

Metodologia de Pesquisa Científica Tecnológica
Aluno: **MARCOS VINICIO PEREIRA**
Fatec SJC - 2020/2 - Professor: Giuliano Araujo Bertoti

Faculdade de Tecnologia de São José dos Campos

Professor Jassen Vidal

MARCOS VINICIO PEREIRA

**GERÊNCIA DINÂMICA E AUTOMÁTICA DO MODELO DE
DADOS NAS BASES DOS CLIENTES
(Gedam)**

São José dos Campos 2020/2

O PROBLEMA

“Manter a estrutura de dados atualizada em todas as bases de clientes, homologações e desenvolvimentos. Fazer as manutenções de forma rápida e eficiente”.

Nos dias atuais todo desenvolvimento de sistema, seja lá quais forem as linguagens utilizadas, encontram um ponto em comum: O Modelo de Dados em Bancos de Dados Relacionais e sua implementação.

Muitos programadores e empresas produtoras de sistemas (*Softwares*) fazem programas que precisam atender a clientes que possuem diferentes Bancos de Dados (BD) e precisam manter atualizados os seus Modelos de Entidades Relacionais, bem como suas respectivas regras (*Constraints*) e índices.

* * *

MOTIVAÇÃO

O uso de Bancos de Dados é o caminho mais utilizado como a forma mais eficiente e segura de persistir seus dados e metadados, porém para ter tal eficiência no dia a dia são necessários experiência e profundos conhecimentos para manipulação e criação de modelos de dados de forma rápida e eficiente em todos os diferentes bancos de dados existente.

Embora os Bancos de Dados tenham a tendência de seguirem, o máximo possível, os padrões ANSI em suas respectivas linguagens SQL, há diferenças em cada um que torna o trabalho de manutenção nas bases dos clientes um trabalho árduo, demorado, suscetíveis a erros principalmente quando um produto possui a premissa de atenderem a mais de um banco de dados.

Também devemos levar em conta que a manutenção das tabelas, seus índices e regras requerem sintaxe repetitiva, alguns cuidados de preparação e sequência que podem muito bem serem colocados em um roteiro.

Hoje os programadores têm que gastar um bom tempo para criar e manter o Diagrama de Entidades Relacionais, um Modelo de Dados com a estrutura exata que precisa ser criada e transcrever o DML (*Data Manager Language*) em parte do sistema ou *scripts* que possam ser utilizados na manutenção das bases dos clientes. Isso ainda piora em sistemas que permitam personalizações específicas de cada instalação.

Os desenvolvedores têm que gastar um bom tempo para criar e manter o Diagrama de Entidades Relacionais, um Modelo de Dados com a estrutura que precisa ser criada e transcrever o DML (*Data Manager Language*) em partes do seu sistema ou *scripts* que possam ser utilizados na manutenção das bases dos clientes. Isso ainda fica pior em sistemas que permitam personalizações específicas de cada instalação.

Essa manutenção acaba sendo feita de forma manual em muitos casos, pois é mais rápido fazer desta forma do que criar diferentes aplicações para cada tipo de banco e testá-las antes de enviar atualizações para estas bases.

PROPOSTA DE SOLUÇÃO

A solução se dará através de uma aplicação de uso remoto que permita fazer os cadastros das Tabelas (Tables) e Visões (Views), gerar pacotes de atualização, enviar para as bases dos clientes e executar tais atualizações.

Cada base a ser mantida, seja de cliente, homologação, testes ou desenvolvimento, terá que ser cadastrada no Gedam com seu respectivo acesso. Na base de destino teremos a replicação dos cadastros deste sistema o que permitirá análise de impacto das alterações antes de aplicá-las, bem como permitir ter suas próprias peculiaridades, como colunas personalizadas.

A aplicação será feita em linguagem que permita sua execução via TCP/IP sempre, pois não há motivos para existir mais de uma fonte de programação. Desta forma teremos também uma aplicação que poderá ser instalada e executada em uma mesma máquina ou em máquinas diferentes que estejam em *hosts* diferentes.

