizadas para criar a listagem de funcionários são também classificadas como um AIE para a aplicação de RH.

Regras para Classificação da Função de Dados	A regra se aplica?
1. Classificar como um ALI se os dados são mantidos pela aplicação sendo medida.	As informações de Funcionário são mantidas pela aplicação de RH. Classifique-a como um ALI.
2. Classificar como um AIE, se: <ul style="list-style-type: none"> • É referenciado, mas não mantido, pela aplicação sendo medida e • É identificado em um ALI em uma ou mais outras aplicações 	<p>Funcionário foi classificado como um ALI; consequentemente, não existem AIEs identificados.</p> <p>Não se aplica.</p>

As informações de Funcionário usadas para criar a Listagem de Funcionários não é um AIE para a aplicação de RH.

Página intencionalmente deixada em branco.

Exemplos de Contagem de Funções de Transação

Introdução

Esta seção utiliza uma aplicação de Recursos Humanos (RH) para ilustrar os procedimentos aplicados na contagem do tamanho de funções de transação. Além desta seção, os exemplos estão descritos nos Estudos de Casos que são parte da documentação suplementar do IFPUG.

Nota: Os exemplos utilizados nesta seção e ao longo deste manual possuem duas finalidades:

1. Ilustrar como as regras de contagem de pontos de função são aplicadas para um determinado conjunto de requisitos do usuário.
2. Possibilitar a prática de contagem a partir dos procedimentos de contagem apresentados.

Cada contador deve:

- Analisar os requisitos específicos do usuário que são aplicáveis para cada projeto ou aplicação que será contado, e
- Contar baseado nesses requisitos

Conteúdo

Esta seção descreve como os exemplos estão organizados e inclui exemplos detalhados de cada função de transação.

Tópico	Página
Organização dos Exemplos de Contagem	2-2
Regras Gerais dos Procedimentos de Contagem das Funções de Transação	2-5
Exemplos de Identificação de Processo Elementar	2-7
Exemplos de Contagem de EE	2-63
Exemplos de Contagem de SE	2-105
Exemplos de Contagem de CE	2-129

Organização dos Exemplos de Contagem

Esta seção explica como os exemplos são apresentados.

Resumo da Organização

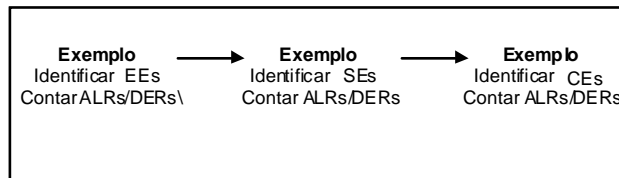
A lista abaixo descreve, em linhas gerais, a sequência de informações dos exemplos detalhados.

Para cada exemplo:

- As EEs, SEs e as CEs são identificadas
- Os ALRs e DERs que contribuem para a definição da complexidade funcional são contados.

Diagrama da Organização

O diagrama abaixo ilustra como os exemplos estão organizados:



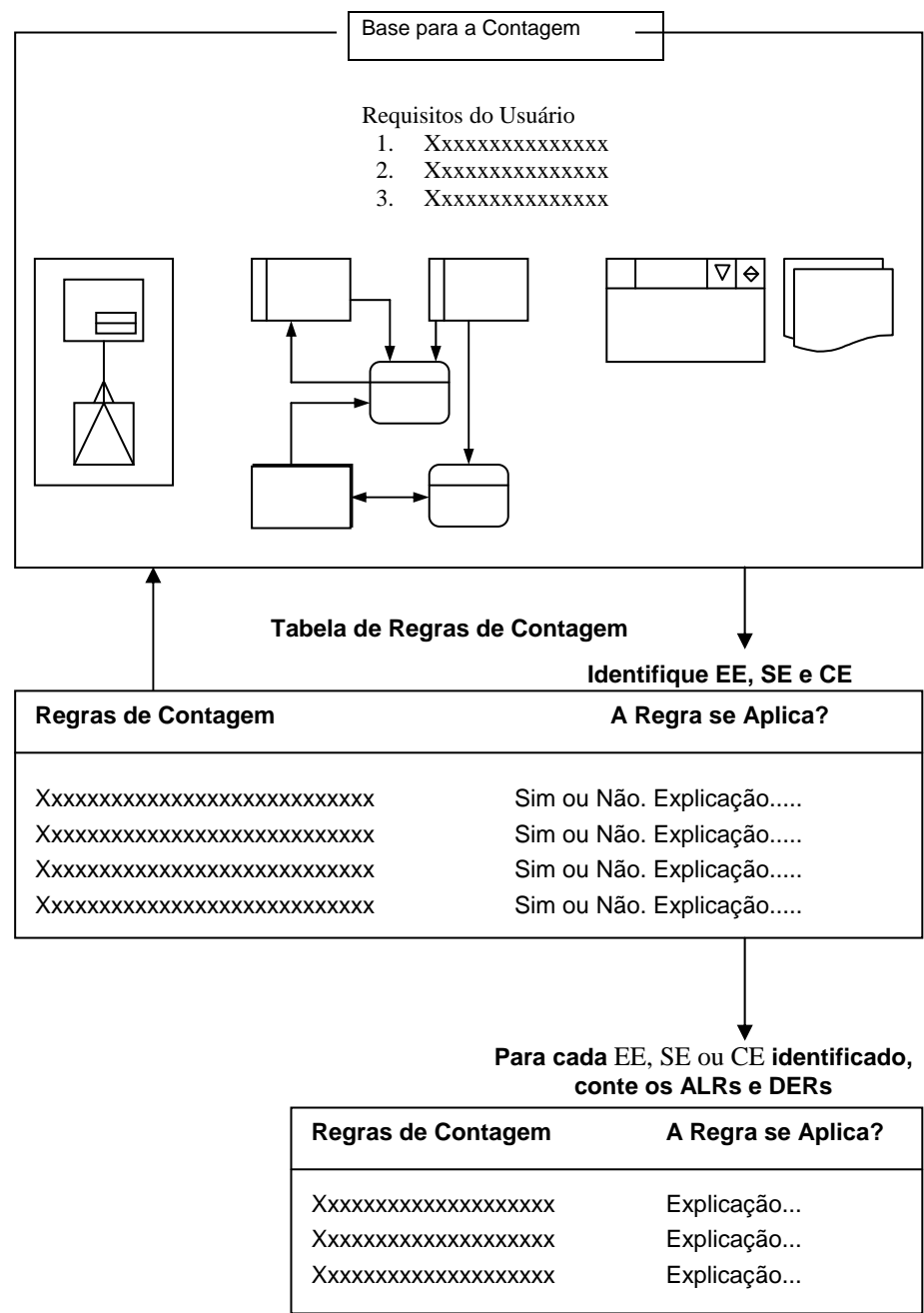
Componentes de Contagem de cada Exemplo

Cada exemplo inclui os seguintes componentes:

- As bases para a contagem
- Tabelas de regras de contagem que foram aplicadas

Diagrama de Componentes

O diagrama abaixo ilustra os componentes de cada exemplo e o fluxo de informação.



Base para a Contagem

Cada exemplo inicia com a definição da base para a contagem. Conforme apresentado no diagrama de componentes, a medição do tamanho funcional pode ser baseada nos seguintes componentes incluídos nos exemplos:

- Requisitos do usuário
- Modelo de dados e de processos
- Janelas, telas ou relatórios

Nota: Nem todos os componentes acima estão incluídos nos diagramas de todos os exemplos. Em alguns exemplos, os requisitos são suficientes para serem utilizados como base para a contagem. Já outros exemplos incluem um modelo de dados ou de processo, janelas, telas e/ou relatórios.

Tabelas de Regras de Contagem

A análise para identificar as funções é apresentada em uma tabela que contém a lista de regras de contagem por tipo de função. As regras são aplicadas aos componentes que compõem a base para a contagem. A análise realizada é descrita na coluna “A Regra se Aplica?” da tabela de regras de contagem.

Nota: Se todas as regras se aplicarem, o exemplo será contado como uma EE, SE ou CE.

As tabelas apresentadas ao longo do capítulo mostram as regras aplicadas e suas respectivas explicações para os tipos que determinam a complexidade de cada tipo de função identificada.

Regras Gerais dos Procedimentos de Contagem das Funções de Transação

O processo de análise de todos os exemplos segue os procedimentos descritos no início deste capítulo. Os passos do processo estão relacionados à aplicação das regras para medir as funções de transação conforme definido na Parte 1, e inclui os itens a seguir:

- Identificar cada Processo Elementar solicitado pelo usuário
- Determinar Processos Elementares únicos
- Classificar cada Processo Elementar como uma Entrada Externa (EE), Saída Externa (SE), ou uma Consulta Externa (CE)
- Contar os Arquivos Lógicos Referenciados (ALRs) para cada função de transação
- Contar os Dados Elementares Referenciados (DETs) para cada função de transação
- Determinar a complexidade funcional para cada função de transação
- Determinar o tamanho funcional de cada função de transação.

Esta página foi deixada intencionalmente em branco.

Exemplos de Identificação de Processo Elementar

Introdução Esta seção utiliza diversos exemplos para ilustrar os procedimentos de identificação de processos elementares, de acordo com as características específicas de cada um.

Nota: Cada exemplo apresenta apenas o requisito específico à situação ilustrada, apesar de na prática ser necessária a avaliação de todos os requisitos e seus respectivos impactos funcionais. Ocasionalmente, são feitas algumas referências às funções relacionadas que também existem, mas não estão ilustradas (i.e., Incluir Funcionário e Atualizar Funcionário).

Conteúdo Esta seção inclui os seguintes exemplos:

Tópico	Página
Descrição Geral dos Exemplos de Contagem de Identificação de Processo Elementar	2-8
Exemplo: Dados de um Novo Funcionário / Dependentes	2-9
Exemplo: Imprimir um Cheque / Marcá-lo como Pago	2-16
Exemplo: Exibir Lista das Funções Assinaladas	2-21
Exemplo: Imprimir as Funções Assinaladas / Salvar os Critérios Seleccionados	2-26
Exemplo: Funcionário - Informações da Entrevista	2-31
Exemplo: Funcionário - Informações sobre a Carteira de Habilitação	2-36
Exemplo: Processamento Batch de Carga de Dados dos Funcionário	2-41
Exemplo: Assinalar um Funcionário a uma Função	2-47
Exemplo: Processos Elementares Similares	2-57

Descrição Geral dos Exemplos de Contagem de Identificação de Processo Elementar

Os exemplos de identificação de processos elementares estão descritos na tabela abaixo:

Exemplo	Descrição Resumida	Página
Dados de um Novo Funcionário/ Dependentes	Este exemplo mostra como múltiplos processos podem compor um processo elementar.	2-9
Imprimir um Cheque / Marcá-lo como Pago	Este exemplo ilustra o conceito sobre a intenção primária de um processo elementar.	2-16
Exibir Lista das Funções Assinaladas	Este exemplo mostra que a entrada dos critérios de seleção de um relatório não é um processo elementar.	2-21
Imprimir as Funções Assinaladas / Salvar os Critérios Seleccionados	Este exemplo mostra explicitamente que salvar os critérios seleccionados para uso posterior é um processo elementar separado.	2-26
Funcionário - Informações da Entrevista	Este é outro exemplo de múltiplos processos que compoem um processo elementar.	2-31
Funcionário - Informações sobre a Carteira de Habilitação	Este é um terceiro exemplo de múltiplos processos que compoem um processo elementar.	2-36
Processamento <i>Batch</i> de Carga de Dados dos Funcionários	Este exemplo ilustra como relatórios de erro e relatórios estatísticos produzidos a partir do resultado de um processamento <i>Batch</i> não são considerados processos elementares separados.	2-41
Assinalar um Funcionário a uma Função	Este exemplo ilustra a avaliação de processos elementares similares para determinar se são ou não únicos.	2-47
Processos Elementares Similares	Este exemplo mostra como dois Processos Elementares similares são contados como transações únicas.	2-57

Exemplo: Dados de um Novo Funcionário / Dependentes

Requisitos do Usuário Ao incluir um novo funcionário, é solicitado ao usuário que entre com:

1. dados de identificação (básicos) do funcionário e
2. informação dos dependentes, caso o número de dependentes seja maior do que zero.

O arquivo de transação é criado durante a atualização da informação do funcionário. Este arquivo de transação é enviado periodicamente (i.e., no final do dia) para o Sistema de Benefícios.

Nota: A atualização da informação dos dependentes de funcionários existentes não está incluída neste exemplo. Para este processo, devem-se consultar os Estudos de Casos 1-3 onde está ilustrado um exemplo de contagem de processos de atualização.

Inclusão do Funcionário sem a Informação dos Seus Dependentes

Determinar se a inclusão das informações básicas de um funcionário sem a informação dos dependentes é um processo elementar ou não. A análise realizada está ilustrada na tabela a seguir:

Identificar o Processo Elementar	A Regra se Aplica?
<p>1. Compor e/ou decompor os Requisitos Funcionais do Usuário na menor unidade de uma atividade, que satisfaça todos os critérios abaixo:</p> <ul style="list-style-type: none"> • É significativo para o usuário • Constitui uma transação completa • É auto-contido e • Deixa o negócio da aplicação que está sendo contada em um estado consistente. 	<p>Sim. Incluir um funcionário é parte dos requisitos funcionais do usuário.</p> <p>Não. A transação completa inclui a inclusão da informação do funcionário com a informação dos dependentes associada, caso o número de dependentes seja maior do que zero. Esses passos não podem ser separados logicamente.</p> <p>Não. Incluir a informação dos dependentes associada ao funcionário incluído é um passo subsequente necessário para completar o processo elementar.</p> <p>Não. Incluir o funcionário sem adicionar os dependentes não deixa o negócio da aplicação em um estado consistente. Para mantê-lo em estado consistente, o requisito funcional do usuário deve ser satisfeito sem que haja mais nada a ser feito. Para satisfazer o requisito funcional do usuário, o dependente precisa ser incluído.</p>
<p>2. Identificar um processo elementar para cada unidade de atividade identificada que satisfaça todos os critérios acima.</p>	<p>Incluir um funcionário (sem incluir as informações dos dependentes) não satisfaz todos os critérios.</p>

Conclusão

Incluir um funcionário sem as informações dos dependentes não satisfaz os requisitos de um processo elementar.

Inclusão apenas das Informações Dos Dependentes

Determinar se a inclusão das informações dos dependentes sem incluir a informação dos funcionários é um processo elementar ou não. A análise realizada está ilustrada na tabela a seguir:

Identificar o Processo Elementar	A Regra se Aplica?
<p>1. Compor e/ou decompor os Requisitos Funcionais do Usuário na menor unidade de uma atividade, que satisfaça todos os critérios abaixo:</p> <ul style="list-style-type: none"> • É significativo para o usuário • Constitui uma transação completa • É auto-contido e • Deixa o negócio da aplicação que está sendo contada em um estado consistente. 	<p>Sim. Incluir um dependente é parte dos requisitos funcionais do usuário.</p> <p>Não. Esta atividade não é uma transação completa porque não pode ser executada independente da inclusão de um funcionário.</p> <p>Não. Esta atividade não é auto-contida porque não pode ser executada independentemente da inclusão de um funcionário.</p> <p>Não. Incluir um dependente sem incluir um funcionário não deixa o negócio da aplicação em um estado consistente. Para mantê-lo em estado consistente, o requisito funcional do usuário deve ser satisfeito sem que haja mais nada a ser feito. Para satisfazer o requisito funcional do usuário, tanto funcionário como dependente precisam ser incluídos.</p>
<p>2. Identificar um processo elementar para cada unidade de atividade identificada que satisfaça todos os critérios acima.</p>	<p>Incluir um dependente (sem o respectivo funcionário) não satisfaz a todos os critérios. Por isto, não é um processo elementar.</p>

Conclusão

Incluir somente as informações de dependentes não satisfaz os requisitos de um processo elementar. Neste exemplo, incluir um funcionário (sem incluir as informações dos dependentes) não satisfaz todos os critérios. Outros sistemas podem manter a informação dos dependentes independentemente da dos funcionários.

Inclusão de um Funcionário com as Informações dos Dependentes

Para um funcionário que possui dependente, determinar se a inclusão das informações do funcionário com a informação dos seus respectivos dependentes é um processo elementar ou não. A análise realizada está ilustrada na tabela a seguir:

Identificar o Processo Elementar	A Regra se Aplica?
<p>1. Compor e/ou decompor os Requisitos Funcionais do Usuário na menor unidade de uma atividade, que satisfaça todos os critérios abaixo:</p> <ul style="list-style-type: none"> • É significativo para o usuário • Constitui uma transação completa • É auto-contido e • Deixa o negócio da aplicação que está sendo contada em um estado consistente. 	<p>Sim. Incluir um funcionário com a informação dos dependentes é parte dos requisitos funcionais do usuário.</p> <p>Sim. Juntas, as informações do funcionário e de seus dependentes são utilizadas para incluir um novo funcionário no sistema de RH. Esses passos não podem ser separados logicamente.</p> <p>Sim. Esta atividade é significativa e auto-contida e toda informação necessária é incluída na aplicação de RH.</p> <p>Sim. O negócio é deixado em um estado consistente quando o funcionário é incluído, assim como os dependentes, quando aplicável.</p>
<p>2. Identificar um processo elementar para cada unidade de atividade identificada que satisfaça todos os critérios acima.</p>	<p>Incluir um Funcionário com as Informações dos Dependentes satisfaz todos os critérios acima.</p>

Conclusão

Incluir um Funcionário com as Informações dos Dependentes satisfaz os requisitos de um processo elementar.

Poderia se adotar implementações diferentes do requisito de adicionar dependentes a um funcionário, como por exemplo:

- um campo de entrada de dados chamado Número de Dependentes dentro da tela de funcionários que permite a exibição da tela de dependentes
- um botão de comando que exibe a tela de dependentes
- um item de menu na tela de funcionários que exibe a tela de dependentes
- a possibilidade de incluir os dependentes na tela de inclusão dos funcionários

Independente da implementação adotada para este requisito, existe um processo elementar, de inclusão de um funcionário com os dependentes.

Não foram identificados processos elementares separados em função de dados opcionais, como por exemplo: Incluir Funcionário com dependentes, Incluir Funcionário sem dependentes.

Determinar Processos Elementares Únicos

A tabela abaixo mostra a análise que foi feita para determinar se o processo elementar é único ou não:

Determinar Processos Elementares Únicos	A Regra se Aplica?
<p>1. Quando comparado com um Processo Elementar já identificado, contar dois Processos Elementares similares como o mesmo Processo Elementar, se ambos:</p> <ul style="list-style-type: none"> • Requerem o mesmo conjunto de DERs e • Requerem o mesmo conjunto de ALRs e • Requerem o mesmo conjunto de lógica de processamento para completar o processo elementar 	<p>Sim. Incluir um Funcionário requer o mesmo conjunto de DERs que Atualizar um Funcionário.</p> <p>Sim. Incluir um Funcionário requer o mesmo conjunto de ALRs que Atualizar Funcionário.</p> <p>Não. Incluir Funcionário tem um conjunto diferente de lógica de processamento em relação ao processo Atualizar um Funcionário.</p>
<p>2. Não dividir um processo elementar com múltiplas formas de lógica de processamento em múltiplos processos elementares.</p>	<p>Os requisitos funcionais do usuário consideram que há uma única função. A função Incluir um Funcionário não foi dividida em dois processos elementares (i.e., Incluir Apenas Funcionário, Incluir Apenas os Dependentes).</p>

Conclusão

Incluir um Funcionário é um processo elementar único em relação a todos os outros processos elementares que já foram identificados.

Enviar Arquivo de Transação para o Sistema de Benefícios

Determinar se o envio de arquivos de transação para o Sistema de Benefícios é um processo elementar adicional. A análise realizada está ilustrada na tabela a seguir:

Identificar o Processo Elementar	A Regra se Aplica?
<p>1. Compor e/ou decompor os Requisitos Funcionais do Usuário na menor unidade de uma atividade, que satisfaça todos os critérios abaixo:</p> <ul style="list-style-type: none"> • É significativo para o usuário • Constitui uma transação completa • É auto-contido e • Deixa o negócio da aplicação que está sendo contada em um estado consistente. 	<p>Sim. Enviar o arquivo de transação para o Sistema de Benefícios faz parte dos requisitos funcionais do usuário.</p> <p>Sim. Enviar o arquivo de transação para o Sistema de Benefícios é uma transação completa. É uma função logicamente separada da função Incluir Funcionário.</p> <p>Sim. A atividade de enviar as transações para o Sistema de Benefícios é auto-contida. O Arquivo de Transação é enviado independentemente (i.e., no final do dia) da função Incluir Funcionário.</p> <p>Sim. Após as transações serem enviadas para o Sistema de Benefícios, o negócio da aplicação de RH fica em um estado consistente. O requisito funcional do usuário foi totalmente satisfeito sem que haja necessidade de que algo a mais seja feito.</p>
<p>2. Identificar um processo elementar para cada unidade de atividade identificada que satisfaça todos os critérios acima.</p>	<p>Enviar Arquivo de Transação para o Sistema de Benefícios satisfaz todos os critérios acima.</p>

Conclusão

Enviar Arquivo de Transação para o Sistema de Benefícios satisfaz os requisitos de um processo elementar.

Determinar Processos Elementares Únicos

A tabela abaixo mostra a análise realizada para determinar se o processo elementar é único ou não:

Determinar Processos Elementares Únicos	A Regra se Aplica?
<p>1. Quando comparado com um Processo Elementar já identificado, contar dois Processos Elementares similares como o mesmo Processo Elementar, se ambos:</p> <ul style="list-style-type: none"> • Requerem o mesmo conjunto de DERs e • Requerem o mesmo conjunto de ALRs e • Requerem o mesmo conjunto de lógica de processamento para completar o processo elementar 	<p>Não. Enviar o arquivo de transação para o Sistema de Benefícios requer DERs diferentes de outros Processos Elementares identificados.</p> <p>Sim. Enviar o arquivo de transação para o Sistema de Benefícios requer o mesmo conjunto de ALRs que Incluir um Funcionário.</p> <p>Não. Enviar o arquivo de transação para o Sistema de Benefícios tem um conjunto de lógica de processamento diferente dos outros processos elementares identificados.</p>
<p>2. Não separar um processo elementar com múltiplas formas de lógicas de processamento em múltiplos processos elementares.</p>	<p>Não há nada a ser separado.</p>

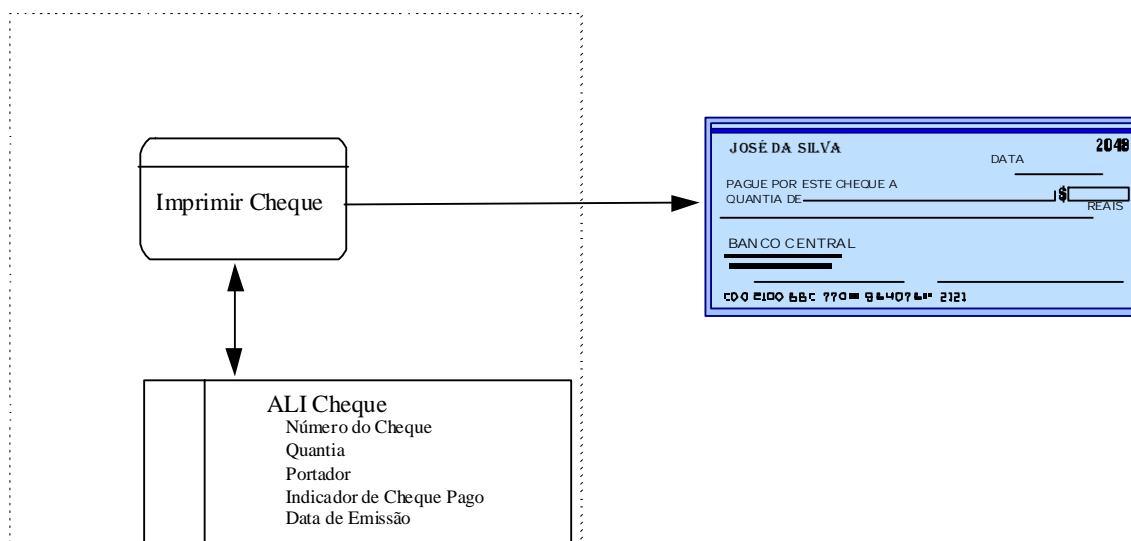
Conclusão

Enviar o Arquivo de Transação para o Sistema de Benefícios é um processo elementar único em relação a todos os outros processos elementares que já foram identificados.

Exemplo: Imprimir um Cheque / Marcá-lo como Pago

Requisitos do Usuário Imprimir um cheque e, como resultado, marcá-lo na conta corrente como pago. Todos os dados impressos no cheque já estão armazenados no arquivo de Cheques.

O diagrama abaixo mostra o fluxo de dados deste exemplo:



Marcar na Conta como Cheque Pago

Determinar se marcar na conta corrente como cheque pago é um processo elementar ou não. A análise realizada está ilustrada na tabela a seguir:

Identificar o Processo Elementar	A Regra se Aplica?
<p>1. Compor e/ou decompor os Requisitos Funcionais do Usuário na menor unidade de uma atividade, que satisfaça todos os critérios abaixo:</p> <ul style="list-style-type: none"> • É significativo para o usuário • Constitui uma transação completa • É auto-contido e • Deixa o negócio da aplicação que está sendo contada em um estado consistente. 	<p>Sim. Marcar na Conta como Cheque Pago faz parte dos requisitos funcionais do usuário.</p> <p>Não. Marcar na Conta como Cheque Pago não representa uma transação completa, a menos que o cheque seja também impresso. Esses passos não podem ser separados logicamente.</p> <p>Não. Marcar na Conta como Cheque Pago sem imprimí-lo não é uma função auto-contida. O cheque não pode ser marcado como pago na conta corrente independentemente da impressão do mesmo.</p> <p>Não. Marcar na Conta como Cheque Pago sem imprimí-lo não deixa o negócio em um estado consistente. Para mantê-lo em estado consistente, o requisito funcional do usuário deve ser satisfeito sem que haja mais nada a ser feito. Para satisfazer o requisito funcional do usuário, a função de marcar na conta como cheque pago necessita ocorrer quando o cheque é impresso.</p>
<p>2. Identificar um processo elementar para cada unidade de atividade identificada que satisfaça todos os critérios acima.</p>	<p>Marcar na Conta como Cheque Pago não satisfaz a todos os critérios. Por isto, não é um processo elementar.</p>

Conclusão

Marcar na Conta como Cheque Pago não satisfaz os requisitos de um processo elementar.

Imprimir um Cheque

Determinar se imprimir um cheque sem marcar na conta como pago é um processo elementar ou não. A análise realizada está conforme tabela abaixo:

Identificar o Processo Elementar	A Regra se Aplica?
<p>1. Compor e/ou decompor os Requisitos Funcionais do Usuário na menor unidade de uma atividade, que satisfaça todos os critérios abaixo:</p> <ul style="list-style-type: none"> • É significativo para o usuário • Constitui uma transação completa • É auto-contido e • Deixa o negócio da aplicação que está sendo contada em um estado consistente. 	<p>Sim. Imprimir um Cheque faz parte dos requisitos funcionais do usuário.</p> <p>Não. Esta atividade não é uma transação completa, a menos que o cheque seja marcado como pago na Conta Corrente. Esses passos não podem ser separados logicamente.</p> <p>Não. Imprimir um Cheque não é uma função auto-contida até que o passo sub-sequente de marcar o cheque como pago na conta corrente seja executado.</p> <p>Não. Imprimir um Cheque sem marcá-lo como pago na conta corrente não deixa o negócio em um estado consistente. Para mantê-lo em estado consistente, o requisito funcional do usuário deve ser satisfeito sem que haja mais nada a ser feito. Para satisfazer o requisito funcional do usuário, o cheque precisa ser marcado como pago na conta corrente.</p>
<p>2. Identificar um processo elementar para cada unidade de atividade identificada que satisfaça todos os critérios acima.</p>	<p>Imprimir um Cheque não satisfaz a todos os critérios. Por isto, não é um processo elementar.</p>

Conclusão

Imprimir um Cheque não satisfaz a os requisitos de um processo elementar.

Imprimir um Cheque e Marcá-lo como Pago na Conta Corrente

Determinar se imprimir um cheque e marcá-lo como pago na conta corrente é um processo elementar ou não. A análise realizada está ilustrada na tabela a seguir:

Identificar o Processo Elementar	A Regra se Aplica?
<p>1. Compor e/ou decompor os Requisitos Funcionais do Usuário na menor unidade de uma atividade, que satisfaça todos os critérios abaixo:</p> <ul style="list-style-type: none"> • É significativo para o usuário • Constitui uma transação completa • É auto-contido e • Deixa o negócio da aplicação que está sendo contada em um estado consistente. 	<p>Sim. Imprimir um Cheque e Marcá-lo como Pago na Conta Corrente fazem parte dos requisitos funcionais do usuário.</p> <p>Sim. Imprimir um Cheque e Marcá-lo como Pago na Conta Corrente constitui uma transação completa. Esses passos não podem ser separados logicamente.</p> <p>Sim. Imprimir um Cheque e Marcá-lo como Pago na Conta Corrente é auto-contido. Não existe nenhuma necessidade de se executar passos anteriores ou sub-sequentes.</p> <p>Sim. Imprimir um Cheque e Marcá-lo como Pago na Conta Corrente deixa o negócio em um estado consistente. O requisito funcional do usuário é totalmente cumprido.</p>
<p>2. Identificar um processo elementar para cada unidade de atividade identificada que satisfaça todos os critérios acima.</p>	<p>Imprimir um Cheque e Marcá-lo como Pago na Conta Corrente satisfaz todos os critérios acima. Portanto, é um processo elementar.</p>

Conclusão

Imprimir um Cheque e Marcá-lo como Pago na Conta Corrente satisfaz os requisitos de um processo elementar.

O requisito do usuário é imprimir o cheque. Marcá-lo como pago na Conta Corrente é parte do processo de impressão do cheque. As funções Imprimir e Marcar juntas correspondem à menor unidade da atividade que é significativa para o usuário. O processo como um todo é significativo para o usuário, constitui uma transação completa, é auto-contido e deixa o negócio da aplicação que está sendo contada em um estado consistente.

Determinar Processos Elementares Únicos A tabela abaixo mostra a análise realizada para determinar se o processo elementar é único ou não.

Determinar Processos Elementares Únicos	A Regra se Aplica?
<p>1. Quando comparado com um Processo Elementar já identificado, contar dois Processos Elementares similares como o mesmo Processo Elementar, se ambos:</p> <ul style="list-style-type: none"> • Requerem o mesmo conjunto de DERs e • Requerem o mesmo conjunto de ALRs e • Requerem o mesmo conjunto de lógica de processamento para completar o processo elementar 	<p>Não. O conjunto de DERs utilizado para Imprimir um Cheque e de Marcá-lo como pago é diferente de qualquer outro processo elementar.</p> <p>Não. O conjunto de ALRs utilizado para Imprimir um Cheque e de Marcá-lo como pago é diferente de qualquer outro processo elementar.</p> <p>Não. O conjunto de lógica de processamento utilizado para Imprimir um Cheque e de Marcá-lo como pago é diferente de qualquer outro processo elementar.</p>
<p>2. Não separar um processo elementar com múltiplas formas de lógicas de processamento em múltiplos processos elementares.</p>	<p>Marcar um cheque como pago na Conta Corrente é parte da lógica de processamento da impressão de um cheque.</p>

Conclusão Imprimir um Cheque e Marcá-lo como Pago na Conta Corrente é um processo elementar único em relação a todos os outros processos elementares que já foram identificados.

Exemplo: Exibir Lista das Funções Assinaladas

Requisitos do Usuário Exibir uma lista das funções assinaladas para um determinado intervalo de datas. O usuário estará habilitado a entrar com os critérios de seleção. Não existe nenhum requisito para armazenar os critérios selecionados uma vez que o relatório tenha sido impresso. O diagrama abaixo mostra o fluxo de dados deste exemplo:

Critérios da Lista das Funções Assinaladas	
Código Funcionário	<input type="text"/>
Data Inicial	<input type="text"/>
Data Final	<input type="text"/>
	<input type="button" value="Imprimir"/>
	<input type="button" value="Cancelar"/>

Entrar com os Critérios de Seleção Determinar se a entrada dos critérios de seleção (sem exibir a lista das funções assinaladas) é um processo elementar ou não. A análise realizada está ilustrada na tabela a seguir:

Identificar o Processo Elementar	A Regra se Aplica?
<p>1. Compor e/ou decompor os Requisitos Funcionais do Usuário na menor unidade de uma atividade, que satisfaça todos os critérios abaixo:</p> <ul style="list-style-type: none"> • É significativo para o usuário • Constitui uma transação completa • É auto-contido e • Deixa o negócio da aplicação que está sendo contada em um estado consistente. 	<p>Sim. Entrar com os Critérios de Seleção faz parte dos requisitos funcionais do usuário.</p> <p>Não. Entrar com os Critérios de Seleção (sem exibir a lista das funções assinaladas) não é uma transação completa. Esses passos não podem ser separados logicamente.</p> <p>Não. Entrar com os Critérios de Seleção (sem exibir a lista das funções assinaladas) não é auto-contido. Exibir Lista das Funções Assinaladas é um passo sub-sequente do processo que é necessário para completar o processo elementar.</p> <p>Não. Entrar com os Critérios de Seleção (sem exibir a lista das funções assinaladas) não deixa o negócio em um estado consistente. Para mantê-lo em estado consistente, o requisito funcional do usuário deve ser satisfeito sem que haja mais nada a ser feito. Para satisfazer o requisito funcional do usuário, a lista das funções assinaladas precisa ser exibida.</p>
<p>2. Identificar um processo elementar para cada unidade de atividade identificada que satisfaça todos os critérios acima.</p>	<p>Entrar com os Critérios de Seleção (sem exibir a lista das funções assinaladas) não satisfaz os critérios acima.</p>

Conclusão Entrar com os Critérios de Seleção não satisfaz os requisitos de um processo elementar.

Visão das Funções Assinaladas

Determinar se a exibir a lista das funções assinaladas (sem entrar com os critérios de seleção) é um processo elementar ou não. A análise realizada está ilustrada na tabela a seguir:

Identificar o Processo Elementar	A Regra se Aplica?
<p>1. Compor e/ou decompor os Requisitos Funcionais do Usuário na menor unidade de uma atividade, que satisfaça todos os critérios abaixo:</p> <ul style="list-style-type: none"> • É significativo para o usuário • Constitui uma transação completa • É auto-contido e • Deixa o negócio da aplicação que está sendo contada em um estado consistente. 	<p>Sim. Exibir a lista das funções assinaladas faz parte dos requisitos funcionais do usuário.</p> <p>Não. Exibir a lista das funções assinaladas (sem entrar com os critérios de seleção) não pode ser executado sem que os critérios de seleção sejam informados. Esses passos não podem ser separados logicamente.</p> <p>Não. Exibir a lista das funções assinaladas (sem entrar com os critérios de seleção) não é auto-contido. A lista das funções assinaladas não pode ser exibida independentemente do passo anterior que corresponde à entrada dos critérios de seleção.</p> <p>Não. Exibir a lista das funções assinaladas (sem entrar com os critérios de seleção) não deixa o negócio em um estado consistente. Para mantê-lo em estado consistente, o requisito funcional do usuário deve ser satisfeito sem que haja mais nada a ser feito. Para satisfazer o requisito funcional do usuário, os critérios de seleção precisam ser informados.</p>
<p>2. Identificar um processo elementar para cada unidade de atividade identificada que satisfaça todos os critérios acima.</p>	<p>Exibir a lista das funções assinaladas (sem entrar com os critérios de seleção) não satisfaz os critérios acima.</p>

Conclusão

Exibir a lista das funções assinaladas (sem entrar com os critérios de seleção) não satisfaz os requisitos de um processo elementar.

Entrada dos Critérios de Seleção e Exibição das Funções Assinaladas

Determinar se a exibição da lista de funções assinaladas com a entrada dos critérios de seleção é um processo elementar ou não. A análise realizada está ilustrada na tabela a seguir:

Identificar o Processo Elementar	A Regra se Aplica?
<p>1. Compor e/ou decompor os Requisitos Funcionais do Usuário na menor unidade de uma atividade, que satisfaça todos os critérios abaixo:</p> <ul style="list-style-type: none"> • É significativo para o usuário • Constitui uma transação completa • É auto-contido e • Deixa o negócio da aplicação que está sendo contada em um estado consistente. 	<p>Sim. Entrar com os Critérios de Seleção e Exibir a Lista de Funções Assinaladas faz parte dos requisitos funcionais do usuário.</p> <p>Sim. Entrar com os Critérios de Seleção e Exibir a Lista de Funções Assinaladas é uma transação completa. Esses passos não podem ser separados logicamente.</p> <p>Sim. Entrar com os Critérios de Seleção e Exibir a Lista de Funções Assinaladas é uma transação auto-contida. Não existe necessidade de se executar passos anteriores ou sub-sequentes</p> <p>Sim. Entrar com os Critérios de Seleção e Exibir a Lista de Funções Assinaladas deixa o negócio em um estado consistente. Para mantê-lo em estado consistente, o requisito funcional do usuário deve ser satisfeito sem que haja mais nada a ser feito. Para satisfazer o requisito funcional do usuário, os critérios de seleção precisam ser informados e a lista exibida.</p>
<p>2. Identificar um processo elementar para cada unidade de atividade identificada que satisfaça todos os critérios acima.</p>	<p>Entrar com os Critérios de Seleção e Exibir a Lista de Funções Assinaladas satisfaz todos os critérios acima.</p>

Conclusão

Entrar com os Critérios de Seleção e Exibir a Lista de Funções Assinaladas satisfaz aos critérios de um processo elementar.

Informação de Controle é a entrada de uma SE ou CE. A solicitação específica de qual e/ou como o dado será recuperado ou gerado faz parte do processo elementar de fornecer os dados do usuário e não do próprio processo elementar.

Entrar com os critérios de seleção não é a menor unidade de uma atividade que seja significativa para o usuário. Não é auto-contida porque não pode ser executada independentemente de gerar o relatório. Entrar com os critérios de seleção e gerar o relatório juntos corresponde à menor unidade de atividade que é significativa para o usuário, constitui uma transação completa, é auto-contida e deixa o negócio da aplicação que está sendo contada em um estado consistente.

Determinar Processos Elementares Únicos

A tabela abaixo mostra a análise que foi executada para determinar se o processo elementar é único ou não:

Determinar Processos Elementares Únicos	A Regra se Aplica?
<p>1. Quando comparado com um Processo Elementar já identificado, contar dois Processos Elementares similares como o mesmo Processo Elementar, se ambos:</p> <ul style="list-style-type: none"> • Requerem o mesmo conjunto de DERs e • Requerem o mesmo conjunto de ALRs e • Requerem o mesmo conjunto de lógica de processamento para completar o processo elementar 	<p>Não. O conjunto de DERs utilizado para Entrar com os critérios de seleção e exibir a lista de funções assinaladas é diferente de qualquer outro processo elementar.</p> <p>Não. O conjunto de ALRs utilizado para Entrar com os critérios de seleção e exibir a lista de funções assinaladas é diferente de qualquer outro processo elementar.</p> <p>Não. O conjunto de lógicas de processamento utilizado para entrar com os critérios de seleção e exibir a lista de funções assinaladas é diferente de qualquer outro processo elementar.</p>
<p>2. Não separar um processo elementar com múltiplas formas de lógicas de processamento em múltiplos processos elementares.</p>	<p>É necessário entrar com os critérios de seleção para ver a lista de funções assinaladas.</p>

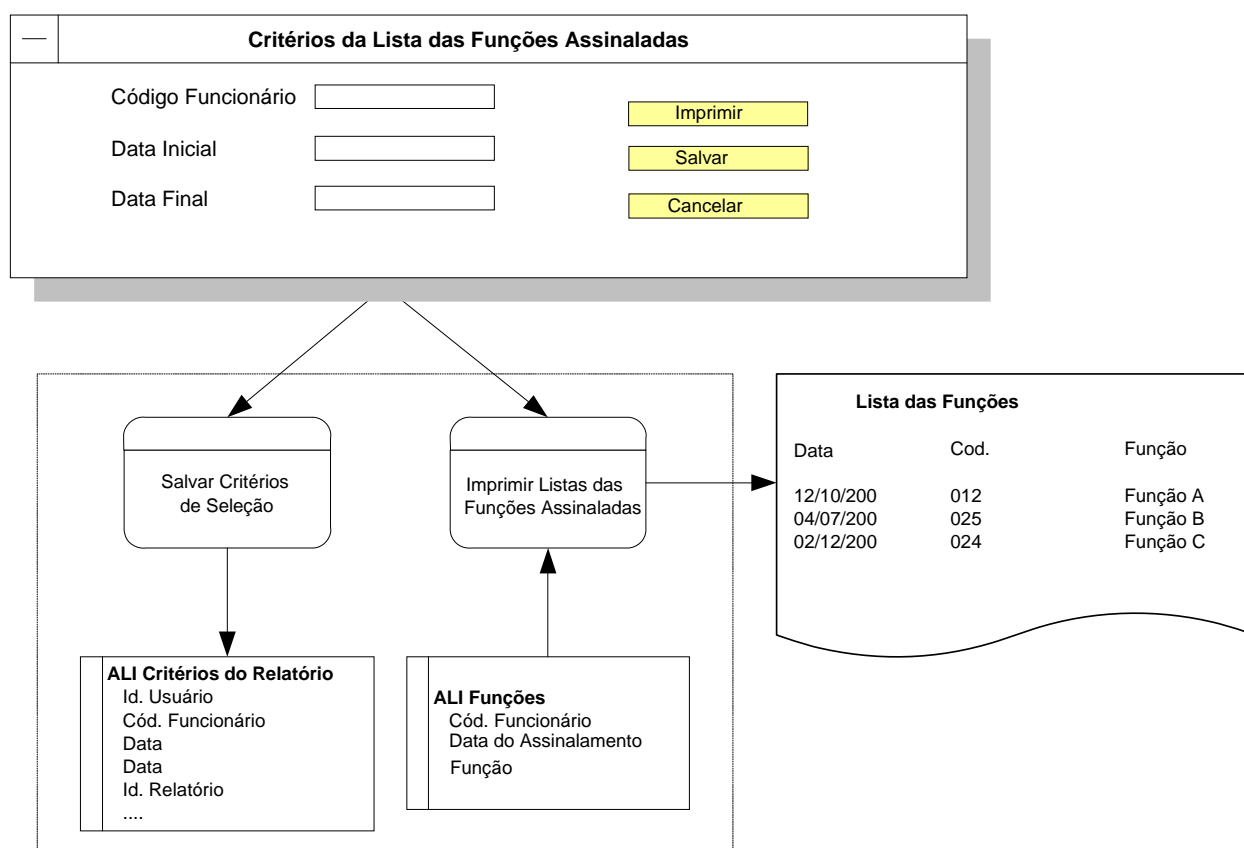
Conclusão

Entrar com os Critérios de Seleção e Exibir a Lista de Funções Assinaladas é um processo elementar único em relação a todos os outros processos elementares que já foram identificados.

Exemplo: Imprimir as Funções Assinaladas / Salvar os Critérios Seleccionados

Requisitos do Usuário Imprimir uma lista das funções assinaladas para um determinado intervalo de datas. O usuário estará habilitado a entrar com os critérios de seleção. Existe um requisito que permite ao usuário armazenar os critérios seleccionados para uso posterior.

O diagrama abaixo mostra o fluxo de dados deste exemplo:



Entrar e Salvar os Critérios de Seleção Informados Determinar se salvar os critérios de seleção informados (sem imprimir a lista de funções assinaladas) é um processo elementar ou não. A análise realizada está ilustrada na tabela a seguir:

Identificar o Processo Elementar	A Regra se Aplica?
<p>1. Compor e/ou decompor os Requisitos Funcionais do Usuário na menor unidade de uma atividade, que satisfaça todos os critérios abaixo:</p> <ul style="list-style-type: none"> • É significativo para o usuário • Constitui uma transação completa • É auto-contido e • Deixa o negócio da aplicação que está sendo contada em um estado consistente. 	<p>Sim. Salvar os critérios de seleção faz parte dos requisitos funcionais do usuário.</p> <p>Sim. Salvar os critérios de seleção é uma transação completa. É uma função separada logicamente da função de impressão da lista de funções assinaladas.</p> <p>Sim. Salvar os critérios de seleção é uma transação auto-contida. Pode ser executada independentemente de imprimir a lista de funções assinaladas.</p> <p>Sim. Salvar os critérios de seleção deixa o negócio em um estado consistente. O requisito funcional do usuário está totalmente satisfeito sem que haja mais nada a ser feito.</p>
<p>2. Identificar um processo elementar para cada unidade de atividade identificada que satisfaça todos os critérios acima.</p>	<p>Salvar os critérios de seleção atende a todos os critérios acima.</p>

Conclusão Entrar e Salvar os Critérios de Seleção informados satisfaz os requisitos de um processo elementar.

Determinar Processos Elementares Únicos

A tabela abaixo mostra a análise realizada para determinar se o processo elementar é único ou não.

Determinar Processos Elementares Únicos	A Regra se Aplica?
<p>1. Quando comparado com um Processo Elementar já identificado, contar dois Processos Elementares similares como o mesmo Processo Elementar, se ambos:</p> <ul style="list-style-type: none"> • Requerem o mesmo conjunto de DERs e • Requerem o mesmo conjunto de ALRs e • Requerem o mesmo conjunto de lógica de processamento para completar o processo elementar 	<p>Não. O conjunto de DERs utilizado para Salvar os critérios de seleção informados é diferente de qualquer outro processo elementar.</p> <p>Não. O conjunto de ALRs utilizado para Salvar os critérios de seleção informados é diferente de qualquer outro processo elementar.</p> <p>Não. O conjunto de lógica de processamento utilizado para salvar os critérios de seleção informados é diferente de qualquer outro processo elementar.</p>
<p>2. Não separar um processo elementar com múltiplas formas de lógicas de processamento em múltiplos processos elementares.</p>	<p>Não há nada a ser separado.</p>

Conclusão

Salvar os Critérios de Seleção Informados é um processo elementar único em relação a todos os outros processos elementares que já foram identificados.

Imprimir a Lista de Funções Assinaladas Determinar se imprimir a lista de funções assinaladas, tendo o critério de seleção sido salvo ou não, é um processo elementar ou não. A análise realizada está ilustrada na tabela a seguir:

Identificar o Processo Elementar	A Regra se Aplica?
<p>1. Compor e/ou decompor os Requisitos Funcionais do Usuário na menor unidade de uma atividade, que satisfaça todos os critérios abaixo:</p> <ul style="list-style-type: none"> • É significativo para o usuário • Constitui uma transação completa • É auto-contido e • Deixa o negócio da aplicação que está sendo contada em um estado consistente. 	<p>Sim. Imprimir a lista de funções assinaladas faz parte dos requisitos funcionais do usuário.</p> <p>Sim. Imprimir a lista de funções assinaladas é uma transação completa. É uma função logicamente separada de salvar os critérios de seleção informados.</p> <p>Sim. Imprimir a lista de funções assinaladas é uma transação auto-contida. Pode ser executada independentemente de salvar os critérios de seleção informados.</p> <p>Sim. Imprimir a lista de funções assinaladas deixa o negócio em um estado consistente. O requisito funcional do usuário está totalmente satisfeito sem que haja mais nada a ser feito.</p>
<p>2. Identificar um processo elementar para cada unidade de atividade identificada que satisfaça todos os critérios acima.</p>	<p>Imprimir a lista de funções assinaladas satisfaz todos os critérios acima</p>

Conclusão Imprimir a Lista de Funções Assinaladas é um processo elementar.

Entrar com os critérios de seleção é significativo ao usuário porque os critérios podem ser salvos pelo mesmo para uso posterior. Imprimir a lista ou Salvar os critérios de Seleção informados podem ser executados independentemente, e ambos deixam o negócio em um estado consistente.

Ambos os processos, armazenar os critérios de seleção e gerar o relatório, são significativos ao usuário, constituem em transações completas, são auto-contidos e deixam o negócio da aplicação que está sendo contada em um estado consistente. De acordo com as Regras de Identificação de Processos Elementares, conclui-se que há dois processos elementares.

Determinar Processos Elementares Únicos A tabela abaixo mostra a análise realizada para determinar se o processo elementar é único ou não.

