



Keynote: Delphi, cosa ci attende nel futuro?

Marco Cantù, RAD Studio PM, Embarcadero
marco.cantu@embarcadero.com

Today Presenter



Marco Cantù

Product Lead, IDE & Tooling at Idera

- Includes being RAD Studio and Delphi Product Manager
- Contact: marco.cantu@embarcadero.com

Delphi Guru

- Book author (20+)
 - Mastering Delphi
 - Object Pascal Handbook
 - Expert Delphi 2nd edition (Focused on FireMonkey)
- Presenter at conferences worldwide, for 30+ years

Embarcadero is part of **Idera, Inc.**

IDE & Tooling division:

- Embarcadero: Delphi, C++Builder, RAD Studio, InterBase
- UltraEdit: UltraEdit, UltraCompare, UltraFTP
- Whole Tomato: Visual Assist



whole tomato
Visual Assist

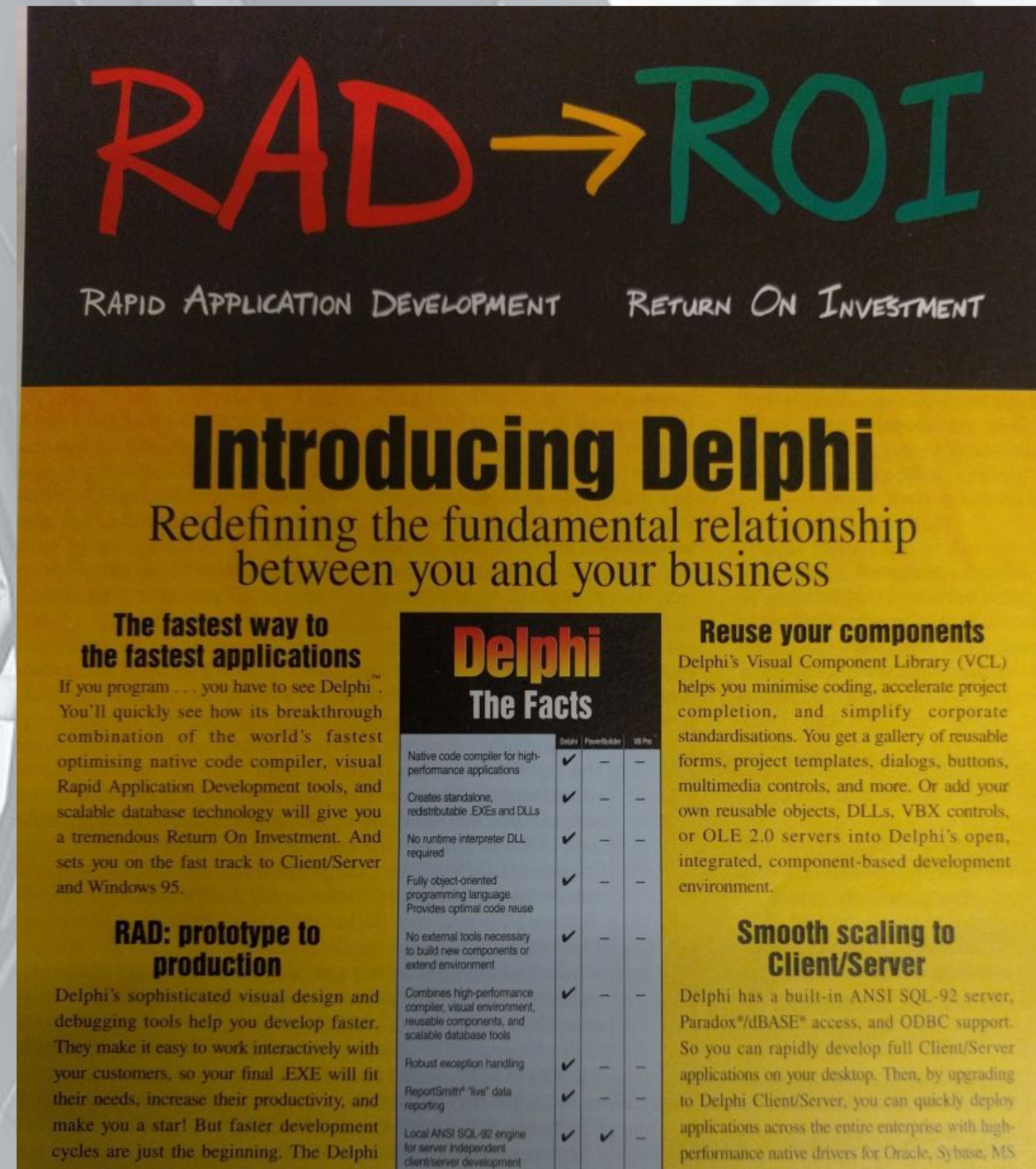
Where is Delphi today and how did we get here?



Delphi = RAD from Day 1

- The fastest way to build the faster applications
- RAD: prototype to production
- Reuse your components
- Smooth scaling to Client/Server

***This has been true for 30 years:
nothing wrong with it,
we are actually proud of it!***



RAD → ROI

RAPID APPLICATION DEVELOPMENT RETURN ON INVESTMENT

Introducing Delphi

Redefining the fundamental relationship between you and your business

The fastest way to the fastest applications

If you program . . . you have to see Delphi™. You'll quickly see how its breakthrough combination of the world's fastest optimising native code compiler, visual Rapid Application Development tools, and scalable database technology will give you a tremendous Return On Investment. And sets you on the fast track to Client/Server and Windows 95.

