ADMINISTRAÇÃO DE DADOS **NORMA**
Processo nº: 44129.007604/2024-91 **N/AD/001/06 (Nº SEI! 0043040)**

PADRÕES PARA GESTÃO E CONSTRUÇÃO DE MODELOS DE DADOS

Nota da versão:

Versão 06 – Alteração no nome da Norma de “Padrões para construção de Modelos de Dados” para “Padrões para Gestão e Construção de Modelos de Dados”; Alteração para inclusão de tipos de dados e classes de atributos espaciais; Adequação de siglas de órgãos da empresa.

SUMÁRIO

1. OBJETIVO

Estabelecer diretrizes, padrões e nomenclaturas para a construção de modelos e bases de dados. Definir premissas e requisitos básicos a serem seguidos para processos de dados na Dataprev.

2. APLICAÇÃO

Aplica-se a todos os órgãos da Empresa.

3. ÓRGÃOS CITADOS NA NORMA

Sigla	Função principal
DEEG	Órgão responsável pela Governança e Engenharia de Dados
DIED	Órgão responsável pela Gestão de Dados

4. CONCEITUAÇÃO

4.1. Gestão de Dados

É a função na organização que cuida do planejamento, controle e entrega de ativos de dados e de informação. Esta função inclui as disciplinas do desenvolvimento, execução e supervisão de planos, políticas, programas, projetos, processos, práticas e procedimentos que controlam,

protegem, distribuem e aperfeiçoam o valor dos ativos de dados e informações. (DAMA-DMBOK ®).

A Gestão de Dados visa a utilização adequada de dados e informações para o alcance dos objetivos estratégicos da organização e pode ser entendida como uma disciplina formada pelo conjunto de onze funções de gerenciamento de dados integradas. A integração dessas funções é feita pela Governança de Dados (DAMA-DMBOK ®).

4.2. Administrador de Dados – AD

Membro da equipe de Gestão de Dados, é o responsável por estabelecer e disseminar padrões de gestão de dados, políticas, boas práticas e procedimentos para a construção e validação dos modelos de dados da empresa. Também é responsável por modelar ou apoiar a construção e/ou alteração dos modelos de dados conceituais, lógicos e físicos, assim como pela implementação e atualização das estruturas de dados nos bancos de dados em ambiente de desenvolvimento.

4.3. Administrador de Banco de Dados – DBA

É o responsável pela criação e manutenção das estruturas de dados nos bancos de dados de missão crítica como homologação e produção. Também é responsável pela criação, manutenção e monitoração dos bancos de dados, suporte aos seus usuários, bem como tomar as ações necessárias para garantir a integridade e disponibilidade destes ambientes.

4.4. Analista Especificador

Analista que especifica os casos de uso em contato com o cliente e participa da construção do modelo de dados junto ao Administrador de Dados.

4.5. Modelagem de Dados

A modelagem de dados pode ser definida como um processo em que, através do levantamento dos requisitos de informação e regras de negócio, aplicando técnicas e mecanismos de abstração, construímos artefatos (modelos de dados) que tem por objetivo representar um conjunto de informações. Estes normalmente são geridos por uma aplicação ou sistema de informação.

4.5.1. Modelo Lógico de Dados

Representa uma visão de alto nível e abstrata, das estruturas de dados, no que se refere a tecnologias de armazenamento. No Modelo Lógico de Dados, as estruturas de dados ainda não possuem as propriedades físicas de um Sistema Gerenciador de Banco de Dados (SGBD) específico.

No Modelo Lógico de Dados, as entidades e relacionamentos, devem representar da forma mais fiel possível, o negócio alvo do Sistema de Informação sendo abordado.

4.5.1.1. Entidade

É a representação de “algo” que existe no mundo real. Uma entidade pode representar um elemento concreto, como uma pessoa ou um livro, ou um elemento abstrato, como um empréstimo, uma viagem ou um conceito. Um conjunto de entidades contém elementos que são similares e compartilham as mesmas características. Cada elemento deste conjunto é dito uma ocorrência da entidade e será identificado de forma unívoca em relação aos outros elementos. As entidades são classificadas em:

a) **Entidade Associativa:** A entidade associativa surge de um relacionamento NxM, em que existe uma associação dos atributos identificadores das duas entidades relacionadas, caracterizando uma nova entidade. A nova entidade gerada possui, normalmente, atributos próprios do relacionamento. Também se utiliza esse tipo de entidade para representar relacionamentos múltiplos (envolvendo mais de 2 entidades) ou agregações (relacionamentos que se relacionam com outras entidades).

b) **Entidade Forte (Independente ou Dominante):** uma entidade x é dita forte quando sua existência não depende da existência de outra entidade.

c) **Entidade Fraca (Dependente ou Subordinada)**: uma entidade x é chamada fraca quando sua existência depende hierarquicamente da existência de outra entidade y. A entidade y é chamada entidade dominante (forte) e a x é chamada entidade subordinada (fraca).

d) **Superclasse (Supertipo):** quando diferentes entidades possuem características comuns, essas características podem ser agrupadas em uma entidade genérica de nível superior chamada superclasse. O processo de agrupamento é chamado generalização.

e) **Subentidade (Subtipo):** quando a partir da subdivisão de uma entidade, segundo um critério de classificação, são obtidas outras entidades com características próprias. Cada entidade obtida com a subdivisão é chamada de subclasse e representa um subconjunto da entidade superior, a superclasse. O processo de subdivisão é dito Especialização.

4.5.1.2. **Atributo**

Uma entidade é representada por um conjunto de atributos. Um atributo é uma propriedade que qualifica, descreve ou identifica um dado dentro da entidade. Os atributos descrevem cada membro de um conjunto de entidades. Um atributo pode ser:

a) **Simples (Atômico):** quando o atributo possui um valor que não é divisível (ex.: sexo);

b) **Composto**: quando o atributo possui um valor que pode ser dividido em partes menores (outros atributos) (ex.: endereço);

c) **Monovalorado**: quando o atributo assume um único valor para cada elemento do conjunto de entidades (ex.: nome);

d) **Multivalorado**: quando o atributo pode ter mais de um valor para uma única entidade (ex.: dependentes, telefones);

e) **Derivado:** quando o valor do atributo é derivado de outros atributos da própria entidade ou a partir de atributos de entidades relacionadas (ex.: idade);

f) **Nulo:** em alguns casos, uma determinada entidade pode não ter um valor aplicável para um atributo, ou este valor é desconhecido (ex.: número do apartamento).

4.5.1.3. **Relacionamento**

Trata-se da associação existente entre uma ou mais entidades. As modalidades de relacionamentos são as seguintes:

a) **Relacionamento Identificador**: é aquele que faz parte da composição do identificador único de uma entidade.

b) **Relacionamento Não Identificador**: é aquele que não faz parte da composição do identificador único de uma entidade.

c) **Auto-relacionamento**: é aquele em que uma entidade se relaciona com ela mesma (relacionamento recursivo).

4.5.1.3.1. **Cardinalidade**

É a quantidade de vezes em que um elemento de um conjunto de entidades pode, em um determinado instante, estar associado, em um dado relacionamento, a outros elementos de outras entidades. Quanto à cardinalidade, os relacionamentos podem ser classificados como:

a) **Um para um (1:1)**: Quando um elemento da entidade A pode estar associado no máximo a um elemento da entidade B, e um elemento da entidade B pode estar associado no máximo a um elemento da entidade A.

b) **Um para muitos (1:N)**: Quando um elemento da entidade A pode estar associado a vários elementos da entidade B. Entretanto, um elemento da entidade B pode estar associado no máximo a um elemento da entidade A.

c) **Muitos para muitos (N:M)**: Quando um elemento da entidade A pode estar associado a vários elementos da entidade B, e um elemento da entidade B pode estar associado a vários elementos da entidade A.