Determinar Processos Elementares Únicos	A Regra se Aplica?
<p>1. Quando comparado com um Processo Elementar já identificado, contar dois Processos Elementares similares como o mesmo Processo Elementar, se ambos:</p> <ul style="list-style-type: none"> • Requerem o mesmo conjunto de DERs e • Requerem o mesmo conjunto de ALRs e • Requerem o mesmo conjunto de lógica de processamento para completar o processo elementar 	<p>Não. O conjunto de DERs utilizado para imprimir a lista de funções assinaladas é diferente de qualquer outro processo elementar.</p> <p>Não. O conjunto de ALRs utilizado para imprimir a lista de funções assinaladas é diferente de qualquer outro processo elementar.</p> <p>Não. O conjunto de lógica de processamento utilizado para imprimir a lista de funções assinaladas é diferente de qualquer outro processo elementar.</p>
<p>2. Não separar um processo elementar com múltiplas formas de lógicas de processamento em múltiplos processos elementares.</p>	<p>Não há nada a ser separado.</p>

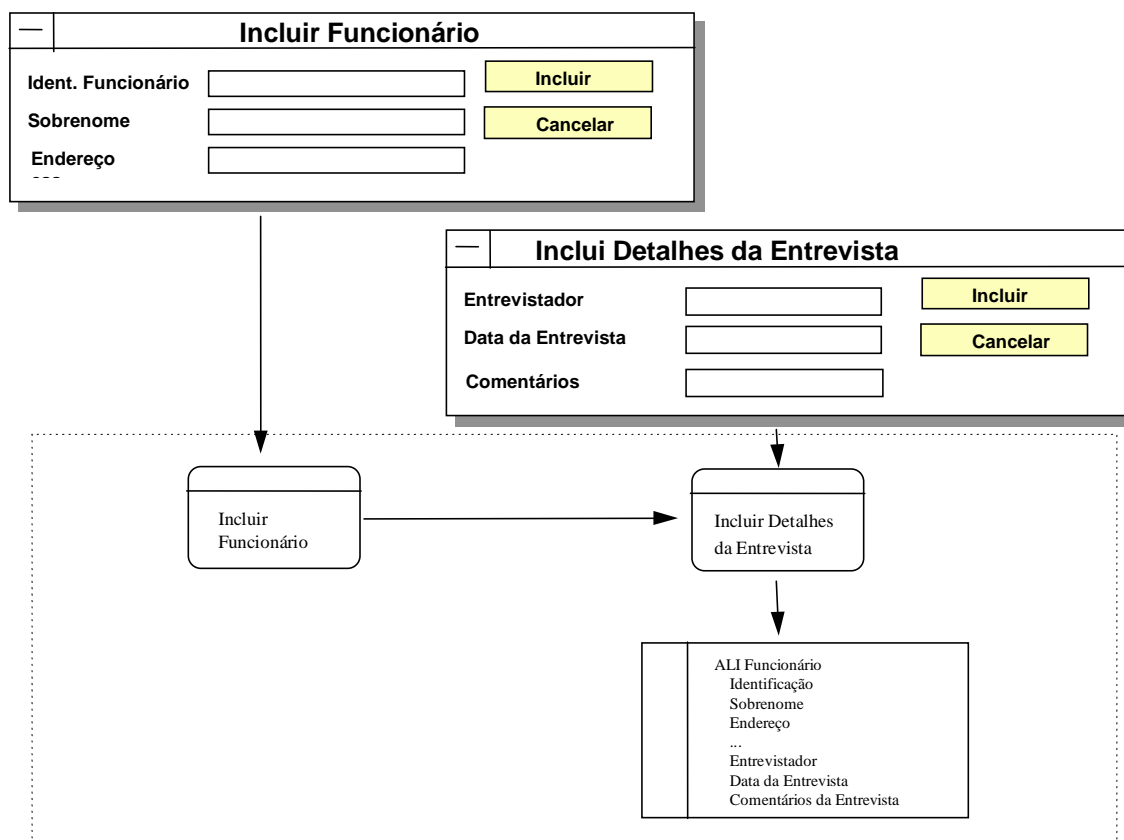
Conclusão Imprimir a Lista de Funções Assinaladas é um processo elementar único em relação a todos os outros processos elementares que já foram identificados.

Exemplo: Funcionário - Informações da Entrevista

Requisitos do Usuário

Quando um funcionário for incluído, além dos dados pessoais (i.e., Identificação, sobrenome, endereço, etc.), é necessário entrar com os detalhes da entrevista feita pelo funcionário. As informações da entrevista incluem o nome do entrevistador, a data da entrevista e os comentários do entrevistador em relação ao candidato.

O diagrama abaixo mostra o fluxo de dados deste exemplo:



Entrar com os Dados Pessoais do Funcionário Determinar se Entrar apenas com os dados pessoais do funcionário é um processo elementar ou não. A tabela abaixo ilustra a análise realizada:

Identificar o Processo Elementar	A Regra se Aplica?
<p>1. Compor e/ou decompor os Requisitos Funcionais do Usuário na menor unidade de uma atividade, que satisfaça todos os critérios abaixo:</p> <ul style="list-style-type: none"> • É significativo para o usuário • Constitui uma transação completa • É auto-contido e • Deixa o negócio da aplicação que está sendo contada em um estado consistente. 	<p>Sim. Entrar com os dados pessoais do funcionário faz parte dos requisitos funcionais do usuário.</p> <p>Não. A transação completa inclui tanto a entrada dos dados pessoais do funcionário quanto os detalhes da entrevista do funcionário. Esses passos não podem ser separados logicamente.</p> <p>Não. Incluir os detalhes da entrevista feita pelo funcionário é um passo sub-sequente necessário para completar o processo elementar.</p> <p>Não. Entrar com os dados pessoais do funcionário sem entrar também com os detalhes da entrevista não deixa o negócio da aplicação em um estado consistente. Para mantê-lo em estado consistente, o requisito funcional do usuário deve ser satisfeito sem que haja mais nada a ser feito. Para satisfazer o requisito funcional do usuário, os detalhes da entrevista precisam ser informados.</p>
<p>2. Identificar um processo elementar para cada unidade de atividade identificada que satisfaça todos os critérios acima.</p>	<p>Entrar apenas com os dados pessoais do funcionário não satisfaz nenhum dos critérios de identificação de um processo elementar.</p>

Conclusão Entrar com os Dados Pessoais do Funcionário (sem incluir também os detalhes da entrevista) não satisfaz aos requisitos de um processo elementar.

Entrar com os Detalhes da Entrevista do Funcionário Determinar se entrar com os detalhes da entrevista do funcionário é um processo elementar ou não. A tabela abaixo ilustra a análise realizada:

Identificar o Processo Elementar	A Regra se Aplica?
<p>1. Compor e/ou decompor os Requisitos Funcionais do Usuário na menor unidade de uma atividade, que satisfaça todos os critérios abaixo:</p> <ul style="list-style-type: none"> • É significativo para o usuário • Constitui uma transação completa • É auto-contido e • Deixa o negócio da aplicação que está sendo contada em um estado consistente. 	<p>Sim. Entrar com os detalhes da entrevista do funcionário faz parte dos requisitos funcionais do usuário.</p> <p>Não. A transação completa inclui tanto a entrada dos dados pessoais do funcionário quanto os detalhes da entrevista do funcionário. Esses passos não podem ser separados logicamente.</p> <p>Não. Entrar apenas com os detalhes da entrevista do funcionário não pode ser executado independentemente da entrada dos detalhes da entrevista em si.</p> <p>Não. Entrar apenas com os detalhes da entrevista do funcionário sem entrar também com suas informações pessoais não deixa o negócio da aplicação em um estado consistente. Para mantê-lo em estado consistente, o requisito funcional do usuário deve ser satisfeito sem que haja mais nada a ser feito. Para satisfazer o requisito funcional do usuário, tanto os dados pessoais do funcionário como os detalhes da sua entrevista precisam ser informados.</p>
<p>2. Identificar um processo elementar para cada unidade de atividade identificada que satisfaça todos os critérios acima.</p>	<p>Entrar apenas com os detalhes da entrevista do funcionário não satisfaz nenhum dos critérios de identificação de um processo elementar.</p>

Conclusão Entrar apenas com os detalhes da entrevista do funcionário (sem entrar também com informações pessoais do funcionário) não satisfaz aos requisitos de um processo elementar.

Entrar com os Dados Pessoais do Funcionário e os Detalhes da Sua Entrevista Determinar se entrar com os dados pessoais do funcionário juntamente com os detalhes da sua entrevista é um processo elementar ou não. A tabela abaixo ilustra a análise realizada:

Identificar o Processo Elementar	A Regra se Aplica?
<p>1. Compor e/ou decompor os Requisitos Funcionais do Usuário na menor unidade de uma atividade, que satisfaça todos os critérios abaixo:</p> <ul style="list-style-type: none"> • É significativo para o usuário • Constitui uma transação completa • É auto-contido e • Deixa o negócio da aplicação que está sendo contada em um estado consistente. 	<p>Sim. Entrar com os dados pessoais do funcionário e os detalhes da sua entrevista, ambos fazem parte dos requisitos funcionais do usuário.</p> <p>Sim. Entrar com os dados pessoais do funcionário e os detalhes da sua entrevista juntos são uma transação completa. Esses passos não podem ser separados logicamente.</p> <p>Sim. Entrar com os dados pessoais do funcionário e os detalhes da sua entrevista juntos são uma função auto-contida. Não existem passos anteriores ou subsequentes cuja execução seja necessária.</p> <p>Sim. Entrar com os dados pessoais do funcionário e os detalhes da sua entrevista deixa o negócio em um estado consistente. Para satisfazer o requisito funcional do usuário, ambos os passos do processo precisam ser executados.</p>
<p>2. Identificar um processo elementar para cada unidade de atividade identificada que satisfaça todos os critérios acima.</p>	<p>Entrar com os dados pessoais do funcionário e os detalhes da sua entrevista satisfaz todos os critérios acima.</p>

Conclusão Entrar com os dados pessoais do funcionário juntamente com os detalhes da sua entrevista satisfaz o critério de um processo elementar.

Se dois processos de entrada são sempre sequenciais e dependentes (onde passo um e passo dois são mandatórios), então existe um processo elementar e uma função.

Um novo funcionário não pode ser registrado até que seus dados pessoais e os detalhes da sua entrevista sejam incluídos. Entrar com os dados pessoais de um funcionário ou com os detalhes da sua entrevista isoladamente não seria considerada como a menor unidade de atividade significativa para o usuário.

Entrar com os Dados Pessoais do Funcionário juntamente com os Detalhes da Sua Entrevista representa a menor unidade de atividade que é significativa para o usuário, constitui uma transação completa, é auto-contida e deixa o negócio da aplicação que está sendo contada em um estado consistente.

Determinar Processos Elementares Únicos

A tabela abaixo mostra a análise realizada para determinar se o processo elementar é único ou não.

Determinar Processos Elementares Únicos	A Regra se Aplica?
<p>1. Quando comparado com um Processo Elementar já identificado, contar dois Processos Elementares similares como o mesmo Processo Elementar, se ambos:</p> <ul style="list-style-type: none"> • Requerem o mesmo conjunto de DERs e • Requerem o mesmo conjunto de ALRs e • Requerem o mesmo conjunto de lógica de processamento para completar o processo elementar 	<p>Não. O conjunto de DERs utilizado para Entrar com os Dados Pessoais do Funcionário juntamente com os Detalhes da Sua Entrevista é diferente de qualquer outro processo elementar.</p> <p>Não. O conjunto de ALRs utilizado para Entrar com os Dados Pessoais do Funcionário juntamente com os Detalhes da Sua Entrevista é diferente de qualquer outro processo elementar.</p> <p>Não. A lógica de processamento utilizada para Entrar com os Dados Pessoais do Funcionário juntamente com os Detalhes da Sua Entrevista é diferente de qualquer outro processo elementar.</p>
<p>2. Não separar um processo elementar com múltiplas formas de lógicas de processamento em múltiplos processos elementares.</p>	<p>Um novo funcionário não pode ser registrado até que os dados pessoais e os detalhes da entrevista sejam informados.</p>

Conclusão

Entrar com os Dados Pessoais do Funcionário e com os Detalhes de Entrevista é um processo elementar único em relação a todos os outros processos elementares que já foram identificados.

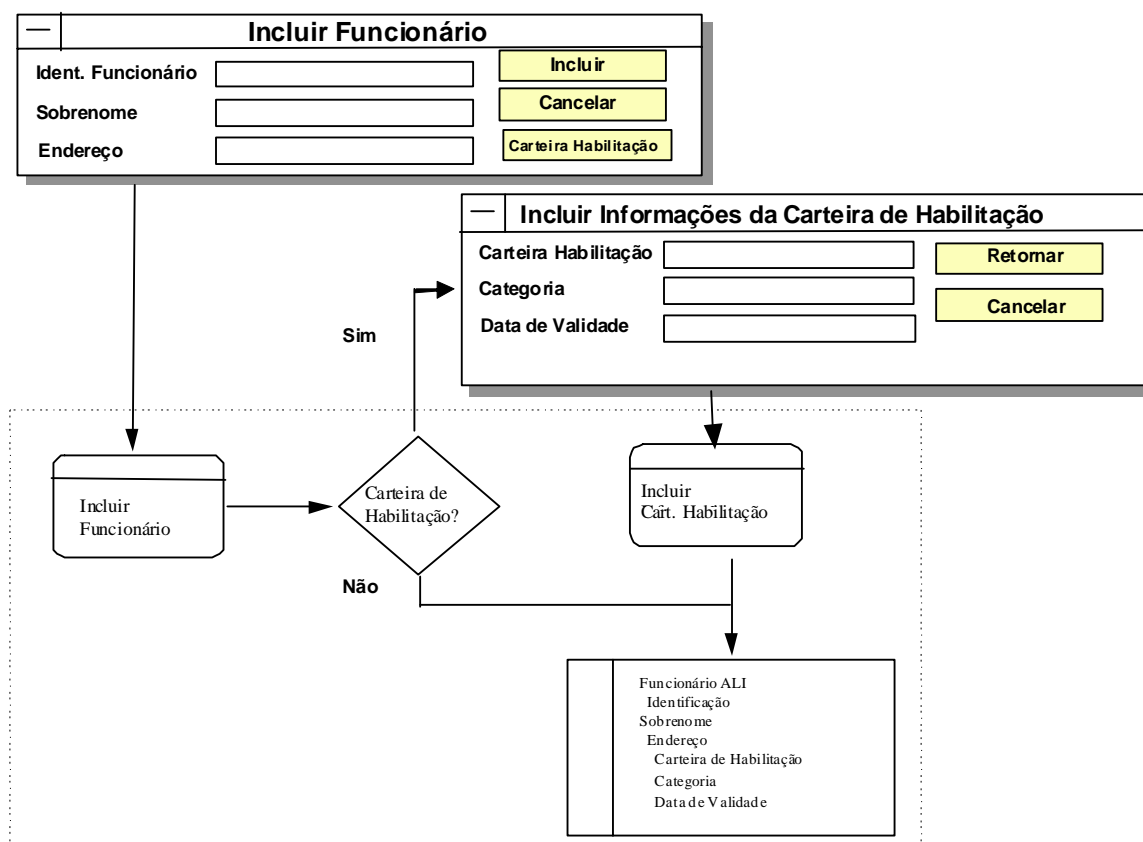
Exemplo: Funcionário - Informações sobre a Carteira de Habilitação

Requisitos do Usuário

Quando um novo funcionário é incluído, deve-se entrar com os seguintes dados pessoais do funcionário: Identificação, sobrenome, endereço, e se o funcionário possui habilitação ou não. Caso o funcionário possua carteira de habilitação, é necessário executar um passo secundário para registrar o número da sua carteira de habilitação, a categoria e a data de validade.

Note: Atualizar as informações pessoais de um funcionário existente incluindo dados de sua carteira de habilitação não está sendo considerado neste exemplo.

O diagrama abaixo mostra o fluxo de dados deste exemplo:



Entrar com os Dados Pessoais do Funcionário Determinar se entrar apenas com os dados pessoais do funcionário é um processo elementar ou não. A tabela seguinte mostra esta análise:

Identificar o Processo Elementar	A Regra se Aplica?
<p>1. Compor e/ou decompor os Requisitos Funcionais do Usuário na menor unidade de uma atividade, que satisfaça todos os critérios abaixo:</p> <ul style="list-style-type: none"> • É significativo para o usuário • Constitui uma transação completa • É auto-contido e • Deixa o negócio da aplicação que está sendo contada em um estado consistente. 	<p>Sim. Incluir um funcionário faz parte dos requisitos funcionais do usuário.</p> <p>Não. A transação completa inclui tanto a entrada dos dados pessoais do funcionário quanto às informações da sua carteira de habilitação (caso exista). Esses passos não podem ser separados logicamente.</p> <p>Não. Entrar com os dados da carteira de habilitação do funcionário (caso exista) é um passo subsequente necessário para completar o processo elementar.</p> <p>Não. Entrar com os dados pessoais do funcionário sem incluir as informações da sua carteira de habilitação (caso exista) não deixa o negócio da aplicação em um estado consistente. Para mantê-lo em estado consistente, o requisito funcional do usuário deve ser satisfeito sem que haja mais nada a ser feito. Para satisfazer o requisito funcional do usuário, as informações da carteira de habilitação precisam ser informados (caso existam).</p>
<p>2. Identificar um processo elementar para cada unidade de atividade identificada que satisfaça todos os critérios acima.</p>	<p>Entrar apenas com os dados pessoais do funcionário não satisfaz nenhum dos critérios de identificação de um processo elementar.</p>

Conclusão Entrar com os Dados Pessoais do Funcionário sem entrar com os dados de sua carteira de habilitação (caso a possua) não satisfaz aos requisitos de um processo elementar.

Entrar com os Dados da Carteira de Habilitação do Funcionário Determinar se Entrar apenas com os dados da carteira de habilitação do funcionário sem entrar também com suas informações pessoais é um processo elementar ou não. A tabela abaixo ilustra a análise realizada:

Identificar o Processo Elementar	A Regra se Aplica?
<p>1. Compor e/ou decompor os Requisitos Funcionais do Usuário na menor unidade de uma atividade, que satisfaça todos os critérios abaixo:</p> <ul style="list-style-type: none"> • É significativo para o usuário • Constitui uma transação completa • É auto-contido e • Deixa o negócio da aplicação que está sendo contada em um estado consistente. 	<p>Sim. Entrar com as informações da carteira de habilitação faz parte dos requisitos funcionais do usuário.</p> <p>Não. A transação completa inclui tanto a entrada dos dados pessoais do funcionário quanto às informações da sua carteira de habilitação (caso exista). Esses passos não podem ser separados logicamente.</p> <p>Não. Entrar com as informações pessoais do funcionário corresponde a um passo anterior necessário para completar o processo elementar.</p> <p>Não. Entrar apenas com as informações da carteira de habilitação do funcionário não deixa o negócio da aplicação em um estado consistente. Para mantê-lo em estado consistente, o requisito funcional do usuário deve ser satisfeito sem que haja mais nada a ser feito. Para satisfazer o requisito funcional do usuário, tanto os dados pessoais do funcionário como a sua carteira de habilitação (caso exista) precisam ser informados.</p>
<p>2. Identificar um processo elementar para cada unidade de atividade identificada que satisfaça todos os critérios acima.</p>	<p>Entrar apenas com as informações da carteira de habilitação do funcionário não satisfaz nenhum dos critérios de identificação de um processo elementar.</p>

Conclusão Entrar com as Informações da Carteira de Habilitação do funcionário (sem entrar com os seus dados pessoais) não satisfaz aos requisitos de um processo elementar.

Entrar com os Dados Pessoais do Funcionário e as Informações de Sua Carteira de Habilitação

Determinar se entrar com os dados pessoais do funcionário juntamente com as informações de sua carteira de habilitação é um processo elementar ou não. A análise realizada está ilustrada na tabela a seguir:

Identificar o Processo Elementar	A Regra se Aplica?
<p>1. Compor e/ou decompor os Requisitos Funcionais do Usuário na menor unidade de uma atividade, que satisfaça todos os critérios abaixo:</p> <ul style="list-style-type: none"> • É significativo para o usuário • Constitui uma transação completa • É auto-contido e • Deixa o negócio da aplicação que está sendo contada em um estado consistente. 	<p>Sim. . Incluir um funcionário e registrar os dados de sua carteira de habilitação fazem parte dos requisitos funcionais do usuário.</p> <p>Sim. A transação completa inclui a entrada dos dados pessoais do funcionário bem como os dados da sua carteira de habilitação (caso exista). Esses passos não podem ser separados logicamente.</p> <p>Sim. Incluir um funcionário e registrar os dados de sua carteira de habilitação é uma função auto-contida. Não existem passos anteriores ou subsequentes cuja execução seja necessária.</p> <p>Sim. Entrar com os dados pessoais de um funcionário e dos dados de sua carteira de habilitação (caso exista) deixa o negócio em um estado consistente. Para satisfazer o requisito funcional do usuário, ambos os passos do processo precisam ser executados.</p>
<p>2. Identificar um processo elementar para cada unidade de atividade identificada que satisfaça todos os critérios acima.</p>	<p>Incluir um funcionário e registrar os dados de sua carteira de habilitação satisfaz todos os critérios acima.</p>

Conclusão

Se dois processos de entrada são sempre sequenciais e dependentes, mas o segundo é opcional (mandatório apenas se aplicável), então existe um processo elementar.

Incluir um Funcionário e Registrar os Dados de Sua Carteira de Habilitação é um Processo Elementar. Se um funcionário não possui carteira de habilitação, o passo “Entrar com os Dados da Sua Carteira de Habilitação” não é relevante. Se um funcionário possui uma carteira de habilitação, uma tela secundária precisa ser preenchida para completar o Processo Elementar e deixar o negócio da aplicação que está sendo contada em um estado consistente.

Determinar Processos Elementares Únicos

A tabela abaixo mostra a análise realizada para determinar se o processo elementar é único ou não.

Determinar Processos Elementares Únicos	A Regra se Aplica?
---	--------------------

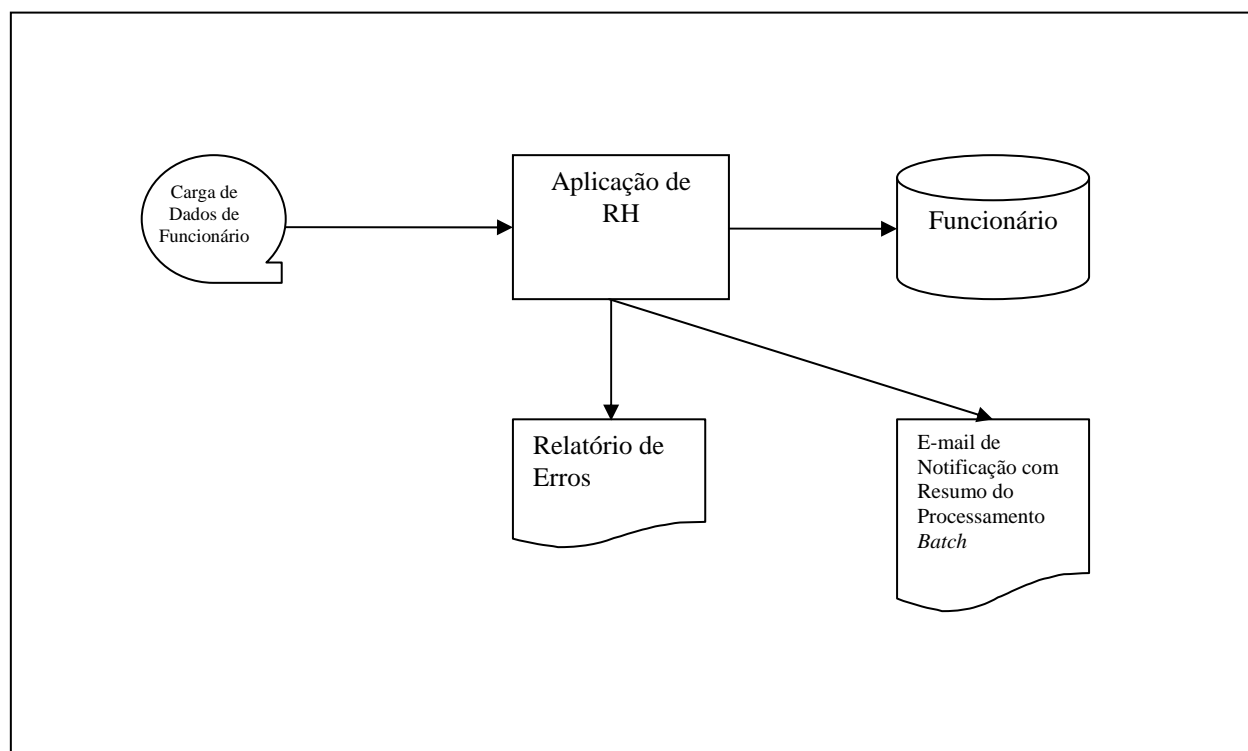
<p>1. Quando comparado com um Processo Elementar já identificado, contar dois Processos Elementares similares como o mesmo Processo Elementar, se ambos:</p> <ul style="list-style-type: none">• Requerem o mesmo conjunto de DERs e• Requerem o mesmo conjunto de ALRs e• Requerem o mesmo conjunto de lógica de processamento para completar o processo elementar	<p>Não. O conjunto de DERs utilizado para Incluir um Funcionário e os Dados de Sua Carteira de Habilitação é diferente de qualquer outro processo elementar.</p> <p>Não. O conjunto de ALRs utilizado para Incluir um Funcionário e os Dados de Sua Carteira de Habilitação é diferente de qualquer outro processo elementar.</p> <p>Não. A lógica de processamento utilizada para Incluir um Funcionário e os Dados de Sua Carteira de Habilitação é diferente de qualquer outro processo elementar.</p>
<p>2. Não separar um processo elementar com múltiplas formas de lógicas de processamento em múltiplos processos elementares.</p>	<p>Se um funcionário possui uma carteira de habilitação , um novo funcionário não pode ser registrado até que os seus dados pessoais e os dados de sua carteira de habilitação sejam informados.</p>

Conclusão Entrar com os Dados Pessoais do Funcionário juntamente com as informações de Sua Carteira de Habilitação é um processo elementar único em relação a todos os outros processos elementares que já foram identificados.

Exemplo: Processamento Batch de Carga de Dados dos Funcionário

Requisitos do Usuário Uma carga dos dados de um novo funcionário em modo *batch* a partir dos dados enviados por outra aplicação deve ser aceita. Os dados do funcionário devem ser validados e armazenados no arquivo Funcionários; um relatório de erros é gerado com todos os erros identificados durante o processamento *batch*. O departamento de RH é notificado via e-mail com o resumo desse processamento.

O diagrama abaixo mostra o fluxo de dados deste exemplo:



Processar Carga de Dados de um Funcionário Determinar se aceitar a Carga de Dados de um Funcionário e processar as transações sem gerar o relatório de erros e o resumo do processamento é um processo elementar ou não. A análise realizada está ilustrada na tabela a seguir:

Identificar o Processo Elementar	A Regra se Aplica?
<p>1. Compor e/ou decompor os Requisitos Funcionais do Usuário na menor unidade de uma atividade, que satisfaça todos os critérios abaixo:</p> <ul style="list-style-type: none"> • É significativo para o usuário • Constitui uma transação completa • É auto-contido e • Deixa o negócio da aplicação que está sendo contada em um estado consistente. 	<p>Sim. Aceitar a Carga de Dados de um Funcionário e processar as transações faz parte dos requisitos funcionais do usuário.</p> <p>Não. A transação completa inclui reportar os erros e o resumo da Carga de Dados de um Funcionário. Esses passos não podem ser separados logicamente.</p> <p>Não. Gerar um Relatório de Erros e Enviar um E-mail de Notificação são passos subsequentes necessários para completar o processo elementar.</p> <p>Não. Processar a Carga de Dados do Funcionário sem gerar um Relatório de Erros e nem um E-mail de Notificação não deixa o negócio da aplicação em um estado consistente. Para mantê-lo em estado consistente, o requisito funcional do usuário deve ser satisfeito sem que haja mais nada a ser feito. Para satisfazer o requisito funcional do usuário, os erros encontrados e o resumo do processamento da Carga de Dados de um Funcionário precisam ser reportados.</p>
<p>2. Identificar um processo elementar para cada unidade de atividade identificada que satisfaça todos os critérios acima.</p>	<p>Nenhum dos critérios foi atingido.</p>

Conclusão Aceitar a carga de Dados de um Funcionário e processar as transações sem gerar o relatório de erros ou enviar um e-mail com o resumo do processamento não satisfaz aos requisitos de um processo elementar.

Gerar um Relatório de Erros

Determinar se gerar um relatório de erros sem processar a Carga de Dados do Funcionário é um processo elementar ou não. A tabela abaixo ilustra a análise realizada:

Identificar o Processo Elementar	A Regra se Aplica?
<p>1. Compor e/ou decompor os Requisitos Funcionais do Usuário na menor unidade de uma atividade, que satisfaça todos os critérios abaixo:</p> <ul style="list-style-type: none"> • É significativo para o usuário • Constitui uma transação completa • É auto-contido e • Deixa o negócio da aplicação que está sendo contada em um estado consistente. 	<p>Sim. Gerar um Relatório de Erros faz parte dos requisitos funcionais do usuário.</p> <p>Não. Gerar um Relatório de Erros está intrinsecamente ligado ao processo de atualização e os erros não podem ser detectados a não ser durante esse processamento. Esses passos não podem ser separados logicamente.</p> <p>Não. Gerar um relatório de erros sem processar a Carga de Dados do Funcionário não é uma função auto-contida. O Relatório de Erros não pode ser gerado independentemente do processamento e validação da Carga de Dados do Funcionário.</p> <p>Não. Gerar um relatório de erros sem processar a Carga de Dados do Funcionário não deixa o negócio da aplicação em um estado consistente. Para mantê-lo em estado consistente, o requisito funcional do usuário deve ser satisfeito sem que haja mais nada a ser feito. Para satisfazer o requisito funcional do usuário, o Relatório de Erros precisa ser produzido a partir do resultado do processamento e da validação da Carga de Dados do Funcionário.</p>
<p>2. Identificar um processo elementar para cada unidade de atividade identificada que satisfaça todos os critérios acima.</p>	<p>Nenhum dos critérios foi atingido.</p>

Conclusão

Gerar um relatório de erros sem processar a Carga de Dados do Funcionário não satisfaz aos requisitos de um processo elementar.

Gerar um E-mail de Notificação Determinar se gerar um E-mail de Notificação sem processar a Carga de Dados do Funcionário é um processo elementar ou não. A tabela abaixo ilustra a análise realizada:

Identificar o Processo Elementar	A Regra se Aplica?
<p>1. Compor e/ou decompor os Requisitos Funcionais do Usuário na menor unidade de uma atividade, que satisfaça todos os critérios abaixo:</p> <ul style="list-style-type: none"> • É significativo para o usuário • Constitui uma transação completa • É auto-contido e • Deixa o negócio da aplicação que está sendo contada em um estado consistente. 	<p>Sim. Gerar um E-mail de Notificação com o resumo do processamento faz parte dos requisitos funcionais do usuário.</p> <p>Não. Gerar um E-mail de Notificação está intrinsecamente ligado ao processo de atualização e as estatísticas do processamento não podem ser calculadas a não ser durante esse processamento. Esses passos não podem ser separados logicamente.</p> <p>Não. Gerar um E-mail de Notificação sem processar a Carga de Dados do Funcionário não é uma função auto-contida. As estatísticas do processamento não podem ser calculadas independentemente do processamento da Carga de Dados do Funcionário.</p> <p>Não. Gerar um E-mail de Notificação sem processar a Carga de Dados do Funcionário não deixa o negócio da aplicação em um estado consistente. Para mantê-lo em estado consistente, o requisito funcional do usuário deve ser satisfeito sem que haja mais nada a ser feito. Para satisfazer o requisito funcional do usuário, o E-mail de Notificação precisa ser gerado a partir do resultado do processamento da Carga de Dados do Funcionário.</p>
<p>2. Identificar um processo elementar para cada unidade de atividade identificada que satisfaça todos os critérios acima.</p>	<p>Nenhum dos critérios foi atingido.</p>

Conclusão Gerar um E-mail de Notificação sem processar a Carga de Dados do Funcionário não satisfaz aos requisitos de um processo elementar.

Processar a Carga de Dados do Funcionário, Gerar Relatório de Erros e E-mail de Notificação

Determinar se aceitar a carga de dados de um funcionário, processar a transação, gerar um relatório de erros e enviar um e-mail de notificação é um processo elementar ou não. A tabela abaixo ilustra a análise realizada:

Identificar o Processo Elementar	A Regra se Aplica?
<p>1. Compor e/ou decompor os Requisitos Funcionais do Usuário na menor unidade de uma atividade, que satisfaça todos os critérios abaixo:</p> <ul style="list-style-type: none"> • É significativo para o usuário • Constitui uma transação completa • É auto-contido e • Deixa o negócio da aplicação que está sendo contada em um estado consistente. 	<p>Sim. Processar a Carga de Dados do Funcionário, gerar Relatório de Erros e E-mail de Notificação fazem parte dos requisitos funcionais do usuário.</p> <p>Sim. Processar a Carga de Dados do Funcionário, gerar Relatório de Erros e E-mail de Notificação é uma transação completa. Esses passos não podem ser separados logicamente.</p> <p>Sim. Processar a Carga de Dados do Funcionário, gerar Relatório de Erros e E-mail de Notificação é uma função auto-contida.</p> <p>Sim. Processar a Carga de Dados do Funcionário, gerar Relatório de Erros e E-mail de Notificação deixa o negócio em um estado consistente. Para satisfazer o requisito funcional do usuário, todos os passos precisam ser executados para processar a Carga de Dados do Funcionário</p>
<p>2. Identificar um processo elementar para cada unidade de atividade identificada que satisfaça todos os critérios acima.</p>	<p>Todos os critérios para identificar um processo elementar foram satisfeitos conforme demonstrado acima.</p>

Conclusão

Processar a Carga de Dados do Funcionário incluindo a geração do Relatório de Erros e do E-mail de Notificação satisfazem os requisitos funcionais do usuário.

Aceitar a Carga de Dados do Funcionário, processar as transações, gerar o Relatório de Erros e o E-mail de Notificação é um Processo Elementar. Se a carga não for aceita e as transações não forem processadas, os passos “Gerar Relatório de Erros” e “Gerar E-mail de Notificação” não serão relevantes. Todos os passos do processamento precisam ser executados para completar o processo elementar e deixar o negócio da aplicação que está sendo contada em um estado consistente.

Determinar Processos Elementares Únicos

A tabela abaixo mostra a análise realizada para determinar se o processo elementar é único ou não.

Determinar Processos Elementares Únicos	A Regra se Aplica?
<p>1. Quando comparado com um Processo Elementar já identificado, contar dois Processos Elementares similares como o mesmo Processo Elementar, se ambos:</p> <ul style="list-style-type: none"> • Requerem o mesmo conjunto de DERs e • Requerem o mesmo conjunto de ALRs e • Requerem o mesmo conjunto de lógica de processamento para completar o processo elementar 	<p>Não. O conjunto de DERs utilizado para Processar a Carga de Dados do Funcionário é diferente de qualquer outro processo elementar.</p> <p>Não. O conjunto de ALRs utilizado para Processar a Carga de Dados do Funcionário é diferente de qualquer outro processo elementar.</p> <p>Não. A lógica de processamento utilizada para Processar a Carga de Dados do Funcionário é diferente de qualquer outro processo elementar.</p>
<p>2. Não separar um processo elementar com múltiplas formas de lógicas de processamento em múltiplos processos elementares.</p>	<p>Conforme discutido anteriormente, não é apropriado subdividir um processo elementar.</p>

Conclusão

O Processamento *Batch* da Carga de Dados dos Funcionários é um processo elementar único em relação a todos os outros processos elementares que já foram identificados.

Exemplo: Assinalar um Funcionário a uma Função

Requisitos do Usuário

Assinalar um funcionário a uma função. Entrar com as informações da função assinalada através da entrada das seguintes informações para cada função e funcionário assinalado:

- Data de Efetivação
- Salário
- Avaliação de Desempenho

Para facilitar o assinalamento, o usuário solicitou uma lista de assinalamento de funcionários e/ou uma lista das funções com os funcionários assinalados.

Janela da Lista de Assinalamentos de Funcionários

Esta janela exibe a lista de funcionários e as funções assinaladas para cada funcionário.

Sistema de Recursos Humanos				▼	⌂																																																																	
Funcionários Funções Assinalamentos Localizações Relatórios Segurança Ajuda																																																																						
<table border="1"> <thead> <tr> <th colspan="5">Assinalamento de Funcionários</th> </tr> <tr> <th>Sobrenome</th> <th>Nome</th> <th>IM</th> <th>Função</th> <th></th> </tr> </thead> <tbody> <tr> <td>Carvalho</td> <td>Carolina</td> <td></td> <td>RD21379305</td> <td>↑</td> </tr> <tr> <td>Costa</td> <td>Fábio</td> <td>B</td> <td>RD21379305</td> <td></td> </tr> <tr> <td>Loureiro</td> <td>Fernanda</td> <td>A</td> <td>RD90379305</td> <td></td> </tr> <tr> <td>Matos</td> <td>Marcos</td> <td>J</td> <td>RD14379305</td> <td></td> </tr> <tr> <td>Matos</td> <td>Marcos</td> <td>J</td> <td>RD15379305</td> <td></td> </tr> <tr> <td>Silva</td> <td>Giovana</td> <td></td> <td>RD14379305</td> <td></td> </tr> <tr> <td>Silva</td> <td>Davi</td> <td>E</td> <td>RD16379305</td> <td></td> </tr> <tr> <td>Silveira</td> <td>Sérgio</td> <td>M</td> <td>RD16379305</td> <td></td> </tr> <tr> <td>Torres</td> <td>Patrícia</td> <td></td> <td>RD90379305</td> <td></td> </tr> <tr> <td>Valentim</td> <td>Sandra</td> <td>B</td> <td>RD21379305</td> <td></td> </tr> <tr> <td>Valentim</td> <td>Sandra</td> <td>B</td> <td>RD90379305</td> <td>↓</td> </tr> </tbody> </table>						Assinalamento de Funcionários					Sobrenome	Nome	IM	Função		Carvalho	Carolina		RD21379305	↑	Costa	Fábio	B	RD21379305		Loureiro	Fernanda	A	RD90379305		Matos	Marcos	J	RD14379305		Matos	Marcos	J	RD15379305		Silva	Giovana		RD14379305		Silva	Davi	E	RD16379305		Silveira	Sérgio	M	RD16379305		Torres	Patrícia		RD90379305		Valentim	Sandra	B	RD21379305		Valentim	Sandra	B	RD90379305	↓
Assinalamento de Funcionários																																																																						
Sobrenome	Nome	IM	Função																																																																			
Carvalho	Carolina		RD21379305	↑																																																																		
Costa	Fábio	B	RD21379305																																																																			
Loureiro	Fernanda	A	RD90379305																																																																			
Matos	Marcos	J	RD14379305																																																																			
Matos	Marcos	J	RD15379305																																																																			
Silva	Giovana		RD14379305																																																																			
Silva	Davi	E	RD16379305																																																																			
Silveira	Sérgio	M	RD16379305																																																																			
Torres	Patrícia		RD90379305																																																																			
Valentim	Sandra	B	RD21379305																																																																			
Valentim	Sandra	B	RD90379305	↓																																																																		
<div> <div>Visualizar</div> <div>Novo</div> <div>Editar</div> <div>Excluir</div> </div>																																																																						

AF-1 Lista de Assinalamento de Funcionários

Visualizar	Inicia exibição dos dados na AF-4 Lista de Assinalamento das Funções
Novo	Retorna Tela em Branco na AF-3 Iniciar Assinalamento de Função (preenchida com informação <i>default</i>)
Editar	Inicia exibição do dado na AF-5 Editar Assinalamento de Função
Excluir	Elimina a entrada de assinalamento da função (após confirmação)

Janela de Assinalamento de Função

A janela seguinte mostra o assinalamento de um funcionário a uma determinada função (por funcionário).

The screenshot shows a web application window titled 'Sistema de Recursos Humanos'. It has a menu bar with 'Funcionários', 'Funções', 'Assinalamentos', 'Localizações', 'Relatórios', 'Segurança', and 'Ajuda'. The main content area displays a window titled 'Assinalamento de Funcionários'. Inside this window is a sub-window titled 'Iniciar Assinalamento de Função'. This sub-window contains the following fields and controls:

- Nome:** Marcos
- IM:** J
- Sobrenome:** Matos
- Ident. Funcionário:** 34567201-9
- Local:** Rio de Janeiro
- Unidade de Negócio:** UFPCA
- Código da Função:** A dropdown menu with a yellow arrow icon.
- Data:** An empty text box.
- Salário:** An empty text box.
- Avaliação de Desempenho:** An empty text box.
- Buttons:** 'OK' and 'Cancelar' buttons.

AF-3 Iniciar Assinalamento de Função

- OK** Salva o novo assinalamento de função e retorna para a tela AF-1 Lista de Assinalamento de Funcionários
- Cancelar** Ignora os dados informados e retorna para a tela AF-1 Lista de Assinalamento de Funcionários

Se o usuário não entrar com os dados corretamente, é exibida uma mensagem de erro.

**Janela de
Exibição da Lista
de Assinalamento
de Funções**

A próxima janela mostra a lista de funções e os funcionários assinalados para cada função.

Sistema de Recursos Humanos			
Funcionários Funções Assinalamentos Localizações Relatórios Segurança Ajuda			
Assinalamento de Funções			
Cód. Função	Sobrenome	Nome	IM
RD14379305	Matos	Marcos	J
RD14379305	Silveira	Sérgio	J
RD15379305	Matos	Marcos	J
RD16379305	Silva	Davi	E
RD16379305	Silva	Giovana	M
RD21379305	Carvalho	Carolina	A
RD21379305	Loureiro	Fernanda	A
RD21379305	Costa	Fábio	B
RD90379305	Valentim	Sandra	A
RD90379305	Loureiro	Patricia	A
RD90379305	Valentim	Sandra	A

AF-1 Lista das Funções Assinaladas

<input type="button" value="Visualizar"/>	Inicia exibição dos dados na AF-4 Lista de Assinalamento dos Funcionários
<input type="button" value="Novo"/>	Retorna Tela em Branco na AF-3 Iniciar Assinalamento de Funcionário (preenchida com informação <i>default</i>)
<input type="button" value="Editar"/>	Inicia exibição do dado na AF 5 Editar Assinalamento de Funcionário
<input type="button" value="Excluir"/>	Elimina entrada do assinalamento do funcionário à uma função (após confirmação)

Janela de Assinalamento de um Funcionário a uma Função

A janela abaixo mostra o assinalamento de um funcionário a uma função (por função).

Sistema de Recursos Humanos

Funcionário Funções Assinalamentos Localização Relatórios Segurança Ajuda

Assinalamento de Funções

Iniciar Assinalamento de Funcionário

RD14379305 May Issue- Print Text

Sobrenome

Nome

Iniciais do Meio

Ident. Funcionário - -

Localização Unidade de Negócio

Data

Salário

Avaliação de Desempenho

OK

Cancel

AF3 Iniciar Assinalamento de Funcionário

OK Salvar o novo assinalamento de Funcionário

Cancelar Ignora os dados informados e retorna para a tela AJ-1 Lista de Funções Assinaladas

Se o usuário não entrar com os dados corretamente, é exibida uma mensagem de erro.

Lista de Assinalamento de Funcionários Determinar se Listar os Assinalamentos por Funcionário é um processo elementar ou não. A tabela abaixo ilustra a análise realizada:

Identificar o Processo Elementar	A Regra se Aplica?
<p>1. Compor e/ou decompor os Requisitos Funcionais do Usuário na menor unidade de uma atividade, que satisfaça todos os critérios abaixo:</p> <ul style="list-style-type: none"> • É significativo para o usuário • Constitui uma transação completa • É auto-contido e • Deixa o negócio da aplicação que está sendo contada em um estado consistente. 	<p>Sim. Listar os Assinalamentos por Funcionário faz parte dos requisitos funcionais do usuário.</p> <p>Sim. Listar os Assinalamentos por Funcionário é uma transação completa. É uma função logicamente independente.</p> <p>Sim. Listar os Assinalamentos por Funcionário é uma função auto-contida. Listar os Assinalamentos por Funcionário é executada independentemente de assinalar um funcionário a uma função.</p> <p>Sim. Listar os Assinalamentos por Funcionário deixa o negócio de uma aplicação em um estado consistente. O requisito funcional do usuário foi totalmente satisfeito e não há mais nada necessário a ser feito.</p>
<p>2. Identificar um processo elementar para cada unidade de atividade identificada que satisfaça todos os critérios acima.</p>	<p>Listar os Assinalamentos por Funcionário satisfaz todos os critérios acima.</p>

Conclusão Listar os Assinalamentos por Funcionário satisfaz os requisitos de um processo elementar.

**Assinalar
Funcionário a
uma Função
(por
Funcionário)**

Determinar se Assinalar Funcionário a uma Função (por Funcionário) é um processo elementar ou não. A tabela abaixo ilustra a análise realizada:

Identificar o Processo Elementar	A Regra se Aplica?
<p>1. Compor e/ou decompor os Requisitos Funcionais do Usuário na menor unidade de uma atividade, que satisfaça todos os critérios abaixo:</p> <ul style="list-style-type: none"> • É significativo para o usuário • Constitui uma transação completa • É auto-contido e • Deixa o negócio da aplicação que está sendo contada em um estado consistente. 	<p>Sim. Assinalar Funcionário a uma Função (por Funcionário) faz parte dos requisitos funcionais do usuário.</p> <p>Sim. Assinalar Funcionário a uma Função (por Funcionário) é uma transação completa. É uma função logicamente independente.</p> <p>Sim. Assinalar Funcionário a uma Função (por Funcionário) é uma função auto-contida. Assinalar Funcionário a uma Função (por Funcionário) é executada independentemente de Listar os Assinalamentos dos Funcionários.</p> <p>Sim. Assinalar Funcionário a uma Função (por Funcionário) deixa o negócio de uma aplicação em um estado consistente. O requisito funcional do usuário foi totalmente satisfeito e não há mais nada necessário a ser feito.</p>
<p>2. Identificar um processo elementar para cada unidade de atividade identificada que satisfaça todos os critérios acima.</p>	<p>Assinalar Funcionário a uma Função (por Funcionário) satisfaz todos os critérios acima.</p>

Conclusão

Assinalar Funcionário a uma Função (por Funcionário) satisfaz os requisitos de um processo elementar.

Listar as Funções Assinaladas Determinar se Listar as Funções Assinaladas é um processo elementar ou não. A tabela abaixo ilustra a análise realizada:

Identificar o Processo Elementar	A Regra se Aplica?
<p>1. Compor e/ou decompor os Requisitos Funcionais do Usuário na menor unidade de uma atividade, que satisfaça todos os critérios abaixo:</p> <ul style="list-style-type: none"> • É significativo para o usuário • Constitui uma transação completa • É auto-contido e • Deixa o negócio da aplicação que está sendo contada em um estado consistente. 	<p>Sim. Listar as Funções Assinaladas faz parte dos requisitos funcionais do usuário.</p> <p>Sim. Listar as Funções Assinaladas é uma transação completa. É uma função logicamente independente.</p> <p>Sim. Listar as Funções Assinaladas é uma função auto-contida. Listar as Funções Assinaladas é executado independentemente de assinalar um funcionário a uma determinada função.</p> <p>Sim. Listar as Funções Assinaladas deixa o negócio de uma aplicação em um estado consistente. O requisito funcional do usuário foi totalmente satisfeito e não há mais nada necessário a ser feito.</p>
<p>2. Identificar um processo elementar para cada unidade de atividade identificada que satisfaça todos os critérios acima.</p>	<p>Listar as Funções Assinaladas satisfaz todos os critérios acima.</p>

Conclusão Listar as Funções Assinaladas satisfaz os requisitos de um processo elementar.

Assinalar Funcionário a uma Função (por Função)

Determinar se Assinalar Funcionário a uma Função (por Função) é um processo elementar ou não. A tabela abaixo ilustra a análise realizada:

Identificar o Processo Elementar	A Regra se Aplica?
<p>1. Compor e/ou decompor os Requisitos Funcionais do Usuário na menor unidade de uma atividade, que satisfaça todos os critérios abaixo:</p> <ul style="list-style-type: none"> • É significativo para o usuário • Constitui uma transação completa • É auto-contido e • Deixa o negócio da aplicação que está sendo contada em um estado consistente. 	<p>Sim. Assinalar Funcionário a uma Função (por Função) faz parte dos requisitos funcionais do usuário.</p> <p>Sim. Assinalar Funcionário a uma Função (por Função) é uma transação completa. É uma função logicamente independente.</p> <p>Sim. Assinalar Funcionário a uma Função (por Função) é uma função auto-contida Assinalar Funcionário a uma Função (por Função) é executada independentemente de Listar as Funções Assinaladas.</p> <p>Sim. Assinalar Funcionário a uma Função (por Função) deixa o negócio da aplicação em um estado consistente.</p>
<p>2. Identificar um processo elementar para cada unidade de atividade identificada que satisfaça todos os critérios acima.</p>	<p>Assinalar Funcionário a uma Função (por Função) satisfaz todos os critérios acima. O requisito funcional do usuário foi totalmente satisfeito e não há mais nada necessário a ser feito.</p>

Conclusão

Assinalar Funcionário a uma Função (por Função) satisfaz os requisitos de um processo elementar.