Delphi The Facts

	Delphi	PascalBuilder	VB Pro
Native code compiler for high-performance applications	✓	—	—
Creates standalone, redistributable EXEs and DLLs	✓	—	—
No runtime interpreter DLL required	✓	—	—
Fully object-oriented programming language. Provides optimal code reuse	✓	—	—
No external tools necessary to build new components or extend environment	✓	—	—
Combines high-performance compiler, visual environment, reusable components, and scalable database tools	✓	—	—
Robust exception handling	✓	—	—
ReportSmith® "live" data reporting	✓	—	—
Local ANSI SQL-92 engine for server independent client/server development	✓	✓	—

Reuse your components

Delphi's Visual Component Library (VCL) helps you minimise coding, accelerate project completion, and simplify corporate standardisations. You get a gallery of reusable forms, project templates, dialogs, buttons, multimedia controls, and more. Or add your own reusable objects, DLLs, VBX controls, or OLE 2.0 servers into Delphi's open, integrated, component-based development environment.

Smooth scaling to Client/Server

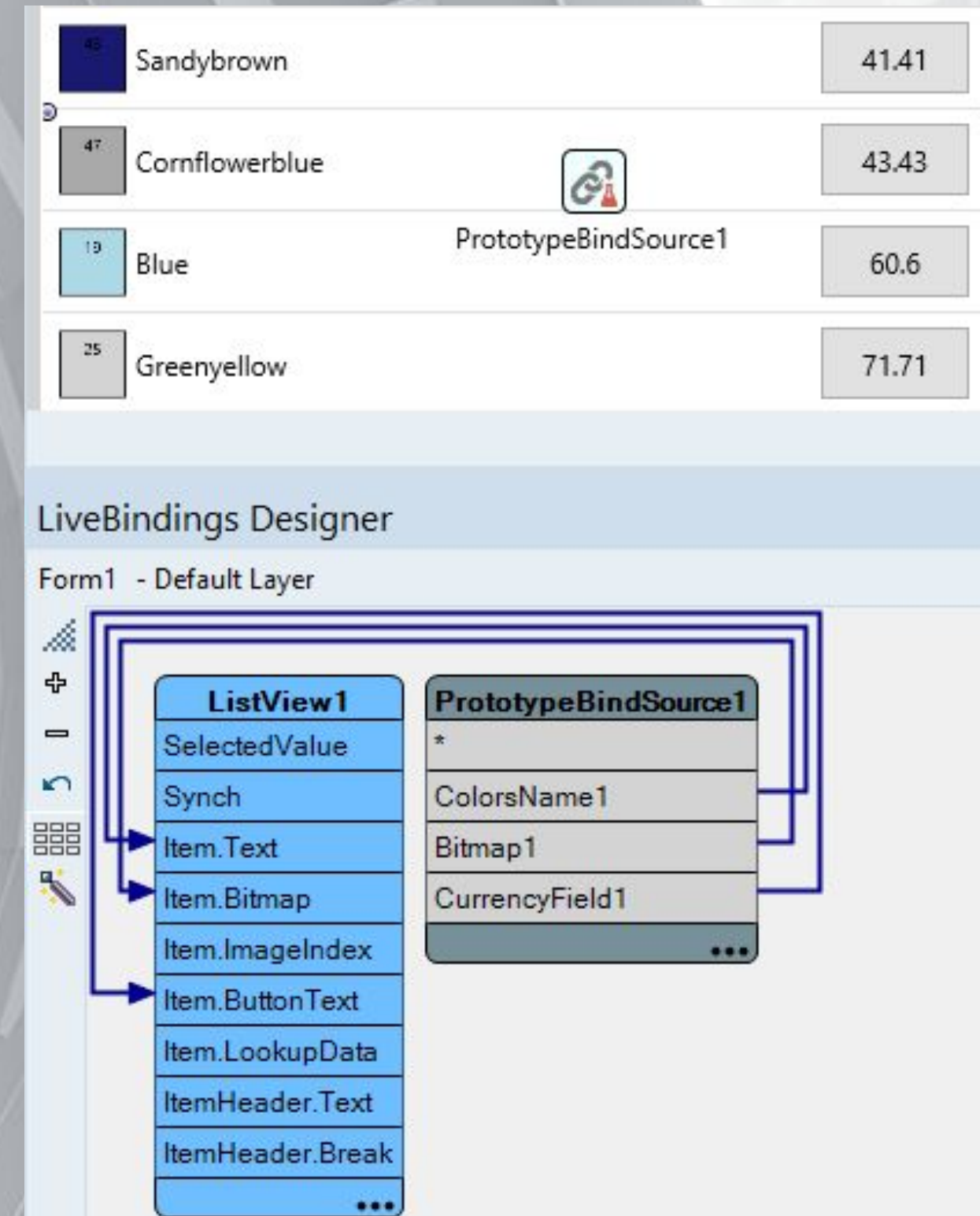
Delphi has a built-in ANSI SQL-92 server, Paradox®/dBASE® access, and ODBC support. So you can rapidly develop full Client/Server applications on your desktop. Then, by upgrading to Delphi Client/Server, you can quickly deploy applications across the entire enterprise with high-performance native drivers for Oracle, Sybase, MS

RAD: prototype to production

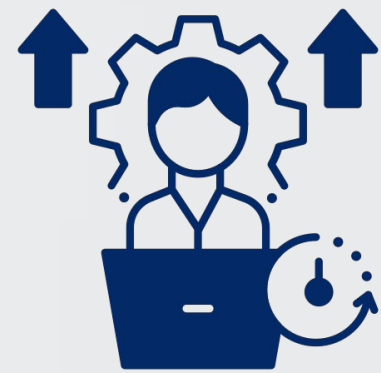
Delphi's sophisticated visual design and debugging tools help you develop faster. They make it easy to work interactively with your customers, so your final .EXE will fit their needs, increase their productivity, and make you a star! But faster development cycles are just the beginning. The Delphi

Delphi is still RAD today

- The fastest way to build the faster applications
 - For multiple platforms
- RAD: prototype to production
 - Xplat - On device preview
- Reuse your components
 - And leverage third party vendors
- Smooth scaling to Client/Server
 - And multi tier with RAD Server



What is Great in Delphi?