4.5.1.4. Restrição de Integridade

É uma regra que deve ser obedecida para que se mantenha a consistência dos dados de um banco de dados. As restrições de integridade podem ser dos seguintes tipos:

a) **Conteúdo:** quando um determinado atributo deve obrigatoriamente possuir um valor ou não. É o que definimos de atributo obrigatório ou opcional.

b) **Domínio**: quando limita os valores que um atributo pode conter. Fica registrada quando se define o formato, tamanho e valores possíveis de um atributo.

c) **Referencial**: representada por um relacionamento entre duas entidades do modelo lógico, é usada para manter a consistência entre os dados relacionados nas duas entidades.

d) **Negócio (Semântica)**: quando representa uma regra característica do negócio em estudo, que não se enquadra nos demais tipos.

4.5.1.5. Relação Normalizada

A normalização de dados consiste na realização de vários passos seguidos no projeto de um banco de dados, que permitem um armazenamento consistente e eficiente acesso aos dados em bancos de dados relacionais. Esses passos reduzem a redundância de dados e as chances dos dados se tornarem inconsistentes.

A técnica de normalização prevê a existência de regras, chamadas de formas normais, que devem ser aplicadas com o objetivo de alcançar certas condições desejáveis que uma relação deve satisfazer.

Uma relação normalizada é a que esteja pelo menos na terceira forma normal, ou seja:

a) Não contenha colunas com elementos multivalorados ou compostos (1ª forma normal);

b) Cada coluna que não faça parte da chave primária depende, funcionalmente, de todas as partes da chave primária (2ª forma normal);

c) Cada coluna que não faça parte da chave primária depende, funcionalmente, apenas da chave primária, e de nenhuma outra coluna fora da chave primária (3ª forma normal).

4.5.2. Modelo Físico de Dados

O modelo de dados físico deve ser criado a partir da transformação de um modelo de dados lógico. Nele a representação dos objetos, suas características e relacionamentos seguem as regras de implementação necessárias para o tipo de tecnologia de armazenamento utilizada para implementar as estruturas de dados deste. Neste nível, são introduzidos conceitos clássicos relacionados a modelos de dados físicos.

a) **Tabela**: estrutura de armazenamento de dados, formada por um conjunto finito de colunas e ilimitado de linhas (tuplas). Uma tabela é equivalente a uma entidade no modelo lógico.

b) **Linha (ou tupla)**: representação equivalente à ocorrência de uma entidade. Cada linha representa um registro dentro da tabela.

c) **Coluna**: são unidades básicas de armazenamento de dados em uma linha dentro de uma tabela. Uma coluna é similar a um atributo no modelo lógico.

d) **Chave Primária**: é um atributo ou um conjunto de atributos cujos valores distinguem ou identificam de forma unívoca cada linha dentro de uma tabela.

e) **Chave Única (Unique Key)**: elemento que especifica que outros valores de coluna(s), além da chave primária, devem ser único(s).

f) **Chave Estrangeira (Foreign Key)**: representa um atributo, ou combinações de atributos, cujos valores aparecem na chave primária ou em uma chave única de outra entidade que se relaciona com esta, onde encontramos a chave estrangeira. A chave estrangeira é oriunda do relacionamento entre as entidades.

g) **Índice**: recurso físico que visa otimizar a recuperação de uma informação, via um método de acesso, fornecendo acesso rápido a linhas específicas no banco de dados, eliminando a necessidade de varreduras completas de tabelas. Um índice é a seleção de uma ou mais colunas de uma tabela para já armazenar os seus dados ordenados, acompanhado de ponteiros que servirão como atalhos para a localização física dos registros.

h) **Regra de Verificação (Check Constraint)**: elemento que especifica regras para os valores que uma coluna poderá receber.

i) **Sequência (Sequence)**: elemento que gera uma numeração serial e única para colunas numéricas de uma tabela.

j) **Visão (View)**: Representa uma visão parcial ou completa de uma tabela ou mais tabelas.

k) **Visão Materializada (Materialized View):** réplica de uma tabela ou visão a partir de um único ponto no tempo. Muito utilizada em modelos multidimensionais para compartilhar estruturas de dados.

4.6. Conceitos relativos a Estruturas de Dados para Aplicações Analíticas

4.6.1. Modelagem de Dados Multidimensional

É uma técnica de projeto na qual a informação reside na interseção de várias dimensões. A modelagem multidimensional permite que o usuário perceba os dados em uma forma próxima de seu entendimento, com várias perspectivas possíveis. É aplicada na elaboração de modelos de dados que visam atender aos objetivos principais de projetos de Data Warehouse (DW) e Data Mart(DM). Estes devem proporcionar meios que facilitem a investigação, resumo e organização de dados para a análise, resultando em relatórios de apoio à tomada de decisões gerenciais e estratégicas.

4.6.1.1. Modelagem de Dados - Esquema Estrela (*Star schema*)

Método de modelagem de dados, no qual os dados são modelados em tabelas dimensionais ligadas a uma tabela de fatos. As tabelas dimensionais contêm as características de um evento. A tabela de fatos armazena os fatos ocorridos e as chaves para as características correspondentes, nas tabelas dimensionais. O nome foi adotado devido à semelhança do modelo com uma estrela. No “centro” da estrela, existem as tabelas de fatos, rodeadas por tabelas auxiliares, chamadas de dimensões.

4.6.1.2. Modelagem de Dados - Esquema Floco de Neve (*Snowflake schema*)

Método de modelagem de dados que representa uma variação do esquema estrela, porém possui ao menos uma de suas tabelas de dimensão normalizadas.

4.6.2. *Data Warehouse* (DW)

É um tipo de banco de dados voltado ao apoio à tomada de decisões gerenciais e estratégicas, cuja tecnologia visa promover melhores negócios à empresa a partir da análise de grande quantidade de informação que se encontra distribuída por diversos sistemas de produção e sistemas externos. O projeto e a implementação de um DW requerem a utilização de conceitos, técnicas, metodologias e ferramentas adicionais das utilizadas nos bancos de dados transacionais.

4.6.3. *Data Mart* (DM)

É um subconjunto departamental de um DW, concentrado em assuntos de negócio (Pedidos, Vendas, Finanças, Concessão de Benefícios aos filiados da Previdência Social, entre outros), ou seja, o DM é um DW com escopo de projeto delimitado a uma área ou assunto específicos de uma empresa.

4.6.4. *Operational Data Store* (ODS)

Um Operational Data Store (ODS) é um conjunto de dados baseado em assuntos, integrado, volátil, atual ou recente, distinto do Data Warehouse, de apoio às decisões operacionais do dia a dia.

4.6.5. *Staging Area* (SA)

Área ou local destinado a representar as entidades copiadas a partir das fontes de dados de origem. Também é utilizada para tratamento das informações necessárias para as operações de ETL.

4.6.6. Área de Trabalho (AT)

Conceito adotado na Dataprev cuja finalidade é similar à SA.