Determinar Processos Elementares Únicos A tabela abaixo mostra a análise realizada para determinar se os processos elementares Listar os Assinalamentos por Funcionário e Listar as Funções Assinaladas são únicos ou não.

Determinar Processos Elementares Únicos	A Regra se Aplica?
<p>1. Quando comparado com um Processo Elementar já identificado, contar dois Processos Elementares similares como o mesmo Processo Elementar, se ambos:</p> <ul style="list-style-type: none"> • Requerem o mesmo conjunto de DERs e • Requerem o mesmo conjunto de ALRs e • Requerem o mesmo conjunto de lógica de processamento para completar o processo elementar 	<p>Sim. O conjunto de DERs utilizado para Listar os Assinalamentos por Funcionário é mesmo do conjunto de DERs utilizado para Listar as Funções Assinaladas.</p> <p>Sim. O conjunto de ALRs utilizado para Listar os Assinalamentos por Funcionário é mesmo do conjunto de ALRs utilizado para Listar as Funções Assinaladas.</p> <p>Sim. O conjunto de lógica de processamento para Listar os Assinalamentos por Funcionário é mesmo que o de Listar as Funções Assinaladas. A única diferença entre eles está na sequência de exibição dos campos e ordem das linhas. Conforme já definido nas Formas de Lógica de Processamento, diferenças na forma de ordenar ou exibir os campos não configura um processo elementar único.</p>
<p>2. Não separar um processo elementar com múltiplas formas de lógicas de processamento em múltiplos processos elementares.</p>	<p>Não há nada a ser separado.</p>

Conclusão Listar os Assinalamentos por Funcionário e/ou Listar as Funções Assinaladas é um processo elementar único.

Determinar Processos Elementares Únicos A tabela abaixo mostra a análise realizada para determinar se os processos elementares Assinalar Funcionário a uma Função (por Funcionário) e Assinalar Funcionário a uma Função (por Função) são únicos ou não.

Determinar Processos Elementares Únicos	A Regra se Aplica?
<p>1. Quando comparado com um Processo Elementar já identificado, contar dois Processos Elementares similares como o mesmo Processo Elementar, se ambos:</p> <ul style="list-style-type: none"> • Requerem o mesmo conjunto de DERs e • Requerem o mesmo conjunto de ALRs e • Requerem o mesmo conjunto de lógica de processamento para completar o processo elementar 	<p>Sim. O conjunto de DERs para Assinalar Funcionário a uma Função (por Funcionário) é o mesmo utilizado para Assinalar Funcionário a uma Função (por Função).</p> <p>Sim. O conjunto de ALRs para Assinalar Funcionário a uma Função (por Funcionário) é o mesmo utilizado para Assinalar Funcionário a uma Função (por Função).</p> <p>Sim. O conjunto de lógica de processamento para Assinalar Funcionário a uma Função (por Funcionário) é mesmo que o Assinalar Funcionário a uma Função (por Função). A única diferença entre eles está na sequência de exibição dos atributos na tela. Conforme já definido nas Formas de Lógica de Processamento, diferenças na forma de ordenar ou exibir os atributos não configura um processo elementar único.</p>
<p>2. Não separar um processo elementar com múltiplas formas de lógicas de processamento em múltiplos processos elementares.</p>	<p>Não há nada a ser separado.</p>

Conclusão Dois processos elementares únicos foram identificados:

- Listar os Assinalamentos por Funcionário e/ou Listar as Funções Assinaladas
- Assinalar Funcionário a uma Função (por Funcionário) e/ou Assinalar Funcionário a uma Função (por Função)

Quando dois processos elementares similares são comparados e identifica-se que eles contém o mesmo conjunto de DERs, ALRs e Lógica de Processamento, eles são identificados como um único processo elementar.

Exemplo: Processos Elementares Similares

Requisitos do Usuário

O usuário que utilize as informações de funcionários requer dois relatórios que são bem similares. Um relatório incluirá o e-mail dos funcionários e será distribuído para a equipe remota que requer comunicação via e-mail ao invés de por telefone. Veja os exemplos de relatórios com seus respectivos detalhamentos.

Todos os dados do relatório vêm do mesmo arquivo lógico exceto o endereço de e-mail que vem de um arquivo lógico diferente que é mantido dentro da mesma aplicação.

Lista de Funcionários (com Endereço de E-mail)

Lista de Funcionários

ID #	Nome	Telefone Resid.	Telefone Trab.	Local	Endereço de E-mail
894253	Fábio Costa	888-555-1212	999-666-0202	Planta A	FCosta@ABCCorp.com
999524	Davi Silva	777-258-489	999-125-3589	Planta B	DSilva@ABCCorp.com
125896	Pat Torres	555-678-6892	999-258-4789	Planta C	PTorres@ABCCorp.com

Lista de Funcionários (sem Endereço de E-mail)

Lista de Funcionários

ID #	Nome	Telefone Resid.	Telefone Trab.	Local
894253	Fábio Costa	888-555-1212	999-666-0202	Planta A
999524	Davi Silva	777-258-489	999-125-3589	Planta B
125896	Pat Torres	555-678-6892	999-258-4789	Planta C

Lista de Funcionários (sem Endereço de E-mail)

Determinar se Exibir a Lista de Funcionários (Sem Endereço de E-mail) é um processo elementar ou não. A tabela abaixo ilustra a análise realizada:

Regras de Contagem de Processo Elementar	A Regra se Aplica?
<p>1. Compor e/ou decompor os Requisitos Funcionais do Usuário na menor unidade de uma atividade, que satisfaça todos os critérios abaixo:</p> <ul style="list-style-type: none"> • É significativo para o usuário • Constitui uma transação completa • É auto-contido e • Deixa o negócio da aplicação que está sendo contada em um estado consistente. 	<p>Sim. Exibir a Lista de Funcionários (sem Endereço de E-mail) é um requisito funcional do usuário.</p> <p>Sim. Exibir a Lista de Funcionários (sem Endereço de E-mail) é uma transação completa que atende a um grupo de usuários.</p> <p>Sim. Exibir a Lista de Funcionários (sem Endereço de E-mail) é significativo por si só.</p> <p>Sim. O negócio é mantido em um estado consistente quando a Lista de Funcionários (Sem o Endereço de E-mail) é criada.</p>
<p>2. Identificar um processo elementar para cada unidade de atividade identificada que satisfaça todos os critérios acima.</p>	<p>Exibir a Lista de Funcionários (Sem o Endereço de E-mail) satisfaz a todos os critérios acima.</p>

Conclusão

Exibir a Lista de Funcionários (Sem o Endereço de E-mail) satisfaz aos requisitos de um processo elementar.

Determinar Processos Elementares Únicos A tabela abaixo mostra a análise realizada para determinar se o processo elementar é único ou não.

Determinar Processos Elementares Únicos	A Regra se Aplica?
<p>1. Quando comparado com um Processo Elementar já identificado, contar dois Processos Elementares similares como o mesmo Processo Elementar, se ambos:</p> <ul style="list-style-type: none"> • Requerem o mesmo conjunto de DERs e • Requerem o mesmo conjunto de ALRs e • Requerem o mesmo conjunto de lógica de processamento para completar o processo elementar 	<p>Sim. A Lista de Funcionários (Sem o Endereço de E-mail) requer um conjunto de DERs diferente da Lista de Funcionários (Com o Endereço de E-mail).</p> <p>Sim. A Lista de Funcionários (Sem o Endereço de E-mail) requer um conjunto de ALRs diferente da Lista de Funcionários (Com o Endereço de E-mail).</p> <p>Sim. A Lista de Funcionários (Sem o Endereço de E-mail) requer um conjunto de lógica de processamento diferente da Lista de Funcionários (Com o Endereço de E-mail).</p>
<p>2. Não separar um processo elementar com múltiplas formas de lógicas de processamento em múltiplos processos elementares.</p>	<p>Não há nada a ser separado.</p>

Conclusão Exibir a Lista de Funcionários (Sem o Endereço de E-mail) é um processo elementar único em relação a todos os outros processos elementares que já foram identificados.

Lista de Funcionários (Com Endereço de E-mail)

Determinar se Exibir a Lista de Funcionários (Com Endereço de E-mail) é um processo elementar ou não. A tabela abaixo ilustra a análise realizada:

Regras de Contagem de Processo Elementar	A Regra se Aplica?
<p>1. Compor e/ou decompor os Requisitos Funcionais do Usuário na menor unidade de uma atividade, que satisfaça todos os critérios abaixo:</p> <ul style="list-style-type: none"> • É significativo para o usuário • Constitui uma transação completa • É auto-contido e • Deixa o negócio da aplicação que está sendo contada em um estado consistente. 	<p>Sim. Exibir a Lista de Funcionários (Com Endereço de E-mail) é um requisito funcional do usuário.</p> <p>Sim. Exibir a Lista de Funcionários (Com Endereço de E-mail) é uma transação completa que atende a um grupo de usuários.</p> <p>Sim. Exibir a Lista de Funcionários (Com Endereço de E-mail) é significativo por si só.</p> <p>Sim. O negócio é mantido em um estado consistente quando a Lista de Funcionários (Com o Endereço de E-mail) é criada.</p>
<p>2. Identificar um processo elementar para cada unidade de atividade identificada que satisfaça todos os critérios acima.</p>	<p>Exibir a Lista de Funcionários (Com o Endereço de E-mail) satisfaz a todos os critérios acima.</p>

Conclusão

Exibir a Lista de Funcionários (Com o Endereço de E-mail) satisfaz aos requisitos de um processo elementar.

Determinar Processos Elementares Únicos

A tabela abaixo mostra a análise realizada para determinar se o processo elementar é único ou não.

Determinar Processos Elementares Únicos	A Regra se Aplica?
<p>1. Quando comparado com um Processo Elementar já identificado, contar dois Processos Elementares similares como o mesmo Processo Elementar, se ambos:</p> <ul style="list-style-type: none"> • Requerem o mesmo conjunto de DERs e • Requerem o mesmo conjunto de ALRs e • Requerem o mesmo conjunto de lógica de processamento para completar o processo elementar 	<p>Sim. A Lista de Funcionários (Com o Endereço de E-mail) requer um conjunto de DERs diferente da Lista de Funcionários (Sem o Endereço de E-mail).</p> <p>Sim. A Lista de Funcionários (Com o Endereço de E-mail) requer um conjunto de ALRs diferente da Lista de Funcionários (Sem o Endereço de E-mail).</p> <p>Sim. A Lista de Funcionários (Com o Endereço de E-mail) requer um conjunto de lógica de processamento diferente da Lista de Funcionários (Sem o Endereço de E-mail).</p>
<p>2. Não separar um processo elementar com múltiplas formas de lógicas de processamento em múltiplos processos elementares.</p>	<p>Não há nada a ser separado.</p>

Conclusão

O processo elementar Exibir a Lista de Funcionários (Com o Endereço de E-mail) é único em relação a todos os outros processos elementares identificados.

Dois processos são determinados para satisfazer os critérios de um processo elementar, então são comparados entre si para determinar se eles contêm DERs, ALRs e Lógicas de Processamento diferentes.

Quando os dois processos elementares são comparados e determina-se que eles contêm DERs, ALRs ou Lógica de Processamento diferentes, eles são identificados como processos elementares separados, se eles são especificados como requisitos funcionais distintos pelo usuário.

Dois processos elementares únicos são identificados:

- Exibir a Lista de Funcionários (Sem o Endereço de E-mail) e
- Exibir a Lista de Funcionários (Com o Endereço de E-mail)

Esta página foi deixada intencionalmente em branco.

Exemplos de Contagem de EE

Introdução Esta seção utiliza uma aplicação de Recursos Humanos (RH) para ilustrar os procedimentos de contagem de entradas externas (EE). Além desta seção, os exemplos estão descritos nos Estudos de Casos que é parte da documentação suplementar do IFPUG.

Conteúdo Esta seção inclui os seguintes exemplos:

Topic	Page
Descrição Geral dos Exemplos de Contagem de EE	2-64
Exemplo: Informações de Controle de Relatório	2-65
Exemplo: Tela de Entrada	2-69
Exemplo: Processamento <i>Batch</i> com Múltiplas EEs e EEs Duplicadas	2-72
Exemplo: Correção de Transações Suspensas	2-75
Exemplo: EE com Múltiplos Arquivos Lógicos Referenciados	2-78
Exemplo: Conversão de Dados	2-82
Exemplo: Referenciando Dados a partir de Outra Aplicação	2-85
Exemplo: EE com Tela de Saída – 1	2-87
Exemplo: EE com Tela de Saída – 1	2-90
Exemplo: EE com Atributos Recuperados de um AIE	2-93
Exemplo: EE Excluir	2-99
Exemplo: Incluir Nível de Segurança de Janelas	2-102

Descrição Geral dos Exemplos de Contagem de EE

Os exemplos de entradas externas estão descritos na tabela abaixo:

Exemplo	Descrição Geral	Página
Informações de Controle de Relatórios	Este exemplo mostra as informações de controle utilizadas para impressão de relatórios.	2-65
Tela de Entrada	Este exemplo ilustra a contagem de uma transação <i>online</i> via tela de entrada.	2-69
Processamento <i>Batch</i> com Múltiplas EEs e EEs Duplicadas	Este exemplo mostra a contagem de um arquivo de transação com múltiplos tipos ou tipos de registros formatados.	2-72
Correção de Transações Suspensas	Este exemplo ilustra a contagem da correção de transações suspensas, registradas em um arquivo de transações suspensas durante um processamento <i>batch</i> de adição ou atualização de funções.	2-75
EE com Múltiplos Arquivos Lógicos Referenciados	Este exemplo ilustra o uso de um diagrama de fluxo de dados para contar uma entrada externa que contém múltiplos arquivos lógicos referenciados (ALRs).	2-78
Conversão de Dados	Este exemplo ilustra a contagem do processo de conversão de um grupo de dados para um novo formato com elementos de dados adicionais.	2-82
Referenciando Dados a partir de Outra Aplicação	Este exemplo mostra como um arquivo de interface externa (discutida no Capítulo 6 da Parte 2) não é contado como uma entrada externa.	2-85
EE com Tela de Saída – 1	Este exemplo ilustra uma EE com a exibição de um campo calculado.	2-87
EE com Tela de Saída – 2	Este exemplo ilustra uma EE com a exibição de um campo calculado e CEs embutidas.	2-90
EE com Atributos Recuperados de um AIE	Este exemplo ilustra uma EE com atributos que são recuperados de um AIE que não atravessa a fronteira da aplicação.	2-93
EE Excluir	Este exemplo ilustra a contagem de DERs de uma transação de exclusão.	2-99
Incluir Nível de Segurança de Janelas	Este exemplo ilustra a contagem de funcionalidade responsável por manter uma aplicação de segurança.	2-102

Exemplo: Informações de Controle de Relatório

Requisitos do Usuário

O usuário requer a habilidade para controlar como e onde os relatórios de assinalamentos serão impressos. A lista abaixo mostra os requisitos específicos do usuário para a criação do relatório:

1. Controlar os seguintes aspectos de processamento do relatório:
 - Classificação (*Sort*)
 - Porta da impressora
 - Tipo de saída (i.e., microficha e/ou papel)
2. Salvar os controles utilizados na criação do relatório de funções assinaladas.
3. Gerar e salvar as mudanças.
4. Enviar uma mensagem para confirmar que os controles usados para criação dos relatórios de funções assinaladas foram adicionados e/ou modificados, e que os mesmos estão sendo gerados.

Nota: Este exemplo mostra apenas o requisito para adicionar o conjunto de informações de controle do relatório de assinalamentos. O Estudo de Casos ilustra a contagem do requisito completo do usuário.

Exemplo da Tela

A tela abaixo é utilizada para Estabelecer os Controles para geração do Relatório das Funções Assinaladas.

—
Sistema de Recursos Humanos
▼ ▲

Funcionário
Funções
Assinalamentos
Locais
Ajuda

—
Relatório das Funções Assinaladas

Classificação

3

Código da Função

2

Código do Funcionário

1

Nome do Funcionário

Identifique com 1, 2 ou 3

Impressora

()

LPT 1

()

LPT 2

()

LPT 3

☐ Gerar Cópia Microficha
☒ Gerar Cópia em Papel

OK

Cancelar

Restaurar

JR-1 Relatório das Funções Assinaladas

OK

Processar o relatório solicitado

Cancelar

Voltar ao menu anterior

Restaurar

Restaurar valores iniciais

Passo 1 Identificar o Processo Elementar

A Função de Transação satisfaz os requisitos de um Processo Elementar?	Sim.
--	------

Passo 2 Determinar se o Processo Elementar é Único ou não

A Função de Transação é única relação aos outros processos elementares?	Sim. Nenhum outro processo elementar executa essa função.
---	---

Passo 3 Classificar cada Processo Elementar

Regras de Contagem de EE	A Regra se Aplica?
1. Possui como intenção primária: <ul style="list-style-type: none"> Manter um ou mais ALIs ou Alterar o comportamento da aplicação. 	Sim. Os dados que entram na fronteira da aplicação são utilizados eventualmente como dados de controle. São dados de negócio, armazenados no ALI de Controle de Relatório. Não.
2. Inclui a lógica de processamento para aceitar dados ou informações de controle que entram na fronteira da aplicação.	Sim. Informações de Controle de Relatórios entram na fronteira da aplicação.

Conclusão Estabelecer os Controles para Geração do Relatório das Funções Assinaladas é uma EE.

Passo 4 Contar ALR - Arquivo Lógico Referenciado (tipo de arquivo referenciado)

Regra de Contagem de ALR	A Regra se Aplica?
1. Um ALR deve ser contado para cada função de dados única que é acessada (lida e/ou mantida) pela função de transação.	O ALI de Controle de Relatório é lido e mantido, mas é contado apenas uma vez.

Passo 5 Contar DER - Dado Elementar Referenciado (tipo de dado elementar)

Regras de Contagem de DER	A Regra se Aplica?
1. Contar um DER para cada atributo não repetido, reconhecido como único pelo usuário, que atravessa (entra e/ou sai) a fronteira durante o processamento de uma função de transação.	Classificação dos Dados (<i>Sort</i>), Porta da Impressora, Tipo de Saída.
2. Contar apenas um DER por função de transação para a habilidade da aplicação de enviar uma mensagem de resposta, mesmo que existam múltiplas mensagens.	Mensagem ao Usuário.
3. Contar apenas um DER por função de transação para a habilidade de iniciar uma ou mais ações, mesmo que existam múltiplas formas para iniciá-la(s).	Botão de Comando <i>OK</i> .
<p>4. Não contar os seguintes itens como DERs:</p> <ul style="list-style-type: none"> • Literais, tais como: títulos de relatórios, identificadores de telas, cabeçalhos de colunas e títulos de atributos. • <i>Application generated stamps</i>, tais como: atributos de data e hora. • Variáveis de paginação, tais como: número de páginas e informações de posicionamento, i.e., 'Linhas 37 a 54 de 211' • Ajudas de navegação, tais como: a habilidade para navegar dentro de uma lista utilizando atalhos como "anterior", "próximo", "primeiro", "último" e suas representações gráficas equivalentes. • Atributos gerados dentro da fronteira da aplicação por uma função de transação e armazenados em um ALI sem sair da fronteira. • Atributos recuperados ou referenciados de um ALI ou AIE para serem usados no processamento sem sair da fronteira. 	<p>Nenhum dos literais existentes na tela de entrada das Informações de Controle é contado.</p> <p>Não existe nenhum item deste tipo.</p> <p>Não existe nenhum item deste tipo.</p> <p>Não existe nenhum item deste tipo.</p> <p>Não existe nenhum item deste tipo.</p> <p>Não existe nenhum item deste tipo.</p>

Passo 6 **Determinar a Complexidade Funcional**

1 ALR e 5 DERs	Complexidade é Baixa
----------------	----------------------

Passo 7 **Determinar o Tamanho Funcional**

O Tamanho Funcional é de 1 EE de Complexidade Baixa	3 PF
---	------

Exemplo: Tela de Entrada

Requisitos do Usuário

O usuário requer a habilidade de:

- Incluir as informações de uma Função em modo *online*
- Gerar uma mensagem de erro e destacar os campos incorretos de forma que o erro possa ser corrigido em modo *online*.
- Salvar as informações da função que foi incluída.

Exemplo da Tela

A tela de Dados de Função abaixo é utilizada para Incluir uma Nova Função.

Ação: _ 7=Anterior 8=Próximo 9=Salvar	
Dados da Função	
Código da Função: <u>RD15379305</u>	
Nome da Função:	<u>Analista de Negócios Sr.</u>
Faixa Salarial:	<u>SR01A</u>
Nr. Linha	Descrição da Função
<u>01</u>	<u>Analista de Negócios Sênior - faixa A</u>
F1=Ajuda F7=Subir F8=Descer F12=Cancelar	

Entra: Retorna para a tela anterior.	F1: Mostra a tela de ajuda no nível do campo ou da tela.
Ação 7: Mostra os dados da função anterior, caso exista	F7: Sobe 10 linhas de descrição.
Ação 8: Mostra os dados da próxima função, caso exista.	F8: Desce 10 linhas de descrição.
Ação 9: Salva os dados da função informados.	F12: Retorna para a tela anterior.

Passo 1 Identificar o Processo Elementar

A Função de Transação satisfaz os requisitos de um Processo Elementar?	Sim.
--	------

Passo 2 Determinar se o Processo Elementar é Único ou não

A Função de Transação é única relação aos outros processos elementares?	Sim. Nenhum outro processo elementar executa essa função.
---	---

Passo 3 Classificar cada Processo Elementar

Regras de Contagem de EE	A Regra se Aplica?
1. Possui como intenção primária: <ul style="list-style-type: none"> Manter um ou mais ALIs ou Alterar o comportamento da aplicação. 	Sim. O ALI de Funções é mantido. Não. O comportamento da aplicação não é alterado.
2. Inclui a lógica de processamento para aceitar dados ou informações de controle que entram na fronteira da aplicação.	Sim. A informação da Função entra na fronteira para que o ALI de Funções seja mantido.

Conclusão Incluir uma Nova Função é uma EE.

Consultar os Estudos de Casos para ver como os requisitos de atualizar e excluir e suas respectivas telas são contados.

Passo 4 Contar ALR - Arquivo Lógico Referenciado (tipo de arquivo referenciado)

Regra de Contagem de ALR	A Regra se Aplica?
1. Um ALR deve ser contado para cada função de dados única que é acessada (lida e/ou mantida) pela função de transação.	O ALI de Funções é mantido e lido, mas é contado apenas uma vez.

Passo 5 Contar DER - Dado Elementar Referenciado (tipo de dado elementar)

Regras de Contagem de DER	A Regra se Aplica?
1. Contar um DER para cada atributo não repetido, reconhecido como único pelo usuário, que atravessa (entra e/ou sai) a fronteira durante o processamento de uma função de transação.	Código da Função, Nome da Função, Faixa Salarial, Descrição da Função (repetido). O Número de Linhas da Descrição da Função existe apenas por razões técnicas e não deveria ser contado como um DER.
2. Contar apenas um DER por função de transação para a habilidade da aplicação de enviar uma mensagem de resposta, mesmo que existam múltiplas mensagens.	Mensagens de Erro.
3. Contar apenas um DER por função de transação para a habilidade de iniciar uma ou mais ações, mesmo que existam múltiplas formas para iniciá-la(s).	Tecla de Ação Incluir.
4. Não contar os seguintes itens como DERs: <ul style="list-style-type: none"> • Literais. • <i>Application generated stamps</i> • Variáveis de paginação • Teclas de Navegação • Atributos gerados sem sair da fronteira e • Atributos recuperados ou referenciados de um ALI ou AIE para serem usados no processamento sem sair da fronteira. 	<p>Literais como “Código da Função” não são contados.</p> <p>Não existe nenhum item deste tipo.</p> <p>Não existe nenhum item deste tipo.</p> <p>F7 e F8.</p> <p>Não existe nenhum item deste tipo.</p> <p>Não existe nenhum item deste tipo.</p>

Passo 6 Determinar a Complexidade Funcional

1 ALR e 6 DERs	Complexidade é Baixa
----------------	----------------------

Passo 7 Determinar o Tamanho Funcional

O Tamanho Funcional é de 1 EE de Complexidade Baixa	3 PF
---	------

Exemplo: Processamento Batch com Múltiplas EEs e EEs Duplicadas

Requisitos do Usuário O usuário requer a habilidade de:

- Incluir as informações de uma Função em modo *batch*
- Atualizar as informações de uma Função em modo *batch*

Nota: O foco deste exemplo é incluir uma Função em modo *batch*. O exemplo anterior apresentou a mesma função só que em modo *online*. Os Estudos de Casos ilustra a contagem de todos os requisitos do usuário para Incluir uma Função, tanto em modo *online* como em *batch*.

Requisitos de Construção Ficou decidido que, durante o processamento *batch*, qualquer Função que não for atualizada com sucesso, será gravada em um arquivo de funções suspensas, que será mantido separadamente. (Veja o próximo exemplo).

Formato dos Registros O diagrama abaixo apresenta o formato dos registros para este exemplo:

```

1234567891012345678901234567890123456789012345678901234567890
1  ADD|01|SRENG|ENGENHEIRO INFORMAÇÃO SISTEMAS SR |05|
2  ADD|02|SRENG|01|INICIO DE FAIXA DE PAGTO 05|
3  ADD|02|SRENG|02|OUTRAS FAIXAS PAGTO:06 E 07|
4  CHG|03|STENG| |04|
5  CHG|04|STENG|02|OUTRAS FAIXAS PAGTO:05 E 06|7
8
9
0

```

Descrição dos Registros A tabela abaixo inclui a descrição de cada tipo de registro.

Registro	Posição	Descrição
01	1-3	Tipo de Transação
	4-5	Tipo de Registro
	6-10	Código da Função
	11-45	Nome da Função
	46-47	Faixa Salarial
02	1-3	Tipo de Transação
	4-5	Tipo de Registro
	6-10	Código da Função
	11-12	Número de Linhas da Descrição
	13-41	Linha da Descrição

Onde os Tipos de Registros são:

- 01 Incluir registro para uma nova função
- 02 Incluir registro para as descrições de uma nova função.

Passo 1 Identificar o Processo Elementar – Transação Tipo 01

A Função de Transação satisfaz os requisitos de um Processo Elementar?	Não. Uma função sem a descrição não é significativa para o usuário.
--	---

Passo 1 Identificar o Processo Elementar – Transação Tipo 02

A Função de Transação satisfaz os requisitos de um Processo Elementar?	Não. A descrição não pode existir sem a função a que está associada. O dado ficaria inconsistente.
--	--

Passo 1 Identificar o Processo Elementar – Transação Tipo 1 + 2

A Função de Transação satisfaz os requisitos de um Processo Elementar?	Sim. Função e sua descrição são significativas para o usuário.
--	--

Passo 2 Determinar se o Processo Elementar é Único ou não

A Função de Transação é única relação aos outros processos elementares?	Sim. Incluir uma Função em modo <i>Batch</i> (Transação Tipo 1 + 2) é similar à transação de Incluir uma Função em modo <i>Online</i> . Entretanto, Incluir uma função em modo <i>Batch</i> mantém um ALI adicional (Funções Suspensas), o que a inclusão de uma função em modo <i>online</i> não faz.
---	--

Passo 3 Classificar cada Processo Elementar – Transação Tipo 1 + 2

Regras de Contagem de EE	A Regra se Aplica?
1. Possui como intenção primária: <ul style="list-style-type: none"> Manter um ou mais ALIs ou Alterar o comportamento da aplicação. 	Sim. Incluir Função tem a intenção primária de manter o ALI de Funções. Não. O comportamento da aplicação não é alterado.
2. Inclui a lógica de processamento para aceitar dados ou informações de controle que entram na fronteira da aplicação.	Sim. Incluir Função inclui lógica de processamento para aceitar as informações da Função.

Conclusão Incluir uma Função (Transação Tipo 1 + 2) é uma EE.

Passo 4 Contar ALR - Arquivo Lógico Referenciado (tipo de arquivo referenciado)

Regra de Contagem de ALR	A Regra se Aplica?
1. Um ALR deve ser contado para cada função de dados única que é acessada (lida e/ou mantida) pela função de transação.	O ALI de Funções é mantido e lido, mas é contado apenas uma vez. O ALI de Funções Suspensas é mantido.

Passo 5 Contar DER - Dado Elementar Referenciado (tipo de dado elementar)

Regras de Contagem de DER	A Regra se Aplica?
1. Contar um DER para cada atributo não repetido, reconhecido como único pelo usuário, que atravessa (entra e/ou sai) a fronteira durante o processamento de uma função de transação.	Código da Função, Nome da Função, Faixa Salarial, Descrição da Função (repetido). Tipo de Transação e o Número de Linhas da Descrição da Função existem apenas por razões técnicas e não deveriam ser contados como um DER.
2. Contar apenas um DER por função de transação para a habilidade da aplicação de enviar uma mensagem de resposta, mesmo que existam múltiplas mensagens.	Não se aplica. Os Erros são armazenados em arquivo de registros suspensos.
3. Contar apenas um DER por função de transação para a habilidade de iniciar uma ou mais ações, mesmo que existam múltiplas formas para iniciá-la(s).	Tipo de Transação.
4. Não contar os seguintes itens como DERs: <ul style="list-style-type: none"> Literais. <i>Application generated stamps</i> Variáveis de paginação Teclas de Navegação Atributos gerados sem sair da fronteira e Atributos recuperados ou referenciados de um ALI ou AIE para serem usados no processamento sem sair da fronteira. 	<p>Não existe nenhum.</p> <p>Não existe nenhum.</p> <p>Não existe nenhum.</p> <p>Não existe nenhum.</p> <p>Não existe nenhum.</p> <p>Não existe nenhum.</p>

Passo 6 Determinar a Complexidade Funcional

2 ALRs e 5 DERs	Complexidade é Média
-----------------	----------------------

Passo 7 Determinar o Tamanho Funcional

O Tamanho Funcional é de 1 EE de Complexidade Média	4 PF
---	------

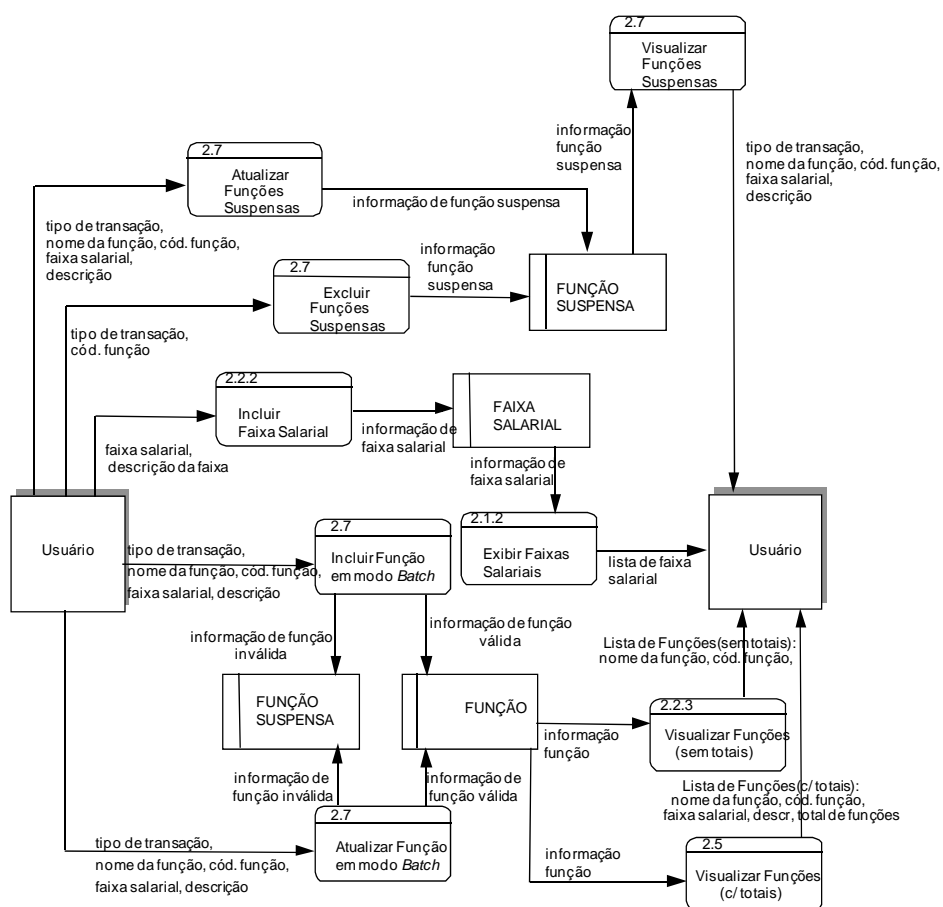
Exemplo: Correção de Transações Suspensas

Requisitos do Usuário Foi decidido que qualquer função cuja atualização não for bem sucedida durante o processamento *batch* deverá ser armazenada em um arquivo de transações suspensas. O usuário requer uma tela para acessar e editar as transações incorretas.

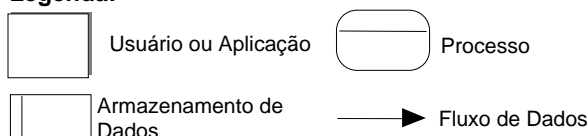
Nota: O foco deste exemplo é apenas em relação ao requisito de corrigir transações suspensas. Os Estudos de Casos ilustram a contagem do requisito completo do usuário.

Diagrama de Fluxo de Dados

O diagrama abaixo apresenta o fluxo de dados deste exemplo:



Legenda:



Passo 1 Identificar o Processo Elementar

A Função de Transação satisfaz os requisitos de um Processo Elementar?	Sim.
--	------

Passo 2 Determinar se o Processo Elementar é Único ou não

A Função de Transação é única relação aos outros processos elementares?	Sim. Nenhum outro processo elementar executa essa função.
---	---

Passo 3 Classificar cada Processo Elementar

Regras de Contagem de EE	A Regra se Aplica?
1. Possui como intenção primária: <ul style="list-style-type: none"> Manter um ou mais ALIs ou Alterar o comportamento da aplicação. 	Sim. Corrigir as Transações Suspensas de Função possui como intenção primária, manter o ALI de Funções Suspensas. Não. O comportamento da aplicação não é alterado.
2. Inclui a lógica de processamento para aceitar dados ou informações de controle que entram na fronteira da aplicação.	Sim. Incluir Função possui lógica de processamento para aceitar as informações de uma Função.

Conclusão Corrigir as Transações Suspensas de Função é uma EE.

Passo 4 Contar ALR - Arquivo Lógico Referenciado (tipo de arquivo referenciado)

Regra de Contagem de ALR	A Regra se Aplica?
1. Um ALR deve ser contado para cada função de dados única que é acessada (lida e/ou mantida) pela função de transação.	O ALI de Funções Suspensas é mantido e referenciado, mas é contado apenas uma vez.

Passo 5 Contar DER - Dado Elementar Referenciado (tipo de dado elementar)

Regras de Contagem de DER	A Regra se Aplica?
1. Contar um DER para cada atributo não repetido, reconhecido como único pelo usuário, que atravessa (entra e/ou sai) a fronteira durante o processamento de uma função de transação.	Tipo de Transação, Código da Função, Nome da Função, Faixa Salarial, Descrição da Função (repetido). O Tipo de Registro o Número de Linhas da Descrição da Função existem apenas por razões técnicas e, portanto, não são contados como um DER. Todos os outros campos são reconhecidos pelo usuário.
2. Contar apenas um DER por função de transação para a habilidade da aplicação de enviar uma mensagem de resposta, mesmo que existam múltiplas mensagens.	Não existe nenhuma mensagem.
3. Contar apenas um DER por função de transação para a habilidade de iniciar uma ou mais ações, mesmo que existam múltiplas formas para iniciá-la(s).	Tecla Entra.
4. Não contar os seguintes itens como DERs: <ul style="list-style-type: none"> • Literais. • <i>Application generated stamps</i> • Variáveis de paginação • Teclas de Navegação • Atributos gerados sem sair da fronteira e • Atributos recuperados ou referenciados de um ALI ou AIE para serem usados no processamento sem sair da fronteira. 	<p>Não existe nenhum.</p> <p>Não existe nenhum.</p> <p>Não existe nenhum.</p> <p>Não existe nenhum.</p> <p>Não existe nenhum.</p> <p>Não existe nenhum.</p>

Passo 6 Determinar a Complexidade Funcional

1 ALR e 6 DERs	Complexidade é Baixa.
----------------	-----------------------

Passo 7 Determinar o Tamanho Funcional

O Tamanho Funcional é de 1 EE de Complexidade Baixa	3 PF
---	------

Exemplo: EE com Múltiplos Arquivos Lógicos Referenciados

Requisitos do Usuário O usuário requer a habilidade de Incluir Assinalamentos de Função.

Nota: O foco deste exemplo é mostrar apenas incluir assinalamentos de função. Os Estudos de Casos ilustram a contagem do requisito completo do usuário.

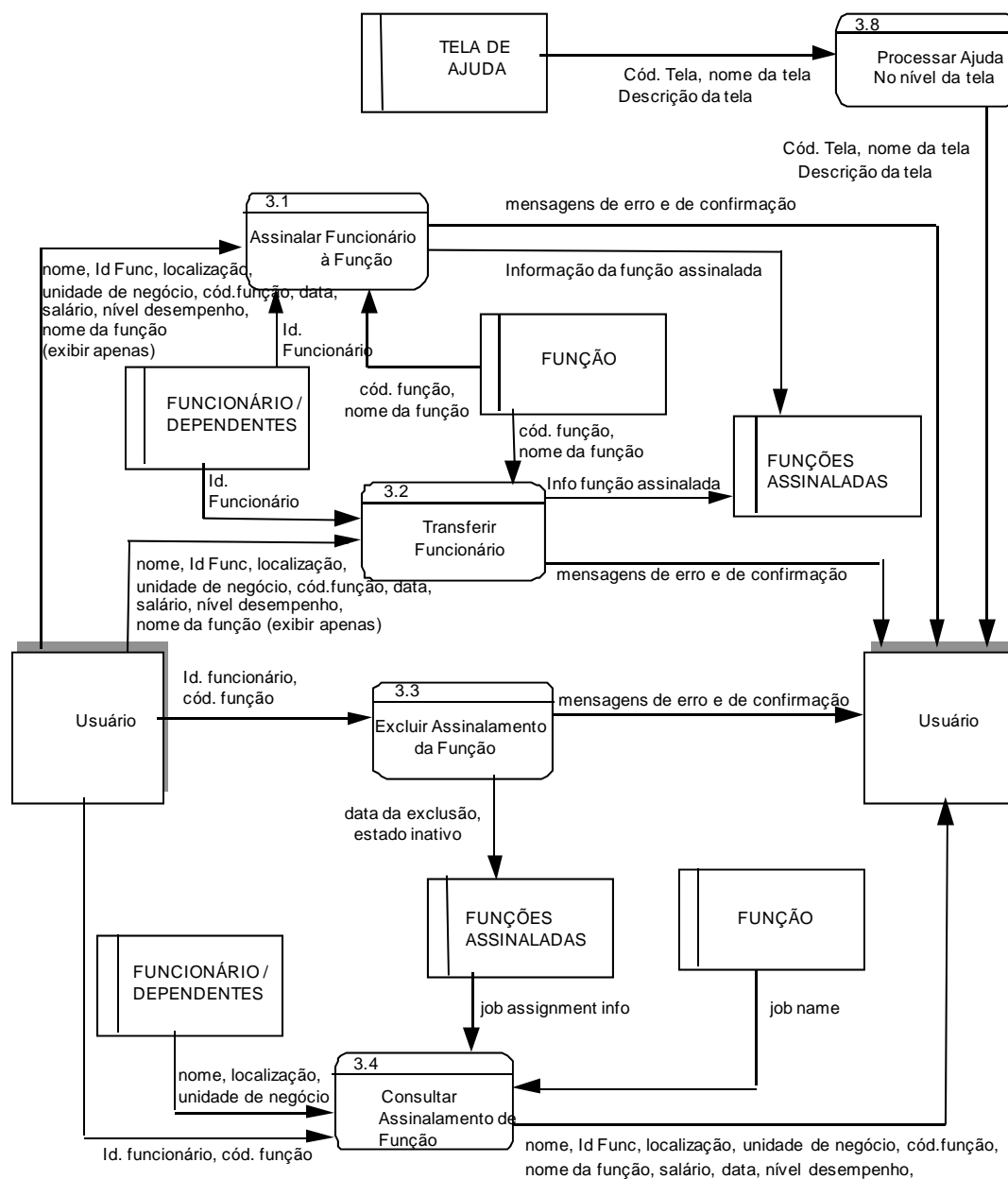
Exemplo da Tela O diagrama a seguir mostra um exemplo da janela de assinalamento de função a um funcionário.

AF-3 Iniciar Assinalamento de Função

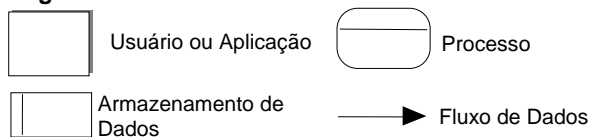
- | | |
|-----------------|--|
| OK | Salva o novo assinalamento de função e retorna para a tela AF-1 Lista de Assinalamento de Funcionários |
| Cancelar | Ignora os dados informados e retorna para a tela AF-1 Lista de Assinalamento de Funcionários |

Diagrama de Fluxos de Dados

O diagrama abaixo mostra o fluxo de dados para o processo de assinalamento de funções.



Legenda:



Passo 1 Identificar o Processo Elementar

A Função de Transação satisfaz os requisitos de um Processo Elementar?	Sim.
--	------

Passo 2 Determinar se o Processo Elementar é Único ou não

A Função de Transação é única relação aos outros processos elementares?	Sim. Nenhum outro processo elementar executa essa função.
---	---

Passo 3 Classificar cada Processo Elementar

Regras de Contagem de EE	A Regra se Aplica?
1. Possui como intenção primária: <ul style="list-style-type: none"> Manter um ou mais ALIs ou Alterar o comportamento da aplicação. 	Sim. Incluir Assinalamentos de Função tem como intenção primária, manter o ALI de Funções Assinaladas. Não. O comportamento da aplicação não é alterado.
2. Inclui a lógica de processamento para aceitar dados ou informações de controle que entram na fronteira da aplicação.	Sim. Incluir o Assinalamento de Função possui lógica de processamento para aceitar as informações de Assinalamento de Função.

Conclusão Incluir Assinalamentos de Função é uma EE.

Passo 4 Contar ALR - Arquivo Lógico Referenciado (tipo de arquivo referenciado)

Regra de Contagem de ALR	A Regra se Aplica?
1. Um ALR deve ser contado para cada função de dados única que é acessada (lida e/ou mantida) pela função de transação.	O ALI Funcionário é lido para garantir que o funcionário existe. O ALI de Funções é lido para garantir que a função existe. O ALI de Funções Assinaladas é mantido e lido, mas é contado apenas uma vez.

Passo 5 Contar DER - Dado Elementar Referenciado (tipo de dado elementar)

Regras de Contagem de DER	A Regra se Aplica?
1. Contar um DER para cada atributo não repetido, reconhecido como único pelo usuário, que atravessa (entra e/ou sai) a fronteira durante o processamento de uma função de transação.	Nome do Funcionário, Identificação do Funcionário, Localização, Unidade de Negócio, Código da Função, Nome da Função, Data de Efetivação, Salário, Avaliação de Desempenho. A tela exibe o Nome do Funcionário com três campos físicos. Entretanto, o diagrama de fluxo de dados trata como um único elemento de dados. Baseado na revisão da funcionalidade da aplicação, o Nome do Funcionário sempre é usado na sua totalidade. Não existe nenhuma tela ou relatório que utilize parte do nome separadamente. Por isto, será contado como um único DER.
2. Contar apenas um DER por função de transação para a habilidade da aplicação de enviar uma mensagem de resposta, mesmo que existam múltiplas mensagens.	Mensagens de Erro.
3. Contar apenas um DER por função de transação para a habilidade de iniciar uma ou mais ações, mesmo que existam múltiplas formas para iniciá-la(s).	Botão de Comando OK.
4. Não contar os seguintes itens como DERs: <ul style="list-style-type: none"> Literais. <i>Application generated stamps</i> Variáveis de paginação Teclas de Navegação Atributos gerados sem sair da fronteira e Atributos recuperados ou referenciados de um ALI ou AIE para serem usados no processamento sem sair da fronteira. 	Literais, tais como: "Código da Função" não são contados. Não existe nenhum. Não existe nenhum. Não existe nenhum. Não existe nenhum. Não existe nenhum.

Passo 6 Determinar a Complexidade Funcional

3 ALRs e 11 DERs	Complexidade é Alta
------------------	---------------------

Passo 7 Determinar o Tamanho Funcional

O Tamanho Funcional é de 1 EE de Complexidade Alta	6 PF
--	------

Exemplo: Conversão de Dados

Requisitos do Usuário

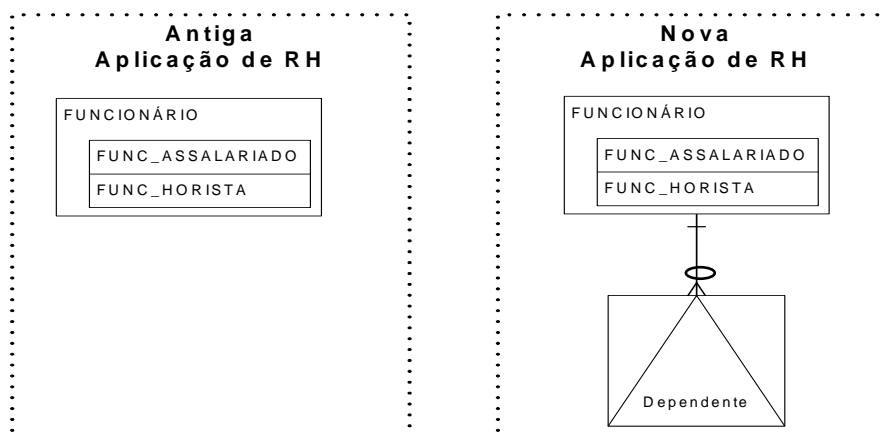
O usuário comprou um novo pacote de uma aplicação de RH. Ele requer a habilidade para Conversão das Informações do Funcionário pela migração das informações existentes dos funcionários (Nome, Identificação do Funcionário, Número de Dependentes, Código de Tipo, Nível de Supervisão, Taxa Padrão por Hora, Unidade de Negócio, Nome do Local) para a nova aplicação.

O sistema antigo não permitia ao usuário a manutenção das informações dos dependentes dos funcionários. As informações dos dependentes poderão ser criadas após a migração dos dados dos funcionários existentes para a nova aplicação.

Nota: O Capítulo 5 da Parte 3 (Atividades de Conversão de Dados) explica como a conversão de dados é mensurada.

Diagramas de Dados

O diagrama abaixo mostra os dados das aplicações de RH nova e antiga:



Legenda:

- Tipo de Entidade
- Tipo de Atributo de Entidade
- Subtipo de Entidade
- Relacionamento Mandatário 1:N
- Relacionamento Opcional 1:N

Passo 1 Identificar o Processo Elementar

A Função de Transação satisfaz os requisitos de um Processo Elementar?	Sim.
--	------

Passo 2 Determinar se o Processo Elementar é Único ou não

A Função de Transação é única relação aos outros processos elementares?	Sim. Nenhum outro processo elementar executa essa função.
---	---

Passo 3 Classificar cada Processo Elementar

Regras de Contagem de EE	A Regra se Aplica?
1. Possui como intenção primária: <ul style="list-style-type: none"> Manter um ou mais ALIs ou Alterar o comportamento da aplicação. 	Sim. O ALI Funcionário é mantido. Não. O comportamento da aplicação não é alterado.
2. Inclui a lógica de processamento para aceitar dados ou informações de controle que entram na fronteira da aplicação.	Sim. Dados do arquivo de funcionários da aplicação de RH antiga atravessam a fronteira da aplicação.

Conclusão Conversão das Informações do Funcionário é uma EE.

Passo 4 Contar ALR - Arquivo Lógico Referenciado (tipo de arquivo referenciado)

Regra de Contagem de ALR	A Regra se Aplica?
1. Um ALR deve ser contado para cada função de dados única que é acessada (lida e/ou mantida) pela função de transação.	O ALI Funcionários é mantido e lido, mas é contado apenas uma vez.

