

Embarcadero Conference 2024

Inovação faz parte do nosso DNA!



Fiorilli S/C Software

Atuando há mais de 48 anos

Uma das maiores empresas do Brasil no desenvolvimento de softwares de gestão pública

Presentes em mais de 1000 municípios facilitando a rotina de mais de 2500 órgãos públicos

Líder de mercado, é a empresa que mais cresce em seu segmento

Delphi em alta velocidade: Como acelerar seu compilador

{ Vinicius de Abreu



Referência

Esta apresentação é baseada no artigo publicado por Marco Cantú, complementado por dicas provenientes da minha experiência e estudos.

[Some Suggestions to Help the Delphi Compiler](#)



Benefícios ao aplicar essas dicas

- Tempo
- Executável menor
- Projeto Legível
- Projeto Limpo
- CodeInsight mais rápido



Entendendo o processo de compilação

Análise Léxica: O compilador lê o código-fonte e converte os caracteres em tokens, que são as menores unidades significativas, como palavras-chave, identificadores e operadores.

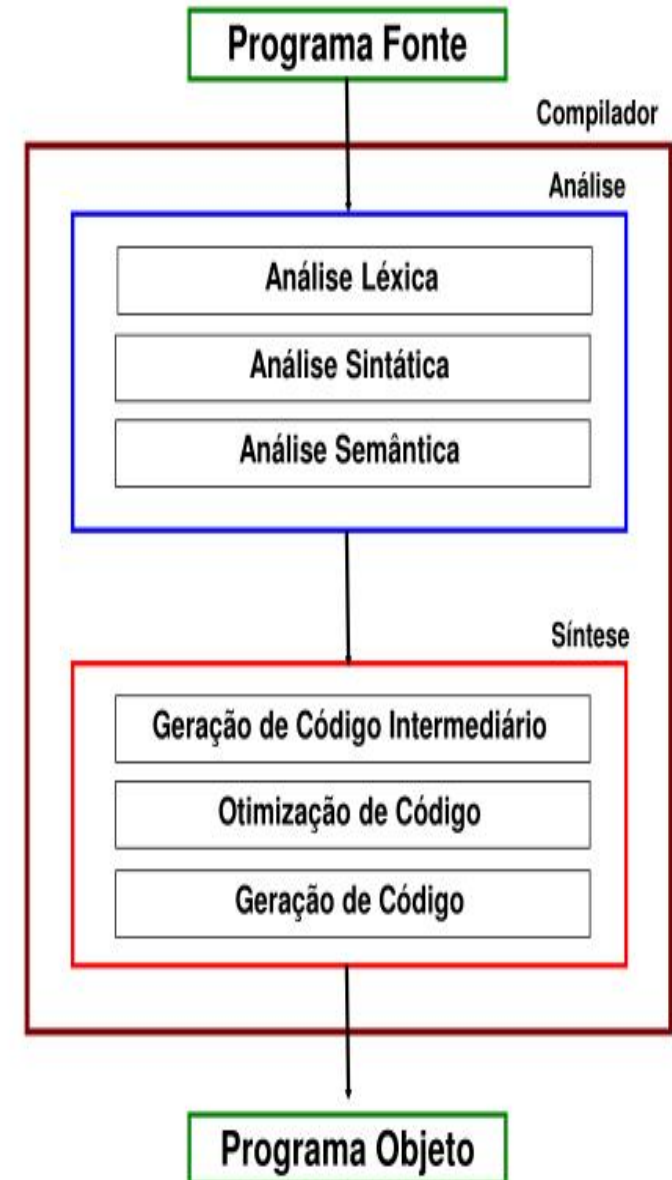
Análise Sintática: Os tokens são organizados em uma estrutura hierárquica, verificando se a sintaxe do código está correta de acordo com as regras da linguagem Delphi.

Análise Semântica: O compilador verifica a semântica do código, garantindo que as operações e declarações façam sentido lógico. Isso inclui verificação de tipos, escopo de variáveis e coerência de operações.

Geração de Código Intermediário: O código é traduzido para uma representação intermediária que é mais fácil de otimizar e transformar em código de máquina.

Otimização de Código: O código intermediário é otimizado para melhorar a eficiência e desempenho do programa final. Isso pode incluir a eliminação de código redundante e a melhoria de loops.

Geração de Código: O código intermediário otimizado é convertido em código de máquina específico para a arquitetura de destino.



