



## **PROCESSO DE ANÁLISE E PROJETO DE BANCO DE DADOS**

### Sumário

<b>1. ASSUNTO/OBJETIVO .....</b>	<b>2</b>
<b>2. FINALIDADE E ÂMBITO DA APLICAÇÃO .....</b>	<b>2</b>
<b>3. UNIDADE GESTORA.....</b>	<b>2</b>
<b>4. PÚBLICO ALVO .....</b>	<b>2</b>
<b>5. RELAÇÃO COM OUTROS NORMATIVOS .....</b>	<b>2</b>
<b>6. REGULAMENTAÇÃO UTILIZADA .....</b>	<b>2</b>
<b>7. DEFINIÇÕES E CONCEITOS BÁSICOS .....</b>	<b>2</b>
<b>8. FLUXOS, FORMULÁRIOS E ORIENTAÇÕES TÉCNICAS .....</b>	<b>4</b>
<b>9. COMPETÊNCIAS E RESPONSABILIDADES .....</b>	<b>4</b>
<b>10. PROCEDIMENTOS .....</b>	<b>6</b>
<b>11. RELATÓRIOS GERENCIAIS E INDICADORES.....</b>	<b>13</b>
<b>12. CONSIDERAÇÕES FINAIS.....</b>	<b>13</b>



## PROCESSO DE ANÁLISE E PROJETO DE BANCO DE DADOS

### 1. ASSUNTO/OBJETIVO

Definir o processo de análise e projeto de banco de dados que apoia a criação e manutenção de modelos lógico e físico dos dados mantidos pelos sistemas desenvolvidos pela Secretaria de Informática do Tribunal de Justiça do Estado do Pará, bem como a consistente manipulação destes dados de acordo com as regras de negócio por meio de procedimentos armazenados na base de dados.

### 2. FINALIDADE E ÂMBITO DA APLICAÇÃO

Um processo de processo de análise e projeto de banco de dados promove a definição de quais atividades devem ser executadas ao longo de um projeto e padroniza as nomenclaturas.

Este processo deve ser seguido pela Secretaria de Informática e por todas as outras unidades que se relacionam com esta como solicitante.

### 3. UNIDADE GESTORA

Coordenadoria de Aplicações – Equipe de Administração de Dados.

### 4. PÚBLICO ALVO

Secretaria de Informática.

### 5. RELAÇÃO COM OUTROS NORMATIVOS

Não se aplica.

### 6. REGULAMENTAÇÃO UTILIZADA

Não se aplica.

### 7. DEFINIÇÕES E CONCEITOS BÁSICOS

**Agregação** – Abstração por meio da qual relacionamentos são tratados como entidades de nível mais alto.

**Atributo** – propriedade da entidade; possui um domínio, ou conjunto de valores.



**Atributo multivalorado** – assume valores múltiplos, por exemplo números de telefone.

**Auto-relacionamento** – atributo que contém um vínculo lógico (chave estrangeira) à chave primária da própria relação (exemplo: na entidade funcionário o código do gerente é um auto-relacionamento que cria uma relação entre um funcionário e outro funcionário que assume o papel de gerente).

**Entidade** - é um objeto que existe e é distinguível de outros objetos; são representadas por atributos; são identificadas por atributos identificadores (chave).

**Entidade Forte** – possui chave primária.

**Entidade Fraca** – não tem atributos suficientes para formar uma chave primária; depende de uma entidade identificador (forte) ou proprietário; depende da existência de um relacionamento identificador; o discriminador ou chave parcial só faz sentido em conjunto com a chave primária do identificador.

**Generalização/Especialização** – O processo de definir hierarquias dentro de um conjunto de entidades que facilitam a reutilização de atributos comuns; pode ser identificada do nível mais acima para os níveis inferiores (especialização) por propriedades adicionais (conjuntos de entidades de nível inferior com recursos distintos que se aplicam apenas dentro de um determinado conjunto de entidades de nível inferior); pode ser identificada nos níveis inferiores para o nível superior (generalização) por propriedades comuns (conjunto de entidades de nível superior com atributos e relacionamentos que se aplica a todos os seus conjuntos de entidades de nível inferior); um subtipo pode participar de relacionamentos que não se aplicam as entidades dos níveis superiores.

**Participação Total** – quando todas as instâncias de uma entidade precisam estar associadas a alguma instância de outra entidade por meio de um relacionamento.