Developer Productivity

Shipping is a feature.
Get to market 5x faster.



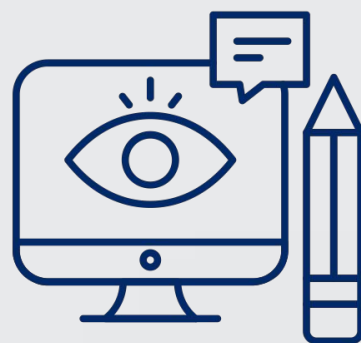
Fast Native Apps

Native compilers give your apps the speed they need.



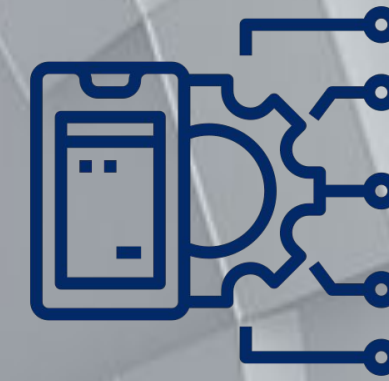
Strong Community

Technology Partners,
MVPs, trainers, authors
and developers –
worldwide



Visual Designers

Forget prototyping in
another tool, just
design it and finish it
in one IDE



Platform API Access

Gives you access to all
the platform APIs on all
platforms.

Database Access

Key to Delphi's initial
design, database
access is
easy and fast.



Backward Compatibility

You have an investment
in your code, keep your
code relevant.

Where is Delphi? Providing value to the industry, silently

- Easy, native, multi-device, and a top Windows solution
- Stability of apps with little or no dependencies
- Single code base to create apps for multiple platforms
- Huge availability of components
- Very easy to use language that any developer can easily understand, yet also very powerful and supports all the modern constructs developers could need
- Delphi is very ‘silently successful’ and is in use in a great number of places!
 - We have so many case studies, but very hidden... Schiphol Airport, New Pier development Amsterdam, F1 at Miniature Wunderland, designing CPUs, the best music software, Long Beach Containers Terminal, Tax returns, government mandated electronic invoicing... and so many more

The Power of Simplicity

- “Power and simplicity, expressiveness and readability, great for learning and for professional development alike, these are some of the traits of today's Object Pascal”
- “A tool for all trades, with compilers and development tools embracing the mobile era. A language ready for the future, but with solid roots in the past”

PIONEERING SIMPLICITY

THE FASCINATING HISTORY
OF *DELPHI* AND *PASCAL*

MARCO GEUZE

Buy on [Amazon.com](https://www.amazon.com)

Native => Compiled to the Metal

- Delphi today includes 8 compilers!
- Stability of apps with little or no dependencies
- XCopy deployment
- No runtime to install, maintain, update
- More performance, more security

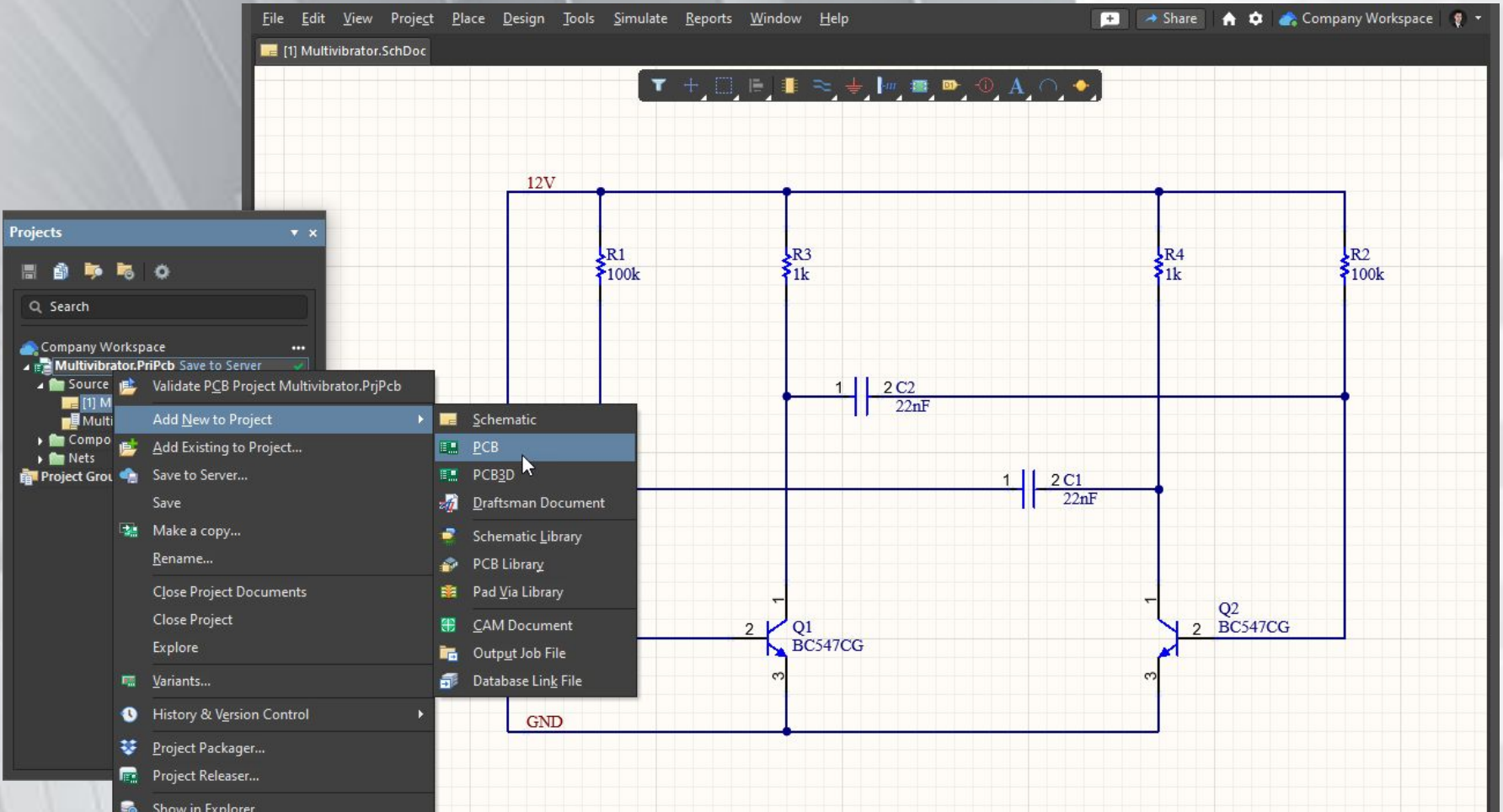
VCL: The Most Effective UI library for Windows