Conhecendo a estrutura de uma Unit

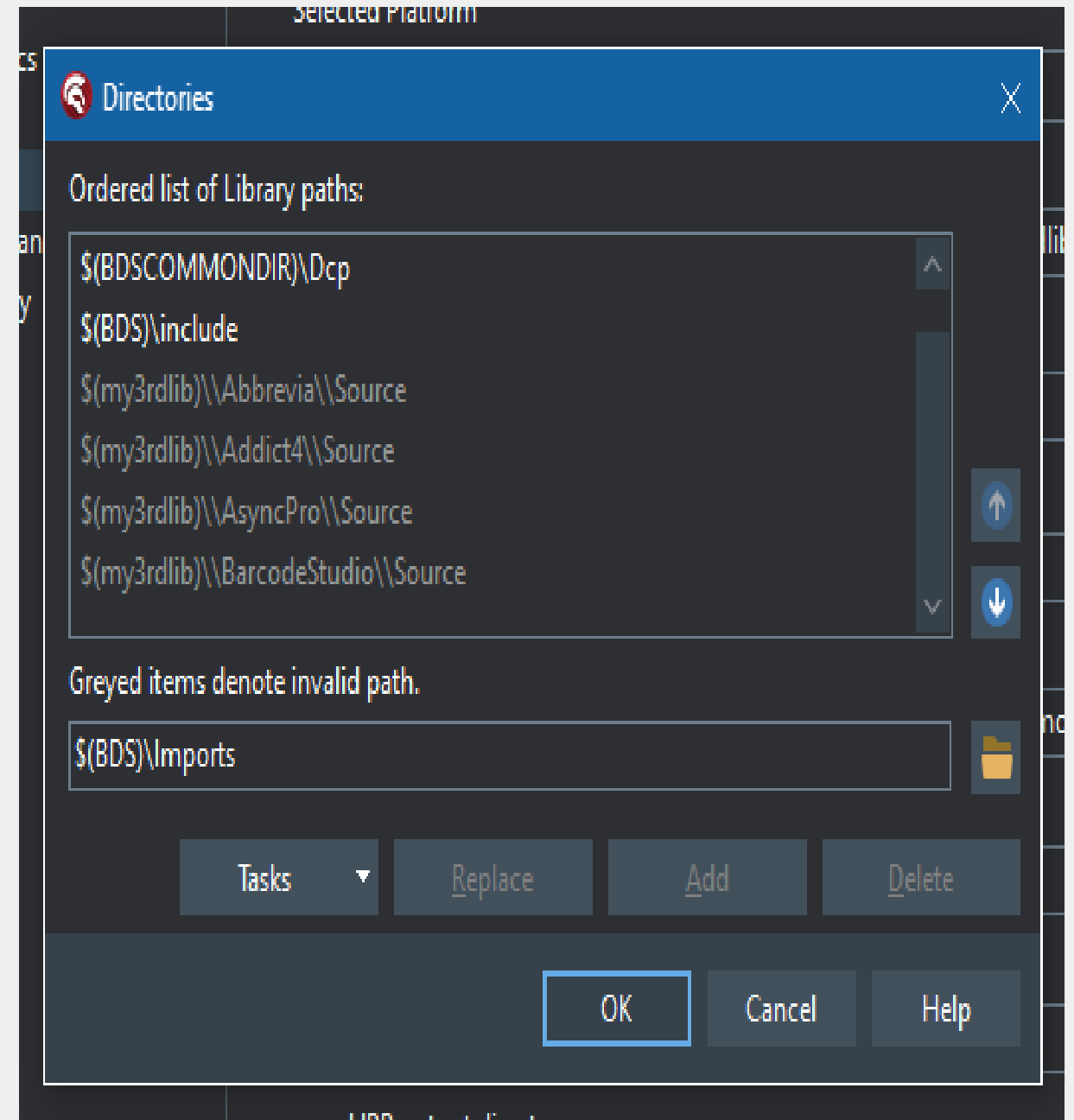
Toda *unit*, por mais básica que ela seja, será composta da seguinte estrutura:

```
unit Exemplo;  
  
interface  
  
uses // Outras units que precisam ser referenciadas  
  
implementation  
  
uses // Outras units que precisam ser referenciadas  
  
initialization // Opcional  
  
finalization // Opcional  
  
end.
```

Diretório de pesquisa ou Diretório de biblioteca inválido

O compilador procura por qualquer unidade referenciada no código por instruções uses na pasta do projeto, no caminho de pesquisa do projeto, no caminho da biblioteca e em alguns outros locais. Percebemos que ter pastas inválidas nesses caminhos faz com que o compilador procure repetidamente nesses locais, uma operação que é muito mais lenta, no nível do sistema operacional, para caminhos não existentes.