4.6.7. Tabela Fato

É uma estrutura que armazena medidas associadas aos eventos de negócio. Cada uma das métricas é obtida pela interseção de todas as dimensões que compõem o esquema estrela. Ela possui uma considerável quantidade de linhas (maior volume de dados) e sua chave primária, geralmente, é constituída das chaves estrangeiras de todas as dimensões, considerando o nível de granularidade mais abrangente. A tabela fato é normalizada e esparsa, dado a grande quantidade de combinações possíveis.

4.6.8. Tabela Dimensão

Estrutura que armazena as descrições textuais das entidades representativas do negócio, tais como tempo, localização geográfica, produto, cliente etc. Possuem uma relação de 1:N com a tabela fato. Ela dispõe de um número significativamente menor de linhas cujas colunas (atributos) são textuais e algumas delas estão dispostas hierarquicamente.

4.6.9. Tabela Ponte

Tabela com uma chave composta que captura um relacionamento de muitos para muitos (n para m) que não pode ser acomodado pela granularidade peculiar de uma tabela de fatos simples ou de uma tabela de uma única dimensão. Funciona como ponte entre a tabela de fatos e a tabela de dimensões para aceitar dimensões com diversos valores ou hierarquias desiguais. Também conhecida como tabela associativa ou auxiliar.

4.6.10. Data Lake

O termo Data Lake se refere a um conjunto de dados armazenados de forma agrupada com o objetivo de análises estratégicas e consultas exploratórias. O Data Lake pode armazenar dados estruturados e não estruturados em seu formato bruto e são projetados para o consumo de dados, apoiando a descoberta de novas perguntas sobre os dados. Como o Data Lake em sua grande maioria armazena os dados em seu formato bruto, cabe àqueles que vão analisar os dados, dar sentido a esses dados para o propósito ao qual a análise se destina. Na Dataprev, este também armazena dados trabalhados/transformados a partir dos dados brutos previamente carregados neste, com o objetivo de aproximar o dado do resultado esperado a ser analisado pelo cliente.

4.7. Conceitos relativos à Estrutura de Dados Georreferenciados

Esses conceitos abrangem desde o campo geral de uma área do conhecimento envolvendo dados georreferenciados, às estruturas específicas desses dados, sendo de grande importância para a normatização das formas de uso desse tipo de dado.

4.7.1. Geoprocessamento

Pode ser entendido como uma disciplina que aborda desde a captação dos dados geográficos, normalmente com o auxílio de recursos computacionais, até a sua disponibilização, que pode ser via Cartografia e/ou resultados de consultas geoespaciais, por exemplo. Para Câmara e Davis (2004, pelo INPE), (...) “o termo Geoprocessamento denota a disciplina do conhecimento que utiliza técnicas matemáticas e computacionais para o tratamento da informação geográfica e que vem influenciando de maneira crescente as áreas de Cartografia, Análise de Recursos Naturais, Transportes, Comunicações, Energia e Planejamento Urbano e Regional”. Essa

disciplina, em sua conceituação, relaciona-se e, até certo ponto, pode se confundir, com o de Sistemas de Informações Geográficas;

4.7.2. Sistemas de Informações Geográficas (SIGs)

São sistemas, computacionais ou, no mínimo, baseados na Matemática e no conhecimento geográfico, que abrangem Profissionais (não apenas geotecnólogos, mas todos que necessitam e saibam operar os recursos existentes a partir da necessidade de responder a perguntas que envolvam o "onde"), Metodologias/Métodos, Hardware, Software e Dados Georreferenciados. Para exemplificar, têm-se um trecho de Câmara e Davis (2004, pelo INPE), com os seguintes projetos que levaram a ferramentas que se baseiam em SIGs: o Sistema de Análise Geo-Ambiental (SAGA); o SPRING, do INPE; e, podem-se acrescentar, para além da citação, o QGIS e o ArcGIS;

4.7.3. Geocomputação

De acordo com Abrahart & See (2014): Geocomputação é, **em livre tradução**, "a adoção em larga escala do computacionalmente intensivo paradigma científico como ferramenta para pesquisas geográficas". Para esta norma, considera-se como um conceito que abrange todo tipo de método, técnica, algoritmo e ferramenta para dados georreferenciados, incluindo bancos de dados geográficos;

4.7.4. Bancos de Dados Geográficos

De forma simples, é um banco de dados voltado a objetos geográficos como a representação de geometrias, e de matrizes cujas células são valores georreferenciados. Trata-se de um banco de dados para consultas que envolvem o "onde" não apenas na descrição do nome de um banco numa praça através de suas coordenadas geográficas, numericamente, e/ou do endereço, mas de dados que ao final são convertidos binariamente como Well-Known Text Representation of Geometry, permitindo uma forma própria de armazenamento e tratamento dos dados e metadados. Conforme Câmara (2005), oferece armazenamento e recuperação dos dados espaciais e seus atributos. Disso, entende-se que tais dados são representações da realidade geográfica, devendo ser pensados não como algo fechado em si, mas como a parte computacional dessa representação, que se inicia na modelagem do fenômeno espacial georreferenciado.

4.7.5. Dados Georreferenciados

São dados referenciados geograficamente, ou seja, com coordenadas, em graus ou em metro, que associam o evento, fenômeno, objeto, ou cada célula de uma matriz, a uma posição na superfície da Terra modelada matematicamente, associada a uma projeção na tela ou no papel, normalmente representada através de um código de Sistema de Referência de Coordenadas, o "EPSG". Conforme Melo et al. (2014), o georreferenciamento pode ser entendido como "às informações pertinentes à localização exata de um objeto na superfície da Terra, identificando o seu posicionamento, sendo comumente utilizado os Sistemas de Referência Terrestre ou Geodésicos (que os autores retiraram do IBGE, 2011). Por sua vez, estão associados a uma superfície que mais se aproxima da forma Terrestre, e sobre a qual são desenvolvidos todos os cálculos das suas coordenadas". Porém, num banco de dados geográfico, é preciso entender a que tipos de dados são referidos, sendo eles os dados do tipo Vetorial e os do tipo Matricial.

4.7.6. Dados Vetoriais

São dados que se baseiam num modelo representacional de vetor. Dessa forma, de acordo com a apostila de Borges (2002): "utiliza pontos, linhas e polígonos para representar a geometria das entidades geográficas. Pontos são representados por um par de coordenadas, linhas por uma sequência de pontos e polígonos por uma sequência de linhas onde a coordenada do ponto inicial e final coincidem. Entidades geográficas lineares, como ruas, divisões político-administrativas e redes de tráfego, são naturalmente representadas em formato vetorial. As redes são casos especiais de dados vetoriais, onde são utilizados arcos e nós conectados na representação do fluxo e da direção da rede. As operações topológicas e métricas são comuns em representações vetoriais.". Sendo assim, têm-se, no mínimo, em WKT, pontos, linhas e polígonos.

4.7.7. Dados Matriciais

Para a representação de valores em superfícies contínuas, de campos, há as matrizes. Esse tipo de representação, bastante útil para imagens de sensores de satélites ou para imagens oriundas de tratamentos via algoritmos, possuem células cujos valores são georreferenciados. Onde há o X e o Y de uma coordenada, é possível o valor Z, com a altitude baseada num valor de luz (radiação) refletida do solo, o que também pode indicar, para outros objetivos além da Altitude, se o ambiente é solo, água ou vegetação, dentre outros tipos, a partir de trechos do espectro eletromagnético. De forma direta, “consiste no uso de uma malha quadriculada regular sobre a qual se constrói, célula a célula, o elemento que está sendo representado. A cada célula, atribui-se um código referente ao atributo estudado, de tal forma que o computador saiba a que elemento ou objeto pertence determinada célula.”, conforme Câmara & Monteiro (2004).