Passo 5 Contar DER - Dado Elementar Referenciado (tipo de dado elementar)

Regras de Contagem de DER	A Regra se Aplica?
1. Contar um DER para cada atributo não repetido, reconhecido como único pelo usuário, que atravessa (entra e/ou sai) a fronteira durante o processamento de uma função de transação.	Nome, Identificação do Funcionário, Número de Dependentes, Código de Tipo, Nível de Supervisão, Taxa Padrão por Hora, Unidade de Negócio, Nome do Local.
2. Contar apenas um DER por função de transação para a habilidade da aplicação de enviar uma mensagem de resposta, mesmo que existam múltiplas mensagens.	Não existe nenhum.
3. Contar apenas um DER por função de transação para a habilidade de iniciar uma ou mais ações, mesmo que existam múltiplas formas para iniciá-la(s).	Não existe nenhum.
4. Não contar os seguintes itens como DERs: <ul style="list-style-type: none"> • Literais. • <i>Application generated stamps</i> • Variáveis de paginação • Teclas de Navegação • Atributos gerados sem sair da fronteira e • Atributos recuperados ou referenciados de um ALI ou AIE para serem usados no processamento sem sair da fronteira. 	<p>Não existe nenhum.</p> <p>Não existe nenhum.</p> <p>Não existe nenhum.</p> <p>Não existe nenhum.</p> <p>Não existe nenhum.</p> <p>Não existe nenhum.</p>

Passo 6 Determinar a Complexidade Funcional

1 ALR e 8 DERs	Complexidade é Baixa
----------------	----------------------

Passo 7 Determinar o Tamanho Funcional

O Tamanho Funcional é de 1 EE de Complexidade Baixa	3 PF
---	------

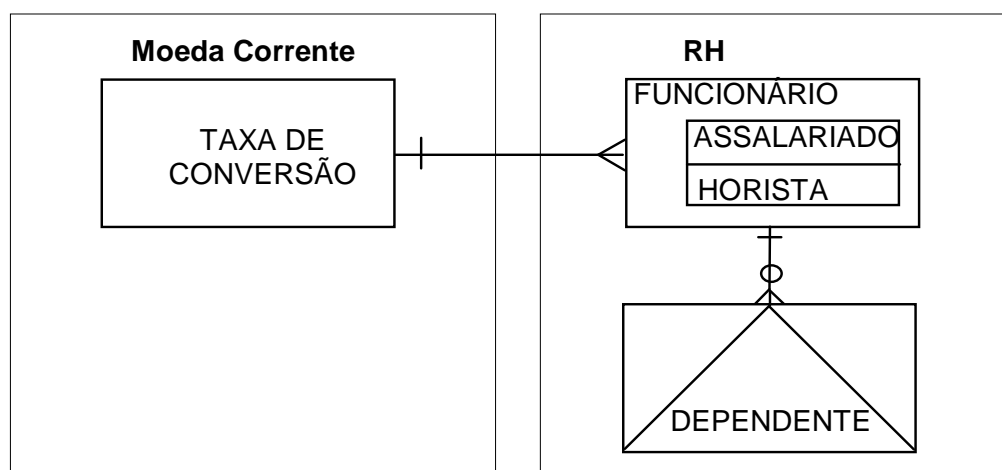
Exemplo: Referenciando Dados a partir de Outra Aplicação

Requisitos do Usuário O usuário requer que a aplicação de Recursos Humanos tenha as seguintes capacidades:






- Todos os funcionários horistas devem ser pagos em Dólares Americanos.
- Quando as informações do funcionário são incluídas ou modificadas, a aplicação de Recursos Humanos deve acessar a aplicação de Moedas Correntes para recuperar a taxa de conversão da moeda. Depois de recuperar a taxa de conversão da moeda, a aplicação de RH converte a taxa padrão de horas local do funcionário para a taxa de horas em Dólares Americanos, utilizando o seguinte cálculo:

$$\frac{\text{Taxa Padrão de Horas}}{\text{Taxa de Conv. Moeda.}} = \text{Taxa de Horas em Dólares Americanos}$$

Diagrama de Dados O diagrama a seguir apresenta o relacionamento para este exemplo.



Legenda:

	Tipo de Entidade
	Tipo de Entidade Atributiva
	Entidade Subtipo
	Relacionamento 1-N Mandatário
	Relacionamento 1-N Opcional

Informação de Conversão de Moeda

A informação de conversão de moeda inclui
MOEDA

- Taxa_Base_Conversão_Moeda
- Moeda

Passo 1 Identificar o Processo Elementar

A Função de Transação atende os requisitos de um Processo Elementar?	Não. Dados de referência são significativos apenas quando associados à inclusão de um funcionário.
--	--

Conclusão

Não existe uma EE para recuperação das informações de conversão de moeda. Veja os exemplos de contagem de AIE nos Exemplos de Contagem de Funções de Dados para verificar porque as informações de conversão de moeda podem ser contadas como AIE quando as informações do funcionário são incluídas ou modificadas. Incluir ou Modificar informações de Funcionário são contados como EEs.

Exemplo: EE com Tela de Saída – 1

Requisitos do Usuário O usuário requer a habilidade de Entrar com uma Transação de Vendas para um cliente. O custo de cada item e o total da transação devem ser exibidos para revisão, antes de a informação ser salva. Se qualquer erro ocorrer, uma mensagem de erro apropriada deve ser exibida.

Exemplo de Tela A seguinte tela de transação de vendas é uma simplificação para ilustrar como os campos de saída são contados. O usuário entra com o nome do cliente e a data da transação. Quando cada item e quantidade requerida são incluídos, o sistema calcula e mostra os custos como apresentado abaixo.

Transação de Vendas			
Nome do Cliente: _____			
Data da Transação: _____			
Item	Qty	Custo Item	Custo Total Item
_____	_____	\$ _____.____	\$ _____.____
_____	_____	\$ _____.____	\$ _____.____
_____	_____	\$ _____.____	\$ _____.____
_____	_____	\$ _____.____	\$ _____.____
_____	_____	\$ _____.____	\$ _____.____
_____	_____	\$ _____.____	\$ _____.____
		Sub Total	\$ _____.____
		Taxa Vendas	\$ _____.____
		Total	\$ _____.____
F1=Salvar			

Passo 1 Identificar o Processo Elementar

A Função de Transação atende aos requisitos de um Processo Elementar?	Sim.
---	------

Passo 2 Determinar se o Processo Elementar é único

A Função de Transação é única em relação a outros processos elementares?	Sim. Nenhum outro processo elementar executa esta função.
--	---

Passo 3 Classificar cada Processo Elementar

Regras de Contagem de EE	A Regra se Aplica?
1. Possui como intenção primária: <ul style="list-style-type: none"> Manter um ou mais ALIs ou Alterar o comportamento da aplicação 	Sim. Entrar com uma Transação de Vendas possui a intenção primária de manter o ALI Transação de Vendas Não. O comportamento da aplicação não é alterado.
2. Inclui lógica de processamento para aceitar dados ou controle de informação que entra na fronteira da aplicação.	Sim. Entrar com uma Transação de Vendas inclui lógica de processamento para aceitar informação de vendas.

Conclusão Entrar com uma Transação de Vendas é uma EE.

Passo 4 Contar ALR - Arquivo Lógico Referenciado (tipo de arquivo referenciado)

Regra de Contagem de ALR	A Regra se Aplica?
1. Um ALR deve ser contado para cada função de dados única que é acessada (lida e/ou mantida) pela função de transação.	O ALI Transação de Vendas é lido e mantido, mas é contado apenas uma vez.

Passo 5 Contar DER - Dado Elementar Referenciado (tipo de dado elementar)

Regras de Contagem de DER	A Regra se Aplica?
1. Contar um DER para cada atributo único, reconhecido pelo usuário e não repetido, que atravessa (entra e/ou sai) a fronteira durante o processamento de uma função de transação.	Os seguintes DERs de entrada são contados: Nome do Cliente Data da Transação Item (repetido) Quantidade (repetido) Os seguintes DERs de saída são contados: Custo do Item (repetido) Custo Total do Item (repetido) Sub Total da Transação Taxa de Vendas Total da Transação
2. Contar apenas um DER para a função de transação, para a habilidade da aplicação de enviar uma mensagem de resposta, mesmo que existam múltiplas mensagens.	Mensagens de Erro.
3. Contar apenas um DER para a função de transação, para a habilidade de iniciar ação(ões), mesmo que existam múltiplas maneiras de fazê-lo.	Tecla F1.
4. Não contar os seguintes itens como DERs: <ul style="list-style-type: none"> Literais <i>application generated stamps</i> Variáveis de paginação Navegação Atributos gerados sem saída da fronteira e Atributos recuperados ou referenciados de um ALI ou AIE sem saída da fronteira 	Literais como "Item" não são contados Não existem. Não existem. Não existem. Não existem. Não existem.

Passo 6 Determinar a Complexidade Funcional

1 ALR e 11 DERs	Complexidade é Baixa
-----------------	----------------------

Passo 7 Determinar o Tamanho Funcional

Tamanho Funcional de 1 EE de Média Complexidade	3 PF
---	------

Exemplo: EE com Tela de Saída - 2

Requisitos do Usuário O usuário requer a habilidade de Assinalar uma Função a um funcionário. Para selecionar um funcionário e função, o usuário requer a habilidade de referenciar o funcionário e arquivos de funções utilizando 2 listas de *drop-down*. A lista de funcionários é requerida para apresentar o código e nome do funcionário. A lista de funções é requerida para apresentar o código e descrição da função. O código dos funcionários assinalados à função é exibido depois do registro salvo. No caso de erro, uma mensagem apropriada é exibida.

Exemplo de Tela A seguinte tela de Funções Assinaladas é uma simplificação de como campos de saída são contados. O usuário seleciona o funcionário de uma lista de *drop-down*, apresentando o nome e código do funcionário. Na seleção, o sistema requer o código do funcionário para o assinalamento. O usuário seleciona a função de uma lista de *drop-down* apresentando o código e descrição da função. O sistema requer o código da função para o assinalamento. Quando o assinalamento é salvo, o sistema determina o número total de funcionários e o apresenta ao usuário.

Sistema de Recursos Humanos			
Funcionários	Funções	Assinalamentos	Localizações
Funções Assinaladas			
Cod. do Funcionário	<input type="text" value="1290"/>	<input type="text" value="1290 James, R.W"/>	<input checked="" type="checkbox"/>
Código da Função	<input type="text" value="0100"/>	<input type="text" value="0100 Apply Lacquer"/>	<input checked="" type="checkbox"/>
Data do	<input type="text" value="12/12/1998"/>		
<input type="button" value="Salvar"/>			
Número total de funcionários assinalados a esta			<input type="text" value="3"/>

As listas de *drop-down* para Funções e Funcionários são CEs e não são analisadas neste exemplo.

Passo 1 Identificar o Processo Elementar

A Função de Transação atende aos requisitos de um Processo Elementar?	Sim.
---	------

Passo 2 Determinar se o Processo Elementar é único

A Função de Transação é única em relação a outros processos elementares?	Sim. Nenhum outro processo elementar executa esta função.
--	---

Passo 3 Classificar cada Processo Elementar

Regras de Contagem de EE	A Regra se Aplica?
1. Possui como intenção primária: <ul style="list-style-type: none"> Manter um ou mais ALIs ou Alterar o comportamento da aplicação 	<p>Sim. Assinalar uma Função possui a intenção primária de manter o ALI de Funções Assinaladas.</p> <p>Não. O comportamento da aplicação não é alterado.</p>
2. Inclui lógica de processamento para aceitar dados ou controle de informação que entra na fronteira da aplicação.	Sim. Assinalar uma Função inclui lógica de processamento para aceitar informações de Funções Assinaladas

Conclusão Assinalar uma Função é uma EE.

Passo 4 Contar ALR - Arquivo Lógico Referenciado (tipo de arquivo referenciado)

Regra de Contagem de ALR	A Regra se Aplica?
1. Um ALR deve ser contado para cada função de dados única que é acessada (lida e/ou mantida) pela função de transação.	<p>O ALI de Funções Assinaladas é mantido.</p> <p>Os ALIs de Funcionário e Função não são contados como ALRs, uma vez que são parte de CEs separadas.</p>

Passo 5 Contar DER - Dado Elementar Referenciado (tipo de dado elementar)

Regras de Contagem de DER	A Regra se Aplica?
1. Contar um DER para cada atributo único, reconhecido pelo usuário e não repetido, que atravessa (entra e/ou sai) a fronteira durante o processamento de uma função de transação.	Os seguintes DERs de entrada são contados: Código do Funcionário Código da Função Data do Assinalamento Os seguintes DERs de saída são contados. Funcionário Assinalado à Função Os DERs de Nome de Funcionário e Nome da Função nas listas de <i>drop-down</i> não são contados como DERs, uma vez que são parte de CEs separados.
2. Contar apenas um DER para a função de transação, para a habilidade da aplicação de enviar uma mensagem de resposta, mesmo que existam múltiplas mensagens.	Uma mensagem é retornada em caso de erro.
3. Contar apenas um DER para a função de transação, para a habilidade de iniciar ação(ões), mesmo que existam múltiplas maneiras de fazê-lo.	Existe apenas uma maneira da função ser invocada, através do botão Salvar.
4. Não contar os seguintes itens como DERs: <ul style="list-style-type: none"> • Literais • <i>application generated stamps</i> • Variáveis de paginação • Navegação • Atributos gerados sem saída da fronteira e • Atributos recuperados ou referenciados de um ALI ou AIE sem saída da fronteira 	Literais como “Código da Função” não são contados Não existem. Não existem. Não existem. Não existem. Não existem.

Passo 6 Determinar a Complexidade Funcional

1 ALR e 6 DERs	Complexidade é Baixa
----------------	----------------------

Passo 7 Determinar o Tamanho Funcional

Tamanho Funcional de 1 EE de Baixa Complexidade	3 PF
---	------

Exemplo: EE com Atributos Recuperados de um AIE

Requisitos do Usuário

O usuário requer a habilidade de Incluir um funcionário entrando com:

- Informação de funcionário
- Informação de salário ou taxa por hora
- Informação de dependente
- A localização deve ser uma localização válida do Sistema de Ativos Fixos.
- A taxa por hora é convertida para Dólares Americanos; os dados de moeda são acessados a partir do Sistema de Moedas Correntes, para converter uma taxa por hora padrão em Dólares Americanos, baseado na moeda do funcionário.

Exemplo de Telas

Os seguintes diagramas são exemplos das janelas para inclusão de um funcionário.

Sistema de Recursos Humanos

Funcionário Funções Assinalamentos Localizações Relatórios Segurança Ajuda

Dados de Funcionário

Sobrenome

Primeiro Nome

Iniciais do Meio

Número de Identificação

Número de Dependentes

Localização

☐ Horista
☐ Assalariado

EN-1 Dados de Funcionário

<input type="button" value="Nível de Supervisão"/>	Navegar para SL-1 Nível do Supervisor
<input type="button" value="Nova Unidade de Negócios"/>	Navegar para BU-1 Unidade de Negócios
<input type="button" value="OK"/>	Navegar para próxima tela: • EN-2H, se tipo de salário horista for selecionado • EN-2S, se tipo de salário assalariado for selecionado
<input type="button" value="Cancelar"/>	Retorna para menu

Sistema de Recursos Humanos		
Funcionário Funções Assinalamentos Localizações Relatórios Segurança Ajuda		
<div style="border: 1px solid black; padding: 5px; margin-bottom: 5px;"> <div style="border-bottom: 1px solid black; padding-bottom: 5px; margin-bottom: 5px;"> <div style="border: 1px solid black; width: 10px; height: 10px; float: left; margin-right: 5px;"></div> <div style="margin-left: 5px;">Dados de Funcionário</div> </div> <div style="padding: 5px;"> <p>Sobrenome <input style="width: 150px;" type="text"/></p> <p>Primeiro Nome <input style="width: 100px;" type="text"/></p> <p>Iniciais do Meio <input style="width: 40px;" type="text"/></p> </div> </div> <div style="border: 1px solid black; padding: 5px; margin-bottom: 5px;"> <div style="border-bottom: 1px solid black; padding-bottom: 5px; margin-bottom: 5px;"> <div style="border: 1px solid black; width: 10px; height: 10px; float: left; margin-right: 5px;"></div> <div style="margin-left: 5px;">Dados de Funcionário Assalariado</div> </div> <div style="padding: 5px;"> <p>Nível do Supervisor Gerente de 1a. Linha </p> </div> <div style="text-align: right; padding-top: 10px;"> <div style="border: 1px solid black; padding: 2px 10px; margin-bottom: 2px;">OK</div> <div style="border: 1px solid black; padding: 2px 10px; margin-bottom: 2px;">Cancelar</div> <div style="border: 1px solid black; padding: 2px 10px;">Dependentes</div> </div> </div>		

EN-2S Dados de Funcionário: Assalariado

- | | |
|---|---|
| <div style="border: 1px solid black; padding: 2px 10px; background-color: #ffff00;">OK</div> | Salva dados de novo funcionário e retorna para EN-1 Dados de Funcionário |
| <div style="border: 1px solid black; padding: 2px 10px; background-color: #ffff00;">Cancelar</div> | Retorna para EN-1 Dados de Funcionário (sem salvar dados) |
| <div style="border: 1px solid black; padding: 2px 10px; background-color: #ffff00;">Dependentes</div> | Navega para EN-3 Dados de Dependente de Funcionário
(dado em EN-2S Dados de Salário de Funcionário é mantido em memória) |

Sistema de Recursos Humanos		
Funcionário Funções Assinalamentos Localizações Relatórios Segurança Ajuda		
<div style="border: 1px solid black; padding: 5px; margin-bottom: 5px;"> <div style="border-bottom: 1px solid black; padding-bottom: 5px; margin-bottom: 5px;"> <div style="border: 1px solid black; width: 10px; height: 10px; float: left; margin-right: 5px;"></div> <div style="margin-left: 5px;">Dados de Funcionário</div> </div> <div style="padding: 5px;"> <p>Sobrenome <input style="width: 150px;" type="text"/></p> <p>Primeiro Nome <input style="width: 100px;" type="text"/></p> <p>Iniciais do Meio <input style="width: 40px;" type="text"/></p> </div> </div> <div style="border: 1px solid black; padding: 5px; margin-bottom: 5px;"> <div style="border-bottom: 1px solid black; padding-bottom: 5px; margin-bottom: 5px;"> <div style="border: 1px solid black; width: 10px; height: 10px; float: left; margin-right: 5px;"></div> <div style="margin-left: 5px;">Dados de Funcionário Horista</div> </div> <div style="padding: 5px;"> <p>Taxa por Hora <input style="width: 100px;" type="text"/></p> <p>Unidade de Negócio UFPCA </p> </div> <div style="text-align: right; padding-top: 10px;"> <div style="border: 1px solid black; padding: 2px 10px; margin-bottom: 2px;">OK</div> <div style="border: 1px solid black; padding: 2px 10px; margin-bottom: 2px;">Cancelar</div> <div style="border: 1px solid black; padding: 2px 10px;">Dependentes</div> </div> </div>		

EN-2H Dados de Funcionário: Horista

- | | |
|---|--|
| <div style="border: 1px solid black; padding: 2px 10px; background-color: #ffff00;">OK</div> | Salva dados de novo funcionário e retorna para EN-1 Dados de Funcionário |
| <div style="border: 1px solid black; padding: 2px 10px; background-color: #ffff00;">Cancelar</div> | Retorna para EN-1 Dados de Funcionário (sem salvar dados) |
| <div style="border: 1px solid black; padding: 2px 10px; background-color: #ffff00;">Dependentes</div> | Navega para EN-3 Dados de Dependente de Funcionário
(dado em EN-2H Dados de Funcionário Horista é mantido em memória) |

EN-3 Dados de Dependente de Funcionário

OK	Salva dados de novo dependente data e retorna para EN-1 Dados de Funcionário
Cancelar	Retorna para EN-2H Dados de Funcionário Horista ou EN-2S Dados de Funcionário Assalariado (sem salvar dados)
Mais Dependentes	Aceita dados de dependente e retorna para EN-3 Dados de Dependente de Funcionário

Passo 1 Identificar o Processo Elementar

A Função de Transação atende aos requisitos de um Processo Elementar?	Sim.
---	------

Passo 2 Determinar se o Processo Elementar é único

A Função de Transação é única em relação a outros processos elementares?	Sim. Nenhum outro processo elementar executa esta função.
--	---

Passo 3 Classificar cada Processo Elementar

Regras de Contagem de EE	A Regra se Aplica?
1. Possui como intenção primária: <ul style="list-style-type: none"> Manter um ou mais ALIs ou Alterar o comportamento da aplicação 	Sim. Incluir um Funcionário possui a intenção primária de manter o ALI Funcionário. Não. O comportamento da aplicação não é alterado.
2. Inclui lógica de processamento para aceitar dados ou controle de informação que entra na fronteira da aplicação.	Sim. Incluir um Funcionário inclui lógica de processamento para aceitar informação de Funcionário.

Conclusão Incluir um Funcionário é uma EE.

Passo 4 Contar ALR - Arquivo Lógico Referenciado (tipo de arquivo referenciado)

Regra de Contagem de ALR	A Regra se Aplica?
1. Um ALR deve ser contado para cada função de dados única que é acessada (lida e/ou mantida) pela função de transação.	O ALI Funcionário é mantido. Os AIEs de Moeda Corrente e Localização são referenciados.

Passo 5 Contar DER - Dado Elementar Referenciado (tipo de dado elementar)

Regras de Contagem de DER	A Regra se Aplica?
1. Contar um DER para cada atributo único, reconhecido pelo usuário e não repetido, que atravessa (entra e/ou sai) a fronteira durante o processamento de uma função de transação.	<p>Nome (Sobrenome, Primeiro Nome, Iniciais do Meio), Número de Identificação, Número de Dependentes, Localização, Tipo de Salário, Nível do Supervisor, Taxa por Hora, Unidade de Negócio, Nome do Dependente (Sobrenome, Primeiro Nome, Iniciais do Meio), Identificação do Dependente, Data de Nascimento do Dependente.</p> <p>A tela divide o Nome do Funcionário em três campos físicos. Baseado na revisão da funcionalidade da aplicação, o Nome do Funcionário é sempre usado na sua totalidade. Não existem telas ou relatórios onde somente um pedaço do nome é usado sem os demais. O mesmo é também verdadeiro para o Nome do Dependente. O Nome do Funcionário e o Nome do Dependente são contados como DERs únicos.</p>
2. Contar apenas um DER para a função de transação, para a habilidade da aplicação de enviar uma mensagem de resposta, mesmo que existam múltiplas mensagens.	Uma mensagem é retornada em caso de erro.
3. Contar apenas um DER para a função de transação, para a habilidade de iniciar ação(ões), mesmo que existam múltiplas maneiras de fazê-lo.	Botão OK.
<p>4. Não contar os seguintes itens como DERs:</p> <ul style="list-style-type: none"> • Literais • <i>application generated stamps</i> • Variáveis de paginação • Navegação • Atributos gerados sem saída da fronteira e • Atributos recuperados ou referenciados de um ALI ou AIE sem saída da fronteira 	<p>Literais como "Sobrenome" não são contados.</p> <p>Não existem.</p> <p>Não existem.</p> <p>Não existem.</p> <p>Taxa por Hora em Dólares Americanos é calculada, mas não sai da fronteira.</p> <p>Taxa de Conversão para Moeda Base é recuperado do AIE Moeda Corrente para calcular a Taxa por Hora em Dólares Americanos, mas não sai da fronteira.</p>

Passo 6 **Determinar a Complexidade Funcional**

3 ARL e 13 DERs	Complexidade é Alta
-----------------	---------------------

Passo 7 **Determinar o Tamanho Funcional**

Tamanho Funcional de 1 EE de Alta Complexidade	6 PF
--	------

Exemplo: EE Excluir

Requisitos do Usuário O usuário requer a habilidade de Excluir um Funcionário.

Excluir todas as informações sobre um Funcionário específico. Ao excluir um funcionário atualmente assinalado a uma função, atualizar o assinalamento da função, configurando o status para inativo.

- Gerar mensagens de erro e destacar campos incorretos se os campos não permitirem edição. Cinco (5) mensagens de erro e uma (1) mensagem de confirmação estão incluídas na transação de exclusão de informações do funcionário.
- Ao excluir um funcionário, atualizar o Status Inativo com um “X” e colocar a data do sistema em Data de Efetivação para cada Função Assinalada associada.

Exemplo de Tela O diagrama seguinte é um exemplo de janela utilizada para excluir um funcionário.

EE2 – Editar Funcionário

OK	Salva alterações realizadas retorna para EE -1 Lista de Funcionários	Excluir	Pedir confirmação, então excluir registro de funcionário & registros de dependentes associados
Cancelar	Ignora alterações, retorna para EE -1 Lista de Funcionários	Dados Salariais	Navegar para: EE-3S Funcionário Assalariado, se o valor é "Assalariado" EE-3H Funcionário Horista, se o valor é "Horista"
Recuperar	Recupera valores anteriores	Dependentes	Apresenta lista em EI-4 Lista de Dependentes
Nível de Supervisão	Navega para SL-1 Nível de Supervisão		
Unidade de Negócios	Navega para BU-1 Unidade de Negócios		

Passo 1 Identificar o Processo Elementar

A Função de Transação atende aos requisitos de um Processo Elementar?	Sim.
---	------

Passo 2 Determinar se o Processo Elementar é único

A Função de Transação é única em relação a outros processos elementares?	Sim. Nenhum outro processo elementar executa esta função.
--	---

Passo 3 Classificar cada Processo Elementar

Regras de Contagem de EE	A Regra se Aplica?
1. Possui como intenção primária: <ul style="list-style-type: none"> Manter um ou mais ALIs ou Alterar o comportamento da aplicação 	Sim. Excluir um Funcionário possui a intenção primária de manter o ALI Funcionário. Não. O comportamento da aplicação não é alterado.
2. Inclui lógica de processamento para aceitar dados ou controle de informação que entra na fronteira da aplicação.	Sim. Excluir um Funcionário inclui lógica de processamento para aceitar informações de Funcionário.

Conclusão Excluir um Funcionário é uma EE.

Passo 4 Contar ALR - Arquivo Lógico Referenciado (tipo de arquivo referenciado)

Regra de Contagem de ALR	A Regra se Aplica?
1. Um ALR deve ser contado para cada função de dados única que é acessada (lida e/ou mantida) pela função de transação.	O ALI Funções Assinaladas é mantido. O ALI Funcionários é referenciado e mantido.

Passo 5 Contar DER - Dado Elementar Referenciado (tipo de dado elementar)

Regras de Contagem de DER	A Regra se Aplica?
1. Contar um DER para cada atributo único, reconhecido pelo usuário e não repetido, que atravessa (entra e/ou sai) a fronteira durante o processamento de uma função de transação.	Número de Identidade. Todos os outros atributos na tela são parte da pesquisa que precede à Exclusão. Eles não são contados como DERs para a Deleção.
2. Contar apenas um DER para a função de transação, para a habilidade da aplicação de enviar uma mensagem de resposta, mesmo que existam múltiplas mensagens.	Uma mensagem é retornada em caso de erro.
3. Contar apenas um DER para a função de transação, para a habilidade de iniciar ação(ões), mesmo que existam múltiplas maneiras de fazê-lo.	Botão Excluir.
4. Não contar os seguintes itens como DERs: <ul style="list-style-type: none"> • Literais • <i>application generated stamps</i> • Variáveis de paginação • Navegação • Atributos gerados sem saída da fronteira e • Atributos recuperados ou referenciados de um ALI ou AIE sem saída da fronteira 	<p>Literais como "Identidade" não são contados.</p> <p>Não existem.</p> <p>Não existem.</p> <p>Não existem.</p> <p>O atributo de Status Inativo é atualizado, mas não sai da fronteira. Não é contado.</p> <p>Não existem.</p>

Passo 6 Determinar a Complexidade Funcional

2 ALR e 3 DERs	Complexidade é Baixa
----------------	----------------------

Passo 7 Determinar o Tamanho Funcional

Tamanho Funcional de 1 EE de Baixa Complexidade	3 PF
---	------

Exemplo: Incluir Nível de Segurança de Janelas

Requisitos do Usuário

O requisito para gerenciar nível de segurança foi incluído durante a fase de Construção. O usuário quer incluir, modificar, excluir e consultar informações de nível de segurança no Sistema de Recursos Humanos. A funcionalidade de inclusão é ilustrada abaixo.

Incluir informações de nível de segurança de janelas entrando com

- Identificador do Usuário.
- Caixas para Funções Assinaladas, Função, Funcionário, Localização e funções de Geração de Relatórios.
- Nível de Segurança de Usuário para as funções acima (p.ex., permitir ou não permitir o acesso).

Gerar mensagens de erro e destacar campos incorretos se os campos não permitirem edição. Duas (2) mensagens de erro e uma (1) mensagem de confirmação estão incluídas para a transação de inclusão de informações de nível de segurança.

Tela de Inclusão de Nível de Segurança de Janelas

A seguinte ilustração apresenta a tela para incluir nível de segurança de janelas.

Sistema de Recursos Humanos

Funcionários Funções Localizações Relatórios Segurança Ajuda

RH Janela de Configuração de Segurança

ID Usuário

Assinalamento de Função

- ☐ Configuração
- ☐ Pesquisar (Função)
- ☐ Pesquisar (Funcionário)
- ☐ Editar
- ☐ Relatórios

Gerador de Relatórios

Definição

- ☐ Definir
- ☐ Editar
- ☐ Pesquisar

Função

- ☐ Configuração
- ☐ Pesquisar
- ☐ Editar
- ☐ Relatórios

Funcionário

- ☐ Configuração
- ☐ Pesquisar
- ☐ Editar
- ☐ Relatórios

Localização

- ☐ Revisar

OK Cancelar

SW-1 RH Janela de Configuração de Segurança

OK Efetivar alterações, retornar para menu principal do Sistema de Recursos Humanos

Cancelar Retornar para menu principal do Sistema de Recursos Humanos

As caixas na ilustração acima representam a Identificação das Janelas.

Passo 1 Identificar o Processo Elementar

A Função de Transação atende aos requisitos de um Processo Elementar?	Sim.
---	------

Passo 2 Determinar se o Processo Elementar é único

A Função de Transação é única em relação a outros processos elementares?	Sim. Nenhum outro processo elementar executa esta função.
--	---

Passo 3 Classificar cada Processo Elementar

Regras de Contagem de EE	A Regra se Aplica?
1. Possui como intenção primária: <ul style="list-style-type: none"> Manter um ou mais ALIs ou Alterar o comportamento da aplicação 	Sim. O ALI Nível de Segurança de Janelas é mantido. Não. O comportamento da aplicação não é alterado. O comportamento da aplicação será alterado pelo controle da funcionalidade que o usuário pode executar quando ele entra no sistema.
2. Inclui lógica de processamento para aceitar dados ou controle de informação que entra na fronteira da aplicação.	Sim. As informações de Nível de Segurança de Janelas entram na fronteira para manter o ALI de Nível de Segurança de Janelas.

Conclusão Incluir Nível de Segurança de Janelas é uma EE.

Veja os Estudos de Casos para verificar como os requisitos de alteração e exclusão e telas associadas a estas funções são contadas.

Passo 4 Contar ALR - Arquivo Lógico Referenciado (tipo de arquivo referenciado)

Regra de Contagem de ALR	A Regra se Aplica?
1. Um ALR deve ser contado para cada função de dados única que é acessada (lida e/ou mantida) pela função de transação.	O ALI Nível de Segurança de Janelas é mantido.

Passo 5 Contar DER - Dado Elementar Referenciado (tipo de dado elementar)

Regras de Contagem de DER	A Regra se Aplica?
1. Contar um DER para cada atributo único, reconhecido pelo usuário e não repetido, que atravessa (entra e/ou sai) a fronteira durante o processamento de uma função de transação.	Identificador do Usuário, Janelas, Nível de Segurança do Usuário.
2. Contar apenas um DER para a função de transação, para a habilidade da aplicação de enviar uma mensagem de resposta, mesmo que existam múltiplas mensagens.	Mensagens de Erro.
3. Contar apenas um DER para a função de transação, para a habilidade de iniciar ação(ões), mesmo que existam múltiplas maneiras de fazê-lo.	Tecla de Ação OK.
4. Não contar os seguintes itens como DERs: <ul style="list-style-type: none"> • Literais • <i>application generated stamps</i> • Variáveis de paginação • Navegação • Atributos gerados sem saída da fronteira e • Atributos recuperados ou referenciados de um ALI ou AIE sem saída da fronteira 	<p>Literais como “ID Usuário” não são contados”</p> <p>Não existem itens deste tipo.</p> <p>Não existem itens deste tipo.</p> <p>Não existem itens deste tipo.</p> <p>Não existem itens deste tipo.</p> <p>Não existem itens deste tipo.</p>

Passo 6 Determinar a Complexidade Funcional

1 ALR e 5 DERs	Complexidade é Baixa
----------------	----------------------

Passo 7 Determinar o Tamanho Funcional

Tamanho Funcional de 1 EE de Baixa Complexidade	3 PF
---	------

Exemplos de Contagem de SE

Introdução Esta seção utiliza uma aplicação de Recursos Humanos (RH) para ilustrar procedimentos usados para contagem de saídas externas. Em adição a esta seção, exemplos estão nos Estudos de Casos incluídos em documentação suplementar do IFPUG.

Conteúdo Esta seção inclui os seguintes exemplos:

Tópico	Página
Descrição Geral dos Exemplos de Contagens de SE	2-106
Exemplo: Relatório Impresso	2-107
Exemplo: Relatório Online	2-110
Exemplo: Transação Enviada para Outra Aplicação	2-113
Exemplo: Mensagens de Erro/Confirmação	2-115
Exemplo: Notificação de Revisão de Desempenho	2-116
Exemplo: SE Disparado sem Dados Entrando na Fronteira	2-119
Exemplo: Intenção Primária de uma SE	2-122
Exemplo: SE como Arquivo de Transação	2-125

Descrição Geral dos Exemplos de Contagens de SE

Os exemplos para SEs estão descritos na tabela a seguir.

Exemplo	Descrição Sumarizada	Página
Relatório Impresso	Este exemplo ilustra a contagem de um relatório impresso em papel.	2-107
Relatório Online	Este exemplo apresenta a contagem de um relatório <i>online</i> .	2-110
Transação Enviada para Outra Aplicação	Este exemplo ilustra uma transação gerada por uma aplicação e enviada para outra aplicação.	2-113
Mensagens de Erro/Confirmação	Este exemplo apresenta que erros ou mensagens de confirmação não são contados como saídas externas.	2-115
Exemplo: Notificação de Revisão de Desempenho	Este exemplo ilustra uma notificação baseada em um cálculo.	2-116
SE Disparada sem Dados Entrando na Fronteira	Este exemplo ilustra o conceito em que uma SE pode ser disparada sem dados entrando na fronteira.	2-119
Intenção Primária de uma SE	Este exemplo ilustra que uma SE pode atualizar um arquivo.	2-122
SE como Arquivo de Transação	Este exemplo ilustra que a existência de cálculos determina que o processo elementar seja uma SE e não uma CE.	2-125

Exemplo: Relatório Impresso

Requisitos do Usuário O usuário do Sistema de Recursos Humanos requer uma listagem dos assinalamentos de funções dos funcionários.

O relatório é gerado a partir da recuperação de:

- Um assinalamento a partir do ALI de funções assinaladas
- Informações adicionais a partir dos ALIs funcionário e função.

O ALI de controle do relatório é referenciado para determinar como gerar o relatório.

Exemplo de Relatório O seguinte relatório de Funções com Funcionários lista funções e funcionários assinalados a elas.

HRS006	Sistema de Recursos Humanos		Página 1
	Funções com Funcionários		Data 99.99.99
Código da Função	Nome da Função	ID Funcionário	Nome do Funcionário
9999	xxxxxxxxxx	xxx-xx-xxxx	xxxxxxxxxxxxxxxxxx
		xxx-xx-xxxx	xxxxxxxxxxxxxxxxxx
		xxx-xx-xxxx	xxxxxxxxxxxxxxxxxx
9999	xxxxxxxxxx	xxx-xx-xxxx	xxxxxxxxxxxxxxxxxx
9999	xxxxxxxxxx	xxx-xx-xxxx	xxxxxxxxxxxxxxxxxx
		xxx-xx-xxxx	xxxxxxxxxxxxxxxxxx
Total de Funções 9999			

Passo 1 Identificar o Processo Elementar

A Função de Transação atende aos requisitos de um Processo Elementar?	Sim.
---	------

Passo 2 Determinar se o Processo Elementar é único

A Função de Transação é única em relação a outros processos elementares?	Sim. Nenhum outro processo elementar executa esta função.
--	---

Passo 3 Classificar cada Processo Elementar

Regras de Contagem de SE	A Regra se Aplica?
1. Possui a intenção primária de apresentar informações ao usuário e	Sim. O relatório de Funções com Funcionários tem a intenção primária de apresentar informações ao usuário.
2. Inclui, no mínimo, uma das seguintes formas de lógica de processamento:	
<ul style="list-style-type: none"> Cálculos matemáticos são executados 	Sim. O número total de funções é considerado tanto calculado quanto derivado.
<ul style="list-style-type: none"> Um ou mais ALIs são atualizados 	Não. Nenhum ALI é atualizado.
<ul style="list-style-type: none"> Dados derivados são criados ou 	Sim. O número total de funções é considerado tanto calculado quanto derivado.
<ul style="list-style-type: none"> O comportamento da aplicação é alterado 	Não. O comportamento da aplicação não é alterado.

Conclusão O Relatório de Funções com Funcionários é uma SE.

Passo 4 Contar ALR - Arquivo Lógico Referenciado (tipo de arquivo referenciado)

Regra de Contagem de ALR	A Regra se Aplica?
1. Um ALR deve ser contado para cada função de dados que é acessada (lida e/ou mantida) pela função de transação.	Os seguintes ALIs são lidos: <ul style="list-style-type: none"> Funcionário Função Funções Assinaladas Controle de Relatório

Passo 5 Contar DER - Dado Elementar Referenciado (tipo de dado elementar)

Regras de Contagem de DER	A Regra se Aplica?
1. Contar um DER para cada atributo único, reconhecido pelo usuário e não repetido, que atravessa (entra e/ou sai) a fronteira durante o processamento de uma função de transação.	Código da Função, Nome da Função, Identificação do Funcionário e Número Total de Funções são exibidos. Contar cada um apenas uma vez.
2. Contar apenas um DER para a função de transação, para a habilidade da aplicação de enviar uma mensagem de resposta, mesmo que existam múltiplas mensagens.	Não existem
3. Contar apenas um DER para a função de transação, para a habilidade de iniciar ação(ões), mesmo que existam múltiplas maneiras de fazê-lo.	Não existem
4. Não contar os seguintes itens como DERs: <ul style="list-style-type: none"> • Literais tais como títulos de relatório, identificadores de tela ou painéis, cabeçalhos de coluna e títulos de atributo. • <i>application generated stamps</i>, tais como atributos de data e hora • Variáveis de paginação, números de página e informações de posicionamento, p.ex., "Linhas 37 a 54 de 211" • Recursos navegacionais tais como habilidade de navegar dentro de uma lista utilizando "anterior", "próximo", "primeiro", "último" e suas equivalências gráficas • Atributos gerados dentro da fronteira por uma função de transação e salvos em um ALI, sem saída da fronteira • Atributos recuperados ou referenciados a partir de um ALI ou AIE para participação no processamento, sem saída da fronteira. 	<p>Identificadores de relatório, títulos de relatório e cabeçalhos de coluna não são contados.</p> <p>A data do relatório não é contada.</p> <p>O número da página não é contado.</p> <p>Não existem.</p> <p>Não existem.</p> <p>Não existem.</p>

Passo 6 Determinar a Complexidade Funcional

4 ALRs e 5 DERs	Complexidade é Média
-----------------	----------------------

Passo 7 Determinar o Tamanho Funcional

Tamanho Funcional de 1 SE de Média Complexidade	5 PF
---	------

Exemplo: Relatório Online

Requisitos do Usuário O usuário requer um relatório de funcionários em ordem decrescente pela duração das funções atuais assinaladas. Este relatório é exibido *online* e contém dados calculados/derivados (por exemplo, duração do assinalamento da função)

Exemplo de Tela A tela Funcionários por Duração do Assinalamento a seguir lista funcionários por duração do assinalamento.

FUNCIONÁRIOS POR DURAÇÃO DO ASSINALAMENTO				
Linhas 1 a 18 de 1.316			MM/DD/AA	
Identificação Funcionário	Nome Funcionário	Código Função	Nome Função	Duração
xxx-xx-xxxx	xxxxxxxxxxx	9999	xxxxxxxxxxx	99 meses
xxx-xx-xxxx	xxxxxxxxxxx	9999	xxxxxxxxxxx	99 meses
Funcionários acima 24 meses 9999				
Funcionários acima 12 meses 9999				
F1=Ajuda F7=Rolar acima F8=Rolar abaixo F16=Fim				

Passo 1 Identificar o Processo Elementar

A Função de Transação atende aos requisitos de um Processo Elementar?	Sim.
---	------

Passo 2 Determinar se o Processo Elementar é único

A Função de Transação é única em relação a outros processos elementares?	Sim. Nenhum outro processo elementar executa esta função.
--	---

Passo 3 Classificar cada Processo Elementar

Regras de Contagem de SE	A Regra se Aplica?
1. Possui a intenção primária de apresentar informações ao usuário e	Sim. Funcionários por Duração do Assinalamento tem a intenção primária de apresentar informações ao usuário.
2. Inclui, no mínimo, uma das seguintes formas de lógica de processamento:	
<ul style="list-style-type: none"> Cálculos matemáticos são executados 	Sim. Três campos calculados são considerados tanto calculados como derivados.
<ul style="list-style-type: none"> Um ou mais ALIs são atualizados 	Não. Nenhum ALI é atualizado
<ul style="list-style-type: none"> Dados derivados são criados ou 	Sim. Três campos calculados são considerados tanto calculados como derivados.
<ul style="list-style-type: none"> O comportamento da aplicação é alterado 	Não. O comportamento da aplicação não é alterado.

Conclusão O relatório de Funcionários por Duração do Assinalamento é uma SE.

Passo 4 Contar ALR - Arquivo Lógico Referenciado (tipo de arquivo referenciado)

Regra de Contagem de ALR	A Regra se Aplica?
1. Um ALR deve ser contado para cada função de dados que é acessada (lida e/ou mantida) pela função de transação.	Os ALIs Funcionário, Função, e Funções Assinaladas são lidos. Nenhum ALI é mantido.

Passo 5 Contar DER - Dado Elementar Referenciado (tipo de dado elementar)

Regras de Contagem de DER	A Regra se Aplica?
1. Contar um DER para cada atributo único, reconhecido pelo usuário e não repetido, que atravessa (entra e/ou sai) a fronteira durante o processamento de uma função de transação.	Identificação do Funcionário, Nome do Funcionário, Nome da Função e Duração do Assinalamento são repetidos. Contar cada um somente uma vez. Identificação de funcionários acima de 24 meses e acima de 12 meses.
2. Contar apenas um DER para a função de transação, para a habilidade da aplicação de enviar uma mensagem de resposta, mesmo que existam múltiplas mensagens.	Não existem.
3. Contar apenas um DER para a função de transação, para a habilidade de iniciar ação(ões), mesmo que existam múltiplas maneiras de fazê-lo.	Uma tecla de função é utilizada para exibir o relatório <i>online</i> .
4. Não contar os seguintes itens como DERs: <ul style="list-style-type: none"> Literais <i>application generated stamps</i> Variáveis de paginação Navegação Atributos gerados sem saída da fronteira Atributos recuperados ou referenciados a partir de um ALI ou AIE sem saída da fronteira 	<p>Literais tais como “Identificação do Funcionário” não são contados.</p> <p>A data do relatório não é contada.</p> <p>“Linhas 1 a 18” não são contados.</p> <p>F7 e F8 não são contados.</p> <p>Não existem.</p> <p>Não existem.</p>

Passo 6 Determinar a Complexidade Funcional

3 ALRs e 8 DERs	Complexidade é Média
-----------------	----------------------

Passo 7 Determinar o Tamanho Funcional

Tamanho Funcional de 1 SE de Média Complexidade	5 PF
---	------

Exemplo: Transação Enviada para Outra Aplicação

Requisitos do Usuário Quando o Sistema de Recursos Humanos inclui dados dos dependentes de um funcionário, o usuário requer que este Arquivo de Dependente seja enviado para aplicação de Benefícios para manter os registros consistentes. Esta informação é enviada para Benefícios diariamente.

Requisitos de Construção Se dados de dependente são adicionados, esta informação é formatada apropriadamente no arquivo de transação de saída.

Durante a implementação da solução, foi decidido incluir um cabeçalho e um registro detalhe com as informações de benefícios. Estes registros são utilizados por Benefícios para garantir que nada está incorreto tecnicamente durante a transmissão do arquivo.

Exemplo de Formato do Registro O formato do registro de dependente de funcionário a seguir contém informações a respeito de dependentes incluídos e modificados.

```
12345678910123456789012345678901234567890123456789012345678901234567890
1 C|NOME DO ARQUIVO      |DATA      |
2 D|FUNC ID  |DEP ID  |NOME DEPENDENTE |NASC|
3 T|TOT. REG |
4
5
6
7
9
0
```

Descrição dos Campos A tabela a seguir inclui descrições para cada campo do registro.

Tipo de Registro	Posição	Descrição
Cabeçalho	1	Tipo de Registro C
	2-13	Nome do Arquivo
	14-19	Data da Criação
Dependente	1	Tipo de Registro D
	2-10	Identificação do Funcionário
	11-19	Identificação do Dependente
	20-39	Nome do Dependente
	40-45	Data de nascimento do Dependente
Total	1	Tipo de Registro T
	2-10	Número total de registros

Passo 1 Identificar o Processo Elementar - Cabeçalho

A Função de Transação atende aos requisitos de um Processo Elementar?	Não. O cabeçalho não contém dados significativos para o usuário.
---	--

Passo 1 Identificar o Processo Elementar - Total

A Função de Transação atende aos requisitos de um Processo Elementar?	Não. O total não contém dados significativos para o usuário.
---	--

Passo 1 Identificar o Processo Elementar - Dependente

A Função de Transação atende aos requisitos de um Processo Elementar?	Sim. A seção dependente do arquivo de transação satisfaz os requisitos para um Processo Elementar.
---	--

Passo 2 Determinar se o Processo Elementar é único

A Função de Transação é única em relação a outros processos elementares?	Sim. Nenhum outro processo elementar executa esta função.
--	---

Passo 3 Classificar cada Processo Elementar

Regras de Contagem de SE	A Regra se Aplica?
1. Possui a intenção primária de apresentar informações ao usuário e	Sim. O arquivo de Dependentes possui a intenção primária de apresentar informações ao usuário.
2. Inclui, no mínimo, uma das seguintes formas de lógica de processamento:	
• Cálculos matemáticos são executados	Não. Cálculos não são executados.
• Um ou mais ALIs são atualizados	Não. Nenhum ALI é atualizado.
• Dados derivados são criados ou	Não. Nenhum dado derivado é criado.
• O comportamento da aplicação é alterado	Não. O comportamento da aplicação não é alterado.

Conclusão O Arquivo de Dependentes não se qualifica como uma SE; seria contado com uma CE (não analisada aqui).

Exemplo de Tela A seguinte tela de Função exibe uma mensagem de confirmação (abaixo na tela).

Ação: _ 7=Anterior 8=Seguinte

Dados de Função

Código da Função: RD15379305

Nome da Função: May Issue - Print Covers

Código de Pgmo: JRNY05A

Linha No	Descrição da Função
<u>01</u>	<u>Print Covers 4-Up, Lacquer Finish.</u>

F1=Ajuda F7=Rolar acima F8=Rolar abaixo F12=Cancelar

Processamento Completado com Sucesso

<p>Entra: Retorna para tela de chamada.</p> <p>F1: Apresenta mapa de ajuda no nível de tela ou campo.</p> <p>F7: Rola acima até 10 linhas de descrição da função.</p> <p>F8: Rola abaixo até 10 linhas de descrição da função.</p>	<p>F12: Retorna para tela de chamada.</p> <p>Ação 7: Apresenta dados anteriores da função, se existentes.</p> <p>Ação 8: Apresenta dados seguintes da função, se existentes.</p>
--	--

Passo 1 Identificar o Processo Elementar

A Função de Transação atende aos requisitos de um Processo Elementar?	Não. A saída de uma mensagem de erro não é uma função auto-contida. É um DER de saída na EE de Inclusão da Função.
---	--

Exemplo: Notificação de Revisão de Desempenho

Requisitos do Usuário O usuário requer notificação automática quando um funcionário completou 12 meses no assinalamento de uma função. Isto indica que uma revisão de desempenho deve ser realizada.

Exemplo de Janela A janela de Notificação de Revisão de Desempenho descreve a mensagem de notificação.