Imagem: <https://quality.embarcadero.com/browse/RSP-39317>



Unit Aliases

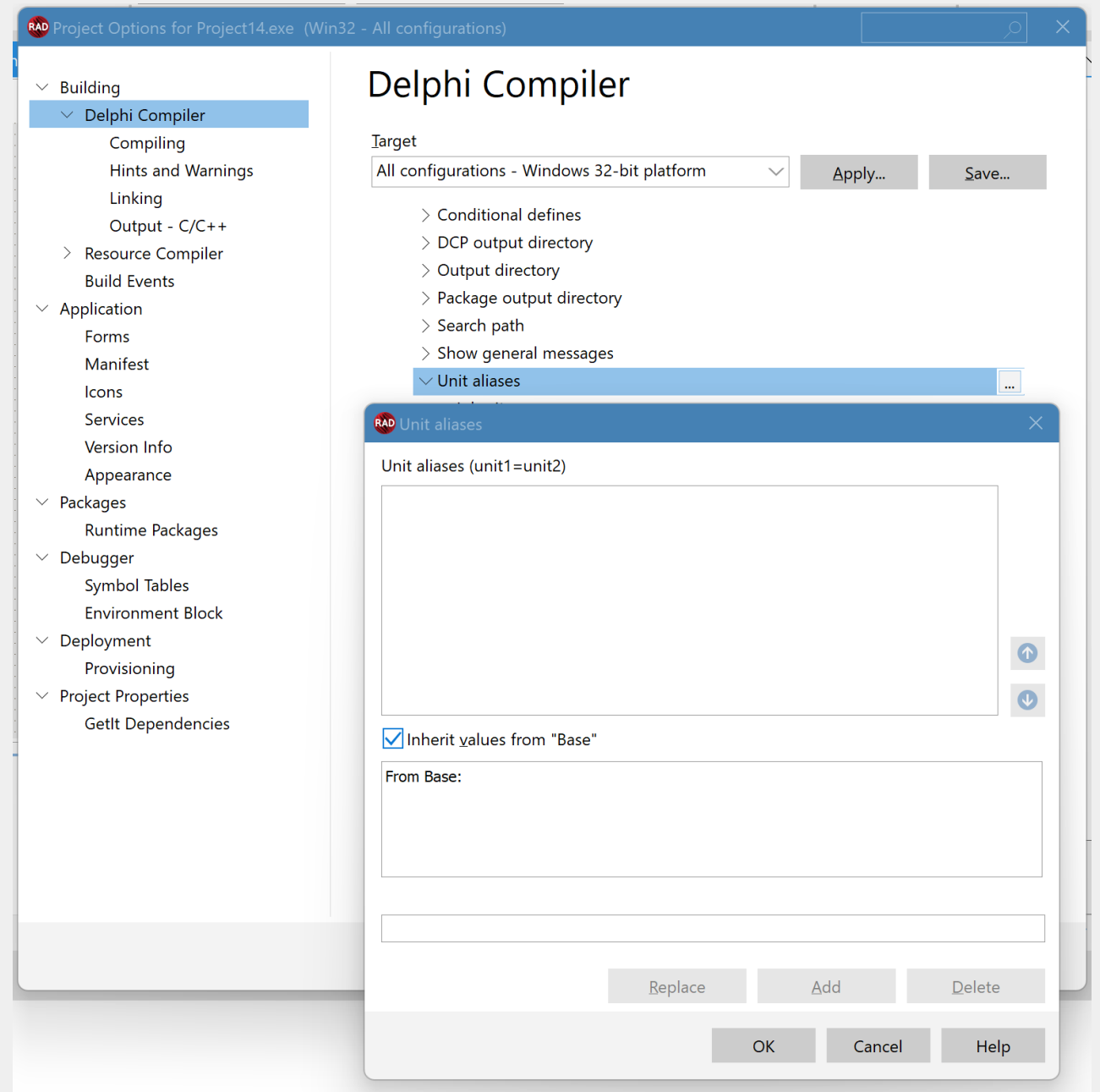
Aliases de unidade oferecem a opção de mapear um nome de unidade em uma declaração uses para um nome de unidade real diferente. Isso pode ser útil ao migrar código antigo e foi usado pelo Delphi ao longo dos anos, quando algumas das unidades principais foram mescladas ou renomeadas.

Por exemplo, em algumas versões iniciais do Delphi você encontraria os seguintes aliases de unidade:

WinTypes=Windows;WinProcs=Windows;DbiProcs=BDE;DbiTypes=BDE;DbiErrs=BDE

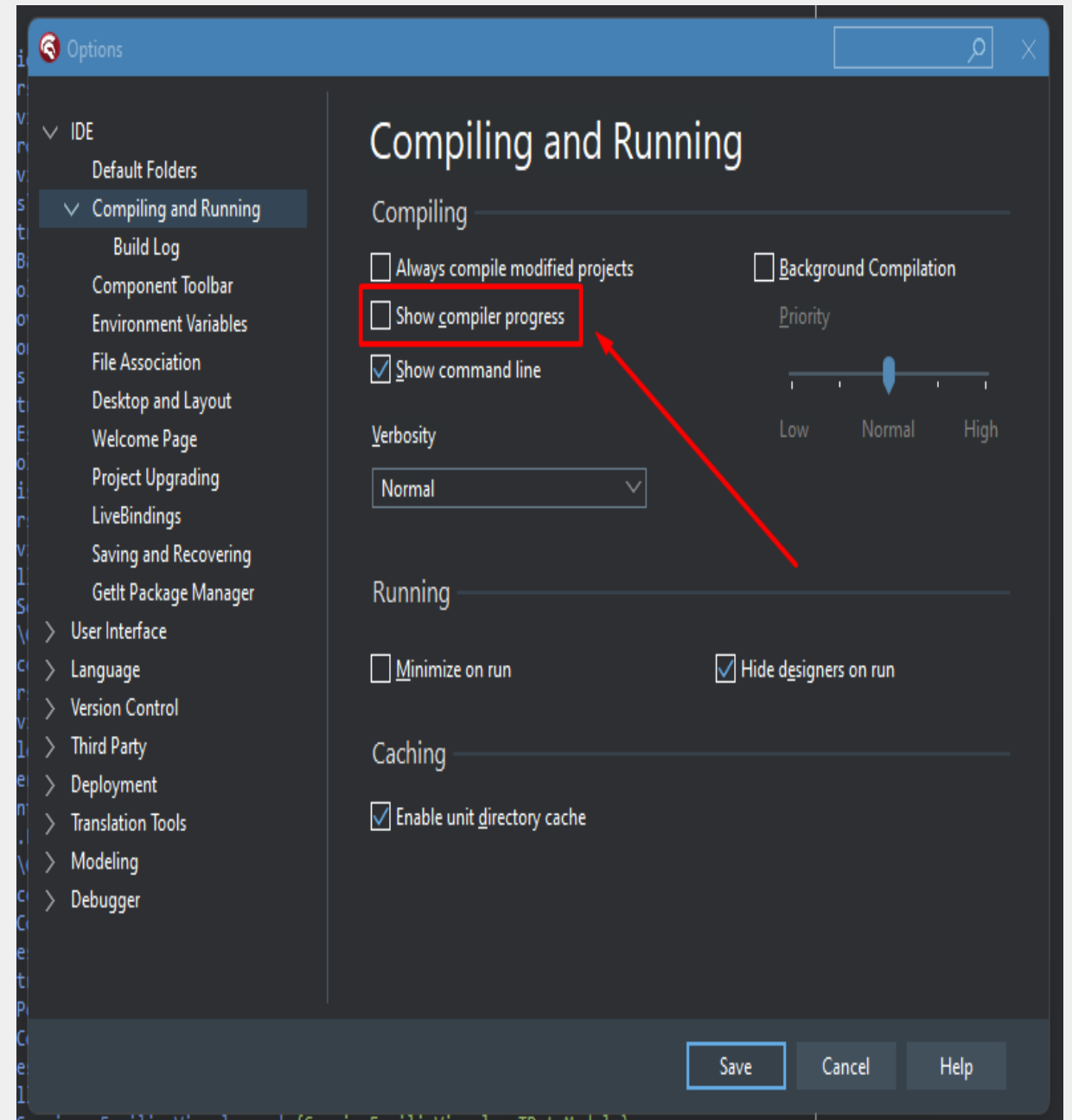
A recomendação é removê-los (possivelmente um por um) e atualizar o código-fonte dos seus projetos com referências às unidades corretas.

Um grande uso de aliases de unidade adiciona algum trabalho extra ao compilador. Embora, até onde sabemos, isso cause apenas uma pequena sobrecarga extra, ainda é um pouco de tempo extra e o esforço também ajuda a manter seu código mais limpo e legível.



Desabilitar visualização do progresso de compilação

Pode diminuir o tempo de compilação em até 30%



Units desnecessárias

As unit declaradas mas não usadas referem-se a unidades que são incluídas na cláusula uses de um programa ou módulo, mas que não são realmente utilizadas no código. Isso pode ocorrer por vários motivos, como a inclusão de unidades por padrão ou a remoção de código que anteriormente dependia dessas unidades.



Uses na Interface x Uses na Implementation

Dependendo de onde a unit for importada, torna ela visível para toda a unit e outras units que a importam, criando dependências de compilação e aumentando o tempo de compilação.