4.8. Gerenciamento de Versão

4.8.1. Projeto de Banco de Dados

Conjunto de modelos de dados lógico com alto grau de dependência entre si e seus modelos de dados físicos derivados. Usualmente este alto grau de dependência indica que suas alterações precisam estar alinhadas, de forma que estes devem necessariamente compor uma versão. Um Projeto de Banco de Dados também pode, de forma opcional, conter outros elementos, como arquivos com dados estatísticos sobre os modelos de dados físicos, *DDLs* (*Data Definition Language*) de versionamento, dentre outras informações.

4.8.2. Projeto de Dados

Conjunto de Projetos de Bancos de Dados que embora possuam linhas evolutivas segregadas, possuem forte relacionamento e até dependência, devendo ser apresentados de forma integrada.

4.8.3. Baseline

Representa uma versão de um Projeto de Dados. Complementarmente, segundo a IEEE (IEEE Std. No. 610.12-12-1990), é uma especificação ou um produto que foi formalmente revisado e acordado, e que depois disso serve como base para futuros desenvolvimentos, e que podem ser alterados somente através de um processo formal de controle de alterações.

4.8.4. Stream

É um objeto do repositório da ferramenta IBM Rational Team Concert utilizado para integrar os trabalhos realizado pelas equipes que participam no desenvolvimento do projeto. Representa um fluxo ou linha de trabalho.

4.8.5. Snapshot

Representa uma versão de uma stream.

4.9. Modelo de Dados Corporativos e Compartilhados

Modelos de Dados Corporativos e Modelos de Dados Compartilhados são ferramentas de integração que auxiliam na conexão entre modelos de dados relacionados, servindo de ponto de integração e reuso.

Os Modelos de Dados Corporativos são modelos de dados onde se agrupam entidades reconhecidas por diversas linhas de negócio da Dataprev. Por sua vez os Modelos de Dados Compartilhados se diferenciam dos Corporativos apenas por possuírem um escopo mais específico, usualmente sendo restrito a assuntos de um cliente ou linha de negócio.

Desta forma, o principal objetivo deste tipo de ferramenta de integração é permitir a entrada de dados por um único modelo de dados e o compartilhamento desta informação com todos os sistemas que necessitem dela, sem a necessidade de manter sincronizações entre dados em diferentes sistemas.

5. DIRETRIZES BÁSICAS

Os elementos do modelo de dados devem estar padronizados de acordo com esta norma.

O modelo de dados pode ser construído em conjunto por analistas de negócios, analistas especificadores e administradores de dados.

O modelo de dados deve ser construído buscando a integração com os demais modelos de dados do cliente, evitando-se redundâncias.

O modelo de dados a ser construído deve se utilizar de elementos de dados do Modelo de Dados Corporativos e Compartilhados, quando couber.

O modelo de dados deve ser construído prezando pelas boas práticas da tecnologia na qual será implementado.

5.1. Nomenclatura para Projetos de Dados

O Projeto de Dados, deve acompanhar a sigla do nome do produto que está sendo desenvolvido. Este pode ser subdividido em mais de um Projeto de Banco de Dados conforme as linhas de desenvolvimento e níveis de integração necessários. Cada Projeto de Banco de Dados, por sua vez, deve ter seu nome relacionado ao Projeto de Dados ou um de seus módulos. Para Projetos de Dados que contenham apenas um Projetos de Banco de Dados, o nome de ambos pode ser o mesmo.

Os Projetos de Banco de Dados feitos para aplicações analíticas receberão o prefixo “BG_” para identificá-lo como uma “Base de Gestão” e um sufixo para classificá-la quando necessária. A saber:

a) BG_..._DW: Para projeto de Data Warehouse.

b) BG_..._DM: Para projeto de Data Mart, dependente ou não do Data Warehouse.

c) BG_..._ODS: Para projeto de Operational Data Store(ODS).

d) BG_…_SA: Para projeto contendo apenas tabelas de Staging Area e Área de Trabalho.

e) BG_..._QVD: para projeto em Qlik View cujos repositórios sejam arquivos QVD.

f) DL_...: Para projeto do Data Lake.

g) HS_...: Para projeto na tecnologia Hyperstage.

h) SAS_...: Para projeto na tecnologia SAS.

5.2. Nomenclatura dos Modelos de Dados

5.2.1. Nomenclatura dos Modelos de Dados Lógicos

Para criação de modelos de dados lógicos na ferramenta CASE, devem ser considerados os seguintes padrões de denominação:

<NomeDaAplicação>, onde:

<NomeDaAplicação>: Nome da Aplicação;

5.2.2. Nomenclatura dos Modelos de Dados Físicos

Para criação de modelos de dados físicos na ferramenta CASE, devem ser considerados os seguintes padrões de denominação:

<NomeDaAplicação>_<CategoriaDaBaseDeDados> onde:

<NomeDaAplicação>: Nome da Aplicação;

<CategoriaDaBaseDeDados>: Categoria da base de dados, que podem ser:

- **DEV**: Base de dados de Desenvolvimento;
- **HML**: Base de dados de Homologação;
- **PRD**: Base de dados de Produção;
- **PRC:** Base de dados de Contingência da Produção;
- **TRN**: Base de dados de Treinamento;
- **TIN**: Base de dados de Teste Integrado;
- **TSA**: Base de dados de Teste Automatizado;
- **TDE**: Base de dados de Teste de Desempenho;
- **MND:** Base de dados de Manutenção em ambiente de Desenvolvimento;
- **MNH:** Base de dados de Manutenção em ambiente de Homologação;
- **SEG:** Base de dados de desenvolvimento seguro.

Exemplo:

SIRC_DEV

5.3. Nomenclatura das Linhas de Desenvolvimento

As linhas de desenvolvimento representam o fluxo de evolução das estruturas de dados no tempo e através delas se faz a gestão de configuração e mudança nos modelos e bancos de dados. Estas são também chamadas de *Streams*.

Por padrão, para cada Projeto de Banco de Dados devem ser criadas ao menos duas *Streams*, a *Stream* de *"DEV"* e a *Stream* de *"HEAD".* Outras *Streams* podem ser criadas conforme necessidade do fluxo de trabalho do produto e suas equipes de desenvolvimento. A *Stream* de "*DEV*" é onde as mudanças são de fato implementadas, enquanto a *Stream* de "*HEAD*" será onde se consolidam as mudanças a cada nova versão gerada. As versões geradas na *Stream* de "*HEAD"* são também chamadas de Baselines.

Para a criação de S*treams*, devem ser considerados os seguintes padrões de denominação:

STR_<NomeDaAplicação>_<TipoDaStream>

Onde:

- **str**: Abreviatura de *Stream*;
- **<NomeDaAplicação>**: Nome da aplicação;
- **<TipoDaStream>**: tipo da stream, que pode ser:
  - **HEAD**: *Stream* principal, onde são geradas todas as Baselines e Snapshots;
  - **DEV:** *Stream* de atuação dos ADs e analistas dos projetos/produtos onde as mudanças são implementadas.

5.4. Nomenclatura de Baselines e Snapshots

Baselines e Snapshots devem ser geradas apenas na Stream de HEAD.