Notificação de Revisão de Desempenho	
<p>Na data: xx/xx/xx</p> <p>O funcionário: xxx-xx-xxxx x_____x</p> <p>Completou 12 meses de assinalamento</p> <p>Na função: xxxx x_____x</p> <p>E deveria ser submetido a uma revisão de desempenho imediatamente</p>	

Passo 1 Identificar o Processo Elementar

A Função de Transação atende aos requisitos de um Processo Elementar?	Sim.
---	------

Passo 2 Determinar se o Processo Elementar é único

A Função de Transação é única em relação a outros processos elementares?	Sim. Nenhum outro processo elementar executa esta função.
--	---

Passo 3 Classificar cada Processo Elementar

Regras de Contagem de SE	A Regra se Aplica?
1. Possui a intenção primária de apresentar informações ao usuário e	Sim. Notificação de Revisão de Desempenho possui a intenção primária de apresentar informação ao usuário.
2. Inclui, no mínimo, uma das seguintes formas de lógica de processamento:	
<ul style="list-style-type: none"> • Cálculos matemáticos são executados 	Sim. A data correspondente a 12 meses do assinalamento a uma função é calculada.
<ul style="list-style-type: none"> • Um ou mais ALIs são atualizados 	Não. Nenhum ALI é atualizado
<ul style="list-style-type: none"> • Dados derivados são criados ou 	Não. Nenhum dado derivado é criado.
<ul style="list-style-type: none"> • O comportamento da aplicação é alterado 	Não. O comportamento da aplicação não é alterado.

Conclusão A Notificação de Revisão de Desempenho é uma SE.

Passo 4 Contar ALR - Arquivo Lógico Referenciado (tipo de arquivo referenciado)

Regra de Contagem de ALR	A Regra se Aplica?
1. Um ALR deve ser contado para cada função de dados que é acessada (lida e/ou mantida) pela função de transação.	Os ALIs Funcionário, Função e Funções Assinaladas são lidos.

Passo 5 Contar DER - Dado Elementar Referenciado (tipo de dado elementar)

Regras de Contagem de DER	A Regra se Aplica?
1. Contar um DER para cada atributo único, reconhecido pelo usuário e não repetido, que atravessa (entra e/ou sai) a fronteira durante o processamento de uma função de transação.	Data, Identificação do Funcionário, Nome do Funcionário, Código da Função, Nome da Função.
2. Contar apenas um DER para a função de transação, para a habilidade da aplicação de enviar uma mensagem de resposta, mesmo que existam múltiplas mensagens.	Não existem.
3. Contar apenas um DER para a função de transação, para a habilidade de iniciar ação(ões), mesmo que existam múltiplas maneiras de fazê-lo.	Não existem.
4. Não contar os seguintes itens como DERs: <ul style="list-style-type: none"> • Literais • <i>application generated stamps</i> • Variáveis de paginação • Navegação • Atributos gerados sem saída da fronteira • Atributos recuperados ou referenciados a partir de um ALI ou AIE sem saída da fronteira 	<p>Literais tais como "Funcionário" não são contados.</p> <p>A data e hora do relatório não são contadas.</p> <p>Não existem.</p> <p>Não existem.</p> <p>Não existem.</p> <p>Não existem.</p>

Passo 6 Determinar a Complexidade Funcional

3 ALR e 5 DERs	Complexidade é Baixa.
----------------	-----------------------

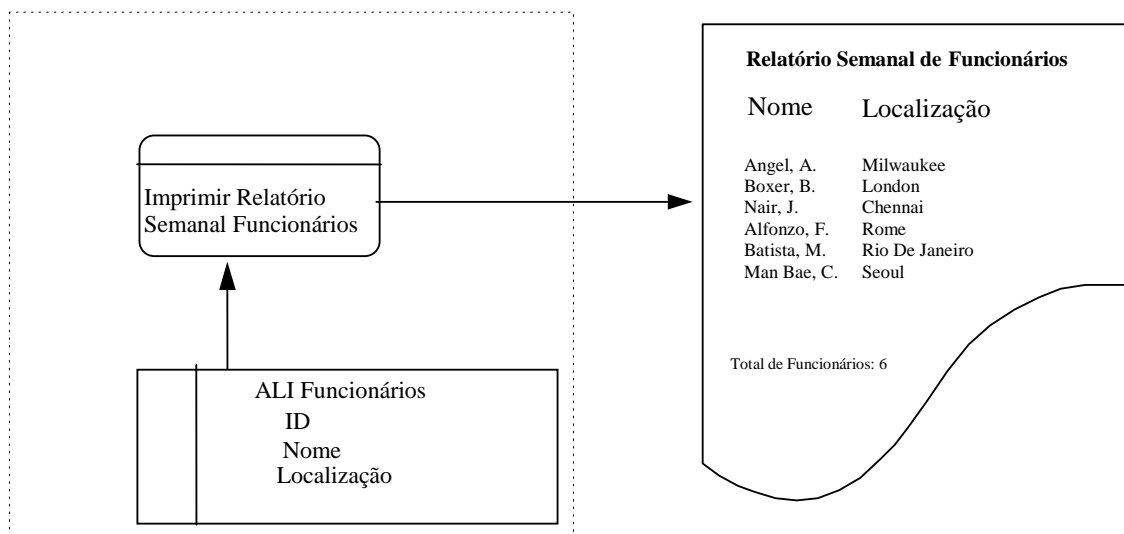
Passo 7 Determinar o Tamanho Funcional

Tamanho Funcional de 1 SE de complexidade Baixa.	4 PF
--	------

Exemplo: SE Disparado sem Dados Entrando na Fronteira

Requisito do Usuário O usuário requer que a aplicação imprima o Relatório Semanal de Funcionários automaticamente todos os Domingos às 23:00. O relatório contém detalhes para cada funcionário com um total dos funcionários.

Modelo de Dados O diagrama a seguir apresenta o fluxo de dados para este exemplo.



Passo 1 Identificar o Processo Elementar

A Função de Transação atende aos requisitos de um Processo Elementar?	Sim.
---	------

Passo 2 Determinar se o Processo Elementar é único

A Função de Transação é única em relação a outros processos elementares?	Sim. Nenhum outro processo elementar executa esta função.
--	---

Passo 3 Classificar cada Processo Elementar

Regras de Contagem de SE	A Regra se Aplica?
1. Possui a intenção primária de apresentar informações ao usuário e	Sim. O Relatório Semanal de Funcionários possui a intenção primária de apresentar informação ao usuário.
2. Inclui, no mínimo, uma das seguintes formas de lógica de processamento:	
<ul style="list-style-type: none"> • Cálculos matemáticos são executados 	Sim. O Total de Funcionários é um campo calculado.
<ul style="list-style-type: none"> • Um ou mais ALIs são atualizados 	Não. Nenhum ALI é atualizado
<ul style="list-style-type: none"> • Dados derivados são criados ou 	Sim. O relatório contém um campo calculado.
<ul style="list-style-type: none"> • O comportamento da aplicação é alterado 	Não. O comportamento da aplicação não é alterado.

Conclusão O Relatório Semanal de Funcionários é uma SE.

Passo 4 Contar ALR - Arquivo Lógico Referenciado (tipo de arquivo referenciado)

Regra de Contagem de ALR	A Regra se Aplica?
1. Um ALR deve ser contado para cada função de dados que é acessada (lida e/ou mantida) pela função de transação.	O ALI Funcionário é lido. Nenhum ALI é mantido.

Passo 5 Contar DER - Dado Elementar Referenciado (tipo de dado elementar)

Regras de Contagem de DER	A Regra se Aplica?
1. Contar um DER para cada atributo único, reconhecido pelo usuário e não repetido, que atravessa (entra e/ou sai) a fronteira durante o processamento de uma função de transação.	Nome, Localização, Total de Funcionários.
2. Contar apenas um DER para a função de transação, para a habilidade da aplicação de enviar uma mensagem de resposta, mesmo que existam múltiplas mensagens.	Não existem
3. Contar apenas um DER para a função de transação, para a habilidade de iniciar ação(ões), mesmo que existam múltiplas maneiras de fazê-lo.	Não existem
4. Não contar os seguintes itens como DERs: <ul style="list-style-type: none"> • Literais • <i>application generated stamps</i> • Variáveis de paginação • Navegação • Atributos gerados sem saída da fronteira • Atributos recuperados ou referenciados a partir de um ALI ou AIE sem saída da fronteira 	<p>Literais tais como "Nome" não são contados.</p> <p>Não existem</p> <p>Não existem</p> <p>Não existem</p> <p>Não existem</p> <p>Não existem</p>

Passo 6 Determinar a Complexidade Funcional

1 ALR e 3 DERs	Complexidade é Baixa.
----------------	-----------------------

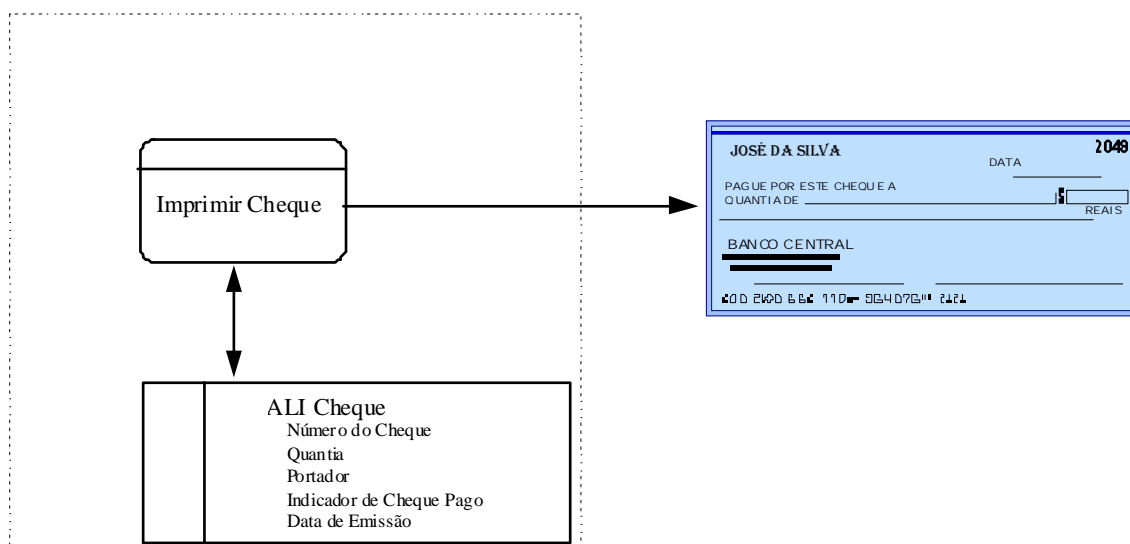
Passo 7 Determinar o Tamanho Funcional

Tamanho Funcional de 1 SE de Complexidade Baixa	4 PF
---	------

Exemplo: Intenção Primária de uma SE

Requisitos do Usuário Imprimir um Cheque e, como resultado, marcar na conta o cheque como pago. Todos os dados impressos no cheque já estão armazenados no arquivo de cheques.

O diagrama a seguir apresenta o fluxo de dados para este exemplo.



Passo 1 Identificar o Processo Elementar

A Função de Transação atende aos requisitos de um Processo Elementar?	Sim.
---	------

Passo 2 Determinar se o Processo Elementar é único

A Função de Transação é única em relação a outros processos elementares?	Sim. Nenhum outro processo elementar executa esta função.
--	---

Passo 3 Classificar cada Processo Elementar

Regras de Contagem de SE	A Regra se Aplica?
1. Possui a intenção primária de apresentar informações ao usuário e	Sim. A intenção primária é imprimir um cheque. A manutenção do ALI é secundária.
2. Inclui, no mínimo, uma das seguintes formas de lógica de processamento:	
• Cálculos matemáticos são executados	Não. Nenhum cálculo é executado.
• Um ou mais ALIs são atualizados	Sim. O ALI Cheques é atualizado.
• Dados derivados são criados ou	Não. Nenhum dado derivado é criado.
• O comportamento da aplicação é alterado	Não. O comportamento da aplicação não é alterado.

Conclusão Imprimir um Cheque é uma SE.

Passo 4 Contar ALR - Arquivo Lógico Referenciado (tipo de arquivo referenciado)

Regra de Contagem de ALR	A Regra se Aplica?
1. Um ALR deve ser contado para cada função de dados que é acessada (lida e/ou mantida) pela função de transação.	O ALI Cheques é lido e mantido, mas é contado somente uma vez.

Passo 5 Contar DER - Dado Elementar Referenciado (tipo de dado elementar)

Regras de Contagem de DER	A Regra se Aplica?
1. Contar um DER para cada atributo único, reconhecido pelo usuário e não repetido, que atravessa (entra e/ou sai) a fronteira durante o processamento de uma função de transação.	Número do Cheque, Quantia, Portador, Data de Emissão do Cheque.
2. Contar apenas um DER para a função de transação, para a habilidade da aplicação de enviar uma mensagem de resposta, mesmo que existam múltiplas mensagens.	Não existem.
3. Contar apenas um DER para a função de transação, para a habilidade de iniciar ação(ões), mesmo que existam múltiplas maneiras de fazê-lo.	Não existem.
4. Não contar os seguintes itens como DERs: <ul style="list-style-type: none"> Literais <i>application generated stamps</i> Variáveis de paginação Navegação Atributos gerados sem saída da fronteira Atributos recuperados ou referenciados a partir de um ALI ou AIE sem saída da fronteira 	<p>Não existem.</p> <p>Existe uma data impressa no cheque, que representa um dado reconhecido pelo usuário e é contada.</p> <p>Não existem.</p> <p>Não existem.</p> <p>O Indicador de Cheque Pago não é contado, uma vez que não atravessa a fronteira.</p> <p>Não existem.</p>

Passo 6 Determinar a Complexidade Funcional

1 ALR e 4 DERs	Complexidade é Baixa
----------------	----------------------

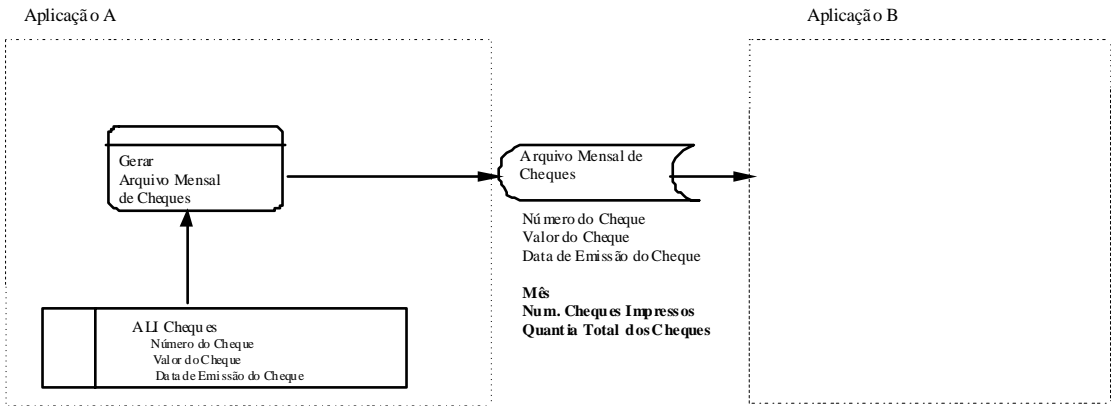
Passo 7 Determinar o Tamanho Funcional

Tamanho Funcional de 1 SE de Complexidade Baixa	4 PF
---	------

Exemplo: SE como Arquivo de Transação

Requisitos do Usuário No final do mês, gerar o Arquivo Mensal de Cheques e enviá-lo para Aplicação B. Os números dos cheques, datas de emissão dos cheques e valor dos cheques são incluídos no arquivo, com um valor computado de contagem de cheques processados e o valor total de todos os cheques impressos no mês. O número de cheques impressos e o valor total de todos os cheques impressos são utilizados pelos usuários da Aplicação B para prover extratos bancários.

Modelo de Dados O diagrama a seguir apresenta o fluxo de dados para este exemplo.



Passo 1 Identificar o Processo Elementar

A Função de Transação atende aos requisitos de um Processo Elementar?	Sim.
---	------

Passo 2 Determinar se o Processo Elementar é único

A Função de Transação é única em relação a outros processos elementares?	Sim. Nenhum outro processo elementar executa esta função.
--	---

Passo 3 Classificar cada Processo Elementar

Regras de Contagem de SE	A Regra se Aplica?
1. Possui a intenção primária de apresentar informações ao usuário e	Sim. A intenção primária é gerar um arquivo de transação.
2. Inclui, no mínimo, uma das seguintes formas de lógica de processamento:	
• Cálculos matemáticos são executados	Sim. O arquivo inclui dois campos calculados.
• Um ou mais ALIs são atualizados	Não. Nenhum ALI é atualizado.
• Dados derivados são criados ou	Sim. O arquivo inclui dois campos calculados.
• O comportamento da aplicação é alterado	Não. O comportamento da aplicação não é alterado.

Conclusão O Arquivo Mensal de Cheques é uma SE.

Passo 4 Contar ALR - Arquivo Lógico Referenciado (tipo de arquivo referenciado)

Regra de Contagem de ALR	A Regra se Aplica?
1. Um ALR deve ser contado para cada função de dados que é acessada (lida e/ou mantida) pela função de transação.	O ALI Cheques é lido. Não existem ALIs mantidos.

Passo 5 Contar DER - Dado Elementar Referenciado (tipo de dado elementar)

Regras de Contagem de DER	A Regra se Aplica?
1. Contar um DER para cada atributo único, reconhecido pelo usuário e não repetido, que atravessa (entra e/ou sai) a fronteira durante o processamento de uma função de transação.	Número do Cheque, Valor do Cheque, Data de Emissão do Cheque, Mês, Número de cheques impressos, Quantia Total dos Cheques.
2. Contar apenas um DER para a função de transação, para a habilidade da aplicação de enviar uma mensagem de resposta, mesmo que existam múltiplas mensagens.	Não existem
3. Contar apenas um DER para a função de transação, para a habilidade de iniciar ação(ões), mesmo que existam múltiplas maneiras de fazê-lo.	Não existem
4. Não contar os seguintes itens como DERs: <ul style="list-style-type: none"> • Literais • <i>application generated stamps</i> • Variáveis de paginação • Navegação • Atributos gerados sem saída da fronteira • Atributos recuperados ou referenciados a partir de um ALI ou AIE sem saída da fronteira 	<p>Não existem</p> <p>Não existem</p> <p>Não existem</p> <p>Não existem</p> <p>Não existem</p> <p>Não existem</p>

Passo 6 Determinar a Complexidade Funcional

1 ALR e 6 DERs	Complexidade é Baixa
----------------	----------------------

Passo 7 Determinar o Tamanho Funcional

Tamanho Funcional de 1 SE de Complexidade Baixa	4 PF
---	------

Esta página foi deixada intencionalmente em branco.

Exemplos de Contagem de CE

Introdução Esta seção utiliza uma aplicação de Recursos Humanos (RH) para ilustrar procedimentos para contar consultas externas. Em adição a esta seção, exemplos estão nos Estudos de Casos incluídos em documentação suplementar do IFPUG.

Conteúdo Esta seção inclui os seguintes exemplos:

Tópico	Página
Descrição Geral dos Exemplos de Contagem de CE	2-106
Exemplo: Menus de Aplicação	2-131
Exemplo: Lista de Dados Recuperados	2-133
Exemplo: Drop-Down List Box	2-138
Exemplo: Ajuda no Nível de Campo – Primeira Ocorrência	2-142
Exemplo: Ajuda no Nível de Campo – Segunda Ocorrência	2-145
Exemplo: Consulta Implícita	2-147
Exemplo: CE Disparada sem Dados Entrando na Fronteira	2-151
Exemplo: Dados Enviados para Outra Aplicação	2-154
Exemplo: Funcionalidade Adicional de Ajuda	2-157
Exemplo: Logon de Nível de Segurança	2-161

Descrição Geral dos Exemplos de Contagem de CE

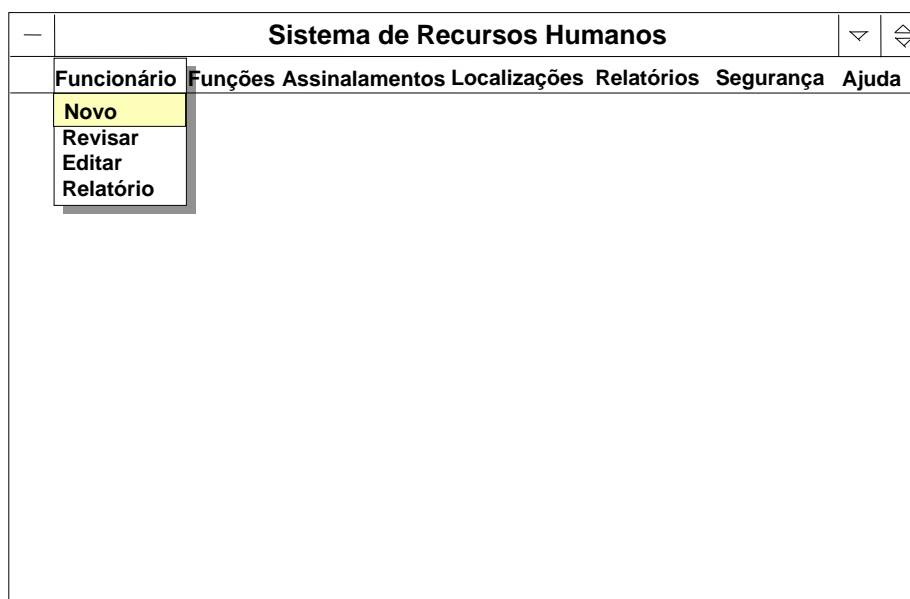
Os exemplos de CEs estão listados e descritos na tabela a seguir.

Exemplo	Descrição Sumária	Página
Menus de Aplicação	Este exemplo mostra que menus navegacionais ou outros recursos navegacionais não são contados como CEs.	2-131
Lista de Dados Recuperados	Este exemplo ilustra a contagem para uma lista.	2-133
Drop-Down List Box	Este exemplo ilustra a contagem de uma <i>drop-down list</i> .	2-138
Ajuda no Nível de Campo – Primeira Ocorrência	Este exemplo ilustra que uma janela de Ajuda no nível de campo é contada para a primeira ocorrência.	2-142
Ajuda no Nível de Campo – Segunda Ocorrência	Este exemplo ilustra que uma segunda instância de uma janela de Ajuda no nível de campo não é contada.	2-145
Consulta Implícita	Este exemplo ilustra contagem de consulta que não está explicitamente definida mas está implícita.	2-147
CE Disparada sem Dados Entrando na Fronteira	Este exemplo ilustra contagem de recuperação e apresentação de dados disparados internamente por tempo.	2-151
Dados Enviados para Outra Aplicação	Este exemplo ilustra contagem de dados enviados para outra aplicação através de um arquivo.	2-154
Funcionalidade Adicional de Ajuda	Este exemplo ilustra contagem de funcionalidade adicional de Ajuda.	2-157
Logon de Nível de Segurança	Este exemplo ilustra contagem de função de <i>logon</i> .	2-161

Exemplo: Menus de Aplicação

Requisitos do Usuário A aplicação de Recursos Humanos requer menus e recursos de navegação.

Exemplo de Janelas O diagrama a seguir apresenta o menu *drop-down* Funcionário no menu principal do Sistema de Recursos Humanos. Esta é a solicitação de entrada.



Quando o usuário seleciona Novo no menu *drop-down*, a seguinte janela de Dados de Funcionário é exibida em branco.

EN-1 Dados de Funcionário

Cancelar

Retorna ao menu inicial

OK

Navega para próxima tela:

• EN-2H Atualizar Funcionário Horista, se tipo de salário horista for selecionado

• EN-2S Atualizar Funcionário Assalariado, se tipo de salário assalariado for selecionado

Passo 1 Identificar o Processo Elementar

A Função de Transação atende aos requisitos de um Processo Elementar?	Não. Menus existem para satisfazer requisitos navegacionais ao invés de requisitos funcionais do usuário.
---	---

Conclusão Menus não satisfazem os requisitos de um processo elementar.

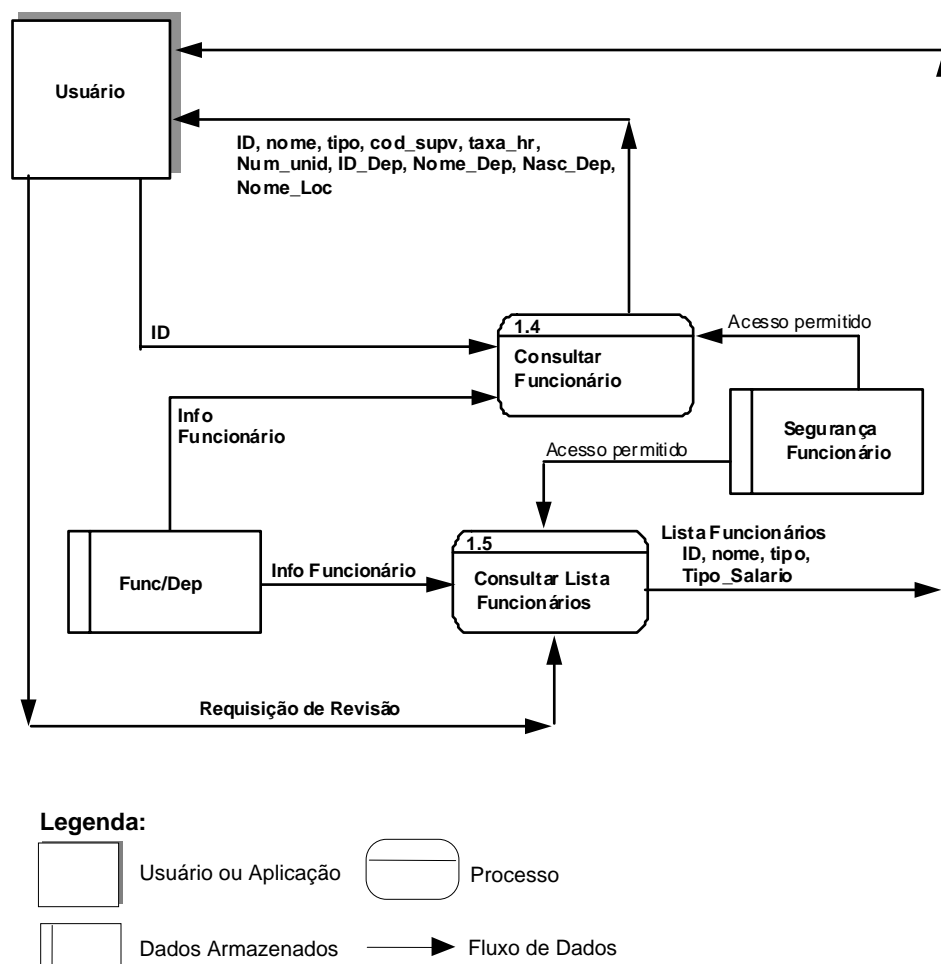
Exemplo: Lista de Dados Recuperados

Requisitos do Usuário O usuário possui os seguintes requisitos:

- Visualizar uma Lista de Funcionários organizada por Sobrenome, Primeiro Nome e as Iniciais do Meio.

Este exemplo foca na visualização de uma lista de funcionários na aplicação de Recursos Humanos.

Diagrama de Fluxo de Dados O seguinte diagrama apresenta o fluxo dados para este exemplo.



Exemplo de Janelas

O seguinte diagrama apresenta o menu *drop-down* para funcionário. O campo Revisar e a tecla Entrar formam a parte de entrada deste exemplo.

Sistema de Recursos Humanos					
Funcionário	Funções	Assinalamentos	Localizações	Relatórios	Segurança Ajuda
<div> <div>Novo</div> <div>Revisar</div> <div>Editar</div> <div>Relatório</div> </div>					

Quando o usuário seleciona Revisar no menu *drop-down* Funcionário, a seguinte janela é exibida com uma lista de funcionários.

Sistema de Recursos Humanos																																																					
Funcionário	Funções	Assinalamentos	Localizações	Relatórios	Segurança Ajuda																																																
<table border="1"> <thead> <tr> <th colspan="6">Lista de Funcionários</th> </tr> <tr> <th>Sobrenome</th> <th>Primeiro Nome</th> <th>IM</th> <th>Identificação</th> <th>Tipo de Salário</th> <th></th> </tr> </thead> <tbody> <tr> <td>Keller</td> <td>Caroline</td> <td></td> <td>123-45-6789</td> <td>Assalariado</td> <td>↑</td> </tr> <tr> <td>Latta</td> <td>Nicky</td> <td>A</td> <td>234-56-7890</td> <td>Horista</td> <td></td> </tr> <tr> <td>Manship</td> <td>Mark</td> <td>J</td> <td>345-67-8901</td> <td>Horista</td> <td></td> </tr> <tr> <td>Schmidt-Taylor</td> <td>Kathleen</td> <td></td> <td>456-78-9012</td> <td>Assalariado</td> <td></td> </tr> <tr> <td>Smith</td> <td>David</td> <td>E</td> <td>567-89-0123</td> <td>Horista</td> <td></td> </tr> <tr> <td>Smith</td> <td>Loretta</td> <td>M</td> <td>678-90-1234</td> <td>Assalariado</td> <td>↓</td> </tr> </tbody> </table> <div> <div>Visualizar</div> <div>Dependentes</div> <div>Cancelar</div> </div>						Lista de Funcionários						Sobrenome	Primeiro Nome	IM	Identificação	Tipo de Salário		Keller	Caroline		123-45-6789	Assalariado	↑	Latta	Nicky	A	234-56-7890	Horista		Manship	Mark	J	345-67-8901	Horista		Schmidt-Taylor	Kathleen		456-78-9012	Assalariado		Smith	David	E	567-89-0123	Horista		Smith	Loretta	M	678-90-1234	Assalariado	↓
Lista de Funcionários																																																					
Sobrenome	Primeiro Nome	IM	Identificação	Tipo de Salário																																																	
Keller	Caroline		123-45-6789	Assalariado	↑																																																
Latta	Nicky	A	234-56-7890	Horista																																																	
Manship	Mark	J	345-67-8901	Horista																																																	
Schmidt-Taylor	Kathleen		456-78-9012	Assalariado																																																	
Smith	David	E	567-89-0123	Horista																																																	
Smith	Loretta	M	678-90-1234	Assalariado	↓																																																

El-1 Lista de Funcionários

Visualizar
Dependentes
Cancelar

Inicia apresentação dos dados em
El-2 Revisar Dados de Funcionário

Inicia lista apresentada em El-4 Lista de Dependentes

Retorna ao menu principal

Passo 1 Identificar o Processo Elementar

A Função de Transação atende aos requisitos de um Processo Elementar?	Sim.
---	------

Passo 2 Determinar se o Processo Elementar é único

A Função de Transação é única em relação a outros processos elementares?	Sim. Nenhum outro processo elementar executa esta função.
--	---

Passo 3 Classificar cada Processo Elementar

Regras de Contagem de CE	A Regra se Aplica?
1. Possui a intenção primária de apresentar informações ao usuário, e:	Sim. A intenção primária é apresentar informações ao usuário.
<ul style="list-style-type: none"> referencia uma função de dados para recuperar dados ou informação de controle e 	Sim. Dados são recuperados do ALI Funcionário.
<ul style="list-style-type: none"> não satisfaz o critério de ser classificado como uma SE 	Sim. Não executa cálculos, não cria dados derivados e não atualiza um ALI.

Conclusão A Lista de Funcionários é uma CE.

Passo 4 Contar ALR - Arquivo Lógico Referenciado (tipo de arquivo referenciado)

Regra de Contagem de ALR	A Regra se Aplica?
1. Um ALR deve ser contado para cada função de dados que é acessada (lida e/ou mantida) pela função de transação.	O ALI Funcionário é lido. Uma CE não pode manter um ALI por definição.

Passo 5 Contar DER - Dado Elementar Referenciado (tipo de dado elementar)

Regras de Contagem de DER	A Regra se Aplica?
1. Contar um DER para cada atributo único, reconhecido pelo usuário e não repetido, que atravessa (entra e/ou sai) a fronteira durante o processamento de uma função de transação.	Os seguintes campos são repetidos e contados apenas uma vez: Sobrenome, Primeiro Nome, Iniciais do Meio, ID, Tipo de Salário. A lista é organizada utilizando elementos individuais do nome (como Sobrenome, Primeiro Nome, Iniciais do Meio). Como resultado, os elementos individuais do Nome são reconhecidos pelo usuário.
2. Contar apenas um DER para a função de transação, para a habilidade da aplicação de enviar uma mensagem de resposta, mesmo que existam múltiplas mensagens.	Não existem.
3. Contar apenas um DER para a função de transação, para a habilidade de iniciar ação(ões), mesmo que existam múltiplas maneiras de fazê-lo.	Sim. O campo Revisar/tecla Entrar iniciam a ação.
4. Não contar os seguintes itens como DERs: <ul style="list-style-type: none"> • Literais tais como títulos de relatório, identificadores de tela ou painéis, cabeçalhos de coluna e títulos de atributo. • <i>application generated stamps</i>, tais como atributos de data e hora • Variáveis de paginação, números de página e informações de posicionamento, p.ex., “Linhas 37 a 54 de 211” • Recursos navegacionais tais como habilidade de navegar dentro de uma lista utilizando “anterior”, “próximo”, “primeiro”, “último” e suas equivalências gráficas • Atributos gerados dentro da fronteira por uma função de transação e salvos em um ALI, sem saída da fronteira • Atributos recuperados ou referenciados a partir de um ALI ou AIE para participação no processamento, sem saída da fronteira. 	<p>Cabeçalhos de tela e coluna não são contados.</p> <p>Não existem.</p> <p>Não existem.</p> <p>A barra de rolagem não é contada.</p> <p>Não existem.</p> <p>Não existem.</p>

Passo 6 Determinar a Complexidade Funcional

1 ALR e 6 DERs	Complexidade é Baixa
----------------	----------------------

Step 7 Determinar Tamanho Funcional

Tamanho Funcional de 1 CE de Complexidade Baixa	3 PF
--	------

Exemplo: Drop-Down List Box

Requisitos do Usuário O usuário requer a capacidade de visualizar uma Lista de Unidades de Negócio adicionada ao Sistema de Recursos Humanos pelo usuário.

Exemplo de Janelas O diagrama a seguir apresenta a janela de Dados de Funcionário Horista com o campo de Unidade de Negócio.

El-3H Dados de Funcionário Horista com Unidade de Negócio

Dependentes Apresenta lista em El-4 Lista de Dependentes

Quando o usuário seleciona a seta, a seguinte lista de *drop-down* é exibida.

Sistema de Recursos Humanos

Funcionário Funções Assinalamentos Localizações Relatórios Segurança Ajuda

Lista de Funcionários

Dados de Funcionários

Mark J Manship

345-67-8901

Dados de Funcionário Horista

Taxa por Hora

Unidade de Negócio

UPFCA
L841
CPLG

Dependentes

EI-3H Dados de Funcionário Horista com Unidade de Negócio

Dependentes Apresenta lista em EI-4 Lista de Dependentes

Passo 1 Identificar o Processo Elementar

A Função de Transação atende aos requisitos de um Processo Elementar?	Sim.
---	------

Passo 2 Determinar se o Processo Elementar é único

A Função de Transação é única em relação a outros processos elementares?	Sim. Nenhum outro processo elementar executa esta função.
--	---

Passo 3 Classificar cada Processo Elementar

Regras de Contagem de CE	A Regra se Aplica?
1. Possui a intenção primária de apresentar informações ao usuário, e:	Sim. A intenção primária é apresentar informações ao usuário.
<ul style="list-style-type: none"> referencia uma função de dados para recuperar dados ou informação de controle e 	Sim. Dados são recuperados a partir do ALI Unidades de Negócio. O ALI Unidades de Negócio contém inúmeros atributos sobre uma Unidade de Negócio.
<ul style="list-style-type: none"> não satisfaz o critério de ser classificado como uma SE 	Sim. Não executa cálculos, não cria dados derivados e não atualiza um ALI.

Conclusão A Lista de Unidades de Negócio é uma CE.

Passo 4 Contar ALR - Arquivo Lógico Referenciado (tipo de arquivo referenciado)

Regra de Contagem de ALR	A Regra se Aplica?
1. Um ALR deve ser contado para cada função de dados que é acessada (lida e/ou mantida) pela função de transação.	O ALI Unidades de Negócio é lido. Uma CE não pode manter um ALI, por definição.

Passo 5 Contar DER - Dado Elementar Referenciado (tipo de dado elementar)

Regras de Contagem de DER	A Regra se Aplica?
1. Contar um DER para cada atributo único, reconhecido pelo usuário e não repetido, que atravessa (entra e/ou sai) a fronteira durante o processamento de uma função de transação.	Unidade de Negócio.
2. Contar apenas um DER para a função de transação, para a habilidade da aplicação de enviar uma mensagem de resposta, mesmo que existam múltiplas mensagens.	Não existem.
3. Contar apenas um DER para a função de transação, para a habilidade de iniciar ação(ões), mesmo que existam múltiplas maneiras de fazê-lo.	Sim. A seta abaixo ativa a <i>listbox</i> .
4. Não contar os seguintes itens como DERs: <ul style="list-style-type: none"> • literais • <i>application generated stamps</i> • variáveis de paginação • navegação • atributos gerados sem saída da fronteira e • atributos recuperados ou referenciados a partir de um ALI ou AIE sem saída da fronteira. 	<p>Não existem.</p> <p>Não existem.</p> <p>Não existem.</p> <p>Não existem.</p> <p>Não existem.</p> <p>Não existem.</p>

Passo 6 Determinar a Complexidade Funcional

1 ALR e 2 DERs	Complexidade é Baixa
----------------	----------------------

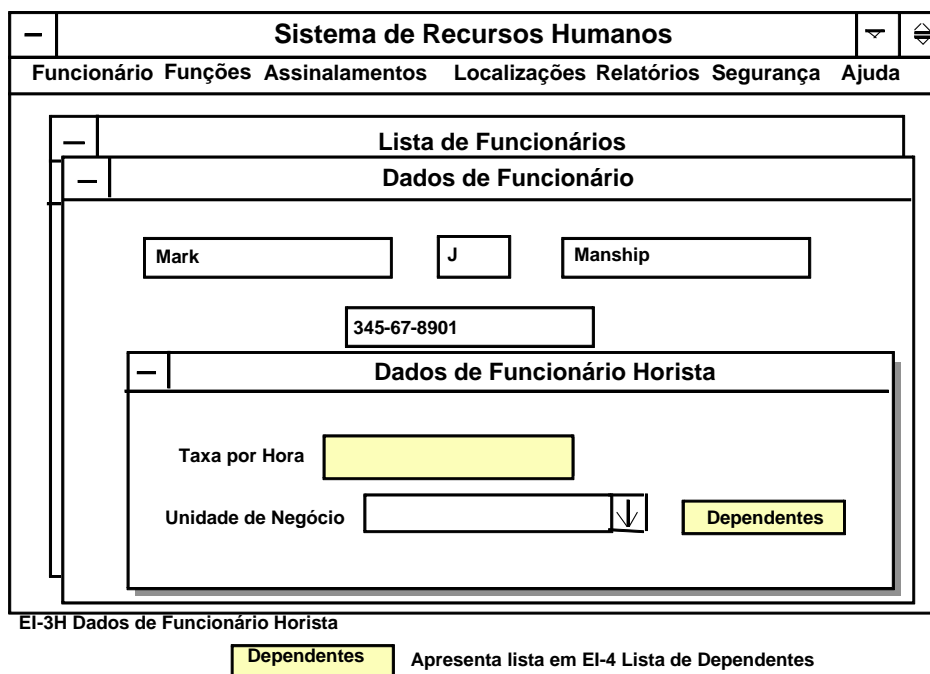
Step 7 Determinar Tamanho Funcional

Tamanho Funcional de 1 CE de Complexidade Baixa	3 PF
---	------

Exemplo: Ajuda no Nível de Campo – Primeira Ocorrência

Requisitos do Usuário Durante a construção do Sistema de Recursos Humanos, um requisito para Ajuda *online* de nível de campo foi adicionado. A informação de Ajuda é mantida por uma aplicação separada. A informação de Ajuda é referenciada pelas aplicações de Recursos Humanos, Moedas Correntes, Ativos Fixos, e Benefícios.

Exemplo de Janelas O diagrama a seguir apresenta a janela de Dados do Funcionário.



Quando o usuário pressiona **F1** enquanto o cursor está sobre o campo de taxa por hora, uma caixa mostra o texto de Ajuda como apresentado no diagrama a seguir.

Sistema de Recursos Humanos

Funcionário Tarefas Assinalamentos Localizações Relatórios Segurança Ajuda

Taxa por hora

A quantia que um funcionário é pago para cada hora de trabalho completo.

Esta informação deve ser provida para cada funcionário horista.

Valores Válidos: Qualquer valor-hora é aceitável.

Valores Default: Este campo não possui valores default.

Taxa por Hora

Unidade de Negócio

Dependentes

EI-3H Taxa por Hora **Dependentes** Apresenta lista em EI-4 Lista de Dependentes

Passo 1 Identificar o Processo Elementar

A Função de Transação atende aos requisitos de um Processo Elementar?	Sim.
---	------

Passo 2 Determinar se o Processo Elementar é único

A Função de Transação é única em relação a outros processos elementares?	Sim. Nenhum outro processo elementar executa esta função.
--	---

Passo 3 Classificar cada Processo Elementar

Regras de Contagem de CE	A Regra se Aplica?
1. Possui a intenção primária de apresentar informações ao usuário, e:	Sim. A Ajuda no nível de Campo possui a intenção primária de apresentar informação ao usuário.
<ul style="list-style-type: none"> referencia uma função de dados para recuperar dados ou informação de controle e 	Sim. Dados são recuperados do AIE Ajuda
<ul style="list-style-type: none"> não satisfaz o critério de ser classificado como uma SE 	Sim. Não executa cálculos, não cria dados derivados e não atualiza nenhum ALI.

Conclusão A Ajuda de Nível de Campo é uma CE.

Passo 4 Contar ALR - Arquivo Lógico Referenciado (tipo de arquivo referenciado)

Regra de Contagem de ALR	A Regra se Aplica?
1. Um ALR deve ser contado para cada função de dados que é acessada (lida e/ou mantida) pela função de transação.	O AIE Ajuda é lido. Uma CE não pode manter um ALI por definição.

Passo 5 Contar DER - Dado Elementar Referenciado (tipo de dado elementar)

Regras de Contagem de DER	A Regra se Aplica?
1. Contar um DER para cada atributo único, reconhecido pelo usuário e não repetido, que atravessa (entra e/ou sai) a fronteira durante o processamento de uma função de transação.	ID da Janela, ID do Campo, Mensagem da Ajuda, Valor <i>Default</i> , Valores Válidos
2. Contar apenas um DER para a função de transação, para a habilidade da aplicação de enviar uma mensagem de resposta, mesmo que existam múltiplas mensagens.	Não existem.
3. Contar apenas um DER para a função de transação, para a habilidade de iniciar ação(ões), mesmo que existam múltiplas maneiras de fazê-lo.	Sim. A tecla F1.
4. Não contar os seguintes itens como DERs: <ul style="list-style-type: none"> • literais • <i>application generated stamps</i> • variáveis de paginação • navegação • atributos gerados sem saída da fronteira e • atributos recuperados ou referenciados a partir de um ALI ou AIE sem saída da fronteira. 	<p>Literais como “Valores Válidos” não são contados.</p> <p>Não existem.</p> <p>Não existem.</p> <p>Não existem.</p> <p>Não existem.</p> <p>Não existem.</p>

Passo 6 Determinar a Complexidade Funcional

1 ALR e 6 DERs	Complexidade é Baixa
----------------	----------------------

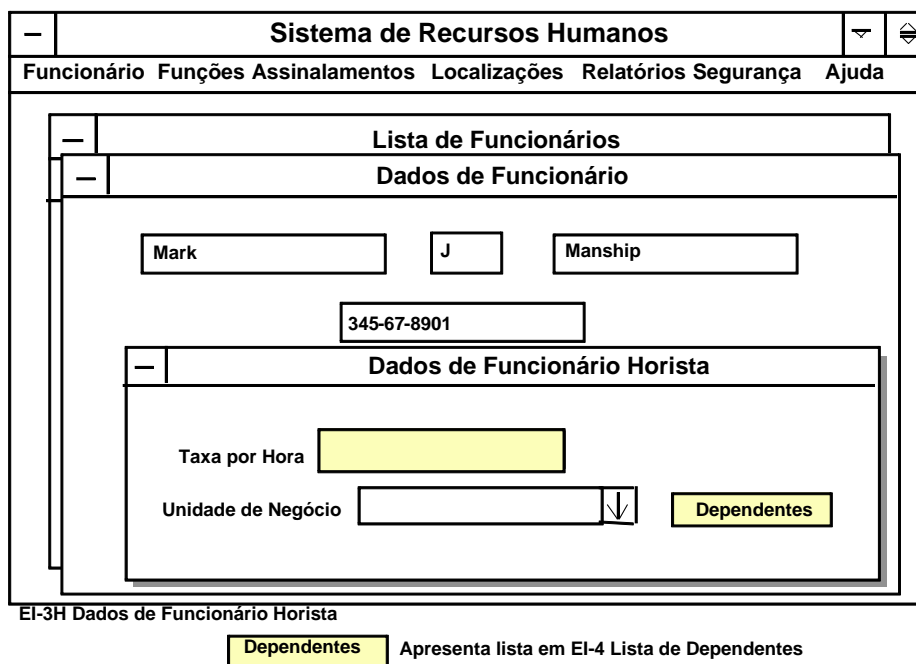
Passo 7 Determinar Tamanho Funcional

Tamanho Funcional de 1 CE de Complexidade Baixa	3 PF
---	------

Exemplo: Ajuda no Nível de Campo – Segunda Ocorrência

Requisitos do Usuário Durante a construção do Sistema de Recursos Humanos, um requisito para Ajuda *online* no nível de campo foi adicionado. A Ajuda *online* está relacionado aos processos de inclusão, exclusão e alteração de informações de Recursos Humanos. As informações de Ajuda são mantidas por uma aplicação separada. As informações de Ajuda são referenciadas pelas aplicações de Recursos Humanos, Moedas Correntes, Ativos Fixos e Benefícios.

Exemplo de Janelas O diagrama a seguir apresenta a janela de Dados de Funcionário.



O usuário posiciona o cursor no campo para o qual a ajuda é necessária, e pressiona a tecla **F1** para visualizar a Ajuda para aquele campo.

Passo 1 Identificar o Processo Elementar

A Função de Transação atende aos requisitos de um Processo Elementar?	Sim.
---	------

Passo 2 **Determinar se o Processo Elementar é único**

A Função de Transação é única em relação a outros processos elementares?	Não. Esta função executa a mesma função que foi descrita no exemplo anterior. Os DERs, ALRs e a lógica de processamento para esta função são as mesmas que o exemplo anterior. Ela não é contada novamente.
--	---

Conclusão A Segunda Ocorrência de Ajuda no Nível de Campo não é um processo elementar único e não é contado como uma CE.

Exemplo: Consulta Implícita

Requisitos do Usuário O usuário requer a habilidade de visualizar informações de assinalamento enquanto edita informações de funções assinaladas. Embora, não esteja explícito, é implícito que as Informações de Funções Assinaladas devem ser recuperadas antes de serem liberadas para alteração.

Exemplo de Janelas O diagrama a seguir apresenta a janela Editar Funções Assinaladas com apenas o nome do funcionário e o código da função.

AE-5

OK	Efetivar alterações, retornar para AE-1
Cancelar	Ignorar alterações, retornar para AE-1
Restaurar	Restaurar valores anteriores
Excluir	Solicitar confirmação, então excluir assinalamento de função

Quando o usuário entra com o nome do funcionário e o código da função, a informação de função é exibida como apresentado no diagrama a seguir.