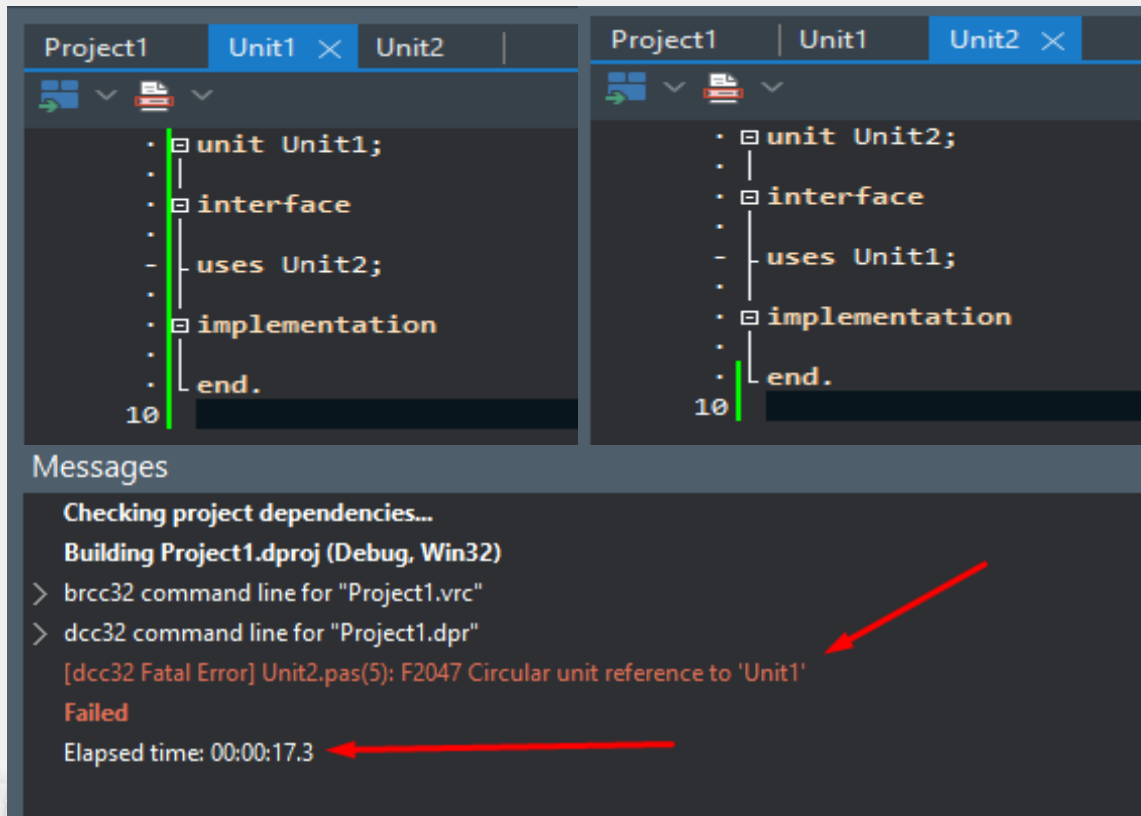


Referência Circular

Referências circulares no Delphi ocorrem quando duas ou mais units ou classes se referenciam mutuamente, criando um ciclo que impede a correta compilação do código. Isso pode acontecer, por exemplo, quando a Unit A usa a Unit B, e a Unit B, por sua vez, usa a Unit A.



