



Introdução FireDAC

Acesso multi-banco para Delphi e C++ Builder

Diego Campos Rosa



Embarcadero Conference

Créditos

“Para aprender Delphi, não basta saber como utilizar os componentes. Deve-se entender como funcionam e, se possível, como foram feitos.”

Gustavo Chaurais

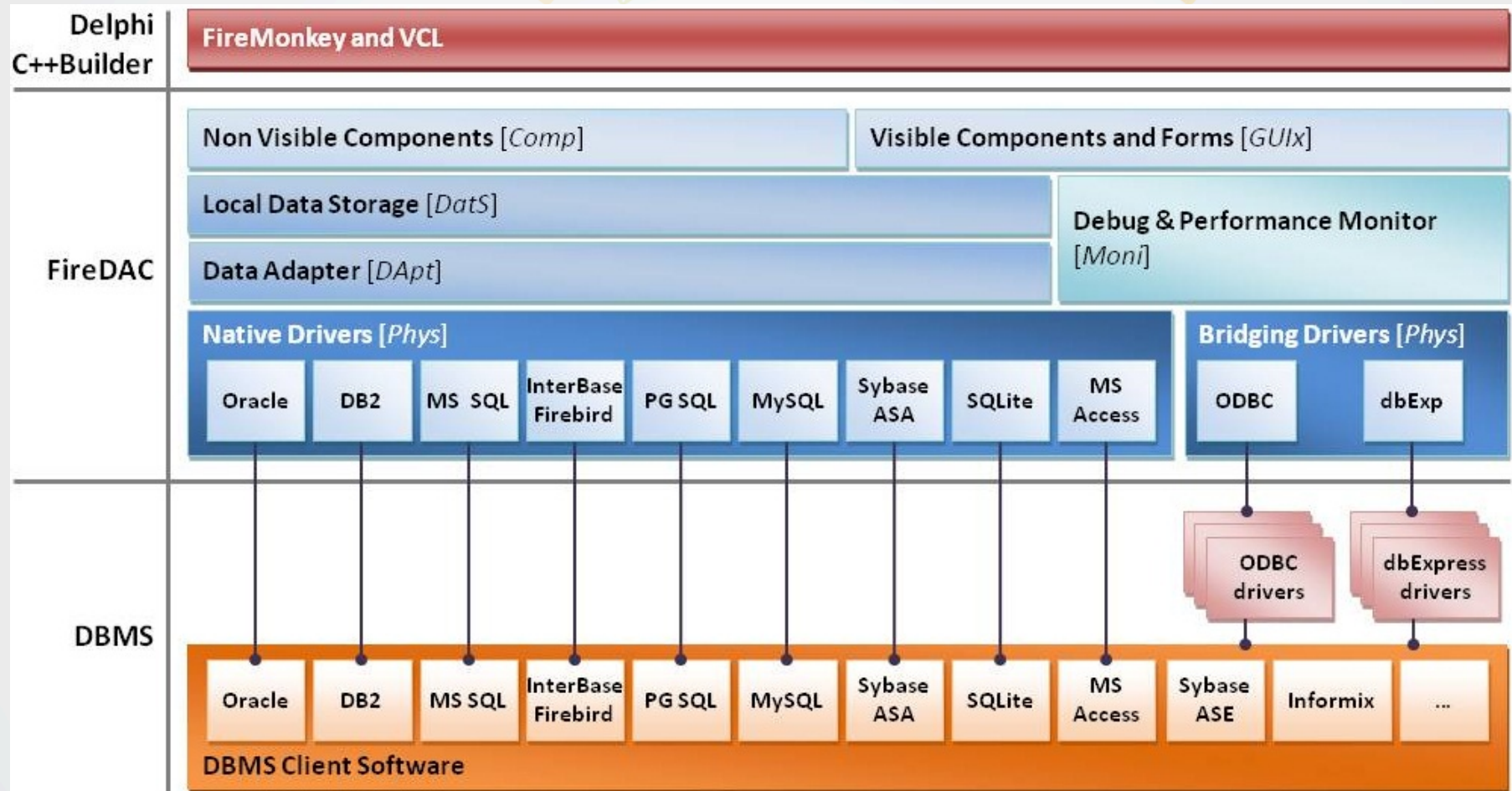
Agenda

- O que é FireDAC
- Visão geral da arquitetura
- Principais características
- Prática
- Perguntas

FireDAC

- Versão Embarcadero do AnyDAC
- Conjunto realmente universal para acesso a banco de dados
 - Possibilitando usar as características específicas de cada banco de dados
- Alta performance de acesso
 - Desde Live Data até Array DML
- API unificada
 - Abstração de SQL
 - Controle de transações e de erros unificados
- Com fácil migração do BDE