**Relacionamento** – é uma associação entre várias entidades.

**Restrição de Integridade de Domínio** – Os valores das linhas para determinada coluna só podem assumir valores dentro do conjunto do domínio (exemplo integer (pré-definidos); 'Sim' ou 'Não' (definidos pelo usuário do SGBD).

**Restrição de chave estrangeira (*foreign key constraint*)** – uma coluna ou uma combinação de colunas que promove o vínculo lógico entre relações;



aponta para o valor de uma chave primária em outra relação ou na mesma relação (auto-relacionamento).

**Restrição de chave primária (*primary key constraint*)** – escolhida para identificar a entidade; uma coluna ou uma combinação mínima de colunas cujos valores distinguem uma linha das demais dentro de uma tabela; valores dos atributos nunca ou raramente mudam.

**Restrição de Chave Única (*unique key constraint*)** – certifica que uma coluna não terá seu valor repetido em mais de uma instância da entidade.

**Restrição de Checagem (*check constraint*)** - define possíveis valores para uma coluna.

**Restrição de Integridade de Vazio (*null e not null*)** – define a obrigatoriedade do preenchimento de valores em um campo de uma tabela.

**Valor Default** – Permite colocar um valor padrão na ausência de valores em um campo de uma tabela.

## 8. FLUXOS, FORMULÁRIOS E ORIENTAÇÕES TÉCNICAS

Não se aplica.

## 9. COMPETÊNCIAS E RESPONSABILIDADES

### 9.1 COORDENADORIA DE APLICAÇÕES

9.1.1 Supervisionar a elaboração e promover a divulgação da normalização de padronização da técnica de modelagem conceitual de dados.

9.1.2 Decidir sobre impasses entre equipes de Desenvolvimento de Sistemas e de Administração de Dados envolvendo a aplicação da norma.

9.1.3 Decidir sobre impasses entre equipes de Administração de Dados e de Administração de Banco de Dados envolvendo a aplicação da norma.

9.1.4 Assegurar o cumprimento da norma.

### 9.2 EQUIPE DE ADMINISTRAÇÃO DE DADOS

9.2.1 Responsável pelo processo de Análise e Projeto de Banco de Dados.

9.2.2 Participar e homologar a análise de requisitos.

9.2.3 Selecionar sistemas de gerenciamento de banco de dados que melhor se apliquem as necessidades da Coordenadoria de Aplicações em conjunto com a equipe de Administração de Banco de Dados.



- 9.2.4 Elaborar as normas, padrões e procedimentos relativos à modelagem lógica/física de dados.
- 9.2.5 Elaborar os modelos lógico/físico dos dados de acordo com a norma.
- 9.2.6 Registrar e acompanhar as pendências de adequação de modelos à norma.
- 9.2.7 Comunicar à Coordenadoria de Aplicações a existência de impasses envolvendo a aplicação da norma.
- 9.2.8 Desenvolver rotinas armazenadas na base de dados que implementem regras de negócio ou auxiliem no tratamento e extração dos dados dos sistemas desenvolvidos pela equipe de desenvolvimento de sistemas.
- 9.2.9 Desenvolver rotinas armazenadas na base de dados para migração de dados.
- 9.2.10 Revisar consultas aos dados confeccionadas pela equipe de Desenvolvimento de Sistemas.
- 9.2.11 Elaborar relatórios não contemplados nos sistemas e extrair informações diretamente na base de dados.
- 9.2.12 Manter o modelo de dados de sistemas já implantados.
- 9.2.13 Elaborar modelos multidimensionais de *datamarts* e *datawarehouse*.
- 9.2.14 Elaborar rotinas e configurar ferramentas para extração, transformação e carga de dados (*ETL - Extract Transform Load*).
- 9.2.15 Configurar ferramentas OLAP para acesso aos dados multidimensionais.
- 9.2.16 Otimizar o modelo físico e consultas para melhor desempenho do sistema, observando sempre o plano de execução gerado pelo otimizador do SGBD de modo a evitar caminhos de acesso onerosos, tais como FULL TABLE SCAN.
- 9.2.17 Avaliar ferramentas de modelagem de dados que melhor se apliquem as necessidades da Administração de Dados observando critérios de edição diagramática, geração de *scripts*, geração de Dicionário de dados, engenharia Reversa (*Reverse Engineering*) e suporte a trabalho cooperativo.
- 9.3 **EQUIPE DE DESENVOLVIMENTO DE SISTEMAS**
  - 9.3.1 Envolver a equipe de Administração de Dados no levantamento de requisitos.
  - 9.3.2 Revisar os modelos lógico/físico dos dados elaborados pela equipe de Administração de Dados.
  - 9.3.3 Confeccionar consultas SQL.