- What's Microsoft offering these days for building UIs on Windows? It's so confusing you hardly know. C# WinUI3? MFX and C++? The good old WinForms? React and JavaScript?
- Microsoft moved from UWP back to native apps, and migration from UWP back to native: Desktop bridger, Project Reunion. Windows App SDK
- We offer the UI library with the best coverage of Windows APIs + integration of WinRT APIs, UWP support, WebView 2 (Chromium Edge), and more. Plus 30 years of compatibility!

Delphi is very 'silently successful'



Delphi is very ‘silently successful’



Technology | ADAS, AV & Safety | Partnerships & M&A | Software-Defined Vehicle

Japan chipmaker Renesas to buy software firm Altium for \$5.9 bln



New Features in 12.X

For Delphi

Highlights of RAD Studio 12 Athens

- Walking over the features summary, mostly to highlight the current product directions
- Three 12 Athens releases, over the last year



Highlights of Delphi 12 Athens

- **Delphi Compilers Specific**

- String literals improvements
- Mobile and other Platforms improvements (Android API level 34 support)
- DelphiLSP quality improvements
- Delphi 64-bit compilers and tools

Highlights of Delphi 12 Athens: IDE

- **RAD Studio IDE Modernization**

- Multi Device Icons Generator
- Editor split views
- Focus Mode and Scrollbar Annotations
- New VCL Designers
- JSON Mapping Wizard

- **AI in IDE**

- Smart CodeInsight (more later)

Highlights of Delphi 12 Athens Libraries

- VCL
 - MDI and Tab based UIs
- FireMonkey
 - Skia UI Controls and rendering and graphic formats and way more (and VCL as well)
- Data
 - JSON Data Binding Wizard
 - FireDAC Secure Coding
 - Sqids in RAD Server
 - WebStencils templates

What's New in 12.2?

- **NEW FEATURES**

- Modernized C++ Compiler and Toolchains
- Delphi 64-bit compilers and tools
- Smart CodeInsight
- Focus Mode and Scrollbar Annotations
- WebStencils

- **QUALITY**

- Improved Mobile Development
- Visual Assist for C++ Builder Enhancements
- Quality, Quality, Quality



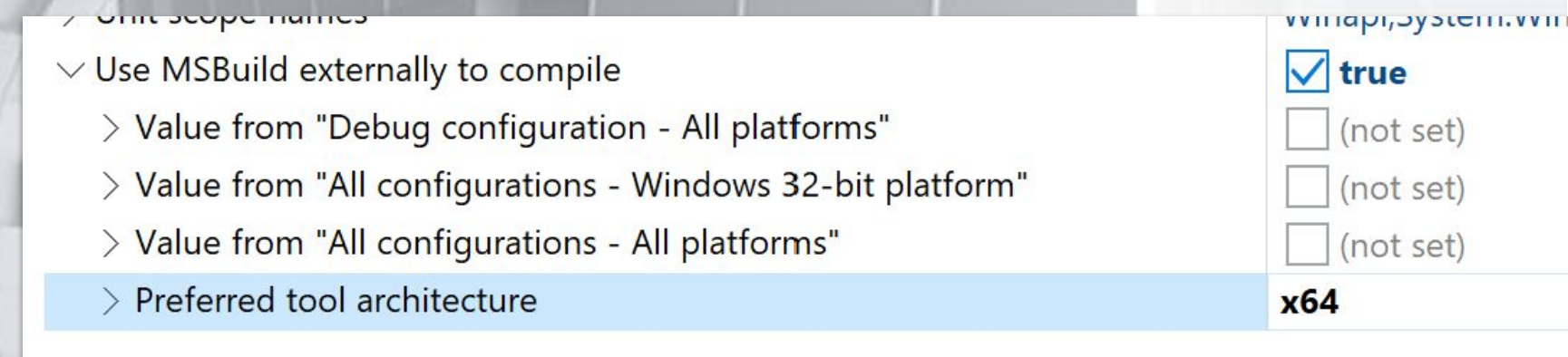
64-bit Version of Delphi Windows compilers