Arquitectura



Principais Componentes

- TADConnection - Gerencia Ligação com BD
- TADTransaction - Gerencia as transações com BD
- TADMmemTable - Conjunto de dados na memória
- TADQuery – Implementação da classe DataSet, que permite execução de SQL.
- TADTable - abre uma tabela para navegação
- TADScript - Executa scripts SQL em lote
- TADUpdateSQL – Aplica atualizações que não podem ser feitas diretamente.
- **Driver Link** - Descreve como configurar o driver do BD

Componentes de UI

- **TFDGUIxErrorDialog:** Caixa de diálogo exibe as exceções FireDAC
- **TFDGUIxLoginDialog:** Caixa de diálogo permite que os usuários insiram suas credenciais de banco de dados
- **TFDGUIxWaitCursor:** Componente permite que o controlo do cursor de espera.
- **3 implementação** - FireMonkey, VCL, console

Componentes de Serviço

- TADXxxxBackup - Backup do BD
- TADXxxxRestore – Restauração do BD
- TADXxxxValidate - Verificação e Correção
- TADXxxxSecurity BD - Gerencie a criptografia do BD
- TADSQLiteFunction, TADSQLiteCollation - Componentes específicos para p SQLite

Opções do FireDAC

- FetchOptions – controla o retorno de registros
- FormatOptions - controlam como os tipos das colunas do DB são mapeados
- ResourceOptions - opções de recursos que controlam como os recursos do sistema são utilizados,
- UpdateOptions - opções que controlam como FireDAC realizara o posts para o DB.
- TxOptions - controlam como as transações são realizadas.

TRACING

- TADMMiniFlatfileClientLink: Arquivo Texto
- TADMMiniRemoteClientLink: ADMonitor
- Ativando Tracing
 - Tracing to true
 - MonytorBy = FlatFile ou Remote in connection setting

Mapeamento de tipos de dados

- Mapeamento personalizado por conexão ou DataSet visuais
- Ou em código

Comandos em lote

- Lotes de comando, permite executar múltiplos comandos de SQL num único passo e processar vários conjuntos de resultados de uma sequência.

```
ADQuery1.FetchOptions.AutoClose := False;  
ADQuery1.SQL.Text := 'select * from orders; select * from customers';  
ADQuery1.Open;
```

```
ADQuery1.FetchAll; // assign orders records to ADMemTable1  
ADMemTable1.Data := ADQuery1.Data; ADQuery1.NextRecordSet;
```

```
ADQuery1.FetchAll; // assign customers records to ADMemTable2  
ADMemTable2.Data := ADQuery1.Data;
```

Desempenho Array DML

- Realizado N insert / update / instruções DELETE parametrizado por chamada
- Flexível, fácil e eficaz
- Cada parâmetro contém uma matriz de valores, `FDQuery1.Execute(FDQuery1.Params.ArraySize);`
- O servidores mais lentos ou mais fracos da rede, criam menos pacotes TCP / IP.
- INSERT registros de 10K:
 - DML array -> 0,03 seg
 - Normal ExecSQL -> 5,50 seg

Execução Assíncrona

- Transações de longa duração pode ser assíntrica ou com um tempo limite
- `ResourceOptions.CmdExecMode = amAsync ,`
- `ResourceOptions.CmdExecMode = amCancelDialog`
 - execução assíncrona do diálogo
TADGUIxAsyncExecuteDialog para cancelamento
- `ResourceOptions.CmdExecTimeout`
- `ADDataSet.AbortJob` - Execução de cancelamento

Execução de SQL específicos

- Função Escape - uma expressão para cada banco de dados:

```
select {left({ucase(RegionDescription)}, 3)},  
RegionDescription from {id Region}
```

- Operações condicionais - permite escrever SQL específico para cada DB:

```
{IF Oracle} SELECT * FROM "Region" {fi}  
{IF MSSQL} SELECT * FROM [Territories] {fi}
```

- Macros – permite a troca de parâmetros na aplicação:

```
ADQuery1.SQL.Text: = SELECT * FROM &TabName;  
ADQuery1.MacroByName ('TabName') AsIdentifier : =  
Edit1.Text; ADQuery1.Open;
```

Migrando de BDE

- Alta compatibilidade entre BDE e FireDAC:
 - as propriedades, métodos, sobre o comportamento

Obrigado!!!

Diego Campos Rosa

dcampos@aquasoft.com.br

Canais Embarcadero

<http://edn.embarcadero.com>

<http://www.embarcadero.com/br>

<http://www.facebook.com/DelphiBrasil>

<http://www.facebook.com/EmbarcaderoBrasil>

<http://www.embarcadero.com/mvp-directory>

<http://www.embarcaderobr.com.br/treinamentos/>