- 9.3.4 Conhecer e executar a padronização da técnica de modelagem conceitual de dados quando necessário.
- 9.4 **EQUIPE DE ADMINISTRAÇÃO DE BANCO DE DADOS**
- 9.4.1 Implementar a infraestrutura no sistema de gerenciamento de banco de dados necessária para o suporte da criação do modelo físico dos dados criado pela equipe de administração de dados.
- 9.4.2 Conhecer e executar a padronização da técnica de modelagem conceitual de dados quando necessário.
- 9.4.3 Compilar objetos em base de dados de produção.
- 9.4.4 Avaliar sistemas de gerenciamento de banco de dados que melhor se apliquem as necessidades da Coordenadoria de Aplicações em conjunto com a equipe de Administração de Dados.

## **10. PROCEDIMENTOS**

- 10.1 O processo de análise e projeto de banco de dados do Tribunal de Justiça do Estado apresenta as seguintes etapas:
  - 10.1.1. Análise de requisitos – No que diz respeito a este processo visa identificar que dados precisam ser modelados no projeto do banco de dados para que seja possível gerar informação de apoio a decisão relevante aos atores do sistema de informação em questão.
  - 10.1.2. Seleção do SGBD - Sistema Gerenciador de Banco de Dados (eventual) – Avaliação e seleção do Sistema de Gerenciamento de Banco de Dados, cujo objetivo é avaliar as ferramentas disponíveis no mercado em conjunto com a equipe de Administração de Banco de Dados de acordo com critérios técnicos, econômicos e políticos, possibilitando a seleção da que melhor se adequa ao contexto do Tribunal de Justiça do Estado.
  - 10.1.3. Projeto Lógico dos Dados – Elaboração de modelos lógico dos dados, visando construir um modelo descritivo das entidades, atributos e relacionamentos (modelo entidade-relacionamento).
  - 10.1.4. Projeto Físico dos Dados - Elaboração de modelos físico dos dados contendo especificação de armazenamento das estruturas, alocação de dados, índices e outras estruturas físicas a serem definidos em conjunto com a equipe de Administração de Banco de Dados.
  - 10.1.5. Implementação - Consiste na implantação do modelo físico no sistema de gerenciamento de banco de dados e desenvolvimento de rotinas para implementação de regras de negócio, tratamento e extração de informações dos sistemas desenvolvidos e migração de dados. Implementação ou revisão de consultas.



10.1.6. Manutenção e Otimização (*Tuning*) – Manter o modelo de dados de sistemas já implantados e otimizar o modelo físico e consultas para melhor desempenho do sistema.

## 10.2 ANÁLISE DE REQUISITOS

10.2.1. Na etapa de Análise de Requisitos os Administradores de Dados devem participar das reuniões e ter acesso aos documentos para melhor compreensão do negócio o que permitirá uma agilidade na confecção dos modelos de dados e das rotinas armazenadas no banco de dados que implementarão as regras de negócio da aplicação. O documento de descrição de requisitos servirá de insumo as próximas etapas do processo.

## 10.3 SELEÇÃO DO SGBD (EVENTUAL)

10.3.1. Na etapa de Seleção do SGBD em conjunto com a equipe de Administração de Banco de Dados devem ser considerados os seguintes aspectos para seleção da ferramentas utilizada atualmente no Tribunal de Justiça do Estado: número de usuários; hardware dos servidores; custo de aquisição; custo de manutenção e suporte; custo de treinamento; propósito (OLTP - *Online Transaction Processing* ou OLAP - *Online Analytical Processing*); plataformas suportadas; tamanho da base de dados; disponibilidade; segurança; ferramentas de administração; desempenho e outros correlatos. Sempre que houver necessidade esta etapa pode ser executada de acordo com estes critérios.

## 10.4 PROJETO LÓGICO DOS DADOS

10.4.1. A construção dos modelos lógico e físicos dos dados deve ser desenvolvida em ferramenta escolhida pela equipe de Administração de Dados e constitui parte essencial do processo de desenvolvimento de software do Tribunal de Justiça do Estado do Pará.