Para criação das Baselines e Snapshots, devem ser considerados os seguintes padrões de denominação:

BSL_<NomeDaAplicação>_<CategoriaDaBaseDeDados>

Onde:

- **BSL**: Abreviação de Baseline;
- **<NomeDaAplicação>**: Nome da Aplicação;
- **<CategoriaDaBaseDeDados>**: Trecho opcional, representa a categoria da base de dados onde uma determinada Baseline pode ser implantada, que podem ser:
    - **DEV**: Base de dados de Desenvolvimento;
    - **HML**: Base de dados de homologação;
    - **PRD**: Base de dados de Produção;
    - **PRC:** Base de dados de Contingência da Produção;
    - **TRN**: Base de dados de Treinamento;
    - **TIN**: Base de dados de Teste Integrado;
    - **TSA**: Base de dados de Teste Automatizado;
    - **TDE**: Base de dados de Teste de Desempenho;
    - **MND:** Base de dados de Manutenção em ambiente de Desenvolvimento;
    - **MNH:** Base de dados de Manutenção em ambiente de Homologação;
    - **SEG:** Base de dados de desenvolvimento seguro.

Observação: *Quando a* ***<CategoriaDaBaseDeDados>*** *não é informada no nome da Baseline, significa que esta Baseline pode ser implantada em todas as bases de dados do produto.*

Exemplo:

BSL_SIRC_HML – Pode ser implantada apenas na base de dados de Homologação.

BSL_SIRC_HML_PRD – Pode ser implantada apenas nas bases de dados de Homologação e Produção.

BSL_SIRC – Pode ser implantada em todas as bases de dados do produto.

5.5. Especificação de Diagramas

a) Para manter legível o diagrama em modelos grandes, é preferencial separar em diagramas menores, contendo as tabelas por grupo de funcionalidade ou negócio;

b) As tabelas do próprio modelo não serão coloridas;

c) As tabelas compartilhadas do modelo de dados corporativo (MDC) devem estar em cinza;

d) As demais tabelas compartilhadas de outros sistemas possuirão cores atribuídas livremente, desde que constem em legenda no diagrama.

e) Todo projeto deve ter um diagrama que represente as dependências deste projeto com os demais projetos.

5.6. Especificação de entidades e tabelas

Na especificação de uma entidade ou tabela, devem ser observados alguns aspectos:

I - **Nome**: deve ser informado um substantivo, função ou evento, no plural, em maiúsculas, sem acentuação, podendo ser simples ou composto separado pelo sublinhado (“_”), com no máximo 30 caracteres.

a) Admite-se nome de tabelas no singular quando não existir plural do nome informado;

b) Toda entidade ou tabela deve possuir uma abreviação do nome normal, com até 5 caracteres. Esta abreviação será usada na composição dos

nomes de outros elementos associados à entidade (Chaves primárias, estrangeiras, índices, etc).

II - **Volume inicial**: quantidade de registros que esta tabela provavelmente armazenará no primeiro ano de uso do sistema. Esta informação será usada no cálculo para dimensionar o espaço físico necessário no banco de dados;

III - **Volume final**: quantidade de registros resultante de uma estimativa de crescimento nos próximos 5 anos;

IV - **Documentação:** Toda tabela e entidade devem possuir obrigatoriamente uma boa documentação sobre sua descrição e função dentro do modelo de dados.

5.6.1. **Prefixo de identificação de tabelas com tipo específico**

Prefixo	Descrição	Uso
LG	Tabelas de Log	OLTP
HT	Tabelas de histórico	OLTP
TP	Tabelas temporárias	OLTP
FT	Tabelas Fato	OLAP
DI	Tabelas de Dimensão	OLAP
SA	Tabelas de Staging Area	OLAP
AT	Tabelas de área de trabalho	OLAP
OD	Tabelas de ODS (Operational Data Store)	OLAP

5.7. **Especificação dos atributos e colunas**

I - **Nome:** O nome deve expressar o que o atributo representa;

a) Devem ser iniciados com duas letras correspondentes à sua *classe de atributo* (as classes são definidas na tabela ao final desta seção), seguido de sublinhado (“_”) e o nome propriamente dito, com no máximo 30 caracteres;

b) Caso haja necessidade de abreviação dos termos que compõem um nome do atributo, deve-se optar por abreviar os termos menos importantes para compreensão do atributo, ou por abreviaturas que fiquem mais inteligíveis, procurando deixar os termos mais importantes na forma extensa;

c) Siglas de amplo uso no ambiente dos clientes, podem ser usadas em sua forma original. Contudo deve existir a devida descrição na propriedade de descrição do atributo;

d) Na nomeação de atributos e colunas, preocupar-se sempre com a coerência do significado e para que não haja redundância, como por exemplo, evitar o uso dos nomes cs_status, cs_tipo, te_descrição, entre outros;

II - **Domínios:** Atributos e colunas da classe IN devem estar obrigatoriamente associadas a um domínio DM_SIM_NAO, com valores 0 e 1, significando respectivamente, NÃO e SIM, sempre do tipo *NUMBER(1).*

III - **Tipificação:** Toda coluna deve ter sua classificação conforme a norma de segurança da Dataprev vigente para Tipificação dos Dados, a qual deve estar em conformidade com a Lei Geral de Proteção dos Dados (LGPD).

IV - **Mapeamento Origem Destino:** Toda coluna que seja derivada de um dado de outro Sistema de Informação deve ter seu Mapeamento Origem Destino definido. O Mapeamento Origem Destino é utilizado para estabelecer o ciclo de vida do dado entre os diferentes sistemas de informação. Este conceito também é conhecido na literatura como linhagem do dado.

V - **Classes de Atributos:** As classes de atributos devem ser abreviadas, utilizando-se as seguintes definições abaixo:

Classe		Descrição
ID	Identificador	Prefixo utilizado em atributo que servirá de identificador da tabela.
CS	Classificador	Representa um número finito de valores que serão incluídos no atributo. É obrigatório a associação com um domínio. O atributo pode ser opcional.
CE	Classificador Externo	Representa um número finito de valores que são desconhecidos do modelo de dados, por pertencerem a fontes de dados externas à Dataprev, sob as quais não se possui gestão. Não terá domínio associado.
NU	Número	Expressão de quantidade, categoria ou identificação.
IN	Indicador	Indica um valor lógico, do tipo sim/não.
DT	Data	Campo que receberá valor de uma data e, opcionalmente, também uma hora.
HR	Hora	Receberá o valor de uma hora.
NM	Nome	Substantivo para denominar, objetos, pessoas, eventos, etc...
TE	Texto	Conjunto de palavras que versa sobre algo.
VL	Valor	O preço monetário atribuído a uma coisa; estimação, valoração.
QT	Quantidade	Indica o número de unidades em determinada grandeza.
TX	Taxa	Índice, imposto e taxa.
SG	Sigla	Abreviação de uma palavra ou expressão
IM	Imagem	figura, fotografia, etc.
VD	Vídeo	Vídeo
SO	Som	Som
DO	Documento	Documento em geral, como planilha, documento, PDF, etc.
DG	Dado Georreferenciado	Dados georreferenciado. Normalmente utilizado para análises espaciais.