Sistema de Recursos Humanos		▼	⚙
Funcionário	Funções	Assinalamentos	Localizações Relatórios Segurança Ajuda

Assinalamento de Funcionários	
Editar Funções Assinaladas	
<div style="display: flex; justify-content: space-between; margin-bottom: 5px;"> Mark J Manship </div> <div style="text-align: center; margin-bottom: 5px;">345-67-8901</div> <div style="display: flex; justify-content: space-between; margin-bottom: 5px;"> Planta Principal UFPCA </div> <div style="display: flex; justify-content: space-between; margin-bottom: 5px;"> Código da Função <div style="border: 1px solid black; padding: 2px;">RD15379305</div> ↓ </div> <div style="display: flex; justify-content: space-between; margin-bottom: 5px;"> Data <div style="border: 1px solid black; padding: 2px;">03/27/93</div> </div> <div style="display: flex; justify-content: space-between; margin-bottom: 5px;"> Salário <div style="border: 1px solid black; padding: 2px;">17.28</div> </div> <div style="display: flex; justify-content: space-between;"> Avaliação de Desempenho <div style="border: 1px solid black; padding: 2px;">Satsfatório</div> </div>	<div style="margin-top: 10px;">Corrigir impressão</div> <div style="margin-top: 5px; text-align: center;"> <div style="background-color: yellow; border: 1px solid black; padding: 2px; width: 50px; margin: 2px auto;">OK</div> <div style="background-color: yellow; border: 1px solid black; padding: 2px; width: 50px; margin: 2px auto;">Cancelar</div> <div style="background-color: yellow; border: 1px solid black; padding: 2px; width: 50px; margin: 2px auto;">Restaurar</div> <div style="background-color: yellow; border: 1px solid black; padding: 2px; width: 50px; margin: 2px auto;">Excluir</div> </div>

AE-5

<div style="background-color: yellow; border: 1px solid black; padding: 2px; width: 50px; margin: 2px auto;">OK</div>	Efetivar alterações, retornar para AE-1
<div style="background-color: yellow; border: 1px solid black; padding: 2px; width: 50px; margin: 2px auto;">Cancelar</div>	Ignorar alterações, retornar para AE-1
<div style="background-color: yellow; border: 1px solid black; padding: 2px; width: 50px; margin: 2px auto;">Restaurar</div>	Restaurar valores anteriores
<div style="background-color: yellow; border: 1px solid black; padding: 2px; width: 50px; margin: 2px auto;">Excluir</div>	Solicitar confirmação, então excluir assinalamento da função

Passo 1 Identificar o Processo Elementar

A Função de Transação atende aos requisitos de um Processo Elementar?	Sim.
---	------

Passo 2 Determinar se o Processo Elementar é único

A Função de Transação é única em relação a outros processos elementares?	Sim. Nenhum outro processo elementar que executasse esta função foi contado até o momento. Se uma consulta direta também existisse, seria considerada uma duplicidade e não seria contada novamente.
--	--

Passo 3 Classificar cada Processo Elementar

Regras de Contagem de CE	A Regra se Aplica?
1. Possui a intenção primária de apresentar informações ao usuário, e:	Sim. A Consulta Implícita Assinalameno de Função possui a intenção primária de apresentar informações ao usuário.
<ul style="list-style-type: none"> referencia uma função de dados para recuperar dados ou informação de controle e 	Sim. Dados são recuperados a partir dos ALIs Funções Assinaladas, Funcionário e Função.
<ul style="list-style-type: none"> não satisfaz o critério de ser classificado como uma SE 	Sim. Não executa cálculos, não cria dados derivados e não atualiza nenhum ALI.

Conclusão Consulta Implícita de Funções Assinaladas é uma CE.

Passo 4 Contar ALR - Arquivo Lógico Referenciado (tipo de arquivo referenciado)

Regra de Contagem de ALR	A Regra se Aplica?
1. Um ALR deve ser contado para cada função de dados que é acessada (lida e/ou mantida) pela função de transação.	Os ALIs Funções Assinaladas, Funcionário e Função são lidos. Uma CE não pode manter um ALI por definição.

Passo 5 Contar DER - Dado Elementar Referenciado (tipo de dado elementar)

Regras de Contagem de DER	A Regra se Aplica?
1. Contar um DER para cada atributo único, reconhecido pelo usuário e não repetido, que atravessa (entra e/ou sai) a fronteira durante o processamento de uma função de transação.	Nome, ID, Localização, Unidade de Negócio, Código da Função, Data, Salário e Avaliação de Desempenho
2. Contar apenas um DER para a função de transação, para a habilidade da aplicação de enviar uma mensagem de resposta, mesmo que existam múltiplas mensagens.	Não existem
3. Contar apenas um DER para a função de transação, para a habilidade de iniciar ação(ões), mesmo que existam múltiplas maneiras de fazê-lo.	Sim. A tecla de comando.
4. Não contar os seguintes itens como DERs: <ul style="list-style-type: none"> • literais • <i>application generated stamps</i> • variáveis de paginação • navegação • atributos gerados sem saída da fronteira e • atributos recuperados ou referenciados a partir de um ALI ou AIE sem saída da fronteira. 	<p>Literais como título da tela não são contados.</p> <p>Não existem</p> <p>Não existem</p> <p>Não existem</p> <p>Não existem</p> <p>Não existem</p>

Passo 6 Determinar a Complexidade Funcional

3 ALRs e 10 DERs	Complexidade é Média
------------------	----------------------

Passo 7 Determinar Tamanho Funcional

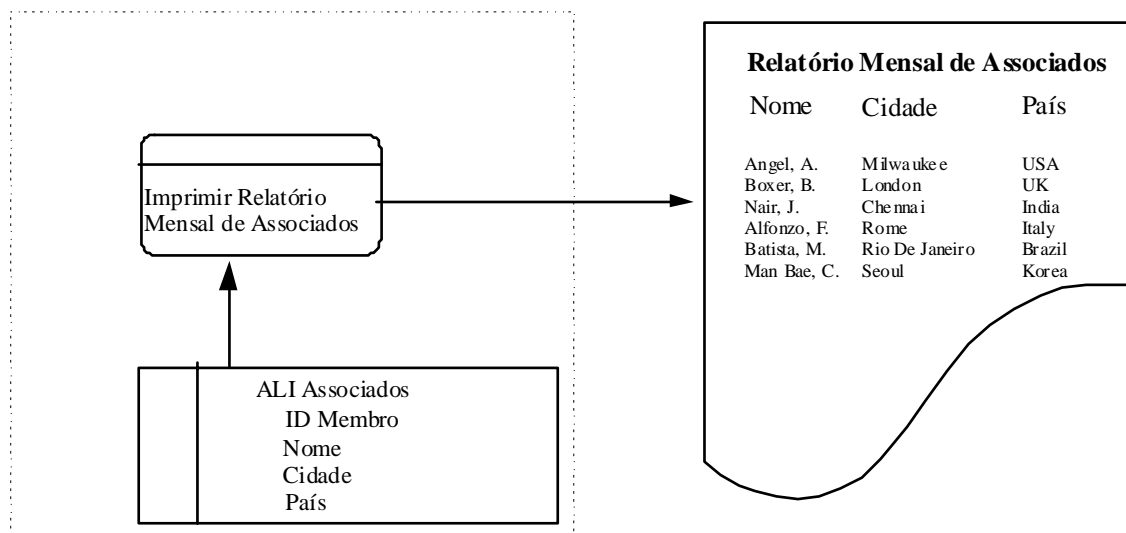
Tamanho Funcional de 1 CE de Complexidade Média	4 PF
---	------

Conclusão A Consulta Implícita de Funções Assinaladas é um processo elementar e é contado como uma CE. Se uma consulta direta também existisse, seria considerada uma duplicidade e não seria contada novamente.

Exemplo: CE Disparada sem Dados Entrando na Fronteira

Requisitos do Usuário O usuário requer que a aplicação imprima o Relatório Mensal de Associados automaticamente todo mês.

O diagrama a seguir apresenta o fluxo de dados para este exemplo.



Passo 1 Identificar o Processo Elementar

A Função de Transação atende aos requisitos de um Processo Elementar?	Sim.
---	------

Passo 2 Determinar se o Processo Elementar é único

A Função de Transação é única em relação a outros processos elementares?	Sim. Nenhum outro processo elementar executa esta função.
--	---

Passo 3 Classificar cada Processo Elementar

Regras de Contagem de CE	A Regra se Aplica?
1. Possui a intenção primária de apresentar informações ao usuário, e:	Sim. O Relatório Mensal de Associados possui a intenção primária de apresentar informações ao usuário.
<ul style="list-style-type: none"> referencia uma função de dados para recuperar dados ou informação de controle e 	Sim. Dados são recuperados a partir do ALI Associados.
<ul style="list-style-type: none"> não satisfaz o critério de ser classificado como uma SE 	Sim. Não executa cálculos, não cria dados derivados e não atualiza nenhum ALI.

Conclusão O Relatório Mensal de Associados é uma CE. Neste exemplo, a transação é disparada por um evento temporal dentro da fronteira.

Passo 4 Contar ALR - Arquivo Lógico Referenciado (tipo de arquivo referenciado)

Regra de Contagem de ALR	A Regra se Aplica?
1. Um ALR deve ser contado para cada função de dados que é acessada (lida e/ou mantida) pela função de transação.	O ALI Associados é lido. Uma CE não pode atualizar um ALI, por definição.

Passo 5 Contar DER - Dado Elementar Referenciado (tipo de dado elementar)

Regras de Contagem de DER	A Regra se Aplica?
1. Contar um DER para cada atributo único, reconhecido pelo usuário e não repetido, que atravessa (entra e/ou sai) a fronteira durante o processamento de uma função de transação.	Nome, Cidade, País
2. Contar apenas um DER para a função de transação, para a habilidade da aplicação de enviar uma mensagem de resposta, mesmo que existam múltiplas mensagens.	Não existem.
3. Contar apenas um DER para a função de transação, para a habilidade de iniciar ação(ões), mesmo que existam múltiplas maneiras de fazê-lo.	Não existem. A função é executada automaticamente todo mês.
4. Não contar os seguintes itens como DERs: <ul style="list-style-type: none"> • literais • <i>application generated stamps</i> • variáveis de paginação • navegação • atributos gerados sem saída da fronteira e • atributos recuperados ou referenciados a partir de um ALI ou AIE sem saída da fronteira. 	<p>Literais como "Nome" não são contados.</p> <p>Não existem.</p> <p>Não existem.</p> <p>Não existem.</p> <p>Não existem.</p> <p>Não existem.</p>

Passo 6 Determinar a Complexidade Funcional

1 ALR e 3 DERs	Complexidade é Baixa
----------------	----------------------

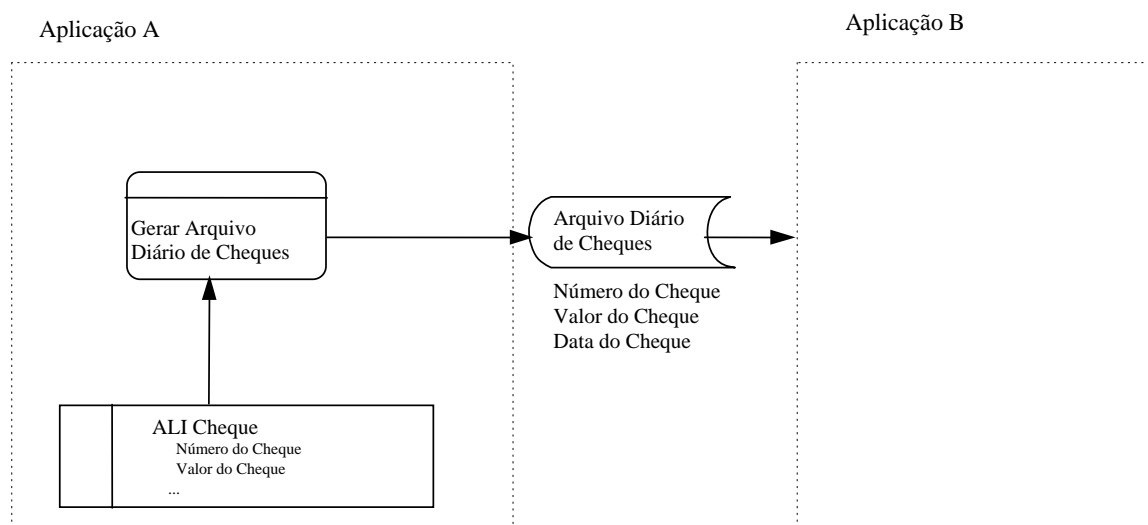
Passo 7 Determinar Tamanho Funcional

Tamanho Funcional de 1 CE de Complexidade Baixa	3 PF
---	------

Exemplo: Dados Enviados para Outra Aplicação

Requisitos do Usuário Ao final de cada dia, enviar um Arquivo Diário de Cheques para Aplicação B listando os números dos cheques, o valor e a data de emissão de cada cheque impresso no dia.

O diagrama a seguir apresenta o fluxo de dados para este exemplo.



Passo 1 Identificar o Processo Elementar

A Função de Transação atende aos requisitos de um Processo Elementar?	Sim.
---	------

Passo 2 Determinar se o Processo Elementar é único

A Função de Transação é única em relação a outros processos elementares?	Sim. Nenhum outro processo elementar executa esta função.
--	---

Passo 3 Classificar cada Processo Elementar

Regras de Contagem de CE	A Regra se Aplica?
1. Possui a intenção primária de apresentar informações ao usuário, e:	Sim. O Arquivo Diário de Cheques possui a intenção primária de apresentar informações. Neste caso, o usuário é a Aplicação B.
<ul style="list-style-type: none"> referencia uma função de dados para recuperar dados ou informação de controle e 	Sim. Dados são recuperados a partir do ALI Cheques.
<ul style="list-style-type: none"> não satisfaz o critério de ser classificado como uma SE 	Sim. Não executa cálculos, não cria dados derivados e não atualiza nenhum ALI

Conclusão O Arquivo de Cheques Diário é uma CE.

Passo 4 Contar ALR - Arquivo Lógico Referenciado (tipo de arquivo referenciado)

Regra de Contagem de ALR	A Regra se Aplica?
1. Um ALR deve ser contado para cada função de dados que é acessada (lida e/ou mantida) pela função de transação.	O ALI Cheques é lido. Uma CE não pode manter um ALI, por definição.

Passo 5 Contar DER - Dado Elementar Referenciado (tipo de dado elementar)

Regras de Contagem de DER	A Regra se Aplica?
1. Contar um DER para cada atributo único, reconhecido pelo usuário e não repetido, que atravessa (entra e/ou sai) a fronteira durante o processamento de uma função de transação.	Número do Cheque, Valor do Cheque, Data de Emissão do Cheque.
2. Contar apenas um DER para a função de transação, para a habilidade da aplicação de enviar uma mensagem de resposta, mesmo que existam múltiplas mensagens.	Não existem
3. Contar apenas um DER para a função de transação, para a habilidade de iniciar ação(ões), mesmo que existam múltiplas maneiras de fazê-lo.	Não existem. A função é executada automaticamente todo mês.
4. Não contar os seguintes itens como DERs: <ul style="list-style-type: none"> • literais • <i>application generated stamps</i> • variáveis de paginação • navegação • atributos gerados sem saída da fronteira e • atributos recuperados ou referenciados a partir de um ALI ou AIE sem saída da fronteira. 	<p>Não existem</p> <p>Não existem</p> <p>Não existem</p> <p>Não existem</p> <p>Não existem</p> <p>Não existem</p>

Passo 6 Determinar a Complexidade Funcional

1 ALR e 3 DERs	Complexidade é Baixa
----------------	----------------------

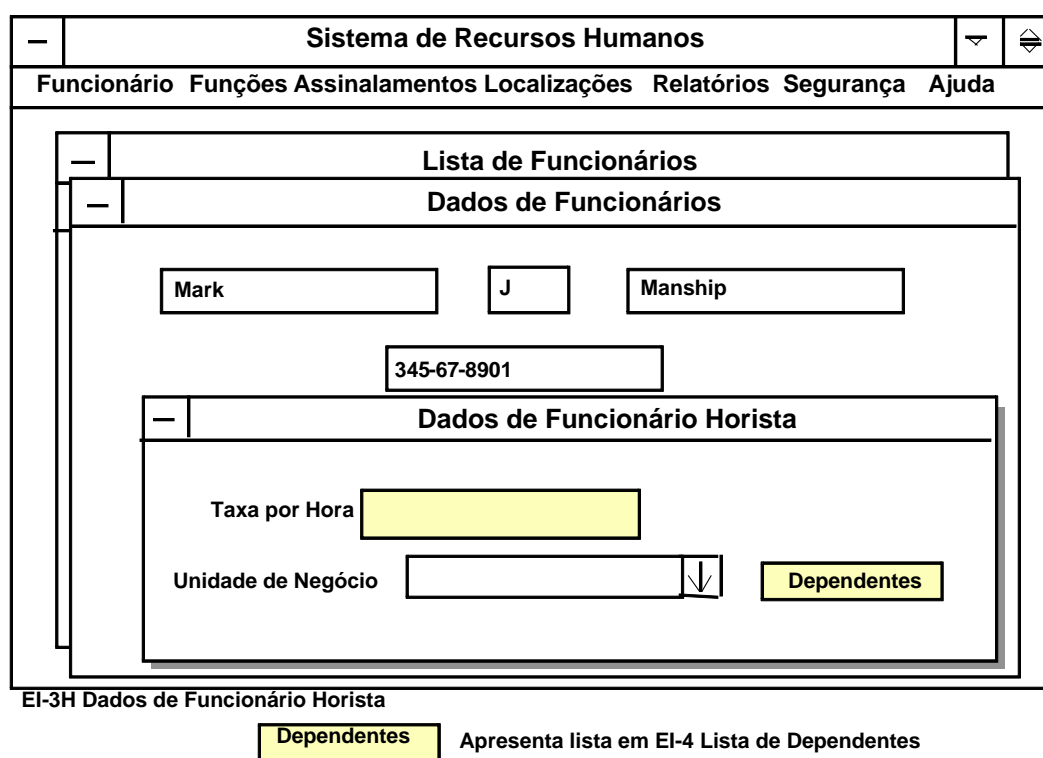
Passo 7 Determinar Tamanho Funcional

Tamanho Funcional de 1 CE de Complexidade Baixa	3 PF
---	------

Exemplo: Funcionalidade Adicional de Ajuda

Requisitos do Usuário Durante a construção do Sistema de Recursos Humanos, um requisito para uma funcionalidade adicional de Ajuda foi incluída. A informação de Ajuda é mantida por uma aplicação separada. A informação de Ajuda é referenciada pelas aplicações de Recursos Humanos, Ativos Fixos e Benefícios.

Processo de Contagem O diagrama a seguir apresenta a janela de Dados de Funcionário



Quando o usuário seleciona a função de Ajuda no cabeçalho de qualquer Janela, três seleções ficam habilitadas: Janela, Pesquisa e Sobre

Passo 1 Identificar o Processo Elementar

A Função de Transação atende aos requisitos de um Processo Elementar?	Sim. Janela, Pesquisa e Sobre aparecem para executar 3 funções independentes.
---	---

Passo 2 Determinar se o Processo Elementar é único

A Função de Transação é única em relação a outros processos elementares?	Sim. Nenhum outro processo elementar executa esta função. Estas funções são contadas apenas uma vez dentro da aplicação.
--	--

Passo 3 Classificar cada Processo Elementar

Regras de Contagem de CE	A Regra se Aplica?
1. Possui a intenção primária de apresentar informações ao usuário, e:	Sim. Janela, Pesquisa e Sobre possuem a intenção primária de apresentar informação ao usuário.
<ul style="list-style-type: none"> referencia uma função de dados para recuperar dados ou informação de controle e 	<p>Dados de Janela são recuperados a partir do AIE Ajuda, baseado na Janela atual</p> <p>Pesquisa habilita o usuário a entrar um tópico e recuperar Dados de Tópico a partir do AIE Ajuda</p> <p>Sobre está recuperando dados estáticos não mantidos no AIE Ajuda. Portanto não atende à definição de um CE.</p>
<ul style="list-style-type: none"> não satisfaz o critério de ser classificado como uma SE 	Sim. Não executa cálculos, não cria dados derivados e não atualiza nenhum ALI

Conclusão

Janela de Ajuda é uma CE e a capacidade de Pesquisa é uma CE. Janela de Ajuda para a seleção “Sobre” não é contada porque não é mantida através de um processo elementar.

É importante garantir que a janela estática de Ajuda não seja contada. Ambientes comuns, especialmente aplicações web, frequentemente contém janela estática de Ajuda.

Também é importante garantir que as fronteiras estejam corretamente avaliadas. Funcionalidade de Ajuda é frequentemente provida por uma aplicação externa, como *RoboHelp*, que provê toda funcionalidade de Ajuda (i.e., manutenção, armazenamento e apresentação de janela de Ajuda). Neste exemplo, nenhuma funcionalidade é creditada para a aplicação a qual a janela de Ajuda pertence.

Janela de Ajuda e Pesquisa de Ajuda são individualmente analisados abaixo.

Passo 4 Contar ALR - Arquivo Lógico Referenciado (tipo de arquivo referenciado) para Janela de Ajuda

Regra de Contagem de ALR	A Regra se Aplica?
1. Um ALR deve ser contado para cada função de dados que é acessada (lida e/ou mantida) pela função de transação.	O AIE Ajuda é lido. Uma CE não pode manter um ALI por definição.

Passo 5 Contar DER - Dado Elementar Referenciado (tipo de dado elementar)

Regras de Contagem de DER	A Regra se Aplica?
1. Contar um DER para cada atributo único, reconhecido pelo usuário e não repetido, que atravessa (entra e/ou sai) a fronteira durante o processamento de uma função de transação.	ID de Janela, Descrição de Janela (apenas texto).
2. Contar apenas um DER para a função de transação, para a habilidade da aplicação de enviar uma mensagem de resposta, mesmo que existam múltiplas mensagens.	Não existem
3. Contar apenas um DER para a função de transação, para a habilidade de iniciar ação(ões), mesmo que existam múltiplas maneiras de fazê-lo.	Sim. Seleção pelo Menu Ajuda
4. Não contar os seguintes itens como DERs: <ul style="list-style-type: none"> • literais • <i>application generated stamps</i> • variáveis de paginação • navegação • atributos gerados sem saída da fronteira e • atributos recuperados ou referenciados a partir de um ALI ou AIE sem saída da fronteira. 	<p>Literais como cabeçalhos de tela não são contados.</p> <p>Não existem</p> <p>Não existem</p> <p>Não existem</p> <p>Não existem</p> <p>Não existem</p>

Passo 6 Determinar a Complexidade Funcional da Janela de Ajuda

1 ALR e 3 DERs	Complexidade é Baixa
----------------	----------------------

Passo 7 Determinar Tamanho Funcional para a Janela de Ajuda

Tamanho Funcional de 1 CE de Complexidade Baixa	3 PF
---	------

Passo 4 Contar ALR - Arquivo Lógico Referenciado (tipo de arquivo referenciado) para a Capacidade de Pesquisa

Regra de Contagem de ALR	A Regra se Aplica?
1. Um ALR deve ser contado para cada função de dados que é acessada (lida e/ou mantida) pela função de transação.	O AIE Ajuda é lido. Uma CE não pode manter um ALI por definição.

Passo 5 Contar DER - Dado Elementar Referenciado (tipo de dado elementar)

Regras de Contagem de DER	A Regra se Aplica?
1. Contar um DER para cada atributo único, reconhecido pelo usuário e não repetido, que atravessa (entra e/ou sai) a fronteira durante o processamento de uma função de transação.	ID da Aplicação, Tópico Cadastrado, Mensagem(ns) de Ajuda retornadas.
2. Contar apenas um DER para a função de transação, para a habilidade da aplicação de enviar uma mensagem de resposta, mesmo que existam múltiplas mensagens.	Não existem
3. Contar apenas um DER para a função de transação, para a habilidade de iniciar ação(ões), mesmo que existam múltiplas maneiras de fazê-lo.	Sim. Seleção pelo Menu Ajuda
4. Não contar os seguintes itens como DERs: <ul style="list-style-type: none"> • literais • <i>application generated stamps</i> • variáveis de paginação • navegação • atributos gerados sem saída da fronteira e • atributos recuperados ou referenciados a partir de um ALI ou AIE sem saída da fronteira. 	<p>Literais como cabeçalhos de tela não são contados.</p> <p>Não existem</p> <p>Não existem</p> <p>Não existem</p> <p>Não existem</p> <p>Não existem</p>

Passo 6 Determinar a Complexidade Funcional

1 ALR e 4 DERs	Complexidade é Baixa
----------------	----------------------

Passo 7 Determinar Tamanho Funcional para a Capacidade de Pesquisa

Tamanho Funcional de 1 CE de Complexidade Baixa	3 PF
---	------

Exemplo: Logon de Nível de Segurança

Requisitos do Usuário O usuário requer uma função de *logon* para controlar o nível de segurança das janelas. A funcionalidade é ilustrada abaixo.

A função de *logon* requer a entrada dos seguintes campos:

- Identificador do Usuário
- Senha do Usuário

Quando o usuário realiza o *logon*, o arquivo de Nível de Segurança é lido para validar o identificador do usuário e a senha, assim como para determinar as janelas que o usuário pode acessar e manter.

Gerar mensagens de erro e destacar campos incorretos se os campos não permitirem edição. Duas (2) mensagens de erro e uma (1) mensagem de confirmação estão incluídas para a transação de *logon* de nível de segurança.

Tela de Logon de Nível de Segurança A ilustração a seguir apresenta a janela para o *logon* no sistema.

Sistema de Recursos Humanos	
Logon de Nível de Segurança	
ID Usuário	<input type="text"/>
Senha	<input type="password"/>

O usuário pressiona a tecla entra depois que toda informação é incluída. O ID do Usuário e a Senha são validados e logicamente todo acesso do usuário é assinalado.

Passo 1 Identificar o Processo Elementar

A Função de Transação atende aos requisitos de um Processo Elementar?	Sim.
---	------

Passo 2 Determinar se o Processo Elementar é único

A Função de Transação é única em relação a outros processos elementares?	Sim. Nenhum outro processo elementar executa esta função.
--	---

Passo 3 Classificar cada Processo Elementar

Regras de Contagem de CE	A Regra se Aplica?
1. Possui a intenção primária de apresentar informações ao usuário, e:	Sim. A intenção primária é apresentar informação ao usuário.
<ul style="list-style-type: none"> referencia uma função de dados para recuperar dados ou informação de controle e 	Sim. Dados são recuperados a partir do ALI Nível de Segurança.
<ul style="list-style-type: none"> não satisfaz o critério de ser classificado como uma SE 	Sim. Não executa cálculos, não cria dados derivados e não atualiza nenhum ALI

Conclusão O *Logon* de Nível de Segurança é uma CE.

Passo 4 Contar ALR - Arquivo Lógico Referenciado (tipo de arquivo referenciado)

Regra de Contagem de ALR	A Regra se Aplica?
1. Um ALR deve ser contado para cada função de dados que é acessada (lida e/ou mantida) pela função de transação.	O ALI Nível de Segurança é lido. Uma CE não pode manter um ALI por definição.

Passo 5 Contar DER - Dado Elementar Referenciado (tipo de dado elementar)

Regras de Contagem de DER	A Regra se Aplica?
1. Contar um DER para cada atributo único, reconhecido pelo usuário e não repetido, que atravessa (entra e/ou sai) a fronteira durante o processamento de uma função de transação.	ID Usuário, Senha.
2. Contar apenas um DER para a função de transação, para a habilidade da aplicação de enviar uma mensagem de resposta, mesmo que existam múltiplas mensagens.	Mensagens de Confirmação e Erro.
3. Contar apenas um DER para a função de transação, para a habilidade de iniciar ação(ões), mesmo que existam múltiplas maneiras de fazê-lo.	Tecla Entra
4. Não contar os seguintes itens como DERs: <ul style="list-style-type: none"> • literais • <i>application generated stamps</i> • variáveis de paginação • navegação • atributos gerados sem saída da fronteira e • atributos recuperados ou referenciados a partir de um ALI ou AIE sem saída da fronteira. 	<p>Literais como cabeçalhos de tela não são contados.</p> <p>Não existem</p> <p>Não existem</p> <p>Não existem</p> <p>Não existem</p> <p>Não existem</p>

Passo 6 Determinar a Complexidade Funcional

1 ALR e 4 DERs	Complexidade é Baixa
----------------	----------------------

Passo 7 Determinar Tamanho Funcional

Tamanho Funcional de 1 CE de Complexidade Baixa	3 PF
---	------

Esta página foi deixada intencionalmente em branco.

Índice da Parte 4

A

Aplicação

exemplo de contagem de menus, 2-129

Arquivos de interface externa

dados referenciados em outras aplicações, 1-39

exemplo de conversão de dados, 1-58

exemplo de help de aplicação, 1-50

help de janela, 1-50

help de nível de campo, 1-50

informação de conversão, 1-43

provendo dados para outras aplicações, 1-48

referenciando dados a partir de outra aplicação, 2-83

referenciando dados a partir de outra aplicação, 1-43

referenciando dados a partir de outras aplicações, 1-39

Arquivos lógicos internos

descrição geral dos exemplos, 1-8, 2-8

exemplo de índice alternativo, 1-20

B

Batch com EEs múltiplos e duplicados, 2-70

C

Consultas externas

exemplo de menus de aplicação, 2-129

Conversão de dados, 1-58, 2-80

Convertendo dados para um novo formato com elementos de dados adicionais, 2-80

D

Descrições geral de exemplos de ALI, 1-8, 2-8

Descrição geral de exemplos de entradas externas, 2-62

Descrição geral de exemplos de saídas externas, 2-104

Diagramas

dados de Funcionário para help de campo na parte de entrada em consultas, 2-143

exemplos de Contagem de componentes em EE/SE/CE, 2-3

exemplos de Contagem de componentes em ALI/AIE, 1-5

exemplo de formato de registro em arquivo de transação de entrada, 1-60

janela de Dados de Assinalamentos de Função, 2-76

modelo de conversão de dados, 1-43

modelo de dados para conversão de dados, 1-58

menu para a parte de entrada em consultas, 2-129

organização dos Exemplos de Contagem para EE/SE/CE, 2-2

tela de relatórios online, 2-108

tela de entrada de Dados de Função, 2-67

E

EE com múltiplos tipos de arquivos referenciados, 2-76

Exemplo de relatório de saída, 2-105

Exemplo de índice alternativo, 1-20

Exemplos de Contagem de EE/SE/CE, 2-2

Exemplos

batch com EEs múltiplos e duplicados, 2-70

conversão de dados, 1-58, 2-80

convertendo dados para novo formato, 2-80

descrições gerais de exemplos de entradas externas, 2-62

- descrições gerais de exemplos de saídas externas, 2-104
- diagrama de fluxo de dados para contagem referente a múltiplos arquivos, 2-76
- EE com múltiplos tipos de arquivos referenciados, 2-76
- help de aplicação, 1-50
- help de janela, 1-50
- help de nível de campo, 1-50
- índice alternativo, 1-20
- informação de controle, 2-63
- informação de conversão, 1-43
- menus de aplicação, 2-129
- provendo dados para outras aplicações, 1-48
- referências para múltiplos arquivos, 2-76
- referenciando dados a partir de uma outra aplicação, 1-43, 2-83
- referenciando dados a partir de outras aplicações, 1-39
- relatório de saída, 2-105
- relatório físico, 2-105
- tela de entrada, 2-67
- transação de inclusão online, 2-67
- transação com tipos de registro formatados, 2-70
- transação com múltiplos tipos, 2-70
- Entradas externas
 - batch com EEs múltiplos e duplicados, 2-70
 - EE com múltiplos tipos de arquivo referenciados, 2-76
 - exemplo de conversão de dados, 2-80
 - exemplo de tela de entrada, 2-67
 - informação de controle, 2-63
- F**
- Formato de registro para transação de arquivo de entrada, 1-60
- H**
- Help de aplicação, 1-50
- Help de campo para Dados de Funcionário, 2-143
- Help de janela, 1-50
- Help de nível de campo, 1-50
- I**
- Informação de Controle
 - para relatórios 2-63
- Informação de Conversão, 1-43
- J**
- Janela do Sistema de Recursos Humanos, 2-129
- Janela de Dados de Assinalamento de Função, 2-76
- Janelas
 - help de nível de campo, 2-143
- M**
- Manual
 - organização dos Exemplos de Contagem de ALI/AIE, 1-4
 - organização dos Exemplos de Contagem de EE/SE/CE, 2-2
- Modelo de Conversão de Dados, 1-43
- Modelos de dados para conversão de dados, 1-58
- Múltiplos tipos, 2-70
- N**
- Novo formato com elementos de dados adicionais, 2-80
- O**
- Organização. Ver Manual
 - exemplos de Contagem de EE/SE/CE, 2-2
- P**
- Provendo dados para outras aplicações, 1-48
- R**
- Referência a múltiplos arquivos, 2-76
- Referenciando dados a partir de outra aplicação, 1-43, 2-83
- Referenciando dados a partir de outras aplicações, 1-39
- Relatório físico, 2-105
- Relatórios
 - cópia física, 2-105
- S**
- Saídas externas
 - exemplo de relatório de saída, 2-105
- T**
- Tela de entrada, 2-67
- Tela de relatório online, 2-108
- Telas
 - funcionários por Duração de Assinalamento, 2-108
 - relatórios online, 2-108
- Tipos de registro, 2-70
- Tipos de registro formatados, 2-70
- Transação de inclusão online via tela, 2-67
- Transação com múltiplos tipos, 2-70
- Transação com tipos de registro formatados, 2-70

Parte 5 – Apêndices e Glossário

Página intencionalmente deixada em branco.

Apêndice A: Tabela de Cálculo de Tamanho Funcional

Introdução O Apêndice A inclui uma tabela para facilitar a medição de tamanho funcional.

Conteúdo Este apêndice inclui a seguinte tabela:

Tópico	Página
Tabela de cálculo de tamanho funcional	A-2

Tabela de cálculo de Tamanho Funcional

A tabela seguinte é fornecida para facilitar o cálculo da contribuição para o tamanho funcional.

Tipo de Função	Complexidade Funcional	Totais Complexidade	Totais Tipos de Função
ALI	Baixa	X 7 =	
	Média	X 10 =	
	Alta	X 15 =	
AIE	Baixa	X 5 =	
	Média	X 7 =	
	Alta	X 10 =	
EE	Baixa	X 3 =	
	Média	X 4 =	
	Alta	X 6 =	
CE	Baixa	X 3 =	
	Média	X 4 =	
	Alta	X 6 =	
SE	Baixa	X 4 =	
	Média	X 5 =	
	Alta	X 7 =	
Tamanho Funcional Total			

Apêndice B: A mudança da versão anterior

Introdução Este apêndice inclui informações sobre as mudanças e melhorias incluídas no CPM 4.3, o processo de tomada de decisão e recomendações para os usuários do novo manual.

Conteúdo Este capítulo inclui o seguinte:

Tópico	Página
Introdução	B-2
Áreas das principais mudanças de estrutura do CPM 4.3	B-3
Controle de Versão	B-3
Visão geral das mudanças	B-4
Background	B-12
Estudo de Impacto	B-12
Conversão do CPM 4.2 para o 4.3	B-12
Impacto nos Usuários do 4.2.1 mudando para o 4.3	B-14
Recomendações	B-14

Introdução

Desde a versão do Manual de Práticas de Contagem do IFPUG (CPM) 4.2 em 2004, o Comitê de Práticas de Contagem (CPC) criou a nova versão da Parte 1 (regras) para substituir o padrão ISO (ISO 20926:2004); isto é, o IFPUG CPM 4.1 não ajustado. A criação do novo padrão ISO exigiu mudanças no texto das demais partes (o guia de implementação) para manter a consistência.

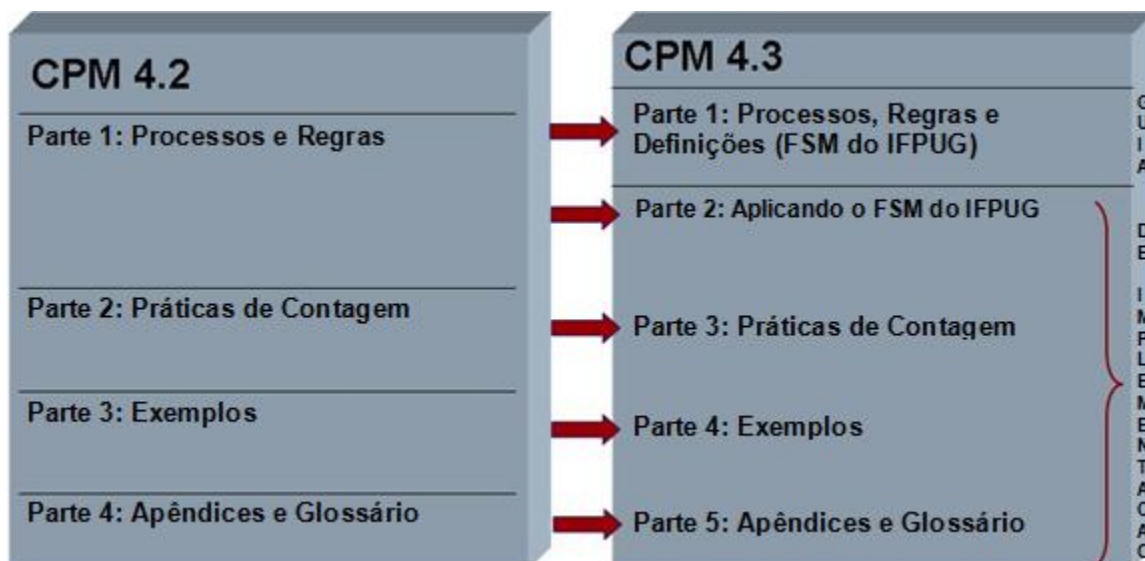
O processo do CPC de revisão do CPM é o seguinte:

1. A edição é submetida ao CPC pelos membros.
2. A edição é designada aos membros CPC para pesquisa.
3. O CPC revisa e discute a edição.
4. O CPC apresenta a solução proposta para os membros.
5. Um estudo de impacto é iniciado.
6. A decisão final é tomada.
7. Os membros do IFPUG são informados da decisão por meio da publicação MetricViews e apresentações nas conferências do IFPUG.
8. As mudanças tornam-se efetivas em um novo CPM.
9. Os estudos de casos são revisados para refletir o novo CPM.

Áreas das principais mudanças de estrutura do CPM 4.3

As áreas das principais mudanças de estrutura do CPM 4.3 são:

- Substituir a Parte 1 existente pela Parte 1 com o novo padrão ISO (ISO/IEC 20926:2010)
- Criar a Ponte – Aplicando Método de Tamanho Funcional do IFPUG (agora Parte 2) que provê um guia na aplicação do processo e regras, tal como definido no Padrão ISO (agora Parte 1)
- Aperfeiçoar as partes restantes a fim de ficarem consistentes com a Parte 1 revisada
 - Práticas de Contagem (Parte 3)
 - Exemplos (Parte 4)
 - Apêndices e Glossário (Parte 5)



Controle de Versão

O CPC escolheu dar o nome IFPUG CPM 4.3 a esta versão, ao invés de 4.2.2 ou 5.0 por duas razões:

- Uma versão com nome de 4.2.2 poderia sugerir que apenas a grafia foi corrigida; versão 4.3 chama mais atenção para a Parte 1 reescrita.
- Uma versão com nome de 5.0 poderia sugerir uma mudança das regras principais. O CPM 4.3 ainda é uma evolução da metodologia de Albrecht que forma a base de todas as versões anteriores do CPM do IFPUG. Esta versão provê esclarecimento adicional das versões anteriores.

Visão geral das mudanças

Além de substituir a Parte 1 existente com o padrão ISO, outros pequenos esclarecimentos foram incluídos no CPM 4.3. Para facilitar usuários que desejam alinhar seu CPM atual escrito em uma língua estrangeira, todas as mudanças foram listadas abaixo.

Parte 1: Processo e Regras

A fim de manter o guia de implementação do método FSM do IFPUG com o Padrão ISO revisado (ISO/IEC 14143-1:2007), parte da terminologia teve que ser revisada. As regras e diretrizes permanecem essencialmente inalteradas; contudo, a seqüência de ações e a redação foi ligeiramente alterada. Não se deve supor que o tamanho funcional seja alterado. Todos os capítulos na Parte 1 (agora Parte 2) incluem alterações de redação, exemplos adicionais e orientação a fim de ser consistente e adequado ao Padrão ISO de FSM atualizado, que foi lançado em 2007 e é agora a Parte 1. As CGSs e o Fator de Ajuste foram retirados desta parte e estão incluídos nos apêndices a fim de se adequar ao Padrão ISO de FSM, que não os reconhece como parte do FSM. Detalhes adicionais são indicados abaixo, por capítulo.

Parte 1, Capítulo 1: Introdução

O título deste capítulo foi alterado para “A Ponte – Aplicando o Método de Medição de Tamanho Funcional do IFPUG”. O capítulo 1 contém apenas uma Introdução; as mudanças que refletem o conteúdo dos demais capítulos estão na nova Parte 2.

Parte 1, Capítulo 2: Visão Geral da Análise de Pontos de Função

O título deste capítulo foi alterado para “Visão Geral do Método FSM do IFPUG”. Houve uma alteração extensiva da redação a fim de ser consistente e em conformidade com o Padrão ISO de FSM atualizado. As seguintes definições e regras foram ligeiramente reescritas:

O diagrama de processo e diretrizes neste capítulo foram alterados para refletir que o primeiro passo no processo de contagem de pontos de função é obter a documentação disponível, de acordo com o Padrão ISO de FSM.

O tamanho funcional agora representa o tamanho do software obtido pela quantificação dos requisitos funcionais do usuário, substituindo o termo “pontos de função não ajustados”. Qualquer discussão sobre “não-ajustado” ou “ajustado” está agora incluída no apêndice a fim de se adequar ao padrão ISO de FSM, que não reconhece as CGSs ou o VAF como parte do FSM.

Parte 1, Capítulo 3: Visão do Usuário

O título deste capítulo foi alterado para “Obter a documentação disponível”. Este capítulo apresenta o conceito de papel do usuário e a abordagem de medição durante o ciclo de vida de uma aplicação; contudo, virtualmente todo o capítulo permanece inalterado exceto pelo título.

Parte 1, Capítulo 4: Determinar o Tipo de Contagem

O título deste capítulo foi alterado para “Determinar o Tipo de Contagem”. Foi revisada a redação para as definições de contagem de pontos de função de projeto de desenvolvimento, contagem de pontos de função de projeto de melhoria e contagem de pontos de função de aplicação, para ser consistente e adequado ao Padrão ISO de FSM atualizado.

Parte 1, Capítulo 5: Identificar o Escopo da Contagem e Fronteira da Aplicação

O título deste capítulo foi alterado para “Determinar o Escopo da Contagem e a Fronteira e Identificar os Requisitos Funcionais do Usuário”. Há algumas pequenas alterações de redação a fim de ser consistente e adequado ao Padrão ISO de FSM atualizado, mas a grande maioria do capítulo permanece inalterada.

Parte 1, Capítulo 6: Contar Funções de Dados

O título deste capítulo foi alterado para “Medir Funções de Dados” para refletir que as regras estão realmente contidas na nova Parte 1 e para refletir que este capítulo provê orientações de implementação para medir Funções de Dados. As regras contidas são repetidas da Parte 1 para facilitar a utilização, e evitar a necessidade de folhear as partes adiante e anteriores.

Parte 1, Capítulo 7: Contar Funções de Transação

O título deste capítulo foi alterado para “Medir Funções de Transação” para refletir que as regras estão realmente contidas na nova Parte 1, e para refletir que este capítulo provê orientações de implementação para medir Funções de Transação. Tal como no Capítulo 6, as regras contidas são repetidas da Parte 1 para facilitar a utilização, e evitar a necessidade de folhear as partes adiante e anteriores. Itens específicos incluem:

- Orientação adicional e esclarecimento sobre as regras do FSM para processos elementares
- Regras simplificadas para DER e ALR

Parte 1, Capítulo 8: Determinar o Fator de Ajuste

O conteúdo completo deste capítulo foi movido para o Apêndice C a fim de alinhar o Guia de Implementação com o FSM do IFPUG que não inclui as CGSs e o VAF.

Parte 1, Capítulo 9: Calcular a Contagem de Pontos de Função Ajustada

As fórmulas previamente contidas neste capítulo foram movidas para o Apêndice C e Parte 3 Capítulo 4 Melhoria e Atividade de Manutenção, a fim de alinhar o Guia de Implementação com o FSM do IFPUG, que não inclui as CGSs e o VAF.

Parte 2: Práticas de Contagem

Todos os capítulos na Parte 2 (agora Parte 3) incluem pequenas alterações de redação a fim de manter a consistência com a reorganização da Parte 1 e /ou manter a conformidade com o Padrão ISO de FSM atualizado. Um novo capítulo (Capítulo 5) foi acrescentado para prover compreensão clara da atividade de contagem de Conversão de Dados. Detalhes adicionais estão indicados por capítulo abaixo.

Parte 2, Capítulo 1: Dados de Código

A APF do IFPUG está em conformidade com o Padrão ISO FSM. A decisão de não contar Dados de Código e de criar o capítulo Dados de Código na Parte 2 do CPM 4.2 teve origem nos requisitos do Padrão ISO de FSM (ISO/IEC 14143-1:1998) de não contar requisitos técnicos e de qualidade.

Em 2007, a ISO publicou uma nova versão da FSM Padrão (ISO/IEC 14143-1:2007). Conseqüentemente, o capítulo Dados de Código precisou ser atualizado para refletir as alterações de redação no Padrão ISO de FSM.

Não há alterações nas regras nem nas orientações deste capítulo, mas há pequenas alterações de redação para manter a conformidade com o Padrão ISO de FSM atualizado.

- Incluída a definição ISO de Tamanho Funcional
- Atualizada a definição Requisitos Funcionais do Usuário
- Substituídos os termos Requisitos de Qualidade e Requisitos Técnicos pelo termo ISO Requisitos Não-Funcionais do Usuário e incluída a definição ISO para este conceito
- A seção Metodologia foi ligeiramente reescrita para refletir as mudanças no passo “Identificar Arquivos Lógicos” no Capítulo 2: Arquivos Lógicos abaixo.

Parte 2, Capítulo 2: Arquivos Lógicos

Este capítulo foi criado no CPM 4.2 para prover práticas de contagem e orientação adicional na identificação e avaliação de Arquivos Lógicos.

No CPM 4.3, a Parte 1 foi substituída pelo padrão ISO de APF do IFPUG

Algumas alterações para o padrão ISO de FSM têm conseqüência (pequena) no capítulo Arquivos Lógicos:

- No processo de identificação de Arquivos Lógicos, o passo1 anterior (“Remoção de Dados de Código antes da avaliação dos Arquivos Lógicos”) tornou-se agora parte do passo1 “Identificar Arquivos Lógicos”, que é o local mais adequado.
- Além disto, o passo 2 anterior (“Identificar Arquivos Lógicos e Classificar”) foi decomposto em dois passos “1. Identificar Arquivos Lógicos” e “2. Classificar Arquivos Lógicos”
- Passos 3 e 4 (identificando RLRs e DERs) foram inter-cambiados
- Os subpassos do passo1 tornaram-se melhor visualizados, através da denominação efetiva dos mesmos como sub-passos.

Estas alterações na estrutura têm algumas conseqüências (pequenas) na estrutura do capítulo Arquivos Lógicos.

Isto é particularmente verdade para o intercambio dos passos Identificando DERs e Identificando RLRS, e tornou necessário o intercambio das páginas relacionadas a estes passos. Isto também foi necessário para combinar as tabelas “Considerando Registros Lógicos Referenciados em conjunto com Arquivos Lógicos via (In-) Dependência de Entidade” (CPM 4.2, página 2-34) e “Considerando Dados Elementares Referenciados em conjunto com Arquivos Lógicos via (In-) Dependência de Entidade” (CPM 4.2, página 2-46) em uma nova tabela “Considerando Registros Lógicos Referenciados e Dados Elementares Referenciados em conjunto com Arquivos Lógicos via (In-) Dependência de Entidade”

Há ligeiras alterações de redação para adequação ao padrão ISO de FSM atualizado (ISO/IEC 14143-1:2007) tal como explicado em mais detalhes na seção acima, dedicada a Parte 2, Capítulo 1 Dados de Código.

Requisitos de Qualidade e Requisitos Técnicos foram substituídos pelo novo termo ISO Requisitos Não-Funcionais do Usuário.

Não se deve supor que, nenhuma destas alterações na estrutura e redação tenha qualquer influência no resultado de qualquer contagem.

Parte 2, Capítulo 3: Dados Compartilhados

Este capítulo foi criado no CPM 4.2 para prover práticas de contagem e orientação adicional na identificação e avaliação de dados compartilhados entre aplicações.

As únicas alterações neste capítulo foram duas referências a outras partes do CPM, que agora são diferentes.

Parte 2, Capítulo 4: Projetos de Melhoria e Atividades de Manutenção

Este capítulo foi criado no CPM 4.2 para prover práticas de contagem e orientação adicional na aplicação da Análise de Pontos de Função para atividades pós desenvolvimento. A contagem de projetos de melhoria, apresentada antes, na Parte 1, Capítulo 9 do CPM 4.1, é agora inteiramente contemplada (incluindo fórmulas aplicáveis) neste capítulo.

Alem das atualizações de referências a outras partes do CPM, as definições e exemplos primários para cada uma das formas de lógica de processamento foram ajustados para serem consistentes com os da nova Parte 2. Termos foram ajustados para consistência com as Partes 1 e 2, como por exemplo, alteração de “campo” para “atributo”.