10.4.2. A construção do modelos lógicos deve estar de acordo com os seguintes padrões de nomenclatura:

- Não utilizar símbolos tais como \_
- Nomes de Entidades
- Prefixo Nome do Sistema + Nome da Entidade. Exemplo: SPPROCESSO



- Nomes de Atributos
  - Chave Primária e Estrangeira: utilizar prefixo CD (Código). Exemplo: CDPROCESSO
  - Alfanuméricos: utilizar prefixo DS (Descrição) ou NM (Nome). Exemplo: NMPESOA
  - Numéricos: utilizar prefixo NO (Número) ou QT (Quantidade). Exemplo: NOBOLETO
  - Valor: utilizar prefixo VL (Valor). Exemplo: VLCAUSA
  - Data: utilizar prefixo DT (Data). Exemplo: DTDISTRIBUICAO
  - Situação, Status ou Flag: utilizar prefixo FL (flag). Exemplo: FLATIVO
  - Enumeração: utilizar prefixo EN. Exemplo: ENTIPOSESSAO
- Nomes de Restrições (constraints)
  - Chave Primária: utilizar prefixo PK (primary key) + Nome da Entidade. Exemplo: PKSPPROCESSO
  - Chave Estrangeira: utilizar prefixo FK (foreign key) + Nome da Entidade Pai + Nome da Entidade Filho. Exemplo: FKSPDOCUMENTOSPPROCESSO
  - Chave Única: utilizar prefixo UK (unique key) + Nome da Coluna + Nome da Entidade. Exemplo: UKDEINSTANCIASPINSTANCIA
  - Restrição de checagem: utilizar prefixo CK (check constraint) + Nome da Coluna + Nome da Entidade. Exemplo: CKFLATIVOSPINSTANCIA
- Nomes de Índices
  - Prefixo IDX + Sequencial numérico + Nome da Tabela. Exemplo: IDX01SPPROCESSO
- Abreviação de Nomes
  - 3 primeiras consoantes + 2 últimas letras. Exemplo: DPTTO (DEPARTAMENTO)



10.4.3. A construção do modelos lógico dos dados deve seguir os procedimentos:

- Mapear Entidades Fortes
- Criar tabela, criar chave primária, criar atributos
- Mapear Entidades Fracas
- Criar tabela, criar chave estrangeira apontando para chave primária da tabela proprietária, criar chave primária, criar atributos
- Mapear Atributos de Chave Estrangeira
- Criar chave estrangeira apontando para a chave primária da tabela que contém o atributo.
- Mapear Atributos Multivalorados
- Criar uma nova tabela com o atributo, criar chave estrangeira apontando para a chave primária da tabela que contém o atributo
- Mapear Relacionamento 1:1
- Criar a chave estrangeira em uma das tabelas do relacionamento (de preferência a de participação total)
- Mapear Relacionamento 1:N
- Criar chave estrangeira na tabela do lado N, criar os atributos do relacionamento nesta tabela
- Mapear Relacionamento N:N
- Criar uma nova tabela, criar chaves estrangeiras para as tabelas do relacionamento que irá compor a chave primária da tabela associativa, criar os atributos do relacionamento nesta tabela.
- Mapear Relacionamento N-ário
- Seguir mesma regra do relacionamento N:N
- Mapear agregações
- 1º passo: Criar uma tabela resultante de um relacionamento N:N
- Seguir mesma regra do relacionamento N:N



- 2º passo: Mapear o relacionamento entre a tabela resultante e a tabela associada a agregação
- Seguir mesma regra do relacionamento 1:1, 1:N, N:N
- Mapear Generalizações/Especializações
- Criar tabela para superclasse, criar chave primária, criar atributos, criar atributo para a categoria da entidade
- Criar cada subclasse, criar chave estrangeira apontando para a chave primária da tabela da superclasse que será a chave primária da tabela da subclasse, criar atributos específicos da subclasse.

## 10.5 PROJETO FÍSICO DOS DADOS

10.5.1. O modelo lógico de dados deverá ser estendido durante a construção do modelo físico dos dados, para tal deve-se adotar os seguintes procedimentos adicionais:

- Para cada atributo criado no modelo lógico:
  - Definir o domínio do mesmo e a restrição de integridade de vazio (null e not null)
  - Se chave estrangeira, adicionar a regra de integridade referencial. Nunca utilizar deleções em cascata.
- Criar restrições (unique key e check constraints)
- Criar índices conforme as consultas mais onerosas da aplicação. Certificar que índices de chave primária e estrangeira serão criados, ou pelo SGBD ou pela ferramenta de modelagem respectivamente.