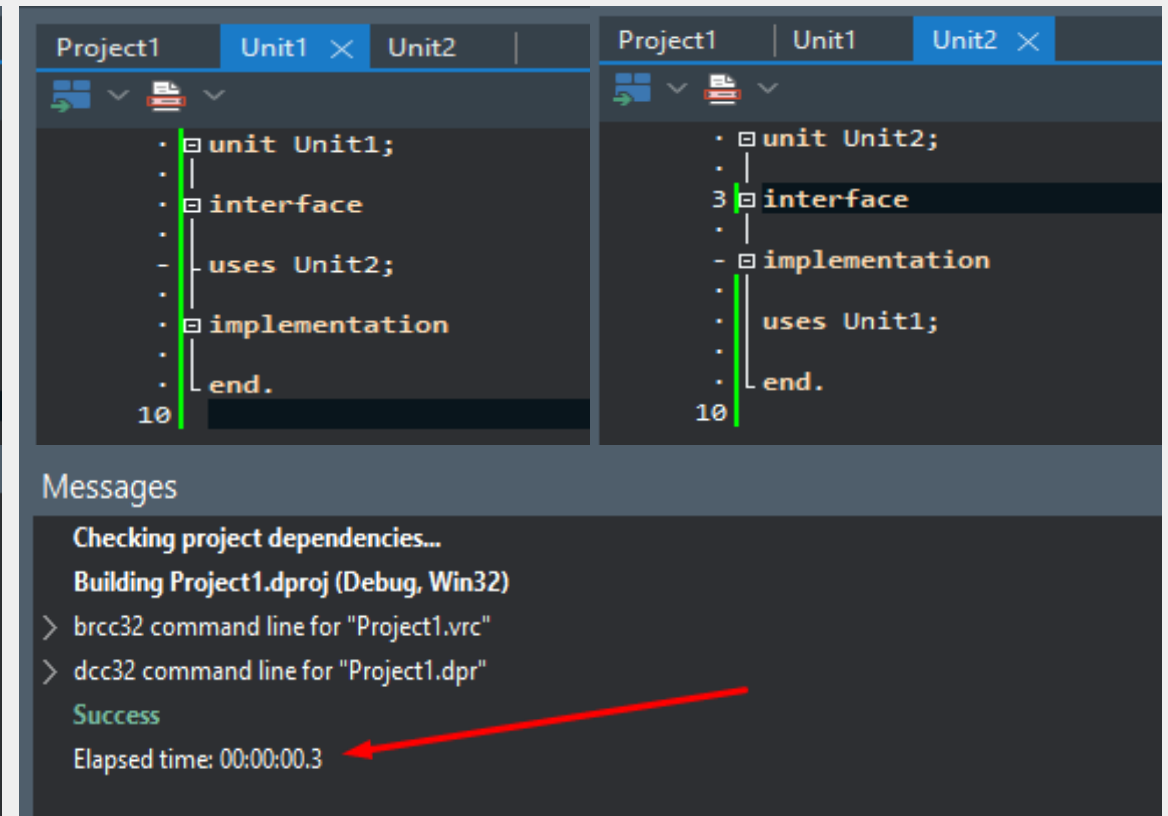
Referência Circular



The screenshot shows the Delphi IDE with two units, Unit1 and Unit2, open. Unit1 contains an interface and an implementation section. Unit2 contains an interface that uses Unit1 and an implementation section. The Messages window at the bottom shows the following output:

```
Checking project dependencies...
Building Project1.dproj (Debug, Win32)
> brcc32 command line for "Project1.vrc"
> dcc32 command line for "Project1.dpr"
[dcc32 Fatal Error] Unit2.pas(5): F2047 Circular unit reference to 'Unit1'
Failed
Elapsed time: 00:00:17.3
```

Red arrows point from the error message to the corresponding code in Unit2.pas, specifically to the `uses Unit1;` line in the interface section.



The screenshot shows the Delphi IDE with the same two units, Unit1 and Unit2, open. The Messages window at the bottom shows the following output:

```
Checking project dependencies...
Building Project1.dproj (Debug, Win32)
> brcc32 command line for "Project1.vrc"
> dcc32 command line for "Project1.dpr"
Success
Elapsed time: 00:00:00.3
```

Red arrows point from the success message to the corresponding code in Unit2.pas, specifically to the `uses Unit1;` line in the implementation section.

**Para combater
o inimigo,
precisamos de
boas armas ...**



GExperts

O GExperts é um conjunto de ferramentas gratuitas e de código aberto projetadas para aumentar a produtividade dos programadores Delphi e C++Builder, adicionando várias funcionalidades ao IDE <https://www.gexperts.org/>. Desenvolvido pela comunidade, ele oferece uma série de recursos úteis



Grep Results

File List View Replace Help

286 AAction.ImageIndex := NTAServices.AddMasked(ReadyBitmap, ReadyBitmap.Transparent);

288 AAction.ImageIndex := NTAServices.AddMasked(ReadyBitmap, GXTransparencyColor);

+ Source\Framework\GX_ClassMgr.pas 3 matches

+ Source\Grep\GX_GrepBackend.pas 21 matches

+ Source\Grep\GX_GrepExpert.pas 20 matches

- Source\Grep\GX_GrepSearch.dfm 3 matches

167 object lblMask: TLabel

174 FocusControl = cbMask

185 object cbMask: TComboBox

+ Source\Grep\GX_GrepSearch.pas 15 matches

+ Source\OpenFile\GX_OpenFile.pas 7 matches

+ Source\RenameComponents\GX_ComprRenameConfig.pas 1 matches

BitmapName := AAction.Name + GxBitmapSuffix;

if ReadyBitmap.Transparent then

 AAction.ImageIndex := NTAServices.AddMasked(ReadyBitmap, ReadyBitmap.TransparentColor);