Alterações específicas na seção Lógica de Processamento neste capítulo incluem o seguinte:

3. Valores Equivalentes: Exemplo alterado em resposta aos comentários do Bulletin Board do IFPUG.
4. Dados são Filtrados: Exemplo existente modificado para excluir a contagem de

uma mudança envolvendo apenas a substituição ou adição de valores, e acrescentados três novos exemplos em resposta aos comentários do Bulletin Board do IFPUG.

11. Preparar e apresentar informações para fora da fronteira: Acrescentados três novos exemplos para refletir as respostas do CPC aos comentários do Bulletin Board do IFPUG.
12. Aceitar informações que entram pela fronteira: Acrescentados dois novos exemplos para refletir as respostas do CPC aos comentários do Bulletin Board do IFPUG.
13. Classificação (Dados são re-classificados ou re-arranjados): Acrescentados dois novos exemplos para refletir as respostas do CPC aos comentários do Bulletin Board do IFPUG.

Em Considerações e Dicas, foi incluída discussão relativa a Funções excluídas, e as dicas sobre CGSs foram movidas para o Apêndice C, onde as CGSs e VAF opcionais são contemplados.

Na seção Melhoria versus Manutenção, qualquer referência as CGSs foi precedida com “opcional”.

Parte 2, Capítulo 5: Atividade de Conversão de Dados (novo capítulo)

Este novo capítulo contempla a funcionalidade a ser avaliada quando existem requisitos para migrar ou converter dados em conjunto com o novo desenvolvimento ou projeto melhoria, ou para migrar uma aplicação para uma plataforma diferente. A Parte 4 do CPM provê outros exemplos de Funções de Dados e de Funções de Transação para conversão de dados.

Parte 3: Exemplos

Em todos os capítulos na Parte 3 (agora Parte 4) foram feitas revisões nos quadros de regras nesta seção para estar consistente com as alterações de redação nas regras de funções de dados, processo elementar e função de transação. Detalhes adicionais são indicados por capítulo abaixo.

Parte 3, Capítulo 1: Exemplos de Contagem de Funções de Dados

- Exemplo de ALI: Dados de Auditoria para consulta e Relatórios – Removidas referências a Manutenção de Segurança do Funcionário do Diagrama de Fluxo de Dados, pois estava confuso e não explicado adequadamente
- Exemplo de ALI: Definição de relatório – Adicionada explicação esclarecendo porque a Definição de relatório não é uma instância de dados de código
- Exemplo de ALI: Dados Compartilhados por aplicações– Exemplo esclarecido para garantir entendimento que a segurança descrita neste exemplo não é aplicação de segurança (ou seja, determinando o que o usuário pode acessar na aplicação)
- Exemplo de AIE: Fornecendo Dados para outras Aplicações – Adicionada explicação esclarecendo porque a Conversão de moeda corrente não é uma instância de dados de código
- Exemplo de AIE: Aplicação de Help – Adicionada explicação, esclarecendo porque o Help não é uma instância de dados de código; e também explicando porque a Janela de Help e o Help de campo são Funções de Dados separadas

Parte 3, Capítulo 2: Exemplos de Contagem de Funções de Transação

- Exemplo de PE: Funcionário Novo / Dados do Dependente – Adicionada explicação, esclarecendo porque o envio do arquivo ao Sistema de Benefícios é um processo elementar separado
- Exemplo de PE: Alimentar Dados de Funcionário via Batch – Este novo exemplo ilustra que a produção de relatórios de erro via batch e relatórios estatísticos não são processos elementares separados
- Exemplo de PE: Designar funcionário a Função - Este novo exemplo ilustra a avaliação de processos elementares similares para determinar se são únicos
- Exemplo de PE: Designar funcionário a Função - Este novo exemplo ilustra dois Processos Elementares similares que são contados como transações únicas
- Exemplo de EE: EE com atributos recuperados de um AIE – Este novo exemplo ilustra uma EE com atributos recuperados de um AIE que não cruzam a fronteira
- Exemplo de EE: EE de Exclusão – Este novo exemplo ilustra a contagem de DERs para transações de exclusão
- Exemplo de EE: Janela de Inclusão de Segurança – Este novo exemplo ilustra a contagem de funcionalidade para manter uma aplicação de segurança
- Exemplo de SE: SE disparada automaticamente sem dados atravessando a fronteira – Exemplo renomeado para eliminar confusão
- Exemplo de CE: CE disparada automaticamente sem dados atravessando a fronteira – Exemplo renomeado para eliminar confusão
- Exemplo de CE: Funcionalidade Adicional de Help – Este novo exemplo ilustra a contagem de funcionalidade adicional de Help
- Exemplo de CE: Segurança para Acesso pelo Usuário – Este novo exemplo ilustra o tratamento de uma aplicação de segurança
- Exemplo de CE: Logon de Aplicação – Este novo exemplo ilustra a contagem de função de logon

Parte 4: Apêndices e Glossário

Todos os capítulos na Parte 4 (agora Parte 5) incluíram revisões. Detalhes são indicados por capítulo abaixo.

Parte 4, Apêndice A: Tabelas de Cálculo

Pequenas alterações de redação para eliminar o uso do termo não-ajustado.

Parte 4, Apêndice B: A mudança da versão anterior

Este novo capítulo inclui o seguinte:

- A principais áreas de mudança funcional no CPM 4.3
- Informação de controle de versão
- Visão Geral das mudanças por capítulo
- O background do processo de mudança
- O processo de estudo de impacto
- O impacto das mudanças nos usuários do 4.3

- Conversão do CPM 4.2.1 para o 4.3
- Recomendações para usuários mudando do 4.2.1 para o 4.3

Parte 4, Apêndice C: Formulário de Solicitação do Leitor

O Formulário de Solicitação do Leitor foi eliminado. Leitores podem sugerir alterações enviando um email ao CPC (cpc@ifpug.org). O título do Apêndice C foi alterado para Tamanho Funcional Ajustado e agora contem orientação para a aplicação das Características Gerais do Sistema e para o Fator de Ajuste. Ele contem todas as fórmulas que utilizam as CGSs e o VAF.

Parte 4, Glossário

Os seguintes novos termos foram acrescentados ao glossário:

- Tamanho funcional ajustado da aplicação (aAFP)
- Tamanho funcional ajustado da aplicação depois de projetos de melhoria. (aAFPA)
- Tamanho funcional ajustado de projeto de desenvolvimento (aDFP)
- Tamanho funcional ajustado de projeto de melhoria (aEFP)
- Tamanho funcional da aplicação
- Arranjo
- Componente Funcional Básico
- Fronteira
- Fronteira da aplicação
- Estado consistente
- Tamanho funcional de projeto de desenvolvimento
- Tamanho funcional de projeto de melhoria
- Tamanho funcional
- Significativo
- Intenção primaria
- Auto-contido
- Classificação
- Tamanho funcional não ajustado

Os seguintes termos foram revisados no glossário:

- Manutenção adaptativa
- Contagem de Pontos de Função Ajustados (AFP) *
- Aplicação
- Fronteira da Aplicação
- Contagem de Pontos de Função da Aplicação
- Contribuição
- Informação de Controle
- Funcionalidade de Conversão
- Manutenção Corretiva
- Escopo da Contagem
- Dados Derivados
- Desenvolvimento *
- Contagem de pontos de função de projeto de desenvolvimento (DFP)
- Melhoria *
- Contagem de Pontos de Função de projeto de Melhoria (EFP)
- Entidade Dependente
- Entidade Independente
- Sistema de Arquivo
- Tipo de Arquivo Referenciado (FTR)
- Complexidade Funcional
- Requisitos Funcionais do usuário
- Ponto de Função (PF)
- Análise de Pontos de Função
- Contagem de Pontos de Função
- Tipo de Função
- Mantido *
- Manutenção
- Múltiplos locais CGS
- Manutenção Perfectiva
- Propósito da Contagem
- Tipo de Registro Elementar (RET)
- Atributo Técnico
- Funções de Transação
- Contagem de Pontos de Função Não-ajustado (UFP)
- Usuário
- Reconhecido pelo Usuário
- Visão do Usuário

* Significa locais onde o nome do termo foi alterado (Ex.: Contagem de Pontos de Função Ajustados (AFP) tornou-se Tamanho Funcional Ajustado).

Background

O processo de tomada de decisão interna do CPC é governado por um conjunto de características do CPM (meta regras) selecionadas e votadas pela diretoria do IFPUG e o CPC. Essas diretrizes principais em ordem de importância são:

1. Deve ser possível modelar a correlação do tamanho do software (derivado usando o CPM) com outros atributos (ex.: esforço, defeitos, custo, etc.).
2. O CPM contém um conjunto de regras consistente.
3. Os resultados da Análise de Pontos de Função são consistentes entre diferentes contadores usando o CPM.
4. O CPM fornece regras em como medir o tamanho de necessidade funcional que esteja definida e acordada pelos usuários e TI.
5. Os resultados da Análise de Pontos de Função usando o CPM podem ser um fator de contribuição na estimativa.
6. O CPM é um método baseado na proposta de Allan Albrecht.
7. Análise de Pontos de Função usando o CPM é fácil.
8. Análise de Pontos de Função usando o CPM é rápida.

Estudo de Impacto

O processo e as Regras da Análise de Pontos de Função (APF) do IFPUG são concisos e fáceis de utilizar. Para refletir isto e tornar o Manual de Práticas de contagem (CPM) cada vez mais atrativo como uma manual de referência, o Comitê de Práticas de contagem (CPC) reestruturou o CPM 4.3 para se adequar ao Padrão ISO de formatação. Além disto, a versão 4.3 contém pequenas modificações e provê novos exemplos, esclarecimentos e interpretações aperfeiçoadas para as regras existentes que irão aumentar a consistência entre contadores.

Para medir a efetividade desta nova versão, um estudo de impacto foi realizado por 44 Especialistas Certificados em Pontos de Função que não tinham ligação direta com o Comitê de Práticas de Contagem. Foi solicitado a estes voluntários contar um estudo de caso, utilizando tanto o CPM 4.2.1 como o CPM 4.3. Os resultados foram idênticos em ambas as versões. Estes participantes contaram projetos que tinham sido executados sob as regras do CPM 4.2.1 usando o novo CPM 4.3. No total mais de 100 contagens incluindo desenvolvimento, aplicação, melhoria e conversão foram consideradas. O fator resultante da conversão foi 1.0; ou seja nenhuma diferença.

Conversão do CPM 4.2 para o 4.3

Considerando que as práticas existentes variam, cada organização deve analisar suas próprias práticas para determinar qual o impacto. Algumas organizações devem encontrar um fator de conversão que seja aplicável em seu portfólio. Outras devem encontrar um fator de conversão que varie através dos diferentes tipos de sistemas, e, em alguns casos, sistemas precisarão ser

recontados.

Impacto nos Usuários do 4.2.1 mudando para o 4.3

Embora certificação adicional não seja requerida para contadores para o CPM 4.3, os testes de certificação serão atualizados para conformidade ao 4.3.

Recomendações

O CPC recomenda as seguintes ações para usuários trocando do CPM 4.2 para o 4.3:

- Atualize todo o material de treinamento desenvolvido internamente para obter conformidade.
- Garanta que todos os contadores em sua organização foram adequadamente treinados nas diferenças entre o 4.2 e o 4.3.
- Verifique todos os materiais oferecidos pelo fornecedor para versão de certificação.
- Notifique qualquer um em sua organização que esteja envolvido com medições de tamanho funcional da mudança, e faça com que o novo manual esteja disponível a estas pessoas.
- Revise todas as ferramentas de contagem de seus usuários, tanto as automáticas como as manuais, para a versão 4.3 de certificação do IFPUG, e, se aplicável, efetue modificações para adequar-se as regras de contagem do 4.3.
- Se estiver provendo serviços baseados em Pontos de Função, garanta que a redação do contrato seja revisada para determinar qual versão do manual CPM será usada; retifique se necessário.
- Especifique na documentação de medição de tamanho funcional executada, e nos resultados, qual versão do CPM foi utilizada.
- Explícite que versão do CPM do IFPUG foi usada para contagem quando submeter dados para benchmarking, mesmo para seu próprio banco de dados benchmark, para o comitê de Benchmarking do IFPUG, ou para o ISBSG.
- Atualize todas as diretrizes internas e outros documentos locais relacionados ao 4.2 para a versão 4.3.

Apêndice C: Tamanho Funcional Ajustado

Introdução

Este capítulo explica as Características Gerais dos Sistemas (CGSs) e o Valor do Fator de Ajuste (VAF).

Nota: Aplicação das CGSs, cálculo do VAF, e cálculo do tamanho funcional ajustado não estão incluídos no FSM do IFPUG e são considerados opcionais no CPM do IFPUG.

De qualquer forma, quando se informa o tamanho funcional medido utilizando o método do IFPUG, o tamanho informado é o resultado antes de qualquer ajuste. O tamanho é informado em unidades de PFs. Quando estiver utilizando tamanhos informados por outras fontes, primeiro determine se eles foram ‘ajustados’ ou estão ‘não ajustados’. Falhar nesta determinação pode resultar em erros ao comparar ou utilizar os dois conjuntos de resultados. Sempre demonstre se o tamanho funcional informado é ‘ajustado’ (aFP) ou ‘não ajustado’ (FP).

Conteúdo

Este capítulo inclui as seguintes seções:

Tópico	Página
Passos para Cálculo do Tamanho Funcional Ajustado	C-3
Determinação do Fator de Ajuste	C-4
Diretrizes para determinar o Grau de Influência das CGSs	C-6
Tabela de Cálculo do Fator de Ajuste	C-31
Pontos de Função Ajustados de Projeto de Desenvolvimento	C-32

Tópico	Página
Fórmula: Tamanho Funcional Ajustado do Projeto de Desenvolvimento (aDFP)	C-32
Exemplo: Tamanho Funcional Ajustado do Projeto de Desenvolvimento (aDFP)	C-33
Tamanho Funcional Ajustado do Projeto de Melhoria (aEFP)	C-37
Formula: Tamanho Funcional Ajustado do Projeto de Melhoria (aEFP)	C-38
Exemplo: Tamanho Funcional Ajustado do Projeto de Melhoria (aEFP)	C-39
Tamanho Funcional Ajustado da Aplicação (aAFP)	C-43
Fórmula: Tamanho Funcional Inicial Ajustado da Aplicação (aAFP)	C-43
Fórmula: Tamanho Funcional Ajustado da Aplicação Depois do Projeto de Melhoria (aAFPA)	C-44
Exemplo: Contagem de Aplicação	C-45

Passos para Cálculo do Tamanho Funcional Ajustado

A lista a seguir inclui os passos da análise de pontos de função apresentados na Parte 1, ampliados para prover o tamanho funcional ajustado.

Passo	Ação
1	Determinar o tamanho funcional utilizando as regras da Parte 1 e a orientação de implementação da Parte 2.
2	Determinar o Fator de Ajuste de acordo com a orientação neste apêndice.
3	Calcular o tamanho funcional ajustado de acordo com as fórmulas neste apêndice.

O restante deste capítulo apresenta as Características Gerais do Sistema, Fator de Ajuste resultante seguidos das fórmulas para calcular o Tamanho Funcional Ajustado. Exemplos de cálculos estão incluídos para cada um dos três tipos de contagem de pontos de função:

- Projeto de Desenvolvimento
- Projeto de Melhoria
- Aplicação

Determinação do Fator de Ajuste

O Fator de Ajuste (VAF) é baseado nas 14 características gerais do sistema (CGSs) que classificam a funcionalidade geral da aplicação sendo contada.

Cada característica tem descrições associadas que ajudam a determinar o nível de influência da característica. O nível de influência de cada característica varia em uma escala de 0 a 5, de sem influência até forte influência.

As 14 características gerais do sistema estão resumidas no Fator de Ajuste. Quando aplicado, o fator de ajuste ajusta o tamanho funcional não ajustado em +/- 35% para produzir o tamanho funcional ajustado.

Procedimentos para Determinar o VAF

Os seguintes passos descrevem os procedimentos para determinar o fator de ajuste.

Passo	Ação
1	Avalie cada uma das 14 características gerais do sistema na escala de 0 a 5 para determinar o nível de influência (NI).
2	Some os níveis de influência das 14 características gerais do sistema para produzir o total do nível de influência (TDI).
3	Insira o TDI na equação abaixo para obter o fator de ajuste.

$$\text{VAF} = (\text{TDI} * 0,01) + 0,65$$

Por exemplo, o seguinte fator de ajuste é calculado se houver o nível de influência 3 para cada uma das 14 CGS (3 * 14)

$$\begin{aligned}\text{VAF} &= (42 * 0,01) + 0,65 \\ \text{VAF} &= 1,07\end{aligned}$$

Uma tabela para facilitar o cálculo é apresentada neste Apêndice.

Características Gerais do Sistema

As características gerais do sistema são um grupo de 14 questões que avaliam a complexidade da aplicação como um todo.

As 14 características gerais do sistema são:

1. Comunicação de Dados
2. Processamento Distribuído
3. Performance
4. Configuração Intensamente Utilizada
5. Volume de Transações
6. Entrada de Dados On-Line
7. Eficiência do Usuário Final
8. Atualização On-Line
9. Processamento Complexo
10. Reusabilidade
11. Facilidade de Instalação
12. Facilidade de Operação
13. Múltiplos Locais
14. Facilidade de Mudança

Níveis de Influência

Com base nos requisitos estabelecidos pelo usuário, cada característica geral do sistema (CGS) deve ser avaliada em termos de seus níveis de influência (NI) em uma escala de 0 a 5.

Pontue como	Influência no Sistema
0	Não presente ou sem influência
1	Influência Mínima
2	Influência Moderada
3	Influência Média
4	Influência Significativa
5	Forte influência

Diretrizes para determinar o Grau de Influência das CGSs

Cada uma das descrições das características gerais do sistema seguintes inclui diretrizes para a determinação do nível de influência.

Cada diretriz contém uma definição da CGS, regras para determinação do nível de influência e, em situações nas quais a regra requer esclarecimento adicional, são fornecidas dicas para ajudar a aplicar as regras consistentemente em todas as plataformas.

Não se pretende que as dicas cubram todas as situações. Ao invés disso, a intenção é que as mesmas forneçam orientação adicional para a determinação do nível de influência apropriado.

1. Comunicação de Dados

Definição A característica Comunicação de Dados descreve até que ponto a aplicação se comunica diretamente com o processador.

Os *dados e informações de controle* utilizados pela aplicação são enviados ou recebidos através de recursos de comunicação. Considera-se que os dispositivos conectados localmente à unidade de controle utilizam recursos de comunicação. O protocolo é um conjunto de convenções que permite a transferência ou intercâmbio de informações entre dois sistemas ou dispositivos. Todos os links de comunicação de dados necessitam de algum tipo de protocolo.

Pontos

Atribua	Descrições para Determinar o Nível de Influência
0	A aplicação é puramente batch ou uma estação de trabalho isolada
1	A aplicação é batch, mas possui entrada de dados <i>ou</i> impressão remota
2	A aplicação é batch, mas possui entrada de dados <i>e</i> impressão remota
3	A aplicação inclui entrada de dados on-line ou front-end de teleprocessamento para um processo batch ou sistema de consulta
4	A aplicação é mais que um front-end, mas suporta <i>apenas um</i> tipo de protocolo de comunicação.
5	A aplicação é mais que um front-end, e suporta <i>mais de um</i> tipo de protocolo de comunicação.

Dicas

Exemplos de protocolo incluem FTP, dial-up, Token Ring, Ethernet, SNA, TCP/IP, IPX/SPX, http, XML, WAP, NTP, ICQ e NETBEUI. Esta lista não deve ser considerada completa.

Dicas para as Regras 1 e 2

- Dispositivos remotos podem incluir um terminal 3270 conectado ao computador mainframe que permita apenas validações simples (numérico versus alfa) ou impressoras conectadas através de portas paralelas (o usuário pode especificar para onde a saída será direcionada).
- A entrada de dados não envolve leitura ou gravação diretamente em um ALI. Os dados são informados on-line, mas as transações são armazenadas em um arquivo temporário para posterior atualização batch do (s) ALI(s).

**Dicas para
a
Regra 3**

- Regras de negócio simples e um mínimo de validações (ex.: alfa/numérico, verificação de faixa, dados obrigatórios, etc.) podem ser executados. Quando os dados forem finalmente processados pela aplicação, validações adicionais serão executadas.
- A entrada de dados não envolve leitura ou gravação diretamente em um ALI. Os dados são informados on-line, mas as transações são armazenadas em um arquivo temporário para posterior atualização batch do (s) ALI(s).

**Dicas para
a
Regra 4**

- Os dados da aplicação são coletados e podem atualizar diretamente os ALI(s), ou serem armazenados para processamento futuro, utilizando um dispositivo de entrada que executa validações baseadas nas regras de negócio.
- Apenas um protocolo de comunicação de dados é utilizado. Normalmente não haverá necessidade de validações adicionais quando os dados forem processados pela aplicação.
- A entrada de dados envolve leitura ou gravação em um ALI.
- Por exemplo, “data entry” de cliente-servidor ou “data entry” de Internet, mas não ambos.

**Dicas para
a
Regra 5**

- Idem ao 4, entretanto, a coleta de dados é executada utilizando vários protocolos de processamento.
- Por exemplo, “data entry” de cliente-servidor e “data entry” de Internet para a mesma transação.

Normalmente

- Aplicações batch pontuam de 0 a 3
- Aplicações on-line pontuam 4
- Aplicações Web pontuam 4 ou 5
- Sistemas real-time, de telecomunicações ou de controle de processos pontuam 4 ou 5.

2. Processamento Distribuído

Definição A característica Processamento Distribuído descreve até que ponto a aplicação transfere dados entre seus componentes físicos.

Funções distribuídas de dados ou de processamento são uma característica da aplicação dentro de sua respectiva fronteira.

Pontos

Atribua	Descrições para Determinar o Nível de Influência
0	Dados não são transferidos ou processados em outro componente do sistema.
1	Dados são preparados para transferência, sendo então transferidos e processados em outro componente do sistema, para processamento pelo usuário.
2	Dados são preparados para transferência, sendo então transferidos e processados em outro componente do sistema, <i>não</i> para processamento pelo usuário.
3	O processamento distribuído e a transferência de dados são online e em apenas uma direção.
4	O processamento distribuído e a transferência de dados são online e em ambas as direções.
5	O processamento distribuído e a transferência de dados são online e executados dinamicamente no componente mais apropriado do sistema.

Dicas

O processamento distribuído de dados, por definição, não é uma aplicação contida em um processador central que envia dados para outra aplicação. Em um ambiente distribuído, a aplicação é vista como requerendo vários componentes (hardware) no qual certo processamento ou dados residem. Um usuário capacitado irá normalmente reconhecer esta configuração.

Dicas para a Regra 0

- Os componentes de apresentação, processamento e I/O estão todos no mesmo lugar (ex.: aplicações stand-alone).

Dicas para a Regra 1

- A aplicação transfere dados para a máquina-cliente de um usuário, então ele pode usar o Excel ou outras ferramentas de relatórios para preparar gráficos e executar outras análises.
- Processo que transfere dados do mainframe para um componente externo para processamento do usuário. Esta transferência é executada utilizando um protocolo simples, como FTP.
- Transferidos para um usuário para processamento.

**Dicas
para a
Regra 2**

- Processo que transfere dados do mainframe para uma plataforma intermediária. Por exemplo: processamento com SAS-PC.
- A aplicação envia dados para o cliente ou para o servidor. Estes dados são então processados ou utilizados para produzir relatórios, etc. Nenhum dado ou confirmação é enviado de volta para o cliente ou servidor.
- Transferidos para um componente para processamento.

**Dicas
para a
Regra 3**

- Os dados são enviados entre o cliente e o servidor em apenas **uma** direção. Estes dados são então processados ou utilizados para produzir relatórios, etc. pela aplicação receptora. Estes dados tipicamente incluem transações que atualizam um ALI no cliente ou servidor.
- Por exemplo, aplicações cliente-servidor ou web.

**Dicas
para a
Regra 4**

- Dados são enviados entre o cliente e o servidor em **ambas** as direções. Estes dados são então processados ou utilizados para produzir relatórios, etc. pela aplicação receptora. Estes dados tipicamente incluem transações que atualizam um ALI no cliente ou servidor.
- Por exemplo, aplicações cliente-servidor ou web.
- A aplicação roda sob um sistema operacional que trata automaticamente a alocação entre componentes, **porém**, o uso do sistema operacional não influencia o projeto e implementação da aplicação.

**Dicas
para a
Regra 5**

- O desenvolvedor deve considerar uma aplicação de software especial que olhe para vários processadores e roda a aplicação em um tipo específico de processador. Isto é invisível para o usuário.
- A aplicação roda sob um sistema operacional que trata automaticamente a alocação dinâmica entre componentes, **e** o uso do sistema operacional influencia especificamente o projeto e implementação da aplicação.

Normalmente

- A maioria das aplicações, incluindo aplicações legadas, recebem 0.
- As aplicações distribuídas primitivas, inclusive aplicações batch em que dados não são transferidos online pontuam 1 ou 2.
- Aplicações cliente-servidor ou web recebem 3 ou 4.
- É raro uma nota 5.
- Existindo múltiplos servidores ou processadores, cada qual seria selecionado dinamicamente de acordo com sua disponibilidade para receber nota 5.

3. Performance

Definição A característica Performance descreve o grau segundo o qual considerações sobre tempo de resposta e performance de throughput (volume de processamento) influenciaram o desenvolvimento da aplicação. Objetivos de performance da aplicação, declarados ou aprovados (ou implícitos) pelo usuário, referentes a tempo de resposta *ou* throughput, influenciam ou influenciarão o projeto, desenvolvimento, instalação e suporte à aplicação.

Pontos

Atribua	Descrições para Determinar o Nível de Influência
0	Nenhum requisito especial de performance foi estabelecido pelo usuário.
1	Requisitos de performance e projeto foram estabelecidos e revisados, mas nenhuma ação especial foi requerida.
2	Tempo de resposta e volume de processamento são críticos durante o horário de pico. Nenhum projeto especial para utilização da CPU foi solicitado. O prazo para processamento é para o próximo ciclo de negócios.
3	Tempo de resposta e volume de processamento são críticos durante todo o horário comercial. Nenhum projeto especial para utilização da CPU foi solicitado. Os requisitos de prazo para processamento das interfaces com sistemas são restritivos.
4	Adicionalmente, requisitos de performance declarados pelo usuário são suficientemente rigorosos para requerer tarefas de análise de performance na fase de <i>design</i> .
5	Adicionalmente, ferramentas de análise de performance foram usadas nas fases de projeto, desenvolvimento, e/ou implementação para satisfazer os requisitos de performance declarados pelo usuário.

Dicas

- As CGS 3, 4 e 5 estão de certa forma relacionadas. Para esta CGS, pense em termos de “O quão rápido nós conseguimos fazer a aplicação rodar e quanto isto impactou o projeto, desenvolvimento e/ou implementação?”.
- Os usuários podem requerer acesso a seus dados em tempo real, estabelecendo, explicitamente ou não, padrões para tempo de resposta e capacidade de processamento (throughput).
- Tempo de resposta normalmente diz respeito ao processamento interativo; Throughput (volume de processamento) refere-se ao processamento batch.

Normalmente

- Aplicações batch recebem nota de 0 a 4.
- Aplicações on-line (incluindo cliente-servidor interativo ou web) recebem de 0 a 4.
- Aplicações web recebem 4 ou 5.
- A maioria dos sistemas on-line MIS (Management Information System - Sistema de Informação Gerencial) recebe 2.
- Sistemas real-time, de telecomunicações ou controle de processos recebem de 0 a 5.
- Uma nota 5 requer o uso de ferramentas de análise de performance.

4. Configuração Intensamente Utilizada

Definição Configuração Intensamente Utilizada descreve o nível segundo o qual as restrições nos recursos do computador influenciam o desenvolvimento da aplicação.

Uma configuração operacional intensamente utilizada pode requerer considerações especiais no projeto da aplicação. Por exemplo, o usuário deseja executar a aplicação em um equipamento existente ou alocado, que será intensamente utilizado.

Pontos

Atribua	Descrições para Determinar o Nível de Influência
0	Nenhuma restrição operacional, implícita ou explícita, foi incluída.
1	Existem restrições operacionais, mas são menos restritivas que em uma aplicação típica. Nenhum esforço especial é necessário para satisfazer as restrições.
2	Existem restrições operacionais, mas são as típicas de qualquer aplicação. É necessário esforço especial para satisfazer as restrições, através de controladores ou programas de controle.
3	As restrições operacionais estabelecidas requerem limites especiais em uma parte da aplicação no processador central ou um processador dedicado.
4	As restrições operacionais estabelecidas requerem limites especiais na aplicação inteira no processador central ou um processador dedicado.
5	Adicionalmente, existem limites especiais na aplicação em componentes distribuídos do sistema.

Dicas

- As CGSs 3, 4 e 5 estão de certa forma relacionadas.
- Para esta CGS pense em termos de “Quanto a infra-estrutura influencia o projeto (*design*)?”.

Exemplos

Exemplos de restrições operacionais podem incluir o seguinte (lista não exaustiva):

- Esta questão indica que a aplicação deve rodar em um computador subdimensionado e que não consegue tratar adequadamente as funcionalidades novas ou alteradas e que os desenvolvedores podem, de alguma forma, superar isto desenvolvendo a aplicação de outro modo.
- Mais de uma aplicação acessando os mesmos dados pode criar restrições operacionais.
- Uma aplicação competindo pelos mesmos recursos e tecnologia, com a possibilidade de *deadlocks*, deve ser ajustada e limitada para evitar degradação de performance.

Normalmente

- A maioria das aplicações recebe nota 2.
- Cliente-servidor, web, real-time, telecomunicações ou sistemas de controle de processos recebem de 3 a 5, mas você precisaria de um processador dedicado, ou de múltiplos processadores processando as mesmas transações e buscando os recursos mais eficientes de processamento.

5. Volume de Transações

Definição A característica Volume de Transações descreve o nível segundo o qual a taxa de transações do negócio influencia o desenvolvimento da aplicação.

O volume de transações é alto e influencia o projeto, desenvolvimento, instalação e suporte da aplicação. Os usuários podem precisar do que eles consideram como tempo de resposta normal mesmo durante as horas de pico de volume.

Pontos

Atribua	Descrições para Determinar o Nível de Influência
0	Não é antecipado nenhum período de pico de transações.
1	Os baixos volumes de transações têm efeito mínimo nas fases de projeto, desenvolvimento e instalação.
2	O volume médio de transações tem algum efeito sobre as fases de projeto, desenvolvimento e instalação.
3	O alto volume de transações afeta as fases de projeto, desenvolvimento e instalação.
4	O alto volume de transações declarado pelo usuário nos requisitos técnicos da aplicação ou no acordo de nível de serviço é suficientemente alto para requerer tarefas de análise de performance nas fases de projeto, desenvolvimento e/ou instalação.
5	O alto volume de transações declarado pelo usuário nos requisitos técnicos da aplicação ou no acordo de nível de serviço é suficientemente alto para requerer tarefas de análise de performance e, adicionalmente, utilização de ferramentas de análise de performance nas fases de projeto, desenvolvimento e/ou instalação.

Dicas

- As CGSs 3, 4 e 5 estão de certa forma relacionadas. Para esta CGS pense em termos de “Quantas transações podem ser processadas pela aplicação em um determinado período de tempo?”
- Muitas vezes esta nota é a mesma para a CGS 3, porque o volume de transação frequentemente influencia os requisitos de performance.

Normalmente

- Aplicações batch recebem de 0 a 3.
- Aplicações on-line (incluindo interações de Cliente-servidor ou Web) recebem de 0 a 4.
- Sistemas real-time, de telecomunicações ou controle de processos recebem de 0 a 5.
- Uma nota 5 requer a utilização de ferramentas de análise de performance.

6. Entrada de Dados On-Line

Definição A característica Entrada de Dados On-line descreve os níveis segundo os quais os dados são informados ou recuperados através das transações interativas.

Interfaces on-line com o usuário para entrada de dados, funções de controle, relatórios e consultas são fornecidos pela aplicação.

Pontos

Atribua	Descrições para Determinar o Nível de Influência
0	Todas as transações são processadas de modo batch
1	1% a 7% das transações são interativas
2	8% a 15% das transações são interativas
3	16% a 23% das transações são interativas
4	24% a 30% das transações são interativas
5	Mais de 30% das transações são interativas

Dicas

- Aqui fazemos referência aos tipos de transações e *não* aos volumes.
- Por exemplo, se uma aplicação tem 45 EEs, SEs e CEs, qual o percentual das EEs, SEs e CEs é executado via transações on-line.

Normalmente

- Aplicações batch recebem 0 ou 1.
- Aplicações on-line, real-time, de telecomunicações ou sistemas de controle de processos recebem 5.
- A maioria das aplicações on-line atuais (incluindo cliente-servidor interativo ou web) recebem 5.
- Sistemas batch com características on-line podem ter a maioria das transações batch, mas o sistema deve ser pelo menos 71 % batch para receber menos do que 5.

7. Eficiência do Usuário Final

Definição

A característica Eficiência do Usuário Final descreve o nível segundo o qual foram considerados os fatores humanos e a facilidade de uso para o usuário na aplicação medida.

As funções on-line fornecidas enfatizam um projeto (“design”) para maior eficiência (fatores humanos/amigabilidade ao usuário). O projeto inclui:

- Auxílio à navegação (ex.: teclas de função, saltos, menus gerados dinamicamente, hiper-links)
- Menus
- Ajuda on-line e documentação
- Movimentação automática do cursor
- Paginação
- Impressão remota (através de transações on-line)
- Teclas de função pré-definidas (ex.: limpeza de tela, solicitação de ajuda, cópia de tela)
- Tarefas batch executadas a partir de transações on-line
- Combos (caixas de combinação)
- Uso intenso de vídeo reverso, brilho, cores, sublinhado e outros indicadores
- Documentação impressa das transações on-line (ex.: print screen)
- Interface de mouse
- Janelas pop-up
- Templates e/ou defaults
- Suporte bilíngüe (Suporte a 2 idiomas: conte como 4 itens)
- Suporte Multi-idíomas (Suporte a mais de 2 idiomas: conte como 6 itens)

Pontos

Atribua	Descrições para Determinar o Nível de Influência
0	Nenhum dos itens acima
1	1 a 3 dos itens acima
2	4 a 5 dos itens acima.
3	6 ou mais dos itens acima, mas não existem requisitos específicos do usuário relacionados à eficiência.
4	6 ou mais dos itens acima, e os requisitos estabelecidos pelo usuário quanto a eficiência são suficientemente fortes para requerer o projeto de tarefas que incluam fatores humanos.
5	6 ou mais dos itens acima, e os requisitos estabelecidos pelo usuário quanto à eficiência são suficientemente fortes para requerer o uso de ferramentas e processos especiais para demonstrar que os objetivos foram alcançados.

Dicas

- Utilize uma convenção de atribuir a nota 4 sempre que a aplicação for implementada em ambiente GUI (a não ser que ela receba 5).
- Normalmente somente ambientes de software que preparam aplicações para “mass-markets” ou usuários não-técnicos recebem 5, e apenas se existirem especialistas em ergonomia e/ou estudos de usabilidade como parte do processo.

Normalmente

- Aplicações puramente batch recebem 0.
- Interface com o usuário em modo character recebe 1 ou talvez 2.
- Interface GUI para ser usada com baixo volume de transações recebe 3.
- Interface GUI para ser usada com alto volume de transações, assim como a maioria das interfaces de Intranet recebem 4 (devem existir tarefas de “design” referentes a fatores humanos).
- Interface com o usuário de Intranet recebe 5 (requer o uso de ferramentas e processos especiais para demonstrar que os objetivos foram alcançados).

8. Atualização On-Line

Definição A característica Atualização On-line descreve os níveis segundo os quais os arquivos lógicos internos são atualizados on-line.

A aplicação fornece atualização on-line dos arquivos lógicos internos.

Pontos

Atribua	Descrições para Determinar o Nível de Influência
0	Nenhuma.
1	A atualização on-line de 1 a 3 arquivos de controle está incluída. O volume de atualizações é pequeno e a recuperação é fácil.
2	A atualização on-line de 4 ou mais arquivos de controle está incluída. O volume de atualizações é pequeno e a recuperação é fácil.
3	A atualização on-line da maioria dos arquivos lógicos internos está incluída.
4	Adicionalmente, a proteção contra perda de dados é essencial e foi especialmente projetada e programada no sistema.
5	Adicionalmente, elevados volumes fazem considerar os custos do processo de recuperação. Procedimentos de recuperação altamente automatizados com um mínimo de intervenção do operador estão incluídos.

Dicas

- A atualização on-line normalmente requer um arquivo chaveado ou banco de dados.
- A recuperação automática fornecida pelo sistema operacional conta se impactar a aplicação.

Normalmente

- As aplicações puramente batch recebem 0
- Atualizações on-line de arquivos que modificam a forma segundo a qual a aplicação processa ou valida dados recebidos recebem 1 ou 2.
- A atualização on-line dos dados persistentes do usuário recebe 3.
- Aplicações MIS (Sistema de Informação Gerencial) recebem 3 ou menos.
- A maioria das aplicações GUI (Interface Gráfica do Usuário) recebem 3 ou mais.
- Aplicações que utilizam recuperação programada como por exemplo “SQL roll back” ou “commit” recebem 4. Backup operacional ou rotineiro não é considerado proteção contra perda de dados.
- Aplicações que executam recuperação de dados, reinicialização ou outras funções autocontidas em caso de erro do sistema recebem 5. A recuperação pode requerer um operador para pressionar “enter” ou executar outra função mínima para iniciar o processo.

9. Processamento Complexo

Definição

A característica Processamento Complexo descreve os níveis segundo os quais a lógica de processamento influenciou o desenvolvimento da aplicação. Os seguintes componentes estão presentes:

- Controle sensível e/ou processamento específico de segurança da aplicação.
- Processamento lógico extensivo.
- Processamento matemático extensivo.
- Muito processamento de exceção, resultando em transações incompletas que devem ser processadas novamente.
- Processamento complexo para manipular múltiplas possibilidades de entrada e saída.

Pontos

Atribua	Descrições para Determinar o Nível de Influência
0	Nenhum dos itens acima
1	Qualquer 1 dos itens acima
2	Quaisquer 2 dos itens acima
3	Quaisquer 3 dos itens acima
4	Quaisquer 4 dos itens acima
5	Todos os 5 itens acima

Dicas

- O controle sensível ou processo de segurança (ex.: usuários individuais teriam diferentes autorizações para acesso em telas onde pudessem ver e/ou alterar dados) pode incluir processamento especial de auditoria (dados de auditoria seriam capturados sempre que dados fossem visualizados e/ou alterados e reportados).
- O processamento específico de segurança da aplicação pode incluir processamento de segurança desenvolvido internamente ou utilizar pacotes de segurança comprados.
- **Processamento Lógico Extensivo** é lógica Booleana (utiliza AND, OR) de dificuldade maior que a média, ou um mínimo de 4 comandos condicionais aninhados (IF, CASE). O processamento lógico extensivo não ocorre na maioria das aplicações MIS (Sistema de Informação Gerencial).
- **Processamento Matemático Extensivo** é a aritmética que está além da capacidade de uma calculadora de 4 operações (soma, subtração, multiplicação, divisão). Isto normalmente não está presente na maioria das aplicações MIS. Todavia, uma aplicação de engenharia pode se qualificar.

Dicas

- O processamento de exceção inclui transações de ATM (caixa automática) incompletas, causadas por interrupção de TP, perda de valores de dados, falhas de validações ou verificação por redundância cíclica (cycle redundancy checks), que podem ser usados para recriar as partes dos dados que foram perdidas.
- As múltiplas possibilidades de entrada/saída incluem multimídia, dispositivos independentes, voz, leitura ótica de caracteres (OCR), leitura de código de barras, leitura da retina e bafômetro.

Normalmente

- Esta pontuação não é dependente de plataforma.

10. Reusabilidade

Definição A característica Reusabilidade descreve os níveis segundo os quais a aplicação e o código da aplicação foram especificamente projetados, desenvolvidos e suportados para serem utilizados em outras aplicações.

Pontos	Atribua	Descrições para Determinar o Nível de Influência
	0	Não há código reutilizável.
	1	É utilizado código reutilizável dentro da aplicação.
	2	Menos de 10% do código desenvolvido da aplicação foi planejado para utilização em mais de uma aplicação.
	3	10% do código desenvolvido da aplicação foi planejado para utilização em mais de uma aplicação da aplicação.
	4	A aplicação foi especificamente empacotada e/ou documentada para fácil reutilização, e está customizada ao nível do código fonte.
	5	A aplicação foi especificamente empacotada e/ou documentada para fácil reutilização, e está customizada para uso através da manutenção dos parâmetros pelo usuário.

Dicas	Dicas para Regra 1	<ul style="list-style-type: none"> • Uma nota 1 é concedida para a reutilização do código, independentemente de onde ele foi desenvolvido. • Código desenvolvido especificamente para reutilização dentro da aplicação e utilizado mais de uma vez dentro da aplicação conta tanto quanto código recuperado de uma biblioteca central e disponível para uso geral.
	Dicas para Regra 2	<ul style="list-style-type: none"> • Para receber 2 ou mais, o código deve ter sido desenvolvido para uso em mais de uma aplicação, armazenada e gerenciada em uma biblioteca central e disponível para uso geral. O código de uma aplicação que é “copiado e colado” em outra aplicação não é considerado reutilização. • O código reutilizado deve estar apoiado por documentação que possibilite e facilite a reutilização.
	Dicas para Regra 5	<ul style="list-style-type: none"> • Exemplos de aplicações customizadas através do uso de parâmetros incluem PeopleSoft e SAP e geralmente receberão 5. • O código reutilizável pode ser modificado <i>levemente</i> na aplicação receptora. • Exemplos de reutilização incluem objetos ou outros códigos estáticos mantidos em uma biblioteca de código/objeto.

Normalmente • Esta pontuação não é dependente de plataforma.

11. Facilidade de Instalação

Definição A característica Facilidade de Instalação descreve os níveis segundo os quais a conversão de ambientes anteriores influenciou o desenvolvimento da aplicação.

A facilidade de instalação e de conversão são características da aplicação. Um plano de conversão e instalação e/ou ferramentas de conversão foram fornecidos e testados durante a fase de teste do sistema.

Pontos

Atribua	Descrições para Determinar o Nível de Influência
0	Nenhuma consideração especial foi estabelecida pelo usuário e nenhum “Setup” especial foi requerido para instalação.
1	Nenhuma consideração especial foi estabelecida pelo usuário, mas um “Setup” especial foi requerido para instalação.
2	Requisitos de conversão e instalação foram estabelecidos pelo usuário, e guias de conversão e instalação foram fornecidos e testados. O impacto da conversão no projeto não é considerado importante.
3	Requisitos de conversão e instalação foram estabelecidos pelo usuário, e guias de conversão e instalação foram fornecidos e testados. O impacto da conversão no projeto é considerado importante
4	Adicionalmente ao item 2, ferramentas automáticas de instalação e conversão foram fornecidas e testadas.
5	Adicionalmente ao item 3, ferramentas automáticas de instalação e conversão foram fornecidas e testadas.

Dicas

- A conversão e instalação incluem a conversão de dados pré-existentes para novos arquivos de dados, carga de arquivos com dados reais ou o desenvolvimento de um software especial de conversão, como no caso da tradução de uma versão para outra.
- Deve ser utilizado software comprado ou desenvolvido para que a aplicação receba pontos referentes à instalação e conversão.

Dica para a Regra 1 • A maioria das aplicações de negócio requerem algum “Setup” especial para instalação da aplicação e recebem 1.

Dica para a Regra 2 • Se a aplicação tem requisitos de conversão e instalação e guias de instalação foram fornecidos, e o fornecimento destas funções e guias não estavam no caminho crítico do projeto, atribua 2.

Dica para a Regra 3

- Se a aplicação tem requisitos de conversão e instalação e guias de instalação foram fornecidos, e o fornecimento destas funções e guias estavam no caminho crítico do projeto, atribua 3.

Dica para as Regras 4 e 5

- Se a aplicação tem requisitos de conversão e instalação e pode ser instalada sem intervenção externa, atribua 4 ou 5, dependendo dos outros requisitos para a pontuação 2 e 3.

Normalmente

- Esta pontuação não é dependente de plataforma.

12. Facilidade de Operação

Definição A característica Facilidade de Operação descreve os níveis segundo os quais a aplicação atende aos aspectos operacionais, tais como os processos de inicialização, backup e recuperação.

A facilidade de operação é uma característica da aplicação. A aplicação minimiza a necessidade de atividades manuais, tais como montagem de fitas, manuseio de papel e intervenção manual direta do operador.

Pontos

Atribua	Descrições para Determinar o Nível de Influência
0	Nenhuma consideração operacional especial, além dos procedimentos normais de backup foi estabelecida pelo usuário.
1 - 4	Um, alguns ou todos os itens seguintes aplicam-se à aplicação. Selecione aqueles que se aplicam. Cada item vale um ponto, exceto quando houver indicação em contrário: <ul style="list-style-type: none"> • Processos de inicialização, de backup e de recuperação foram fornecidos, mas a intervenção humana é necessária. • Processos de inicialização, de backup e de recuperação foram fornecidos, e a intervenção humana não é necessária (conte 2 itens) • A aplicação minimiza a necessidade de montagem de fitas e/ou acesso a dados remotos requerendo intervenção humana. • A aplicação minimiza a necessidade de manuseio de papéis
5	A aplicação é projetada para operação não assistida. Isto é, nenhuma intervenção humana é necessária para operar o sistema, que não seja a inicialização e término da aplicação. A recuperação automática de erros é uma característica da aplicação.

Dicas

- | | |
|-------------------------------|---|
| Dica para a Regra 1-4a | <ul style="list-style-type: none"> • A aplicação tem a habilidade de executar inicialização, backup e recuperação; porém, a resposta humana é requerida para iniciar a função. |
| Dica para a Regra 1-4b | <ul style="list-style-type: none"> • A aplicação tem a habilidade de executar inicialização, backup e recuperação; e nenhuma resposta humana é requerida para iniciar a função. |
| Dica para a Regra 1-4c | <ul style="list-style-type: none"> • A aplicação minimiza a necessidade de acesso a dados que não estejam imediatamente disponíveis. • Isto pode incluir a importação de dados de um processador distribuído para o processador local antes da execução, a fim de eliminar demoras no acesso. |

Dica para a Regra 1-4d

- A aplicação foi projetada para suprir o usuário com dados em um formato condensado, ou através de um outro meio diferente de papel.
- Isto pode incluir a eliminação de informações impressas detalhadas, ou acesso a relatórios on-line, consultas, microfichas, CD ou outra mídia semelhante.

Dica para a Regra 5

- A nota 5 é atribuída a uma aplicação que executa e se recupera automaticamente dos erros, ela mesma - uma operação não assistida.
- Operação não-assistida pode incluir satélite sem tripulação, reator nuclear ou controle de tráfego aéreo.

Normalmente

- Esta pontuação não é dependente de plataforma.

13. Múltiplos Locais

Definição A característica Múltiplos Locais descreve os níveis segundo os quais a aplicação foi desenvolvida para diferentes ambientes de hardware e software.

Pontos

Atribua	Descrições para Determinar o Nível de Influência
0	As necessidades de <i>apenas um</i> local de instalação foram consideradas no projeto.
1	As necessidades de mais de um local de instalação foram consideradas no projeto e a aplicação está projetada para operar apenas em ambientes de hardware e software <i>idênticos</i> .
2	As necessidades de mais de um local de instalação foram consideradas no projeto e a aplicação está projetada para operar apenas em ambientes de hardware e software <i>similares</i> .
3	As necessidades de mais de um local de instalação foram consideradas no projeto e a aplicação está projetada para operar em ambientes de hardware e software <i>diferentes</i> .
4	A documentação e o plano de suporte foram fornecidos e testados para suportar a instalação da aplicação em múltiplos locais e a aplicação é descrita pelo item 2.
5	A documentação e o plano de suporte foram fornecidos e testados para suportar a instalação da aplicação em múltiplos locais e a aplicação é descrita pelo item 3.

Dicas

O termo múltiplos locais é um termo lógico e não necessariamente físico. Podem existir múltiplos locais dentro do mesmo local físico. A determinação do fator está baseada nas necessidades das diversas instalações.

Dicas para Regra 0

- A maioria das aplicações mainframe provavelmente recebem 0.
- Porém, se uma aplicação está instalada em vários computadores mainframe com múltiplas configurações significativamente diferentes ou diferentes sistemas operacionais, ela receberia uma nota maior que 0.

Dicas para Regra 1

- Por exemplo, Windows NT em hardware com exatamente a mesma configuração.