5.8. Tipos de dados para SGBD Oracle

Os tipos de dados a serem utilizados para o SGBD Oracle conforme cada classe de atributo são definidos a seguir:

Classe	Tipo de Dado	Observações
ID	NUMBER(X)	Especificar o tamanho do atributo de acordo com o volume máximo da tabela, acrescentando-se em média dois dígitos conforme a expectativa de crescimento.
CS	NUMBER(X) VARCHAR2(X)	Especificar o valor máximo do atributo
CE	NUMBER(X) VARCHAR2(X)	Especificar o valor máximo do atributo
NU	NUMBER(X) VARCHAR2(X)	Opção usual Em caso excepcional como número de processo, placa de automóvel, número de chassi, ou demais numerais expressos com caracteres alfanuméricos
IN	NUMBER(1)	Valores permitidos 0 para Não e 1 para Sim
DT	DATE NUMBER(X)	Opção usual Quando tipo de dado for mais adequado ao modelo do que DATE, como quando se refere a uma data parcial como uma competência (aaaamm)
HR	DATE NUMBER(X)	Opção usual Quando o tipo de dado for mais adequado ao modelo do que DATE, como quando se refere a uma hora parcial como “mmss”.
NM	VARCHAR2(X)	Especificar o comprimento máximo do atributo
TE	VARCHAR2(X) CLOB STRING	Especificar o comprimento máximo do atributo Utilizado a nível lógico quando o tamanho do texto a ser inserido é maior que 65535 bytes ou não possui limite conhecido

Classe	Tipo de Dado	Observações
		Utilizado a nível físico quando o tamanho do texto a ser inserido é maior que 65535 bytes ou não possui limite conhecido
VL	NUMBER(X)	Especificar o valor máximo do atributo
QT	NUMBER(X)	Especificar o valor máximo do atributo
TX	NUMBER(X)	Especificar o valor máximo do atributo
SG	VARCHAR2(X)	Especificar o comprimento máximo do atributo
IM	BLOB	
VD	BLOB	
SO	BLOB	
DO	BLOB	
DG	SDO_GEOMETRY(X1,X2,X3,X4,X5)	Os parâmetros para o tipo “SDO_GEOMETRY” são: SDO_GTYPE NUMBER = código para o tipo de geometria SDO_SRID NUMBER = código para o Sistema de Referência de Coordenadas SDO_POINT SDO_POINT_TYPE = com as coordenadas X, Y e Z para pontos SDO_ELEM_INFO SDO_ELEM_INFO_ARRAY = serve para mostrar como interpretar os ordinates, que fazem parte da coluna seguinte SDO_ORDINATES SDO_ORDINATE_ARRAY = armazena as coordenadas dos limites do objeto

5.9. Tipos de dados para SGBD PostgreSQL

Os tipos de dados a serem utilizados para o SGBD PostgreSQL conforme cada classe de atributo são definidos a seguir:

Classe	Tipo de Dado	Observações
ID	NUMERIC(X)	Especificar o tamanho do atributo de acordo com o volume máximo da tabela, acrescentando-se em média dois dígitos conforme a expectativa de crescimento.
CS	NUMERIC(X) VARCHAR(X)	Especificar o valor máximo do atributo.
CE	NUMERIC(X) VARCHAR(X)	Especificar o valor máximo do atributo.
NU	NUMERIC(X) VARCHAR(X)	Especificar o valor máximo do atributo e decimais, se houver. Em caso excepcional como número de placa, chassi, processo, ou demais numerais expressos com caracteres alfanuméricos.
IN	NUMERIC(X)	Valores permitidos 0 para Não e 1 para Sim.
DT	DATE NUMERIC(X)	Armazena somente a Data. Quando tipo de dado for mais adequado ao modelo do que DATE, como quando se refere a uma data parcial como uma competência “aaaamm”.
HR	TIME NUMERIC(X)	Armazena somente a hora do dia. Quando o tipo de dado for mais adequado ao modelo do que TIME, como quando se refere a uma hora parcial como “mmss”.
NM	VARCHAR(X)	Especificar o comprimento máximo do atributo
TE	VARCHAR(X) TEXT	Especificar o comprimento máximo do atributo. Utilizado quando o tamanho do texto a ser inserido não possui limite de caracteres.
VL	NUMERIC(X)	Especificar o valor máximo do atributo.
QT	NUMERIC(X)	Especificar o valor máximo do atributo.
TX	NUMERIC(X)	Especificar o valor máximo do atributo.
SG	VARCHAR(X)	Especificar o valor máximo do atributo.
IM	BLOB	Será convertido no modelo físico e no SGBD para BYTEA
VD	BLOB	Será convertido no modelo físico e no SGBD para BYTEA
SO	BLOB	Será convertido no modelo físico e no SGBD para BYTEA
DO	BLOB	Será convertido no modelo físico e no SGBD para BYTEA
DG	Geometry	Conforme Obe e Hsu (2021), em obra no formato e-book, existem esses 4 tipos para

Classe	Tipo de Dado	Observações
	Geography Raster Topology	dados georreferenciados: Geometry é um tipo para superfícies planas. Utiliza-se da matemática cartesiana. Geography é um tipo para superfícies esféricas. Linhas e polígonos são desenhados sobre a superfície curva da Terra. Raster é um tipo para células multi-banda. Ou, mais especificamente, é uma grade de células retangulares, cada uma contendo valores numéricos em arrays. Topology considera a modelagem com uma rede de nós conectados, arcos (como se fossem “linhas”, arestas) e faces (áreas). Serve também para roteamento.

5.10. Tipos de dados para HADOOP/Hive

Os tipos de dados a serem utilizados para os ambientes Hadoop/Hive, conforme cada classe de atributo são definidos a seguir:

Classe	Tipo de Dado	Observações
ID	DECIMAL(X)	Especificar o tamanho do atributo de acordo com o volume máximo da tabela. Admite-se tamanhos de até três vezes o volume máximo ou dois dígitos.
CS	DECIMAL(X) VARCHAR(X)	Especificar o valor máximo do atributo.
CE	DECIMAL(X) VARCHAR(X)	Especificar o valor máximo do atributo.
NU	DECIMAL(X) VARCHAR(X)	Opção usual. Em caso excepcional como número de processo, placa de automóvel, número de chassi, ou demais numerais expressos com caracteres alfanuméricos.
IN	DECIMAL(1)	Valores permitidos 0 para Não e 1 para Sim.
DT	DATE DECIMAL(X)	Opção usual. Quando tipo de dado for mais adequado ao modelo do que DATE, como quando se refere a uma data parcial como uma competência “aaaamm”.
HR	TIMESTAMP DECIMAL(X)	Opção usual. Quando o tipo de dado for mais adequado ao modelo do que TIMESTAMP, como quando se refere a uma hora parcial como “mmss”.
NM	VARCHAR(X)	Especificar o comprimento máximo do atributo.