64-bit binaries (large memory address) for Delphi Windows compilers for both Win32 and Win64 **command line compilers**

- Can be enabled in the IDE, when using *external MSBuild*
- Allows building **very large** applications as a single executable
- Available only in **Enterprise** and **Architect**

64-bit binary (large memory address) version of the **DelphiLSP** executable

- Can be activated in the IDE
- Also available only in Enterprise and Architect



“Enterprise only” stirred negative criticism, but larger companies should be on Enterprise

Smart CodeInsight: AI Powered Coding

Use **AI LLMs for coding** in the RAD Studio IDE

- An **open architecture** for AI plugins

Ready-to-use support for **three online solutions**

- OpenAI
- Gemini by Google
- Claude by Anthropic

and one **offline solution**

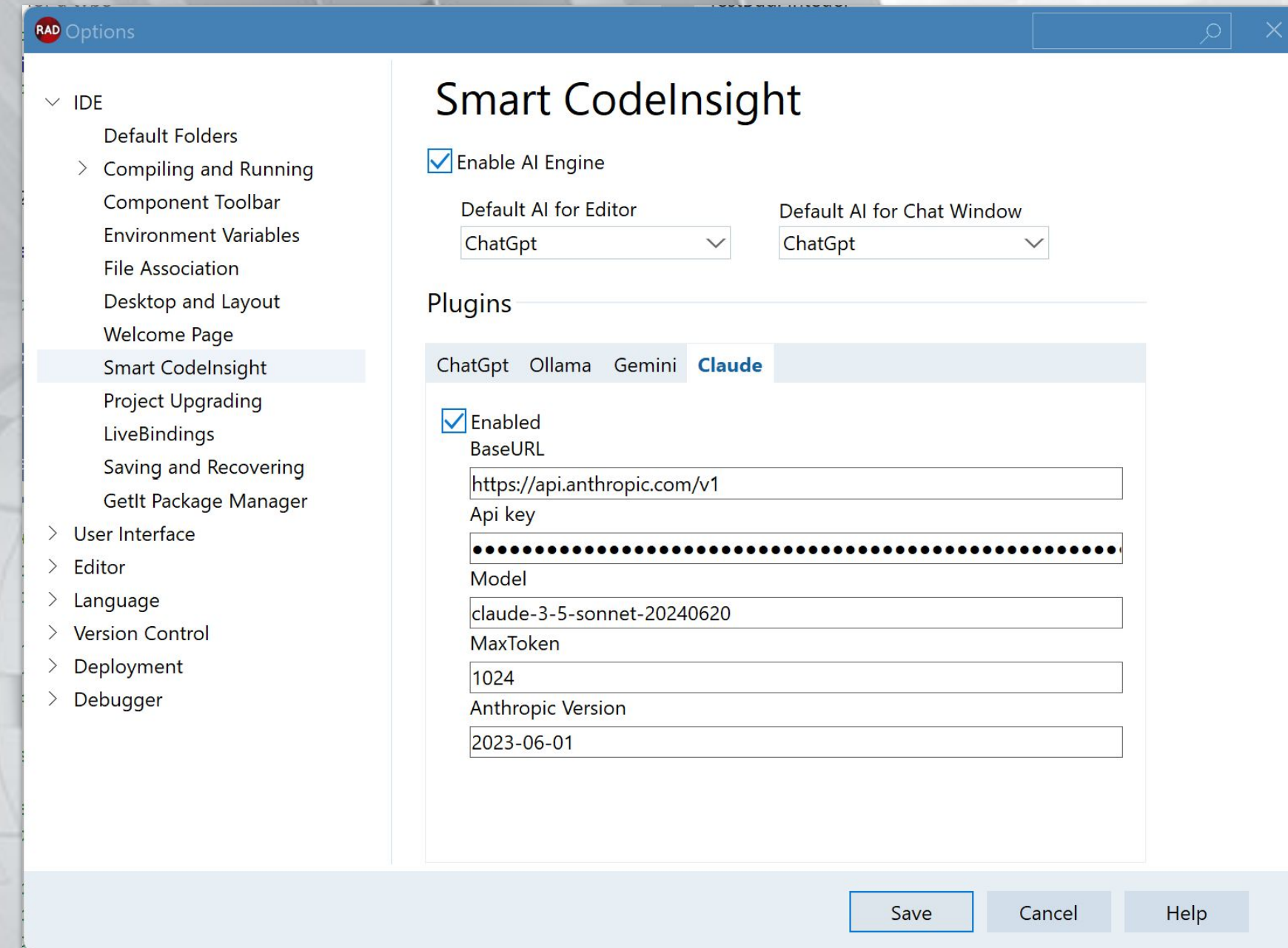
- Ollama

Ready to use docker image, works in WSL2



Smart CodeInsight: AI Powered Coding

- Everything is **optional** and **disabled** by default
- **Multiple** providers, so you can **choose** which one or ones **you want** to enable
- A locally installed, **offline** solution for **maximum privacy**
- You decide **if** and **how** to use **AI**
... and you can even **create additional plugins** for your preferred solution



Smart CodeInsight: AI Powered Coding

IDE: chat window and editor commands

Project21 - RAD Studio 12 - Unit25

File Edit Search View Refactor Project Run Component

Smart CodeInsight

Answer>
In Delphi, "strict private" is a visibility specifier used in object-oriented programming to control access to class members (fields, methods, properties). It is one of the most restrictive visibility levels available in Delphi. Here's what you need to know about "strict private":

1. Scope: Members declared as "strict private" are only accessible within the class where they are declared.

2. Inheritance: Strict private members are not inherited by subclasses and cannot be accessed by descendant classes.

3. Unit scope: Unlike regular "private" members, strict private members are not accessible even within the same unit where the class is defined.