Visão Geral

O sistema será bem simples, pois a complexidade exigida estará na integração das tabelas do seu modelo de dados e não na sua criação e manutenção no dia a dia. Por isso teremos alguns cadastros orientados aos componentes que já utilizamos normalmente na criação destas tabelas, como, o cadastro da tabela (Tables), colunas de tabela, validação da coluna, valor padrão (*default*), Unique Key, Not Null e índices. Na mesma modelagem da tabela teremos modelagens para visões (Views), já considerando que este modelo poderá ser expandido para outro nível de ferramenta que possa integrar diretamente as tabelas ou visões a outras ferramentas como geradores de relatórios, por exemplo.

A forma simples como esta visão é apresentada realmente expressa o que teremos neste sistema, considerando não só a minha visão pessoal do que será como minha experiência em ferramentas parecidas com esta proposta.

Como estamos falando de um sistema que conterà tabelas que servirá para reger a construção e manutenção de outras tabelas, devemos estabelecer que a citação “cadastro” está se referindo à tabela deste sistema (metadados) e a citação “tabela” está se referindo às tabelas que estamos criando e mantendo que são o objeto deste trabalho de graduação.

Metodologia de Pesquisa Científica Tecnológica
Aluno: **MARCOS VINICIO PEREIRA**
Fatec SJC - 2020/2 - Professor: Giuliano Araujo Bertoti

Bases / Clientes

Este cadastro será a definição do destino do que for feto no Gedam. Aqui teremos que cadastrar as bases que serão atendidas o que nos permite cadastrar tanto bases internas de desenvolvimento, testes e homologações como bases externas como de clientes ou de outra equipe de desenvolvimento.

Por se tratar de informações de acesso os dados serão todos criptografados. Esta criptografia não irá impactar no uso do Gedam porque será lida uma única vez quando necessária.

Para este cadastro teremos que informar os campos:

- Nome (Único)
- Banco de Dados Utilizado
- Versão do Banco de Dados
- TCP/IP de acesso ao Banco de Dados
- Usuário de acesso ao Banco de Dados
- Senha de acesso ao Banco de Dados
- Atualizar via Script (A base não será atualizada diretamente, mas através de scripts que serão enviados para o seu respectivo DBA).

Tipos de Coluna

Os tipos de coluna poderão ter dois níveis de definição.

- Tipos portáteis que representarão tipos de colunas comuns a todos os tipos de bancos de dados como Número Decimal (NUMBER), Caracteres (VARCHAR2), Lógico (Boolean) e Data (DATE).
- Tipos específicos que representarão tipos que podem conter tratamentos específicos em bancos de dados, mas não de forma padrão.

Identificadores

De um modo geral todos os cadastros terão um identificador descritivo, sempre em caixa alta, não permitindo espaços e nem caracteres especiais. Desta forma garantimos a legibilidade dos programadores ao utilizarem estes cadastros de forma integrada.

Por isso não iremos citar os identificadores como parte integrante de todos os cadastros que trataremos, mas eles sempre existirão.

Metodologia de Pesquisa Científica Tecnológica
Aluno: **MARCOS VINICIO PEREIRA**
Fatec SJC - 2020/2 - Professor: Giuliano Araujo Bertoti

Dados de Versão e Auditoria

Todo cadastro terá automaticamente a informação de usuário, data e hora em que foi criado e, separadamente, os mesmos dados para registrar a alteração do registro. Estas informações serão utilizadas para a geração de pacotes de atualização.

Estas informações também serão padrões em todas as tabelas de cadastro deste sistema.

Módulos

Este será um cadastro para o programador definir os módulos de seu programa e que será informado no cadastro das tabelas. Desta forma teremos uma organização de todas as tabelas criadas.

Este cadastro terá os campos:

- Identificador
- Prefixo (Que poderá ser obrigatório nos nomes das tabelas)
- Validar prefixo nas novas tabelas

Bancos de Dados

Cadastro dos diferentes Bancos de Dados que serão atendidos pelo Gedam. Este cadastro será estático da aplicação e será alterado conforme novas versões de banco de dados forem sendo criadas. Esta informação somente será utilizada em casos de configurações específicas para algum banco de dados quando for necessário.

