Alteração da Estrutura de Banco de Dados

Regra para Inativação de Campos ou Tabelas

Quando um campo ou tabela deixar de ser utilizado, ele **não** deve ser removido da unit de criação e **não** deve ser executado nenhum comando SQL de exclusão (`DROP`).

A estrutura deve permanecer intacta no banco de dados, mesmo sem utilização pelo sistema.

Definição de Escopo e Escolha da Unit

A unit utilizada para a criação de uma tabela é definida pelo seu escopo de uso:

- **`CriaTabEsp.pas`:** Utilizada para tabelas específicas de um único módulo.
- **`CriaTab.pas`:** Utilizada para tabelas compartilhadas entre dois ou mais módulos do sistema.

Vínculo da Tabela ao Módulo (`DDL.Classes.pas`)

Ao criar uma nova tabela, também é obrigatório registrá-la na unit `DDL.Classes.pas`.

A tabela deve ser adicionada na procedure `CriaMatrizSistemas__` correspondente ao módulo ao qual ela pertence, utilizando o método `AddTabela`.

- **Exemplo Prático (Módulo de Compras - CC):** Para uma tabela pertencente ao módulo de Compras, adicione o registro na procedure `CriaMatrizSistemasCC`.

```
class procedure CriaMatrizSistemasCC(var AMatriz: TMatrizSistema); static;

AddTabela(AMatriz, 'PROPNR', 'Prop. dos itens das Notificações de Recebimento');
AddTabela(AMatriz, 'PROPNRM', 'Prop. dos itens das Notificações de Recebimento no Arquivo Morto', 'NULO',
True, False, True);
```

Esse registro é utilizado pelo sistema para identificar a quais módulos a tabela pertence, sendo também a referência utilizada para determinar quais units `uMatrizAlt__.pas` devem ser atualizadas quando houver alterações em tabelas compartilhadas.

Mapeamento de Chaves Estrangeiras e Integridade

Sempre que uma nova tabela possuir chave estrangeira ou depender de outra tabela do sistema, também é obrigatório registrar essa relação de integridade referencial na unit `DDL.Classes.pas`.

O relacionamento deve ser adicionado na procedure `CriaMatrizIntegRef__` correspondente ao módulo ao qual a tabela pertence, utilizando o método `AddIntegRef`.

- **Exemplo Prático (Módulo de Compras - CC):** Para uma tabela pertencente ao módulo de Compras, adicione o relacionamento na procedure `CriaMatrizIntegRefCC`.

```
class procedure CriaMatrizIntegRefCC(var AMatriz: TMatrizIntegRef); static;

AddIntegRef(
  AMatriz, 'ITENSNR', 'Abrev;Numero;NumeroFor;Serie;NumeroCli;Item', 'PROPNR',
  'Abrev;Numero;NumeroFor;Serie;NumeroCli;Item', tcCascade);

AddIntegRef(
  AMatriz, 'ITENSNRM', 'Abrev;Numero;NumeroFor;Serie;NumeroCli;Item', 'PROPNRM',
  'Abrev;Numero;NumeroFor;Serie;NumeroCli;Item', tcCascade);
```

Esse registro define os relacionamentos entre as tabelas do sistema, permitindo que as regras de integridade referencial, como exclusão ou atualização em cascata, sejam aplicadas corretamente.

Versionamento de Alterações na Base de Dados

Cenário A: Tabelas Exclusivas de Módulo (`CriaTabEsp.pas`)

Ao criar ou modificar uma tabela de módulo único, é obrigatório registrar a alteração na unit `uMatrizAlt__.pas` correspondente ao módulo modificado, chamando o método de atualização com o número da versão do módulo.

- **Exemplo Prático (Módulo de Compras - CC):** Se uma nova coluna for adicionada a uma tabela exclusiva de Compras, você deve acessar a unit `uMatrizAltCC.pas` e incluir a chamada:

```
AddMatrizAlt('PLANFISC', '5.194');
```

Cenário B: Tabelas Compartilhadas (`CriaTab.pas`)

Quando uma tabela compartilhada for criada ou modificada, o desenvolvedor deve identificar todos os módulos afetados e replicar a chamada de atualização em **todas** as units `uMatrizAlt__.pas` desses respectivos módulos.

Como identificar os módulos afetados:

1. Acesse a unit `DDL.Classes.pas`.
2. Verifique, dentro das procedures `CriaMatrizSistemas__`, quais delas fazem uso dessa tabela. Você encontrará uma estrutura semelhante a esta:

```
AddTabela(AMatriz, 'PLANFISC', 'Plano Fiscal');
```

3. Para cada módulo identificado, adicione a chamada `AddMatrizAlt` com o número da versão do módulo na sua respectiva unit de alteração.

- **Exemplo Prático:** Você adicionou um novo campo na tabela `PLANFISC` (alterando a `CriaTab.pas`). Ao buscar por `PLANFISC` na unit `DDL.Classes.pas`, você identificou que ela é utilizada pelas seguintes procedures:
  - `CriaMatrizSistemasCC` (Módulo CC)

- `CriaMatrizSistemasEX` (Módulo EX)
- `CriaMatrizSistemasVD` (Módulo VD)

Portanto, você deverá abrir as units `uMatrizAltCC.pas`, `uMatrizAltEX.pas` e `uMatrizAltVD.pas` e adicionar a chamada de atualização em cada uma delas:

```
AddMatrizAlt('PLANFISC', '5.194');
```

> **Atenção:** Cada módulo possui um número de versão diferente. Caso tenha dúvidas, consulte o guia de como identificar o número da versão de cada módulo.

Revision #13
Created 25 June 2026 18:55:13 by Vinicius Pereira
Updated 2 July 2026 14:29:54 by Carlos Filho