4. Usage: It's typically used for implementation details that should be completely hidden from outside the class and its descendants.

Claude>What's the meaning of strict private in Delphi?

Claude

Structure Smart CodeInsight

function TestBug: Integer;
var
P: Pointer;
begin
GetMem (P, 100);
Result := 100;
end;

//***** Find Bugs *****
1. Memory Leak: The allocated memory isn't freed after been used. After calling GetMem the programmer should call FreeMem(P) before the function ends to avoid leaking memory.
2. Uninitialized pointer: The allocated memory is not initialized. In order to avoid it is recommended to initialize the memory after allocating it. The allocated memory using FillChar or ZeroMemory functions.
3. Unused pointer: The pointer 'P' was allocated memory but it's not used anywhere in the function.
4. No error handling: The function GetMem may fail if the system cannot provide the requested memory. This failure is not handled in the code.

The improved version of the code could look like this:
``delphi
function TestBug: Integer;
var
P: Pointer;
begin
GetMem(P, 100);
try
// use 'P' for some operations here...
// ...
FillChar(P^, 100, 0);
Result := 100;
finally
FreeMem(P);
end;
end;
``

Smart CodeInsight


Open File at Cursor Ctrl+Enter
Topic Search F1
Complete Class at Cursor Ctrl+Shift+C
Add To-Do Item... Ctrl+Shift+T
Cut Ctrl+X
Copy Ctrl+C
Paste Ctrl+V
Toggle Bookmarks
Goto Bookmarks
Clear Bookmarks
Debug
Toggle Comment Ctrl+/
Fold
Unfold
Refactor
Find
Find Symbols
Sync Prototypes Ctrl+Alt+Shift+P
Surround
Add to Repository
View Messages
Read Only
Editor Options

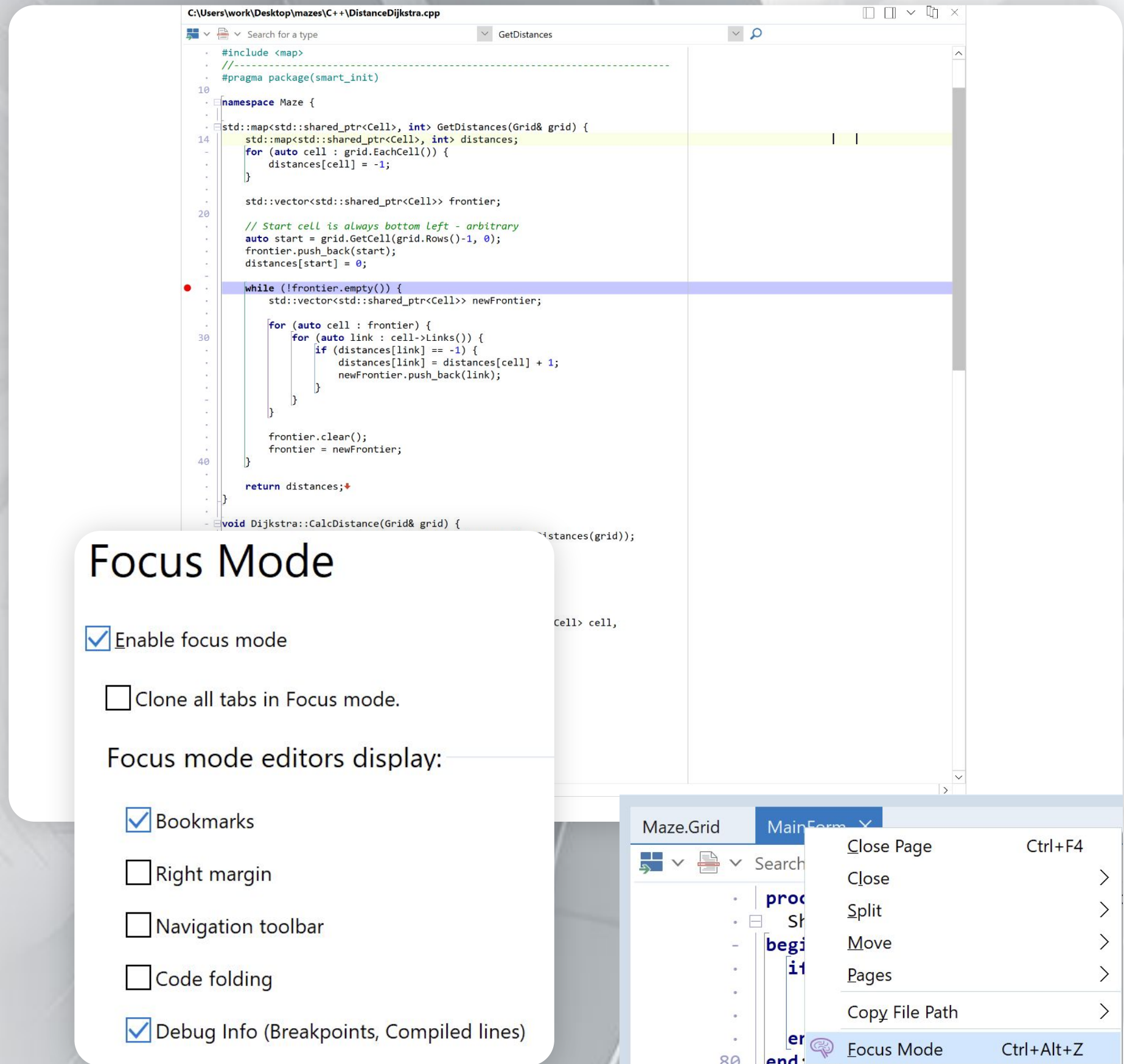
AI Chat
Find Bugs
Explain the Code
Add Comment
Complete the Code
Optimize Code
Add Unit Test
Convert to Assembly
Convert to Delphi
Convert to C++