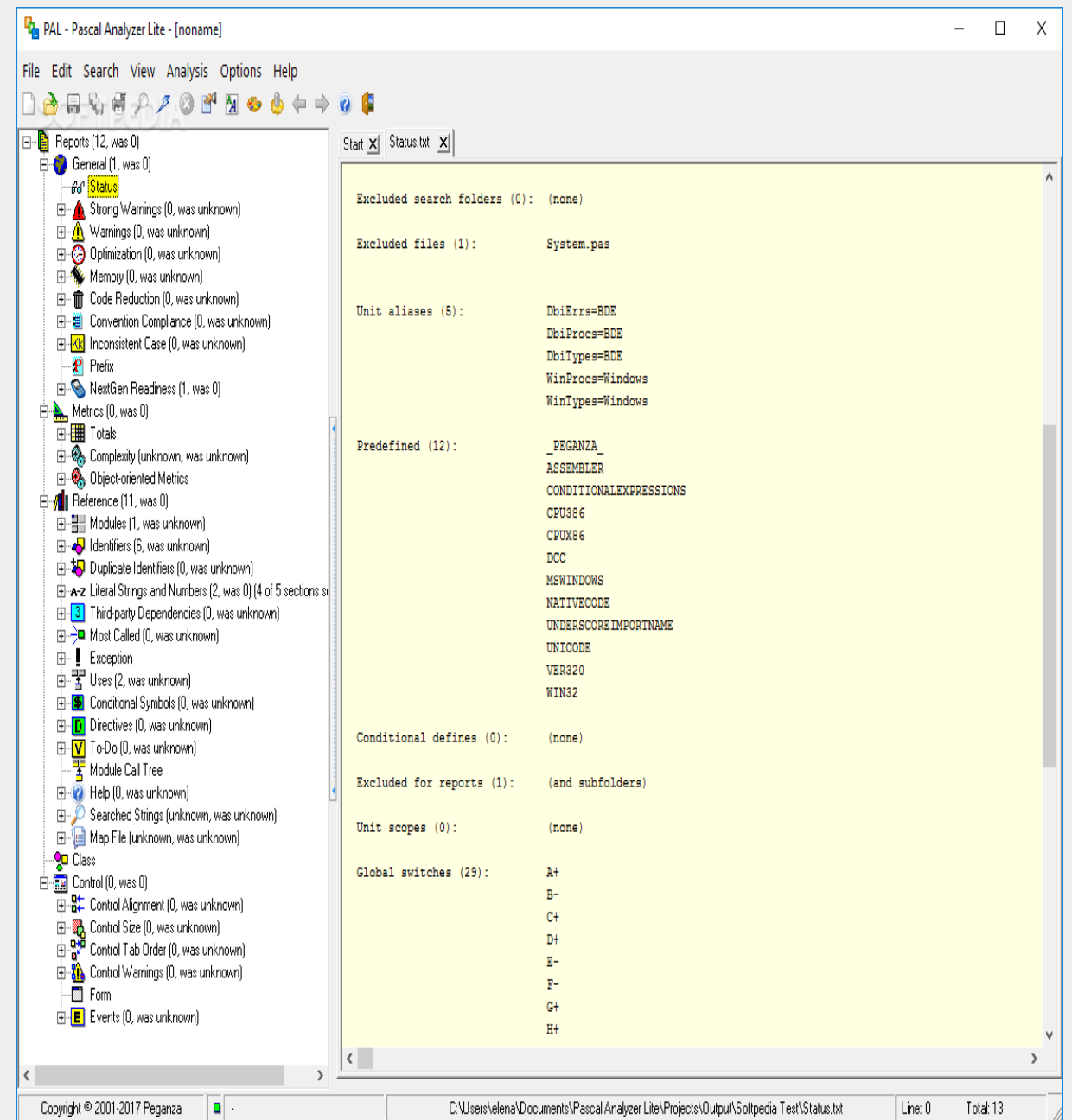
else

 AAction.ImageIndex := NTAServices.AddMasked(ReadyBitmap, GXTransparencyColor, BitmapName);