Dicas para Regra 2

- Por exemplo, Windows 95, 98 e NT em hardware com uma configuração similar.
- As variações podem incluir diferentes tamanhos de memória, várias capacidades de armazenamento, diferentes velocidades de processadores e diferentes tipos de impressoras.

**Dicas para
Regra 3**

- Por exemplo, Windows, OS X, UNIX, Linux e VOS3 em diferentes tipos de hardware.
- As diferenças podem incluir PC baseado em Intel MAC, Tandem, Sun e AS400.

Normalmente A pontuação depende do número de plataformas diferentes.

14. Facilidade de Mudança

Definição

A característica Facilidade de Mudança descreve os níveis segundo os quais a aplicação foi desenvolvida para fácil modificação da lógica de processamento ou estrutura de dados.

As seguintes características podem ser aplicáveis à aplicação:

A: Consulta Flexível:

1. Consultas e/ou relatórios flexíveis são fornecidos, permitindo a manipulação de pedidos *simples*. (conte como 1 item)
2. Consultas e/ou relatórios flexíveis são fornecidos, permitindo a manipulação de pedidos de complexidade *média*. (conte como 2 itens)
3. Consultas e/ou relatórios flexíveis são fornecidos, permitindo a manipulação de pedidos *complexos*. (conte como 3 itens)

B: Dados de controle do negócio:

1. Dados de controle do negócio são guardados em tabelas mantidas pelo usuário através de processos on-line interativos, mas as alterações só têm efeito no próximo dia útil. (conte como 1 item)
2. Dados de controle do negócio são guardados em tabelas mantidas pelo usuário através de processos on-line interativos, e as alterações têm efeito imediato. (conte como 2 itens)

Pontos

Atribua	Descrições para Determinar o Nível de Influência
0	Nenhum dos itens acima
1	Qualquer 1 dos itens acima
2	Quaisquer 2 dos itens acima
3	Quaisquer 3 dos itens acima
4	Quaisquer 4 dos itens acima
5	Quaisquer 5 dos itens acima

Dicas

Relatórios e Consultas Flexíveis:

- A facilidade de consulta e relatório flexível significa mais do que uma lista de seleções em uma consulta ou relatório “enlatado”.
- É a capacidade do usuário de controlar os dados, fonte de dados, sequência e formato das suas consultas ou relatórios solicitados.

Dicas

- Significa a liberdade (autonomia) de projetar o layout da tela, classificação horizontal e vertical, apresentação dos itens de dados, seleção dos critérios tanto dos arquivos quanto dos itens de dados.
- Inclui programação do usuário para consultas, às vezes denominada consulta ou relatório "ad-hoc".
- A utilização de filtros que controlam a quantidade de dados vistos ou impressos em um formato fixo não considerada consulta/relatório flexível.
- A capacidade de escrever uma consulta e/ou relatório é muitas vezes fornecida por linguagens como SQL ou Focus, ou por algumas das ferramentas mais dinâmicas para a geração de relatórios "ad-hoc" (ex. Crystal Reports) .

Dica para a Regra A1

- Pedidos simples podem incluir lógica "e/ou" aplicada a **apenas um** arquivo lógico interno.

Dica para a Regra A2

- Pedidos de complexidade média podem incluir lógica "e/ou" aplicada a **mais de um** arquivo lógico interno.

Dica para a Regra A3

- Pedidos complexos podem incluir combinações de lógica "e/ou" em um ou mais arquivos lógicos internos.

Dados de Controle do Negócio:

- Dados de Controle do Negócio (Dados Referenciados) são armazenados para suportar as regras de negócio para a manutenção dos Dados do Negócio; ex.: em uma aplicação de folha de pagamento, seriam as alíquotas governamentais para cada faixa salarial e a data em que a alíquota entrou em vigor.
- Veja a Parte 2, Dados de Códigos para informações adicionais.

Normalmente A pontuação não depende de plataforma

Tabela de Cálculo do Fator de Ajuste

A tabela a seguir é fornecida para facilitar o cálculo do Fator de Ajuste.

Características Gerais do Sistema (CGSs)	Grau de Influência (DI) 0 - 5
1. Comunicação de Dados	_____
2. Processamento Distribuído	_____
3. Performance	_____
4. Configuração Intensamente Utilizada	_____
5. Volume de Transações	_____
6. Entrada de Dados On-Line	_____
7. Eficiência do Usuário Final	_____
8. Atualização On-Line	_____
9. Processamento Complexo	_____
10. Reusabilidade	_____
11. Facilidade de Instalação	_____
12. Facilidade de Operação	_____
13. Múltiplos Locais	_____
14. Facilidade de Mudança	_____
Total do nível de influência (TDI)	_____
Fator de Ajuste (VAF)	_____
$VAF = (TDI * 0.01) + 0.65$	

Pontos de Função Ajustados de Projeto de Desenvolvimento

Funcionalidades da Aplicação

Funcionalidades da aplicação consistem em funções usadas depois da instalação do software para satisfazer as necessidades correntes de negócio do usuário.

Funcionalidades de Conversão

Funcionalidades de conversão consistem em funções fornecidas apenas na instalação para converter dados e/ou atender outros requisitos de conversão especificados pelo usuário, tais como relatórios especiais de conversão.

Por exemplo, se uma aplicação de Recursos Humanos (RH) estava em uso e uma nova aplicação é instalada, os usuários desejam que as informações dos funcionários sejam convertidas e carregadas dentro da nova aplicação. O requisito de conversão especificado pelo usuário é para transferir os dados atuais dos funcionários para o novo sistema de RH.

Fator de Ajuste da Aplicação

O fator de ajuste é determinado utilizando-se as 14 características gerais do sistema para avaliar a complexidade funcional da aplicação.

Fórmula: Tamanho Funcional Ajustado do Projeto de Desenvolvimento (aDFP)

Utilize a seguinte fórmula para calcular o tamanho funcional ajustado do Projeto de Desenvolvimento.

$$aDFP = DFP * VAF$$

Onde:

aDFP é o tamanho funcional ajustado do Projeto de Desenvolvimento

DFP é o tamanho funcional do Projeto de Desenvolvimento (DFP = ADD + CFP; Veja Parte 1)

VAF é o Fator de Ajuste

Nota: Depois da instalação do software, o tamanho funcional da Aplicação é calculado utilizando-se os componentes do tamanho funcional do Projeto de Desenvolvimento.

Exemplo: Tamanho Funcional Ajustado do Projeto de Desenvolvimento (aDFP)

Esta seção mostra um exemplo da contagem para um projeto de desenvolvimento. O projeto inclui tanto funcionalidades da aplicação como de conversão.

Nota: a seguinte contagem é destinada a ser apenas um exemplo e não está especificamente relacionada com outras seções deste manual. Todas as funcionalidades relacionadas podem não estar incluídas (isto é, pode haver funcionalidades faltantes).

Funcionalidade da Aplicação

As seguintes tabelas mostram as funcionalidades da aplicação medidas para um projeto de desenvolvimento.

Funções de Dados	RLRs	DERs	Complexidade Funcional
Arquivos Lógicos Internos			
• Informações da Tarefa	2	5	Baixa
• Tarefas Suspensas	2	6	Baixa
• Definições de Relatórios	1	4	Baixa
• Informações dos Funcionários	1	6	Baixa
Arquivos de Interface Externa			
• Informações de Locais	1	6	Baixa
• Informações de Conversão	1	2	Baixa
• Informações de Ajuda de Telas	1	2	Baixa
• Informações de Ajuda de Campos	1	5	Baixa

Funções de Transação	ALRs	DERs	Complexidade Funcional
Entradas Externas			
Definição do Relatório de Alocação	1	5	Baixa
Inclusão das informações da Tarefa (tela entrada)	1	7	Baixa
Inclusão das informações da Tarefa (entrada batch)	2	6	Média
Correção de Tarefas Suspensas	1	7	Baixa
Atribuição da Tarefa ao Funcionário	3	7	Alta
EE com tela de saída – 1	2	11	Média
EE com tela de saída – 2	1	6	Baixa
Saídas Externas			
Relatório das Tarefas com Funcionários	4	5	Média
Relatório dos Funcionários por Duração da Tarefa	3	7	Média
Notificação de Avaliação da Performance	3	4	Baixa
Relatório Semanal de Funcionários	1	3	Baixa
Impressão do Cheque	1	3	Baixa
Arquivo de Transação do Cheque	1	4	Baixa
Consultas Externas			
Lista de Dados Recuperados	1	4	Baixa
Unidade de Negociação	1	2	Baixa
Ajuda Nível Campo	1	6	Baixa
Relatório Semanal de filiação	1	3	Baixa
Arquivo Diário de Cheques	1	2	Baixa

Funcionalidade de Conversão

A seguinte tabela mostra as funcionalidades de conversão para o projeto de desenvolvimento.

Função de Transação	ALRs	DERs	Complexidade Funcional
Entrada Externa			
Migração do Funcionário	1	11	Baixa

Contribuição da Aplicação para o Tamanho Funcional

A seguinte tabela mostra a contribuição ao tamanho funcional não ajustado.

Tipo de Função	Complexidade Funcional			Totais por tipo de Complexidade	Totais por Tipo de Função
ALIs	4	Baixa	X 7 =	28	28
	0	Média	X 10 =	0	
	0	Alta	X 15 =	0	
AIEs	4	Baixa	X 5 =	20	20
	0	Média	X 7 =	0	
	0	Alta	X 10 =	0	
EEs	4	Baixa	X 3 =	12	26
	2	Média	X 4 =	8	
	1	Alta	X 6 =	6	
SÉS	4	Baixa	X 4 =	16	26
	2	Média	X 5 =	10	
	0	Alta	X 7 =	0	
CEs	5	Baixa	X 3 =	15	15
	0	Média	X 4 =	0	
	0	Alta	X 6 =	0	
					Incluídos 115

Contribuição da Conversão para o Tamanho Funcional

A seguinte tabela mostra a contribuição das funcionalidades de conversão para o tamanho funcional.

Tipo de Função	Complexidade Funcional			Total Complexidade	Totais por Tipo de Função
EEs	1	Baixa	X 3 =	3	CFP 3
	0	Média	X 4 =	0	
	0	Alta	X 6 =	0	

Cálculo Final

Utilizando as contagens de complexidade e contribuição para este exemplo, o tamanho ajustado do Projeto de Desenvolvimento é demonstrado abaixo. O Fator de Ajuste para este exemplo é 1.05.

$$aDFP = (ADD + CFP) * VAF$$

$$aDFP = (115 + 3) * 1.05$$

$$aDFP = 123.9 \text{ ou } 124$$

Tamanho Funcional Ajustado do Projeto de Melhoria (aEFP)

O tamanho funcional ajustado do Projeto de Melhoria consiste de três componentes:

- Funcionalidades da Aplicação incluídas nos requisitos do usuário para o projeto
- Funcionalidades de Conversão incluídas nos requisitos do usuário para o projeto
- Fator de Ajuste da Aplicação

Considerações sobre a CGS para o Projeto de Melhoria

As 14 Características Gerais do Sistema (CGSs) opcionais, definidas neste apêndice, devem ser revistas para alteração. Pequenas melhorias normalmente não exigem tal revisão. Exemplos de mudanças que podem indicar uma necessidade de revisão das CGSs incluem:

- Adição de funções on-line em uma aplicação batch;
- Aumento do volume de transações e/ou redução de tempo de resposta que agora necessitem de projeto de performance e atividades de teste;
- Novas características de usabilidade solicitadas;
- Adição de uma interface Web em uma aplicação on-line existente;
- Adição de um novo protocolo de comunicação em uma aplicação existente.

Funcionalidade da Aplicação

As funcionalidades da aplicação consistem em:

- Pontos de função identificados a partir das funcionalidades que são incluídas pela melhoria
- Pontos de função contados devido a funcionalidades existentes que são alteradas durante o projeto de melhoria
- Pontos de função contados para as funcionalidades excluídas durante o projeto de melhoria

Funcionalidades de Conversão

As funcionalidades de conversão consistem em pontos de função fornecidos por qualquer funcionalidade de conversão solicitada pelo usuário.

Fator de Ajuste

Os dois fatores de ajuste são:

- Fator de ajuste da aplicação antes do projeto de melhoria iniciar
- Fator de ajuste da aplicação depois do projeto de melhoria estar concluído

Formula: Tamanho Funcional Ajustado do Projeto de Melhoria (aEFP)

Utilize a seguinte formula para calcular o tamanho funcional ajustado do Projeto de Melhoria.

Nota: Requisitos de conversão de dados *estão incluídos* nesta contagem.

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

Onde:

aEFP É o tamanho funcional ajustado do Projeto de Melhoria

ADD É o tamanho das funções sendo adicionadas pelo Projeto de Melhoria

CHGA É o tamanho das funções sendo alteradas pelo Projeto de Melhoria – tal como elas são/serão *depois* da implementação

CFP É o tamanho da Funcionalidade de Conversão

VAFA É o Fator de Ajuste da Aplicação *depois* do Projeto de Melhoria estar concluído

DEL É o tamanho das funções sendo excluídas pelo Projeto de Melhoria

VAFB É o Fator de Ajuste da Aplicação *antes* do Projeto de Melhoria iniciar

Nota: Quando um projeto de melhoria é instalado, o tamanho funcional da Aplicação deve ser atualizado para refletir as alterações nas funcionalidades da aplicação.

Exemplo: Tamanho Funcional Ajustado do Projeto de Melhoria (aEFP)

Esta seção mostra um exemplo para um projeto de melhoria. Os requisitos para o projeto de melhoria incluem as seguintes mudanças:

- O usuário não precisa mais incluir uma tarefa on-line, portanto, esta funcionalidade é para ser ou foi excluída.
- O usuário precisa receber um relatório adicional sobre tarefas que foram incluídas no total.
- DERs adicionais são requeridos para incluir tarefas em modo batch e para corrigir transações suspensas. Uma referência à segurança também é incluída na transação de inclusão de tarefa.

Funcionalidade da Aplicação

Os seguintes parágrafos explicam as funcionalidades da aplicação contadas para o exemplo do projeto de melhoria. Cada funcionalidade é descrita como incluída, alterada ou excluída.

Funcionalidades Incluídas

A seguinte tabela mostra a complexidade funcional da funcionalidade incluída contada quando o projeto estiver concluído.

Nota: O fornecimento de um novo relatório foi uma saída externa adicional.

Funções de Transação	ALRs	DERs	Complexidade Funcional
Saída Externa			
Relatório de Tarefas	1	15	Baixa

Funcionalidades Alteradas

A seguinte tabela mostra a complexidade funcional das funcionalidades alteradas, depois que o projeto de melhoria estiver concluído.

Nota: a complexidade de incluir uma tarefa foi aumentada porque um tipo de arquivo adicional foi referenciado. A complexidade de correção de transações suspensas continua baixa.

Depois

Funções de Transação	ALRs	DERs	Complexidade Funcional
Entrada Externa			
Inclusão de informações de tarefa (modo batch)	3	8	Alta
Correção de transações suspensas	1	8	Baixa

Antes

Funções de Transação	ALRs	DERs	Complexidade Funcional
Entrada Externa			
Inclusão de informações de tarefa (modo batch)	2	6	Média
Correção de transações suspensas	1	7	Baixa

Funcionalidades Excluídas

A seguinte tabela mostra a complexidade funcional da funcionalidade excluída identificada no final do projeto.

Funções de Transação	ALRs	DERs	Complexidade Funcional
Entradas Externas			
Inclusão de informação de tarefa (tela entrada)	1	7	Baixa

Contribuição da Aplicação para o Tamanho Funcional

Os seguintes parágrafos explicam a contribuição das funcionalidades da aplicação para o tamanho funcional total não ajustado.

Funcionalidades incluídas

A seguinte tabela mostra a contribuição para o tamanho funcional das funcionalidades incluídas, identificadas ao final do projeto.

Tipo de Função	Complexidade Funcional			Totais por tipo de Complexidade	Totais por Tipo de Função
SEs	1	Baixa	X 4 =	4	
	0	Média	X 5 =	0	
	0	Alta	X 7 =	0	
					ADD 4

Funcionalidades alteradas

A seguinte tabela mostra a contribuição das funcionalidades alteradas para o tamanho funcional das funcionalidades alteradas tal como existirão depois do Projeto de Melhoria estar concluído.

Depois

Tipo de Função	Complexidade Funcional			Totais por tipo de Complexidade	Totais por Tipo de Função
EEs	1	Baixa	X 3 =	3	
	0	Média	X 4 =	0	
	1	Alta	X 6 =	6	
					CHGA 9

Antes

Tipo de Função	Complexidade Funcional			Totais por tipo de Complexidade	Totais por Tipo de Função
EEs	1	Baixa	X 3 =	3	
	1	Média	X 4 =	4	
	0	Alta	X 6 =	0	
					CHGB 7

Funcionalidades excluídas

A seguinte tabela mostra a contribuição ao tamanho funcional das funcionalidades excluídas.

Tipo de Função	Complexidade Funcional			Totais por tipo de Complexidade	Totais por Tipo de Função
EEs	1	Baixa	X 3 =	3	
	0	Média	X 4 =	0	
	0	Alta	X 6 =	0	
					DEL 3

Cálculo Final

O fator de ajuste da aplicação era 1,05 antes do início do projeto. O fator de ajuste continua o mesmo depois do término do projeto.

Utilizando a contagem de complexidade e contribuição para este exemplo, o tamanho funcional ajustado do Projeto de Melhoria está demonstrado abaixo:

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

$$aEFP = [(4 + 9 + 0) * 1.05] + (3 * 1.05)$$

$$aEFP = 16.8 \text{ ou } 17$$

Tamanho Funcional Ajustado da Aplicação (aAFP)

Esta seção fornece fórmulas para calcular o tamanho funcional ajustado da Aplicação. Existem duas variações desta fórmula:

- Fórmula para determinar o tamanho funcional inicial ajustado para uma Aplicação
- Fórmula para recalcular o tamanho funcional ajustado para uma Aplicação *depois* de um Projeto de Melhoria ter alterado as Funcionalidades da Aplicação

Fórmula: Tamanho Funcional Inicial Ajustado da Aplicação (aAFP)

Utilize a fórmula desta seção para determinar o tamanho funcional inicial ajustado para uma Aplicação. Inicialmente, o usuário está recebendo novas funcionalidades. Não existem alterações nas funcionalidades existentes ou exclusões de funcionalidades obsoletas ou desnecessárias. O tamanho funcional ajustado da Aplicação *não inclui* requisitos de conversão.

$$\text{aAFP} = \text{ADD} * \text{VAF}$$

Onde:

aAFP É o tamanho funcional inicial ajustado da Aplicação

ADD É o tamanho das funções do Projeto de Desenvolvimento a serem entregues ao usuário ou a Funcionalidade que existe em qualquer momento que a Aplicação seja medida.

VAF É o Fator de Ajuste da Aplicação.

Fórmula: Tamanho Funcional Ajustado da Aplicação Depois do Projeto de Melhoria (aAFPA)

Quando um projeto de melhoria é instalado, o tamanho funcional da Aplicação existente deve ser atualizado para refletir as modificações na Aplicação. As funcionalidades da aplicação podem ser alteradas de várias maneiras:

- Funcionalidades incluídas (novas) aumentam o tamanho da aplicação
- Funcionalidades alteradas aumentam, diminuem ou não afetam o tamanho da aplicação
- Funcionalidades excluídas diminuem o tamanho da aplicação
- Mudanças no fator de ajuste aumentam ou diminuem o tamanho da aplicação.

Nota: Como as funcionalidades de conversão não afetam o tamanho funcional ajustado da Aplicação, qualquer funcionalidade de conversão associada a um projeto de melhoria está inteiramente excluída do cálculo do tamanho funcional ajustado da Aplicação.

Utilize a seguinte fórmula para calcular o tamanho funcional ajustado da Aplicação após o projeto de melhoria:

$$aAFPA = [(AFPB + ADD + CHGA) - (CHGB + DEL)] * VAFA$$

Onde:

aAFPA É o tamanho funcional ajustado da Aplicação *depois* do Projeto de Melhoria

AFPB É o tamanho funcional da Aplicação *antes* do Projeto de Melhoria começar.

ADD É o tamanho das funções sendo incluídas pelo Projeto de Melhoria.

CHGA É o tamanho das funções sendo alteradas pelo Projeto de Melhoria – tal como elas são / serão após a implementação

CHGB É o tamanho das funções sendo alteradas pelo Projeto de Melhoria – tal como elas são / eram antes do início do projeto

DEL É o tamanho das funções sendo excluídas pelo Projeto de Melhoria.

VAFA É o fator de ajuste da aplicação depois que o projeto de melhoria estiver concluído.

Nota: Se este valor estiver indisponível, pode ser calculado utilizando-se a fórmula $AFPB = aAFPB / VAFB$; Onde $aAFPB$ é o tamanho funcional ajustado da Aplicação *antes* do Projeto de Melhoria e $VAFB$ é o Fator de Ajuste da Aplicação *antes* do Projeto de Melhoria.

Exemplo: Contagem de Aplicação

Esta seção mostra um exemplo de tamanho funcional inicial ajustado e o tamanho funcional ajustado que reflete um Projeto de Melhoria. Os números para estas medições de tamanho funcional provem do exemplo de Aplicação, anterior neste apêndice.

Tamanho Funcional Inicial Ajustado da Aplicação (aAFP)

O tamanho funcional inicial ajustado da Aplicação é demonstrado abaixo. O Fator de Ajuste é 1.05.

$$aAFP = ADD * VAF$$

$$aAFP = 115 * 1.05$$

$$aAFP = 120.75 \text{ ou } 121$$

Nota: Apenas o tamanho das funcionalidades da aplicação instaladas para o usuário são incluídas no tamanho inicial.

Tamanho Funcional Ajustado da Aplicação Depois (aAFPA)

O tamanho funcional ajustado da Aplicação para refletir o projeto de Melhorias é demonstrado abaixo. O Fator de Ajuste é 1.05.

$$aAFPA = [(AFPB + ADD + CHGA) - (CHGB + DEL)] * VAFA$$

$$aAFPA = [(115 + 4 + 9) - (7 + 3)] * 1.05$$

$$aAFPA = 123.9 \text{ ou } 124$$

Página intencionalmente deixada em branco.

Glossário IFPUG

Este é um glossário de termos utilizados nas publicações do IFPUG.

Albrecht 1984. Documento original do conceito de ponto de função, escrito por Allan J. Albrecht em Novembro de 1984. Também conhecido como “313” devido ao seu número de documento.

Análise de Pontos de Função. É o método de medição de tamanho funcional, tal como definido no método FSM (Medição de Tamanho Funcional) do IFPUG.

Aplicação. Uma coleção coesa de procedimentos automatizados e dados suportando um objetivo de negócio. Ela consiste em um ou mais componentes, módulos ou subsistemas.

Área de Aplicação. Um termo geral para um agrupamento de aplicações que manipulam uma área específica de negócio. Ela corresponde a um nível administrativo para propósitos gerenciais

Arquivo. Para funções de dados, um grupo de dados logicamente relacionados, não a implementação física destes grupos de dados.

Arquivo de Interface Externa (AIE). Um arquivo de interface externa (AIE) é um grupo de dados logicamente relacionados ou informações de controle, reconhecido pelo usuário, referenciado pela aplicação sendo contada, mas mantido dentro da fronteira de 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 medida. Isto significa que um AIE contado para uma aplicação deve estar em um ALI em outra aplicação.

Arquivo Lógico. Veja Função de Dados.

Arquivo Lógico Interno (ALI). Um arquivo lógico interno (ALI) é um grupo de dados logicamente relacionados ou informações de controle, reconhecido pelo usuário, mantido dentro da

fronteira da aplicação sendo medida. A intenção primária de um ALI é armazenar dados mantidos através de um ou mais processos elementares da aplicação que está sendo medida.

Arquivo Lógico Referenciado (ALR). Ver Tipo de Arquivo Referenciado.

Arranjo. A atividade de seqüenciar atributos em uma função de transação.

Asset. (1) Um bem capital de uma empresa. (2) Uma vantagem ou recurso.

Atributo. Veja atributo de Projeto / Aplicação e atributo de dado.

Atributo de dado. Uma característica de uma entidade. Atributos de dados são geralmente análogos aos Dados Elementares Referenciados (Tipos de Dados Elementares) (DERs). Também conhecido como campo.

Atributo de Projeto/Aplicação. Características de um projeto ou de uma aplicação que podem ter um impacto significativo na produtividade. Exemplos incluem: plataforma de hardware, experiência do pessoal, ferramentas e metodologia. O atributo do projeto/aplicação é utilizado para categorizar dados do projeto durante uma análise.

Atributo Técnico. Atributo não funcional que é um resultado de considerações do projeto ou da implementação.

Atualização On-Line CGS. Uma das 14 características gerais do sistema que descreve os níveis segundo os quais os arquivos lógicos internos são atualizados on-line.

Auto-contido. Nenhum processamento anterior ou subsequente é necessário para iniciar ou completar o(s) requisitos(s) funcional(is) do usuário.

Campo. Veja atributo de dados.

Características Gerais do Sistema (CGSs). As *características gerais do sistema* são um conjunto de 14 questões que avaliam a complexidade global da aplicação.

Carga. Copiar instruções ou dados do computador de um armazenamento externo para um interno. (IEEE)

Chave Estrangeira. Dados em um ALI ou AIE que existem porque o usuário requer um relacionamento com outro ALI ou AIE.

Classificação. A atividade de sequenciar linhas ou registros em uma função de transação.

Comitê de Práticas de Contagem (CPC). O comitê de trabalho que mantém o Manual de Práticas de Contagem do IFPUG.

Complexidade Funcional. Taxa de complexidade específica associada a uma função utilizando as regras definidas neste padrão internacional.

Componente Funcional Básico. Unidade elementar de Requisitos Funcionais do Usuário definido e utilizado pelo método FSM para propósitos de medição (ISO/IEC 14143-1:2007). ALIs, AIEs, EEs, SEs, CEs são os tipos de CFBs (BFCs).

Comunicação de Dados CGS. Uma das 14 características gerais do sistema que descreve o nível em que a aplicação se comunica diretamente com o processador.

Configuração Intensamente Utilizada CGS. Uma das 14 características gerais do sistema que descreve o nível em que as restrições de recursos computacionais influenciam no desenvolvimento da aplicação.

Consulta Externa (CE). 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 CE é apresentar informações ao usuário através da recuperação de dados ou informações de controle de um ALI ou AIE. O processamento lógico não contém fórmula matemática ou cálculos, não cria dados derivados. Nenhum ALI é mantido durante o processamento, nem o comportamento do sistema é alterado.

Contagem de Pontos de Função. A atividade de aplicar as regras do método de Medição de Tamanho Funcional (FSM) do IFPUG para

medir o tamanho funcional de uma aplicação ou projeto. NOTA: Existem três tipos de contagem de Pontos de Função: Aplicação, Projeto de Desenvolvimento e Projeto de Melhoria.

Contagem de Pontos de Função de Baseline. Veja Contagem de Pontos de Função da Aplicação.

Contagem de Pontos de Função da Aplicação. A atividade de aplicar as regras do método de Medição de Tamanho Funcional (FSM) do IFPUG para medir o tamanho funcional de uma aplicação.

Contagem de Pontos de Função de Projeto de Desenvolvimento (DFP). A atividade de aplicar as regras do método de Medição de Tamanho Funcional (FSM) do IFPUG para medir o tamanho funcional de um projeto de desenvolvimento.

Contagem de Pontos de Função de Projeto de Melhoria (EFP). A atividade de aplicar as regras do método de Medição de Tamanho Funcional (FSM) do IFPUG para medir o tamanho funcional de um projeto de melhoria.

Contagem de Pontos de Função Instalados. Veja Contagem de Pontos de Função da Aplicação.

Contagem de Pontos de Função Não-Ajustada (UFP). Veja contagem de Pontos de Função.

Contribuição. Contribuição dos tipos de função (ALI, AIE, EE, SE, CE) para o tamanho funcional.

Cópia. (1) Ler dados de uma origem, deixando os dados originais sem alterações, e escrever os mesmos dados em outro lugar em uma forma física que pode ser diferente da forma do dado original. Por exemplo, copiar dados de um disco magnético em uma fita magnética. (2) O resultado de um processo de cópia como o descrito acima. Por exemplo, uma cópia de um arquivo de dados. (IEEE)

Dado Elementar Referenciado (DER). Ver Tipo de Dado Elementar.

Dado derivado. Dado criado como resultado de processamento que envolve etapas, , no lugar ou além da recuperação direta e validação da informação em funções da dados.

Dados de Código. O usuário nem sempre especifica diretamente os Dados de Código, algumas vezes referenciados como Lista de Dados ou Dados de Tradução. Em outros casos, ele é identificado

pelo desenvolvedor em resposta a um ou mais requisitos técnicos do usuário. Dados de Código fornecem uma lista de valores válidos que um atributo descritivo pode ter. Tipicamente, os atributos de Dados de Código são Código, Descrição e/ou outros atributos "padrão" que descrevem o código; ex.: abreviação padrão, data efetiva, data de término, dados de trilhas de auditoria, etc.

Dados de Negócio. Pode ser também referenciado como os dados principais do usuário ou objetos de negócio. Este tipo de dado reflete a informação necessária para ser armazenada e recuperada pela área funcional destinada pela aplicação. Dados de Negócio normalmente representam um percentual significativo das entidades identificadas.

Dados de Referência. Este tipo de dado é armazenado para suportar as regras de negócio para a manutenção dos Dados de Negócio; ex. em uma aplicação de folha de pagamento estes seriam os dados armazenados nas taxas governamentais para cada escala salarial e data efetiva da taxa. Dados de referência freqüentemente representam um pequeno percentual das entidades identificadas.

Defeito. Um problema que, se não for corrigido, pode fazer com que uma aplicação falhe ou produza resultados incorretos. A ausência de funcionalidade que foi especificada ou requisitada também é considerada um defeito.

Despesa de capital. Uma forma de gasto no qual uma empresa negocia com dinheiro (capital) para aquisição de objetos tangíveis tais como mobiliário, computadores e assim por diante.

Eficácia. Produção do resultado pretendido ou desejado.

Eficiência. Produção de um resultado com um mínimo de esforço extra ou redundante.

Eficiência do Usuário Final CGS. Uma das 14 características gerais do sistema que descreve o nível de considerações do fator humano e da facilidade de uso pelo usuário da aplicação que está sendo medida.

Entidade (ou tipo de entidade). Uma coisa fundamental relevante para o usuário, sobre a qual uma coleção de fatos é mantida. Uma associação entre entidades que contém atributos próprios é também uma entidade.

Entidade de Dados. Veja Entidade.

Entidade Dependente. Uma entidade que não é significativa ou que não tem significado para o negócio por si mesma sem a presença de outras entidades, tais como:

- Uma ocorrência da entidade X deve estar ligada a uma ocorrência da entidade Y.
- A exclusão de uma ocorrência da entidade Y resulta na exclusão de todas as ocorrências relacionadas da entidade X.

Entidade Independente. Uma entidade que é significativa e importante por si mesma sem a presença de outras entidades.

Entidade Subtipo. Uma subdivisão de um tipo de entidade. Um subtipo herda todos os atributos e relacionamentos de seus tipos de entidade pai, e pode ter atributos adicionais e relacionamentos próprios.

Entrada de Dados On-Line CGS. Uma das 14 características gerais do sistema que descreve o nível em que os dados entram através de transações interativas.

Entrada Externa (EE). Uma entrada externa (EE) é um processo elementar que processa dados ou informações de controle que vem 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.

Escopo da Contagem. O escopo da contagem define o conjunto de requisitos funcionais do usuário a ser incluído na contagem de pontos de função.

Esforço de Trabalho. Recursos de trabalho requeridos para a produção de uma saída específica. Aqui se refere ao esforço necessário. Recursos de trabalho são usualmente expressos em horas de trabalho.

Estado consistente. O ponto no qual o processamento foi completamente executado, o requisito funcional do usuário foi satisfeito e não há nada mais a ser feito.

Facilidade de Instalação CGS. Uma das 14 características gerais do sistema que descreve o nível em que a conversão de ambientes anteriores influencia no desenvolvimento da aplicação.

Facilidade de Mudança CGS. Uma das 14 características gerais do sistema que descreve o nível em que a aplicação tem sido desenvolvida

para facilitar a modificação da lógica de processamento ou estrutura de dados.

Facilidade de Operação CGS. Uma das 14 características gerais do sistema que descreve o nível em que a aplicação atende aos aspectos operacionais, como os processos de start-up, de backup e de recuperação.

Fator de Ajuste (VAF). O fator que indica a funcionalidade geral fornecida ao usuário da aplicação. O VAF é calculado baseado na avaliação das 14 características gerais do sistema (CGSs) para uma aplicação.

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

Fronteira da Aplicação. Veja Fronteira

Função. Capacidades ou particularidades de uma aplicação, conforme observadas pelo usuário.

Função de Dados. Uma função de dados representa a funcionalidade fornecida ao usuário para satisfazer seus requisitos de armazenamento de dados internos e externos. Uma função de dados é um arquivo lógico interno (ALI) ou um arquivo de interface externa (AIE).

Função de Transação. É um processo elementar que provê funcionalidade para o usuário processar dados. Uma função de transação é uma entrada externa, saída externa ou consulta externa

Funcionalidade. Veja Função.

Funcionalidade de Conversão. Funções de dados ou de transação providas para converter dados e/ou fornecer outros requisitos de conversão especificados pelo usuário.

Gerente de Aplicação. Pessoa responsável por gerenciar projetos e fornecer suporte às atividades de uma ou mais áreas da aplicação.

Gerente de Projetos. Uma pessoa que gerencia um ou mais projetos ou grupo de projetos.

Identificável pelo usuário. O termo *identificável pelo usuário* se refere aos requisitos definidos para processos e/ou dados que estão acordados e entendidos tanto pelo(s) usuário(s) quanto pelo(s) desenvolvedor(es) de software.

IBM CIS & A Guideline 313. Veja Albrecht 1984.

IFPUG. O "International Function Point Users Group" é uma organização governada por membros, sem fins lucrativos, com o compromisso de promover e fornecer suporte a

análise de pontos de função e outras técnicas de medição de software.

Imagem. Uma réplica exata de um outro objeto, arquivo, ou tabela que é geralmente criada através de um utilitário.

Informação de Controle. Informação de Controle são dados que influenciam um processo elementar especificando o que, quando ou como os dados serão processados.

Intenção primaria. Intenção que é a primeira em importância.

Investimento de Capital. Uma forma de despesa em que uma empresa negocia capital para aquisição de objetos tangíveis tais como mobília, computadores e outros similares.

Líder de Projeto. Uma pessoa que gerencia ou lidera projetos. Pode ser um sinônimo para Gerente de Projeto.

Lógica de Processamento. Qualquer requisito especificamente solicitado pelo usuário para completar um processo elementar, tais como validações, algoritmos ou cálculos, e leitura ou manutenção de uma função de dados.

Mantido. O termo *mantido* refere-se a habilidade de incluir, alterar ou excluir dados através de um processo elementar.

Manutenção. O esforço para manter uma aplicação executando de acordo com suas especificações, geralmente sem mudanças nas suas funcionalidades (ou tamanho funcional). Manutenção inclui reparos, pequenas melhorias, conversão, suporte ao usuário e atividades de manutenção preventiva. Atividades incluem remoção de defeitos (veja Reparo), upgrades de hardware ou software (veja conversão), otimização ou melhoria da qualidade (veja manutenção preventiva), e suporte ao usuário.

Manutenção adaptativa. A modificação de um produto de software, executada depois da entrega, para manter o produto de software utilizável em um ambiente alterado ou em vias de alteração. Manutenção adaptativa fornece as melhorias necessárias para acomodar mudanças no ambiente no qual um produto de software deve operar. Estas mudanças são as que devem ser feitas para manter-se em dia com o ambiente alterado. Por exemplo, o sistema operacional deve sofrer upgrade e algumas mudanças devem

ser feitas para acomodar o novo sistema operacional. (ISO/IEC 14764:2006).

Manutenção corretiva. Modificação reativa de um produto de software executada depois da entrega para corrigir problemas identificados. A modificação corrige os produtos de software para satisfazer os requisitos. (ISO/IEC 14764:2006)

Manutenção Perfectiva. Modificação de um produto de software depois da entrega para detectar e corrigir falhas latentes no produto de software antes que ele manifeste estas falhas. Manutenção perfectiva fornece melhorias para usuários, melhorias de documentação de programas e recodificação para melhorar a performance do software, manutenibilidade e outros atributos do software. Contrasta com : Manutenção adaptativa e manutenção corretiva. (ISO/IEC 14764:2006)

Manutenção Preventiva. Mudanças no hardware ou software executadas para prevenir futuros defeitos ou falhas. Por exemplo, reestruturação dos programas ou dados para melhorar a manutenibilidade ou prevenir defeitos.

Maturidade Software Engineering Institute (SEI). A habilidade de uma organização realizar um processo medido e controlado como uma fundação para a melhoria contínua (Humphrey). O nível de experiência de uma organização ou projeto com uma ferramenta, um recurso, uma técnica ou uma metodologia específica.

Medição. Associação de um valor relativo. Normalmente, em um processo de melhoria, medidas resultantes desta atividade são combinadas para formarem métricas.

Medidas de Processo. Informação capturada sobre o processo de desenvolvimento.

Medidas de Produto. Informação capturada sobre a aplicação de software desenvolvida.

Mídia. Um canal de comunicação ou informação, por exemplo, um relatório em papel ou microficha.

Medida. Como um substantivo, um número que associa um valor relativo. Alguns exemplos podem incluir volume, altura, pontos de função ou esforço de trabalho. Como um verbo, para averiguar ou avaliar pela comparação a um padrão.

Merge. Múltiplos arquivos com os mesmos dados elementares consolidados em um único arquivo. (IEEE)

Métrica. Não existe uma única definição universal de uma métrica. No contexto deste documento, uma métrica é uma combinação de duas ou mais medidas ou atributos. Exemplos incluem (1) densidade de defeitos (defeitos por ponto de função) e (2) taxa de entrega (pontos de função por hora).

Múltiplos Locais CGS. Uma das 14 características gerais do sistema que descreve o nível em que a aplicação tem sido desenvolvida para múltiplos ambientes de hardware e software.

NESMA. Netherlands Software Metrics Association (www.nesma.org). Uma organização governada por membros, sem fins lucrativos na Holanda, comprometida para promover e suportar a análise de pontos de função e outros métodos de medição de software.

Nível de Influência (DI). Um indicador numérico da quantidade do impacto de cada uma das 14 características gerais do sistema, variando de 0 a 5. Estes indicadores são usados para calcular o fator de ajuste.

Nível de Área de Aplicação. O nível de gestão responsável pelo gerenciamento de atividades de manutenção bem como novos desenvolvimentos ou principais projetos de melhoria para uma ou mais aplicações.

Nível de Projeto. O nível gerencial responsável por gerenciar individualmente projetos de novos desenvolvimentos ou os principais projetos de melhoria.

Nível de Organização. Os níveis gerenciais responsáveis pelo gerenciamento de um ou mais de um sistema organizacional de processamento de dados ou sistemas de informação.

Nível Executivo Corporativo. O nível gerencial responsável pela empresa, incluindo o Quadro de Diretores.

Normalização. Processo pelo qual qualquer estrutura de dados pode ser transformada por um projetista de banco de dados em um conjunto de relações normalizadas que não possuem grupos repetidos.

Orçamento. Uma seqüência planejada de custos ao longo do tempo com os custos monetários designados para as tarefas ou funções específicas. Frequentemente é também utilizado para referir-se ao esforço de trabalho ao invés de dinheiro.

Performance CGS. Uma das 14 características gerais do sistema que descreve o nível em que o tempo de resposta e as considerações de performance de throughput influenciam o desenvolvimento da aplicação.

Ponto de Função (PF). Unidade de medida de tamanho funcional tal como definido no método de Medição de Tamanho Funcional (FSM) do IFPUG.

Primeira Forma Normal. Resultado de um processo de normalização que transforma grupos de dados para que eles tenham um identificador único, um ou mais atributos, e nenhum atributo repetido.

Processamento Complexo CGS. Uma das 14 características gerais do sistema que descreve o nível em que a lógica de processamento influencia no desenvolvimento da aplicação.

Processamento Distribuído CGS. Uma das 14 características gerais do sistema que descreve o nível em que a aplicação transfere dados entre os componentes físicos da aplicação.

Processo Elementar. Um *processo elementar* é a menor unidade de atividade que é significativa para os usuários.

Produtividade. Razão entre o produto de trabalho e o esforço de trabalho. Veja também Taxa de Entrega.

Produto de trabalho. O produto que é criado pelo trabalho do sistema de informação, o resultado de um esforço de desenvolvimento de software.

Projeto. Uma coleção de tarefas de trabalho com um prazo e um produto de trabalho a ser entregue.

Projeto de desenvolvimento. Um projeto de desenvolvimento é um projeto para desenvolver e entregar a primeira versão de uma aplicação de software.

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

Propósito da Contagem. A razão para executar a contagem de pontos de função.

Qualidade. Qualidade inclui: conformidade às expectativas do usuário, conformidade aos requisitos do usuário, satisfação do cliente, confiabilidade, nível de defeitos apresentados. O contexto e a política decidirão a melhor definição para uma determinada situação.

Reconhecido pelo usuário. O termo reconhecido pelo usuário refere-se aos requisitos para processos e/ou dados que estão acordados e entendidos tanto pelo(s) usuário(s) quanto pelo(s) desenvolvedor(es) de software.

RECUP. (RMCSP) Acrônimo para Reparo/Melhoria/Conversão/Suporte ao usuário / Prevenção(Repair/Enhancement/Conversion/User support/Prevention). Veja também taxa de manutenção (suporte).

Refresh. O processo de recriação de um grupo de dados para torná-lo atualizado com sua fonte.

Registro. Um grupo de itens relacionados que são tratados como uma unidade.

Registro Lógico Referenciado (RLR) – Ver Tipo de Registro Elementar.

Relacionamento. Uma associação de interesse entre duas entidades. Um relacionamento que não tem atributos e não é contado como RLR na contagem de pontos de função.

Release. Versão entregue de uma aplicação que pode incluir toda ou parte de uma aplicação.

Remoção de Defeitos. Veja Reparo.

Reparo. A correção de defeitos que tem resultado de erros em projeto externo, projeto interno ou código. Exemplos são funções faltantes que não resultam na falha da aplicação (erro de projeto externo) ou erros resultante de uma situação de parada de execução da aplicação (erro de código).

Requisitos Funcionais do Usuário. Um subconjunto de requisitos do usuário especificando o que o software deve fazer, em termos de tarefas e serviços (ISO 14143-1:2007)

Requisitos de Qualidade. Qualquer requisito relativo à qualidade de software conforme definido na ISO 9126:1991.

Requisitos Técnicos. Requisitos que estão relacionados com a tecnologia e o ambiente para o desenvolvimento, manutenção, suporte e execução do software.

Reusabilidade CGS. Uma das 14 características gerais do sistema que descreve o nível em que a aplicação e o código da aplicação tem sido especificamente projetado, desenvolvido e suportado para ser utilizável em outras aplicações.

Saída Externa (SE). 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 ao de uma consulta externa. A intenção primária de uma SE é apresentar informações para o usuário através de um processamento lógico adicional ou substituto à recuperação de dados ou de informações de controle. O processamento lógico deve conter pelo menos uma fórmula matemática ou um cálculo, criar dados derivados, manter um ou mais ALIs, e/ou alterar o comportamento do sistema.

Scoope Creep/gallop. Funcionalidade adicional que não foi especificada nos requisitos originais, mas é identificada como o escopo que está sendo esclarecido e as funções definidas.

Segunda forma normal. Resultado de um processo de normalização que transforma grupos de dados para que cada atributo não chave dependa dos atributos chave do grupo de dados e de todas as partes dos atributos chave.

Significativo. Reconhecido pelo usuário e satisfaz um requisito funcional do usuário.

Sistema. Veja Aplicação.

Sistema de Arquivo. Composto de registros e atributos de dados.

Subgrupo Obrigatório. Um dos dois tipos de subgrupos para os tipos de registros elementares (RLRs). Subgrupos obrigatórios significam que o usuário deve usar um dos subgrupos durante o processo elementar que cria uma instância do dado.

Subgrupo Opcional. *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 ou o dado.

Subtipos. Veja Entidade Subtipo.

Suporte. Veja manutenção.

Tamanho funcional ajustado da aplicação (aAFP). Veja apêndice C para a fórmula.

Tamanho funcional ajustado da aplicação depois de projetos de melhoria. (aAFPA). Veja apêndice C para a fórmula.

Tamanho funcional ajustado de projeto de desenvolvimento (aDFP). Veja apêndice C para a fórmula.

Tamanho funcional ajustado de projeto de melhoria (aEFP). Veja apêndice C para a fórmula.

Tamanho funcional ajustado. O tamanho funcional ajustado é o resultado do tamanho funcional não ajustado multiplicado pelo fator de ajuste. O tamanho funcional ajustado é calculado utilizando-se uma fórmula específica para projeto de desenvolvimento, projeto de melhoria e aplicação, o que está ilustrado no apêndice C.

Tamanho funcional da aplicação. Tamanho funcional da aplicação é a medida da funcionalidade que uma aplicação fornece ao usuário, determinada pela contagem de pontos de função da aplicação através da atividade de aplicação do método de Medição de Tamanho Funcional (FSM) do IFPUG. Veja Parte 1, Seção 5.8 para a fórmula.

Tamanho funcional de projeto de desenvolvimento. Tamanho funcional de projeto de desenvolvimento é a medida da funcionalidade fornecida ao usuário com a primeira versão do software medido pela contagem de pontos de função de projeto de desenvolvimento através da atividade de aplicação do método de Medição de Tamanho Funcional (FSM) do IFPUG. Veja Parte 1, Seção 5.8 para a fórmula.

Tamanho funcional de projeto de melhoria. Tamanho funcional de projeto de melhoria é a medida da funcionalidade incluída, alterada ou excluída na conclusão de um projeto de melhoria medida pela contagem de pontos de função de projeto de melhoria, através da atividade de aplicação do método de Medição de Tamanho Funcional (FSM) do IFPUG. Veja Parte 1, Seção 5.8 para a fórmula.

Tamanho funcional não ajustado. Veja tamanho funcional.

Tamanho Funcional. Tamanho do software resultante da quantificação dos requisitos funcionais do usuário. (ISO 14143-1:2007)

Taxa. No contexto deste documento, taxa é definida como o resultado da divisão de uma quantidade medida por uma outra.

Taxa de Entrega. A produtividade medida para a criação ou melhoria de uma aplicação. Ela é expressa pelos Pontos de Função do Projeto dividido pelo Esforço de Trabalho do projeto de desenvolvimento ou de melhoria.

Taxa de Manutenção (ou suporte). A produtividade medida para manter uma aplicação. Ela é expressa como o Esforço de Trabalho pela categoria de manutenção dividida por 1000 Pontos de Função da aplicação em um período de tempo.

Tendência. Uma análise ao longo do tempo mostrando ocorrências repetidas de uma medida ou métrica específica.

Terceira forma normal. Resultado de um processo de normalização que transforma grupos de dados para que cada atributo não-chave não dependa de qualquer outro atributo não-chave.

Tipo de Arquivo Referenciado (Arquivo Lógico Referenciado ALR). Um tipo de arquivo referenciado é uma função de dados lida e/ou mantida por uma função de transação.

Tipo de Dado Elementar (Dado Elementar Referenciado DER). Um *tipo de dado elementar* é um atributo único, reconhecido pelo usuário e não repetido.

Tipo de Entidade Associativa. Um tipo de entidade que contém atributos e que descreve um relacionamento muitos-para-muitos entre dois outros tipos de entidades.

Tipo de entidade atributiva. Um tipo de entidade que descreve um ou mais características de um outro tipo de entidade.

Tipo de Função. Os cinco componentes funcionais básicos identificados no método de Medição de Tamanho Funcional (FSM) do IFPUG. NOTA: Os cinco tipos de função são entrada externa, saída externa, consulta externa, arquivo lógico interno, e arquivo de interface externa. Também conhecidos como componentes funcionais básicos.

Tipo de Registro Elementar (Registro Lógico Referenciado RLR). Um *tipo de registro elementar* (RLR) é um subgrupo de dados elementares reconhecidos pelo usuário dentro de uma função de dados.

Total do Nível de Influência (TDI). Somatório dos níveis de influência das 14 CGSs.

Transação. Veja função de transação.

Visão do Usuário. Visão do usuário são os requisitos funcionais do usuário tal como são compreendidos pelo usuário.

Volume de Transações CGS. Uma das 14 características gerais do sistema que descreve o nível em que o volume das transações de negócio influencia o desenvolvimento da aplicação.

Usuário. Qualquer pessoa ou coisa que se comunica ou interage com o software em qualquer momento.

313. Veja Albrecht 1984.

Pagina intencionalmente deixada em branco.