Classe	Tipo de Dado	Observações
TE	VARCHAR(X) CLOB	Especificar o comprimento máximo do atributo. Utilizado quando o tamanho do texto a ser inserido é maior que 4000 bytes ou não possui limite.
VL	DECIMAL(X)	Especificar o valor máximo do atributo.
QT	DECIMAL(X)	Especificar o valor máximo do atributo.
TX	DECIMAL(X)	Especificar o valor máximo do atributo.
SG	VARCHAR(X)	Especificar o comprimento máximo do atributo.
IM	BINARY	
VD	BINARY	
SO	BINARY	
DO	BINARY	
DG	Point Rectangle Polygon	Consideram o espaço da Matemática Euclidiana. *(Necessita da biblioteca Spatial Hadoop)*

5.11. Nomenclatura de demais elementos dos modelos

Elemento	Nome padronizado	Exemplo
Chave Primária	{abreviação da tabela}_PK	ACA_PK
Índice de Chave Primária	{nome da chave primária}	ACA_PK
Chave Estrangeira	{abreviação da tabela}_{abreviação da tabela “pai”}_FK	ACA_CEPS_FK
Índice de Chave Estrangeira	{nome da chave estrangeira}_I	ACA_CEPS_FK_I
Índice	{abreviação da tabela}_{nome significativo}_I	ACA_ENDERECO_I
Chave Única	{abreviação da tabela}_{nome significativo}_UK	MUNI_NOME_UK

Regra de Verificação	{abreviação da tabela}_{nome significativo}_CK	FUN_SALARIO_CK
Domínio	DM_{nome significativo}	DM_ESTADO_CIVIL
Sequence	SQ_{nome da tabela associada no plural}	SQ_ACOES
Visão	VI_{nome significativo}	VI_FUNCIONARIOS_ATIVOS
Visão Materializada	MV_{nome significativo}	MV_MUNICIPIOS
Procedure	PR_{nome significativo}	PR_ATUALIZA_HISTORICO
Trigger	TG_{nome significativo}	TG_ATUALIZA_LOG
Function	FC_{nome significativo}	FC_IDENTIFICA_USUARIO
Package	PKG_{nome significativo}	PKG_CARGA_HISTORICO

6. O PROCESSO DE INTEGRAÇÃO DE MODELOS DE DADOS

As bases de dados de um cliente devem ser entendidas como peças de um modelo de dados maior que busca representar o negócio do cliente, independente da tecnologia de armazenamento adotada. A construção de modelos de dados ou processo de modelagem de dados, como também pode ser chamado, privilegiando a construção de modelos de dados altamente integrados apresenta os seguintes benefícios:

> *(i)* Entrada de dados em pontos únicos, evitando a necessidade de sincronização cruzada entre modelos de dados;
>
> *(ii)* A redundância de dados é evitada, economizando espaço de armazenamento;
>
> *(iii)* Os modelos de dados são mantidos coesos, coerentes e responsáveis por um subconjunto dos dados que fazem parte do negócio do cliente.

Desta forma pregasse que a integração dos modelos de dados deve ser um dos principais objetivos no processo de construção de cada modelo de dado. De fato, a análise dos dados que comporão um novo modelo e como ele pode ser incorporado ao modelo de dados do cliente deve ser uma das primeiras atividades do processo de modelagem de dados. Dentre as principais atividades quanto ao processo de integração, destacam-se:

(1) Novos atributos e entidades devem ser avaliadas quanto a que assunto ou negócio do cliente estes pertencem e se estes já são tratados em alguns dos modelos de dados já existentes.

(2) Uma vez entendido que de fato representam um conceito não abordado em algum dos modelos de dados já existentes, as novas entidades e atributos devem ser avaliadas quanto ao seu uso pelo demais modelos de dados existentes do cliente. Caso possam ser utilizadas por vários dos demais modelos, estas devem ser incluídas em um dos modelos de dados compartilhados do cliente ou até cogitada sua incorporação ao Modelo de Dados Corporativo caso sejam de interesse de outros clientes.

Uma exceção a ser observada quanto a evitar a redundância de dados, diz respeito às aplicações onde a comunicação entre as fontes de dados seja um gargalo tecnológico e o desempenho seja um requisito crítico. Nesta situação se considera a clonagem dos dados como uma técnica adequada. No entanto, este artifício deve ser utilizado com cautela, não permitindo que se crie outro ponto de entrada de informações para os dados clonados e jamais permitindo que estes sofram atualizações.

6.1. Processo de Integração nas Aplicações Analíticas

Aplicações analíticas usualmente precisam lidar com grandes volumes de dados, desta forma a exceção quanto a evitar a redundância de dados explanada na seção anterior é justamente uma das técnicas empregadas como meio de garantir a viabilidade desta categoria de sistemas de informação.

Nas aplicações analíticas a consulta às fontes de dados de forma online inviabilizaria as aplicações. A comunicação entre aplicações analíticas e suas fontes de dados, devido ao desempenho na comunicação de grandes volumes de dados, é justamente um gargalo tecnológico, onde a técnica de clonagem se torna necessária.

Desta forma, quanto ao processo de integração quando referenciamos a aplicações analíticas, temos um terceiro passo:

(3) Analisar as fontes de dados a serem utilizadas na construção do modelo de dados e cloná-las semanticamente idênticas ao dado presente na origem e apenas a partir destas entidades clonadas se recomenda implementar as transformações necessárias.

Por semanticamente idênticas ao dado presente na origem, quer se dizer que traduções necessárias na migração do dado entre tecnologias de armazenamento são toleradas.

Com relação às técnicas de clonagem disponíveis, na Dataprev estas tem evoluído com o tempo e conforme as tecnologias de armazenamento foram sendo incorporadas. Cada tecnologia de armazenamento possui suas indicações quanto à forma mais adequada de implementar a clonagem, mas independente da técnica de clonagem adotada, os principais pontos de atenção são:

> *(i)* Não permitir atualizações nos dados clonados que necessitem ser retornados às fontes de dados de origem;
>
> *(ii)* Não permitir operações de transformação entre a fonte de dados de origem e o dado clonado, preservando a semântica original do dado.

De fato, impedir que haja atualizações nos dados clonados evita a necessidade de implementação de complexas sincronizações cruzadas. Por sua vez, preservar a semântica original do dado permite que este seja reusado por vários processos dentro de uma mesma aplicação ou por várias aplicações, dentre de uma mesma tecnologia de armazenamento ou servidor. No mais, também reduz o ônus das extrações nas fontes de dados de origem. As alterações necessárias às aplicações nas entidades clonadas devem ser tratadas em uma segunda camada de dados específica, usando estas entidades clonadas como fonte de dados.

A exceção à regra é quando não há possibilidade ou interesse no reuso do dado da fonte de origem por outras aplicações, neste caso não haverá ônus desnecessário de outros processos de extração e carga de dados e a transformação pode ser tolerada que aconteça em conjunto com o processo de extração e carga de dados.

7. ARQUITETURA DE DADOS PARA APLICAÇÕES ANALÍTICAS

A arquitetura de Projetos de Dados para aplicações analíticas na Dataprev pode ser entendida como uma arquitetura multicamada. Este aspecto se deve em parte aos requisitos de integração mencionados na seção anterior. Esta arquitetura pode ser entendida como adequada para todas as tecnologias de armazenamento utilizadas na Dataprev, sejam estas baseadas em SGBDs Objeto/Relacionais ou mesmo tecnologias de armazenamento “Não SQL”, conforme será discutido a seguir.

A primeira camada diz respeito aos dados clonados a partir das fontes de dados da origem e é persistida em uma área ou local usualmente chamado de *“Staging Area”* (SA) ou “Área de Trabalho” (AT), quando nos referimos a termos de aplicações analíticas. Nesta primeira camada se entende que deve haver apenas as estruturas de dados clonadas semanticamente conforme presentes na origem. Esta camada segue a regra mencionada como forma de propiciar a integração através de seu reuso, pelos motivos explanados na seção anterior. Como também

mencionado anteriormente a exceção à regra se dá quando não há possibilidade ou interesse no reúso do dado.