Este cadastro terá os campos

- Identificador
- Nome do Banco de Dados (Oracle, Sql Server, MySQL, Firebird, etc).
- Versão do Banco de Dados (Tabela relacionada que permitirá indicar versões diferentes que tenham recursos diferentes do mesmo banco de dados).

Modelagem de Domínio de Tipo de Dado

Um domínio de tipo de dado será um cadastro que permitirá definir características padrões de uma coluna que poderá ser vinculada a diversas colunas de diversas tabelas.

Os campos que definirão este cadastro:

- **Identificador**
- **Nome padrão** (Sugestão ao criar nova coluna)
- **Tipo** (Número, Data, Caracter, etc)
- **Tamanho**
- **Decimais** (Quando se tratar de número)

Metodologia de Pesquisa Científica Tecnológica

Aluno: **MARCOS VINICIO PEREIRA**

Fatec SJC - 2020/2 - Professor: Giuliano Araujo Bertoti

Todas as colunas de tabela que estiverem vinculadas a um domínio de tipo de dado não poderão ter seu tipo e tamanho modificados, garantindo que todas estas colunas estejam definidas de forma igual em todas as suas respectivas tabelas.

Também teremos outra definições que poderão ser utilizadas por aplicações como máscara que poderão ser utilizadas para definir como os dados serão exibidos em tela e prefixo obrigatório que definirá parte inicial do nome da coluna que será obrigatória.

A ideia de utilizar tipos de colunas irá reger o trabalho dos programadores impedindo que estes utilizem tipos não definidos para uso no sistema.

Modelagem de Domínio de PK/FK

De forma parecida com o que teremos no domínio de tipo de dado, teremos um domínio que definirá uma regra de PK. Este domínio será vinculado automaticamente às colunas da sua referida tabela. Em colunas de tabela que se referenciarem a esta PK iremos vincular este código para definirmos a criação da FK.

O cadastro de domínio de PK/FK terá as informações:

- Identificador
- Tabela Relacionada
- Campos Relacionados (A ordem de cadastro dos campos definirão a ordem de criação da PK e suas respectivas FKs)
- Domínio de Tipo de Dado (Opcional)
- **Nome padrão** (Sugestão ao criar nova coluna)
- **Tipo** (Número, Data, Caracter, etc)
- **Tamanho**
- **Decimais** (Quando se tratar de número)

Notamos que este domínio se assemelha ao domínio de tipo de dados, pois também irá reger a forma como uma coluna será criada, mas com o adicional que irá criar as Primary Keys e Foreign Keys automaticamente. Quando a coluna de tabela se referenciar a este domínio não será possível fazer o vínculo a um Domínio de Tipo de Dado.

Metodologia de Pesquisa Científica Tecnológica
Aluno: **MARCOS VINICIO PEREIRA**
Fatec SJC - 2020/2 - Professor: Giuliano Araujo Bertoti

Modelagem de Tabelas e Visões (*Table/View*)

Este será o cadastro principal da criação de tabelas e Views

O cadastro de tabelas terá como seu identificador o próprio nome da tabela que será criada no banco de dados. Por este motivo a descrição do nome não poderá conter caracteres especiais que serão regradados pelo Gedam.

Campos deste cadastro:

- Identificador
- Módulo
- Personalizado (Indicar que foi criada para uso específico de um cliente)
- Nome (Da tabela ou visão)
- Sinalização que é uma visão

Tabelas relacionadas a este cadastro, mas que fazem parte da tabela

- Colunas
- Índices
- Expressão SQL (Para visões)
- Tipo (Cadastro, Lançamento ou Estática)

O cadastro de tabelas será diferente do cadastro de visões. Por este motivo o cadastro de visão exigirá apenas o cadastro da **Expressão SQL** que fará a sua geração.

Modelagem de Colunas das Tabelas

Este será um cadastro vinculado diretamente a uma nova tabela, ou seja, terá como chave secundária uma tabela criada. O seu acesso será de dentro de uma Modelagem de Tabelas e Visões, mas será obrigatória somente no cadastro de Tabelas.