Searched 346 files in 3.09 seconds for "Mask" 471 matches in 31 files

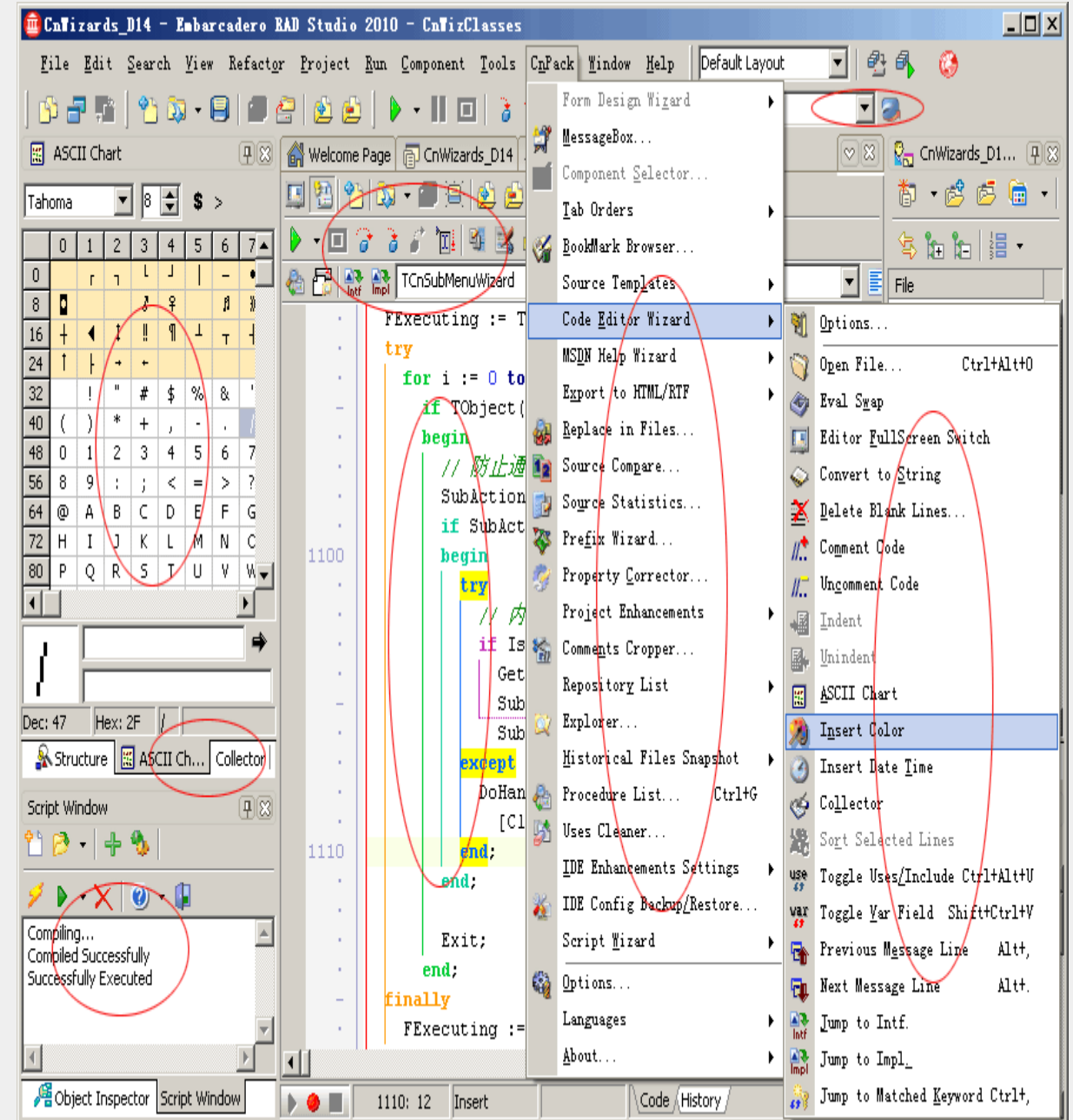
(PAL) – Pascal Analyzer Lite

O Pascal Analyzer Lite (PAL) é uma versão gratuita e simplificada do Pascal Analyzer, uma ferramenta de análise de código estático para Delphi e Pascal. Desenvolvido pela Peganza, o PAL ajuda a analisar, documentar, depurar e otimizar o código-fonte.



CnWizard

O CnWizards (CnPack IDE Wizards) é um conjunto de ferramentas de código aberto para Delphi e C++ Builder, desenvolvido pela equipe CnPack. Ele visa melhorar a eficiência dos desenvolvedores, oferecendo uma série de assistentes e melhorias para o ambiente de desenvolvimento integrado (IDE).



DelphiLint

O DelphiLint é um pacote de IDE para RAD Studio que oferece análise de código em tempo real e linting, utilizando o SonarDelphi <https://github.com/integrated-application-development/delphilint>. Ele ajuda a identificar e corrigir problemas no código antes mesmo de serem confirmados, proporcionando um ciclo de feedback mais curto.



The screenshot displays the DelphiLint IDE interface. The top pane shows the source code of a Delphi unit, `FooUnit.pas`, with several linting issues highlighted in yellow. The bottom pane shows the DelphiLint results, listing six issues with their locations and descriptions. The right pane provides detailed information about the selected issue, including its reliability, why it is an issue, how to fix it, and resources.

DelphiLint

FooUnit.pas
6 issues

- Move this import to the implementation section.
(5, 5) • New issue
- Move this import to the implementation section.
(5, 21) • New issue
- Rename this constructor to match the expected naming convention
(9, 10) • New issue
- Avoid mixing names (found: "makeNew" expected: "MakeNew").
(14, 20) • New issue
- Remove this unused local variable.
(16, 2) • New issue
- Change this formatting argument to match the type of the corresponding specifier (%s).
(18, 45) • New issue

Intentionality rule | Logical
'Format' calls should be supplied arguments of the correct type

Reliability: ●

Why is this an issue?

Passing arguments of the wrong type to a call to `System.SysUtils.Format` raises a runtime exception. In particular:

- `%d`, `%u`, and `%x` must be an integer value
- `%e`, `%f`, `%g`, `%n`, and `%m` must be a floating-point value
- `%p` must be a pointer value
- `%s` must be a string, character, `PChar`, static array of `Char`, or variant value
- Floating-point width and precision must be integer values

How to fix it

Ensure that the arguments to the `Format` call match the specifiers in the format string:

```
Format('Id got %.2f percent on the test.', ['Bob', 74.599]);
```

```
Format('%s got %.2f percent on the test.', ['Bob', 74.599]);
```

Resources

- [RAD Studio API documentation System.SysUtils.Format](#)

Idle (last analysis succeeded at 4:31pm)

Dúvidas ???



Melhores da Conference - 2023



Não deixe
essa
sequência
acabar 🔥

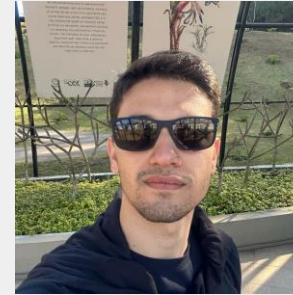


Embarcadero Conference 2024


Inovação faz parte do nosso DNA!





Quer me ver na
#ECON25?
Acesse o QRCode
e avalie minha palestra!




Vinicius Abreu

 [@vinib.abreu](https://www.instagram.com/vinib.abreu)

 [/vinicius-abreu-batista](https://www.linkedin.com/company/vinicius-abreu-batista)

 abreuvinicius87@gmail.com

 [\(11\) 9 9999 9999](tel:(11)999999999)