A segunda camada representa as entidades e atributos resultantes ou utilizadas em transformações necessárias à aplicação analítica. Esta deve utilizar a primeira camada como fonte de dados e assim construir as entidades e atributos específicos que sejam necessários à aplicação analítica em questão sendo construída. Esta pode ser entendida ainda como sendo persistida na área chamada de SA, pois seu objetivo ainda é preparar os dados para a próxima camada a qual conterá os dados disponibilizados ao cliente.

Por sua vez, a terceira camada tem por objetivo representar as entidades do negócio objetivo da aplicação analítica propriamente dita. O local onde a camada pode ser persistida se confunde com os termos utilizados para definir o próprio tipo da aplicação analítica, uma vez que as entidades persistidas nesta são o principal objetivo desta. Desta forma, podemos chamá-la de diferentes formas conforme o tipo de aplicação analítica sendo abordada, mas as denominações mais comuns são: DM, DW. ODS ou Data Lake. Esta representa o objetivo final da aplicação analítica que é dispor os dados ao cliente, seja através de uma ferramenta, aplicação ou diretamente dependendo do tipo da aplicação em questão, no formato adequado para a geração de informações analíticas. Assim, ela representa de fato a aplicação analítica final, conforme exemplificado na figura 1.

Importante ressaltar que pode haver mais camadas conforme as necessidades de transformação do dado ou ainda visões deste que o usuário necessite. No mais, embora conceitualmente apartadas, a forma como as entidades de cada uma destas camadas será fisicamente implementada pode variar conforme a tecnologia de armazenamento adotada, podendo estar alocadas em unidades de armazenamento separadas ou até fisicamente implementadas no mesmo nível hierárquico/esquema.

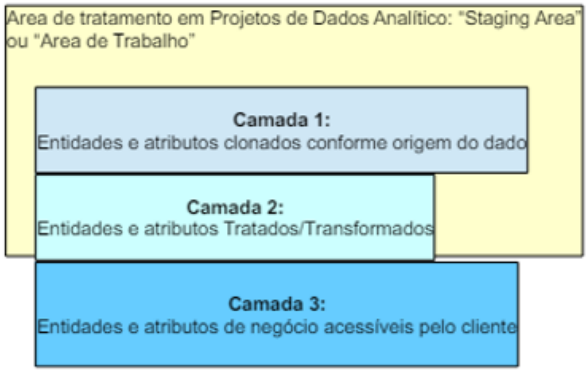

Figura 1: Arquitetura de Dados multicamada para Aplicações Analíticas x Conceitos

8. VIGÊNCIA

Esta Norma entra em vigor a partir desta data e revoga a N/AD/001/05.

VINICIUS DE ARAÚJO PORTO

GERENTE EXECUTIVO

Responsável pela elaboração

ISABEL LUIZA RAFAEL MACHADO DOS SANTOS

SUPERINTENDENTE JURÍDICA E DE COMPLIANCE

Responsável pela chancela

FLAVIO RONISON SAMPAIO

DIRETOR

Responsável pela aprovação

ANEXO I

REFERÊNCIAS

ABRAHART & SEE. ***Geocomputation***. CRC Press, 23 de jun. de 2014.

BARBIERI, Carlos. **BI – Business Ingelligence: Modelagem e Tecnologia**. 1ª Edição. Rio de Janeiro: Axcel Books, 2001.

BORGES, Karla Albuquerque de Vasconcelos. **Modelagem de dados geográficos:** curso de especialização em Geoprocessamento (Apostila). UFMG, 2002. Link: cursogeo2002.PDF (ufmg.br). Acessado em julho de 2024.

CÂMARA et al. **Introdução à Ciência da Geoinformação**. Instituto Nacional de Pesquisas Espaciais (INPE), 2004. Link: Geoprocessamento: Teoria e Aplicacoes (inpe.br). Acessado em julho de 2024.

_____. **Bancos de Dados Geográficos.** Instituto Nacional de Pesquisas Espaciais (INPE), 2005. Link: https://www.dpi.inpe.br/livros/bdados/. Acessado em julho de 2024.

COUGO, Paulo. **Modelagem Conceitual e Projeto de Banco de Dados**. 1ª Edição. Rio de Janeiro: Campus, 1997.

DATE, C J. **Introdução a Sistemas de Bancos de Dados**. 8ª Edição. Rio de Janeiro: Elsevier, 2003.

KORTH, Henry F.; SILBERSCHATZ, Abraham; SUDARSHAN, S. **Sistema de Banco de Dados**. 3ª Edição. São Paulo: Makron Books, 1999.

MELO, D. H. C. T. B. et al. **Decifrando o Georreferenciamento**. Geografia Ensino & Pesquisa, v. 18, n. 3, 25 nov. 2014.

ORACLE. ***Oracle® Spatial and Graph**: Developer's Guide*. Link: Developer's Guide (oracle.com). Acessado em julho de 2024.

ELMASRI, Ramez; NAVATHE, Shamkant B. **Sistema de Banco de Dados**. 4ª Edição. São Paulo: Addison Wesley, 2005.

REGO, Bergson Lopes. **Gestão e governança de dados: promovendo os dados como ativo de valor nas empresas**. 1ª Edição. Rio de Janeiro: Brasport, 2013.

SPATIALHADOOP. Link: SpatialHadoop (umn.edu). Acessado em julho de 2024.

TEOREY, Tobey J. **Projeto e modelagem de banco de dados**. 2ª Edição. Rio de Janeiro: Elsevier, 2014.

MILANI, André. **PostgreSQL: Guia do programador**. 1ª Edição. São Paulo: Novatec Editora, 2008.

OBE, Regina; HSU, Leo S.. **PostGIS in action**. 3a ed. Estados Unidos: Simon and Schuster, 2021.

INMON, W.H.; HACKARTHORN, R. D. **Como usar o Data Warehouse**. 2ª edição. Rio de Janeiro: IBPI Press, 1997.

INMON, W.H.; IMHOFF, Claudia; BATTAS, Greg. **Building the Operational Data Store**. 1ª Edição. New York: John Wiley & Sons, Inc., 1996.

INMON, W.H.; IMHOFF, Claudia; SOUSA Ryan. **Corporate Information Factory**. 2ª edição. New York: Wiley, 2001.

*** Este documento se torna válido a partir da assinatura de todos os signatários indicados. Estando automaticamente invalidadas assinaturas posteriores realizadas por usuários não indicados.**

Documento assinado eletronicamente por **Isabel Luiza Rafael Machado dos Santos**, **Superintendente**, em 16/07/2024, às 15:40, conforme horário oficial de Brasília, com fundamento no Decreto nº 8.539, de 8 de outubro de 2015 e no Decreto n° 10.543, de 13 de novembro de 2020.

Documento assinado eletronicamente por **Vinicius de Araujo Porto**, **Gerente de Departamento**, em 17/07/2024, às 10:11, conforme horário oficial de Brasília, com fundamento no Decreto nº 8.539, de 8 de outubro de 2015 e no Decreto n° 10.543, de 13 de novembro de 2020.

Documento assinado eletronicamente por **Flavio Ronison Sampaio**, **Diretor(a)**, em 23/07/2024, às 15:01, conforme horário oficial de Brasília, com fundamento no Decreto nº 8.539, de 8 de outubro de 2015 e no Decreto n° 10.543, de 13 de novembro de 2020.

A autenticidade deste documento pode ser conferida no site https://dataprev.sei.gov.br/sei/controlador_externo.php?acao=documento_conferir&id_orgao_acesso_externo=0, informando o código verificador **0043040** e o código CRC **700B7FCA**.