10.5.2. O Administrador de Dados visando uma melhor modelagem e manutenção dos dados deverá adotar as seguintes boas práticas:

- Assegurar que o modelo promova a consistência dos dados, através da implementação de regras de integridade (de domínio, de vazio, de chave e referencial) e controle de acesso aos dados (encapsulamento) de modo que os mesmos sempre que possível sejam manipulados através de procedimentos armazenados no banco de dados



- Utilizar as regras de normalização para validar as estruturas conceituais do modelo de dados.
- O processo de normalização é conduzido, no mínimo, até a terceira forma normal.
- Prover a reutilização de dados e compartilhamento de informações, diminuindo redundâncias e aumentando a confiabilidade dos dados
- Disponibilizar informações atualizadas para todas as fontes, o que será possível sempre que for utilizado o compartilhamento das informações pois um dado atualizado será automaticamente visível
- Prover agilidade no acesso a dados através da otimização de consultas e procedimentos armazenados
- Adotar padrões de nomenclatura para melhor compreensão do dicionário de dados
- Desenvolver rotinas visando a redução do tempo de desenvolvimento de aplicações
- Diminuir o desperdício em atividades redundantes de processamento
- Desenvolver rotinas e procedimentos para recuperação (restore) ágil de dados tais como mecanismos de auditoria (log do sistema), bem como estimular o uso de deleção lógica dos dados nos sistemas, e não físicas
- Sempre que possível implementar restrições de estado ou dependência funcional por triggers para garantir a consistência dos dados
- Sempre que possível utilizar tipos para prover o uso de boas práticas da programação OO – Orientação a Objetos , tais como encapsulamento de dados e método construtor. Utilizar tipos também em conjunto com table functions
- Para manutenções em produção evitar desabilitar regras de integridade referencial, triggers e auditoria (log) bem como alterar definições de atributos pois dados inválidos podem ser inseridos no momento da manutenção



10.6.1. Na etapa de implementação em conjunto com a equipe de Administração de Banco de Dados serão criados os objetos no SGBD e desenvolvidas as rotinas a serem armazenadas no banco de dados. Nesta etapa também são desenvolvidos, quando necessário, rotinas para migração de dados entre sistemas.

10.6.2. A etapa de implementação deve estar de acordo com os seguintes padrões de nomenclatura:

- Nomes de Funções:
  - Prefixo FNC + Nome da Função. Exemplo: FNCALTERACLASSE
- Nomes de Procedimentos:
  - Prefixo PRC + Nome do Procedimento. Exemplo: PRCPREENCHEMODELO
- Nomes de Pacotes:
  - Prefixo PCK + Nome do Sistema + Nome do Pacote. Exemplo: PCKSPDOCUMENTO
- Nomes de Tipos:
  - Classe: Prefixo TABLEOF + Nome do Tipo. Exemplo: TABLEOFBOLETO
  - Objeto: Prefixo TP + Nome do Tipo. Exemplo: TPBOLETO
- Nomes de Triggers:
  - Trigger de regra de negócio: Nome da Entidade + \_ + Identificador do disparo (B para before; A para after) + Identificador da operação (I para insert; U para update; D para delete) + R (row). Exemplo: SPTIPOAGENTE\_BIUR
  - Trigger de log: Nome da Entidade + \_LOG. Exemplo: SPTIPOAGENTE\_LOG
- Nomes de Views:
  - Prefixo V\_ + Nome da View. Exemplo: V\_DISTRIBUICAO
- Nomes de Sequences:



- Prefixo SEQ + Nome da Entidade. Exemplo: SEQSPBOLETO

## 10.7 MANUTENÇÃO E OTIMIZAÇÃO (*TUNNING*)

10.7.1. Na etapa de manutenção e otimização sempre que necessário serão feitos ajustes nas consultas e rotinas para adaptações em regra de negócio ou para a otimização do sistema. Nesta fase poderão ser criados índices, views materializadas para otimização de consultas e jobs para processamento de dados fora do expediente.

## **11. RELATÓRIOS GERENCIAIS E INDICADORES**

Não se aplica.

## **12. CONSIDERAÇÕES FINAIS**

Esse normativo define os procedimentos empregados no Processo de análise e projeto de banco de dados. Os procedimentos são constantemente submetidos a melhorias continuadas. Demais esclarecimentos devem ser dirigidos à Equipe de Administração de Dados.