Editor: Focus Mode

Progression from 12.1's split views,
and known as 'zen mode' in other
IDEs

Focus Mode leaves the IDE with **only**
your code editor

- Removes other UI chrome
- Multiple side by side editors
- 'Code only'
- Distraction free... *focus!* 



Z:\projects\Embarcadero\mazes\mazes\Delphi\MainForm.pas

Search for a type

TForm1.RecreateGrid

```
procedure TForm1.edtCellSizeKeyUp(Sender: TObject; var Key: Word; var KeyChar: Char;
  Shift: TShiftState);
begin
  if Key = vkReturn then begin
    RecreateGrid;
    PaintGrid;
  end;
80 end;

procedure TForm1.FormShow(Sender: TObject);
begin
  PaintGrid;
end;

procedure TForm1.RecreateGrid;
var
  R, C : Integer;
90 begin
  C := Trunc(ImageViewer.Width / CellSize) - 1;
  R := Trunc(ImageViewer.Height / CellSize) - 1;

  // Limit the small size
  C := Max(C, 2);
  R := Max(R, 2);

98 FGrid := CreateGrid(R, C);
end;
100
function TForm1.CellSize : Integer;
begin
  Result := 50;
  TryStrToInt(edtCellSize.Text, Result);
end;

procedure TForm1.PaintGrid;
var
  CellDistPainting : TCellPainting;
110 begin
  CellDistPainting := TCellPainting(cmbDistance.ItemIndex);
  ImageViewer.Bitmap.Assign(FGrid.Repaint(CellSize, CellDistPainting));
  Invalidate;
end;

procedure TForm1.btnNewClick(Sender: TObject);
begin
  RecreateGrid;
  PaintGrid;
120 end;

function TForm1.GetMazeGenerationProc : TMazeGenerationProc;
begin
  Result := nil;
  case cmbAlgorithm.ItemIndex of
  0 : Result := BinaryTree;
  1 : Result := AldousBroder;
  2 : Result := RecursiveBacktracker;
  end;
130 end;

procedure TForm1.btnPlayClick(Sender: TObject);
var
```

Code History


Editor: Scrollbar Annotations

- Editor Scrollbar Annotations
 - See at a glance sections of code modified, with bookmarks, errors, warnings...
- Delphi Inline Assembly Highlighting
- Delphi LSP: code completion optionally in editor colors
- Split Editor Improvements around resizing, stability and IDE toolbars in high DPI quality

```
distances[start] = 0;

while (!frontier.empty()) {
    std::vector<std::shared_ptr<Cell>> newFrontier;

    for (auto cell : frontier) {
        for (auto link : cell->Links()) {
            if (distances[link] == -1) {
                distances[link] = distances[cell] + 1;
                newFrontier.push_back(link);
            }
        }
    }
}
```



```
procedure TForm1.FormCreate(Sender: TObject);
begin
    Self.
end;
end.
```

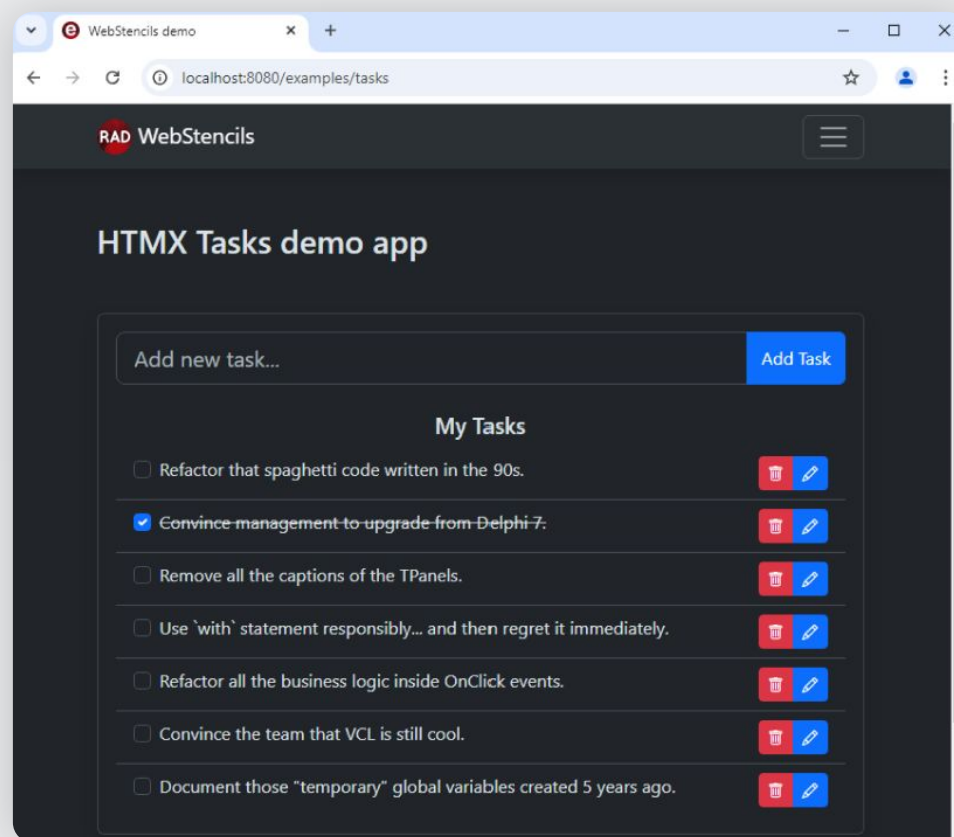
procedure **FormCreate**(Sender: TObject);
procedure **Arrangelcons**;
procedure **Cascade**;
procedure **Next**;
procedure **Previous**;
procedure **Tile**;
property **ActiveMDIChild**: TForm;