Ao cadastrar uma coluna teremos que informar os campos:

- Identificador (FK da Tabela)
- Sequencial (Utilizado para permitir ordem constante dos campos ou mesmo sua inversão com outros campos)
- Domínio de PK/FK (Opcional)
- Domínio Tipo de Dado (Opcional)
- **Nome**
- **Tipo** (Será automático quando um domínio for informado)
- **Tamanho** (Será automático quando um domínio for informado)
- **Decimais** (Quando se tratar de número e será automático quando um domínio for informado)
- **Valor Padrão** (Default executado pelo Banco de Dados)
- **Not Null** (Sinalização que indicará a criação desta regra)

Metodologia de Pesquisa Científica Tecnológica
Aluno: **MARCOS VINICIO PEREIRA**
Fatec SJC - 2020/2 - Professor: Giuliano Araujo Bertoti

Modelagem de Índices (Gerais ou por banco)

O cadastro de índices está vinculado diretamente a uma nova tabela e, como as colunas, tem seu acesso diretamente de dentro do cadastro de Tabelas e Visões. Porém este cadastro somente será realizado quando se tratar de uma tabela.

A criação dos índices será automática e, este cadastro, também permitirá a recriação dos índices do banco de dados quando o DBA sentir necessidade de fazer tal atualização por questões de degradações percebidas.

Ao cadastrar um índice teremos que informar os campos:

- Identificador (FK da Tabela)
- Sequencial (Utilizado para permitir ordem constante dos campos ou mesmo sua inversão com outros campos)
- Coluna (FK do cadastro de colunas)
- Descendente (Indica que esta coluna é descendente)
- Expressão Alternativa (Campo para digitar expressões alternativas ao nome da coluna como funções como UPPER()).

Modelagem de Sequencias (Sequences)

O cadastro de sequencias que poderão ser utilizadas, por exemplo, em expressão default das colunas.

Ao cadastrar teremos que informar os campos:

- Identificador
- Iniciar em
- Valor de incremento (Padrão 1)

A criação de um sequência não irá atrelá-la diretamente a uma tabela, pois seu uso será por definição dentro de Expressão Padrão de uma coluna.

Gerador de Atualização

Será uma rotina que permitirá agrupar todos os componentes, separadamente, em um lote de atualização tomando por base de filtro algumas informações como módulos, usuários, período (data/hora) e componentes específicos.

Poderemos gerar um pacote de atualização que será enviado às bases pré-definidas e enviar somente o que foi alterado nas modelagens de tabela.

Poderemos também enviar conteúdo de tabelas estáticas para as bases. Este tipo de tabela foi pensado para aquelas tabelas que não são de cadastro dos usuários, mas de uso da sua aplicação como, por exemplo, um cadastro de estados do Brasil.

Metodologia de Pesquisa Científica Tecnológica
Aluno: **MARCOS VINICIO PEREIRA**
Fatec SJC - 2020/2 - Professor: Giuliano Araujo Bertoti

Conclusão

O uso desta aplicação será muito intuitivo, pois espelha de forma clara e visual o que os programadores fazem no seu dia a dia de trabalho. O programador irá apenas definir os nomes de tabelas, nomes de colunas, regras e índices e o Gerador de Atualização vai detectar tudo o que foi alterado para ser enviado às outras bases. A ideia é que realmente seja tão simples de executar como este texto.

O trabalho de definição do modelo de dados será feita de forma simples sem a preocupação com quais sintaxes utilizar para cada banco de dados. O Gedam cuidará disto.

Apêndice

Este trabalho tem um objetivo específico, mas que dará abertura para continuação posterior para outras frentes de trabalho que irão agregar e melhorar o trabalho dos programadores. Vejamos alguns exemplos:

- ✓ Gerador de Relatórios
- ✓ Gerador de MER (Modelo de Entidades Relacionais) gráfico e dinâmico.
- ✓ Migração de bases entre diferentes bancos de dados.

* * *

Marcos Vinicio Pereira