✓ Tab Enter — ↑ ↓ ✗ Space " # \$ ' () * + , - . : ; < = > [] { } Esc →

WebStencils Template Library

Server-side script-based integration and processing of HTML files for WebBroker and RAD Server. **You** can develop modern websites

- Based on **any JavaScript library**, CSS, and web technology
- Powered by the data extracted and processed by a RAD Studio **server-side application**



- Foundation for **HTMX** as a web development solution.
- HTMX pages benefit from server-side code generation and hook into REST servers for content updates

WebStencils Template Library

Core Web Scripting Syntax

- Uses **@** for special processing of symbols and **{ }** to define blocks
- **@object.value** to read data from server objects
- **@keywords** for special processing
- Supports simple expressions

```
ProcessorGeneric.AddVar( 'obj2',  
    TSimpleObject.Create( 'marc', 9 ) );
```

```
<h4>Obj2</h4>  
  
@if obj2.ValueBelowTen {  
    <p>@obj2.name <span> has a value of </span>  
        @obj2.value, which is under 10</p>  
}
```

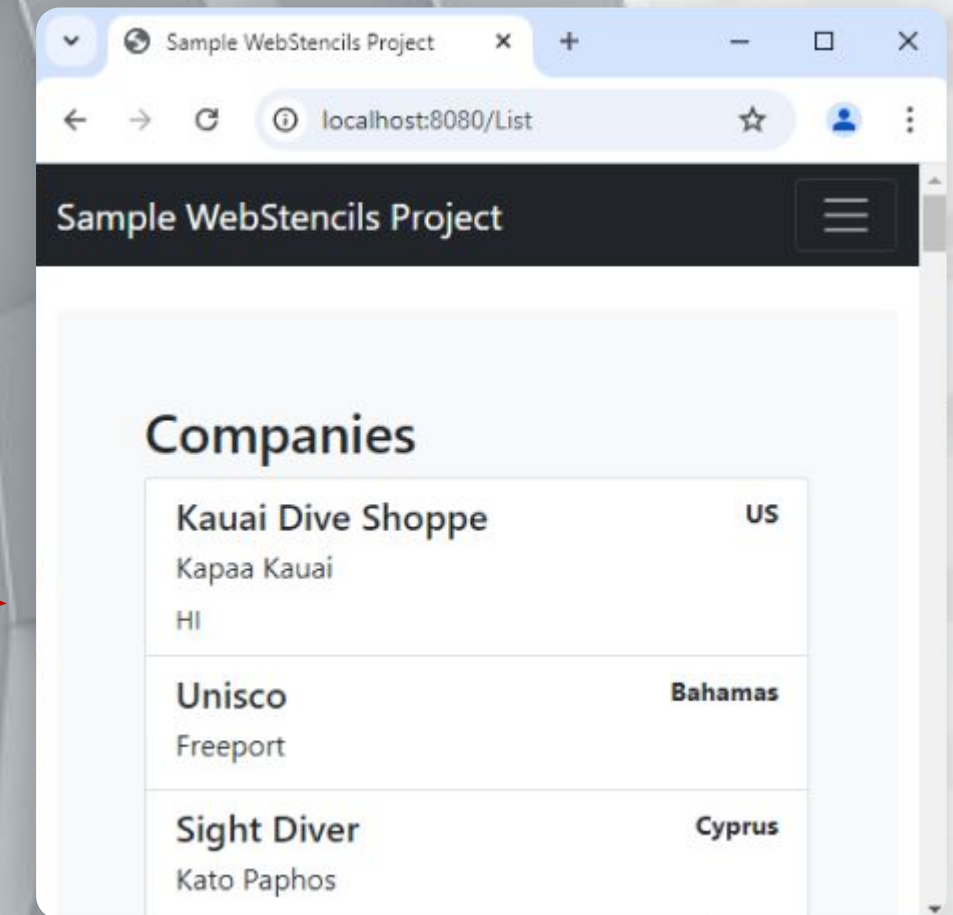
Obj2

marc has a value of 9, which is below 10

WebStencils Template Library

Data access for the script

- RTTI based (but not only)
- `@object.value`
- `OnValue` event handler
- Use objects, lists, or datasets registered as **script variables**
- Define modules variables with attributes



```
<div class="list-group w-65">
  @foreach dataset {
    <a href="/company?id=@loop.custno" class="list-group-item
      list-group-item-action" aria-current="true">
      <div class="d-flex w-100 justify-content-between">
        <h5 class="mb-1">@loop.company</h5>
        <small><b>@loop.country</b></small>
      </div>
      <p class="mb-1">@loop.city</p>
      <small>@loop.state</small>
    </a>
  }
</div>
```


Improved Mobile Development

Significant improvements for Delphi mobile development

Apple platforms

- Linker updated to the most recent Apple version
- Link iOS static libraries written in Xcode, like **Firebase iOS SDK**

Android support

- Options to **compress Dex files** and **compress native library** files
- The **targetSdkVersion** of the manifest file can be configured in the IDE
- Import Android libraries ('**.aar**' files)
- Works with the **Facebook SDK** for Android and **Firebase Analytics**

Questions?

