

Delphi Keynote

Il futuro del nostro strumento di sviluppo





Today's Presenter

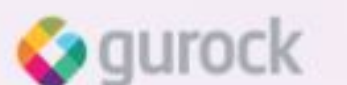
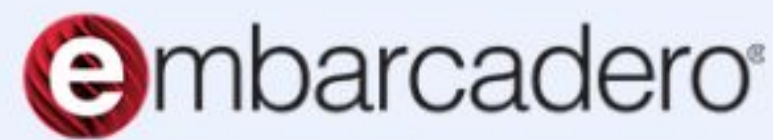
Marco Cantu, Product Lead, IDE & Tooling

marco.cantu@embarcadero.com

Embarcadero & Idera

Embarcadero is

- Owned by Idera, www.ideracorp.com
- Part of the Dev Tools division



What is Delphi?

The ultimate IDE for building

- multi-platform

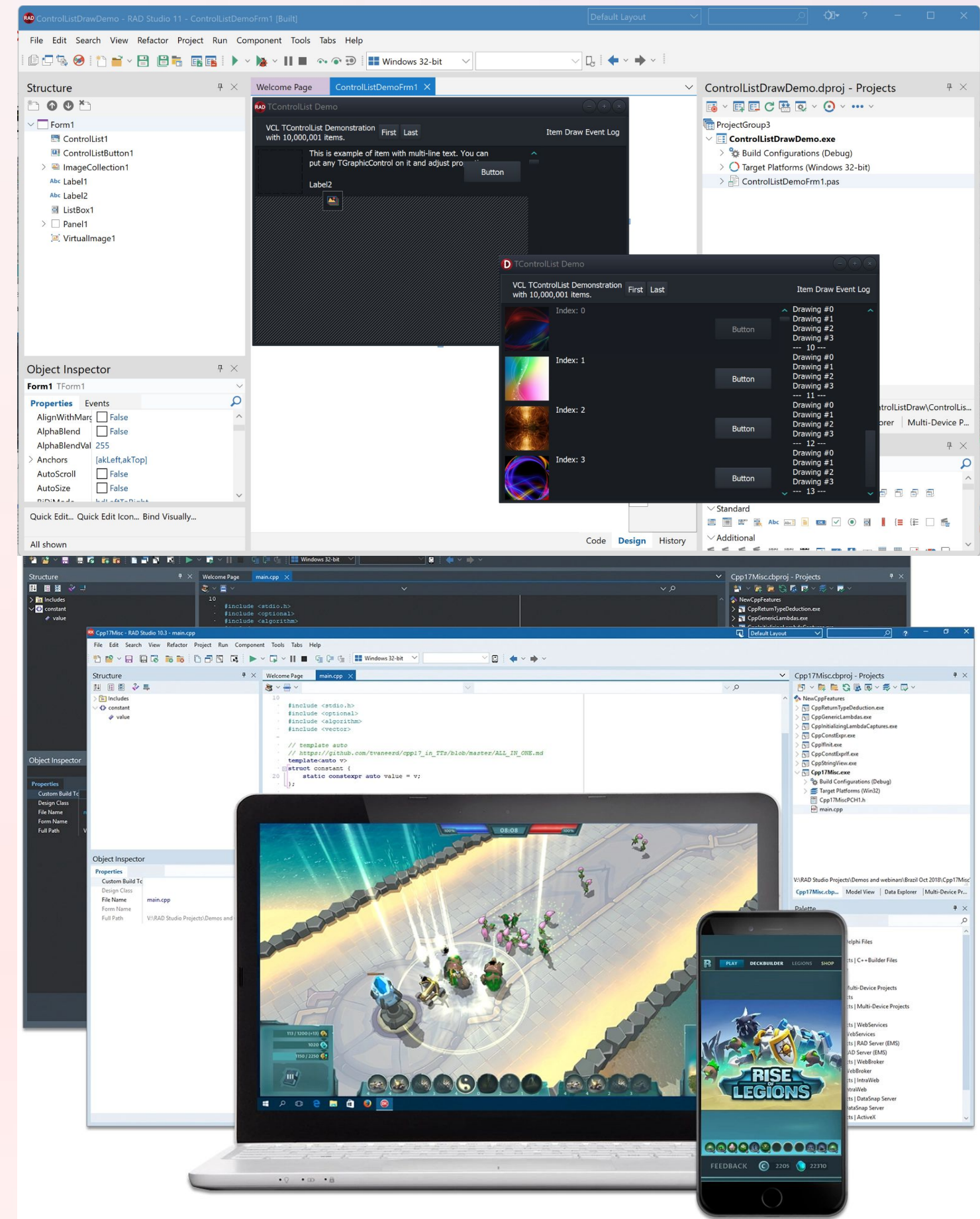
- high-performance

- native applications

in Modern Object Pascal

with powerful visual design tools

and integrated toolchains



What's Great in Delphi?



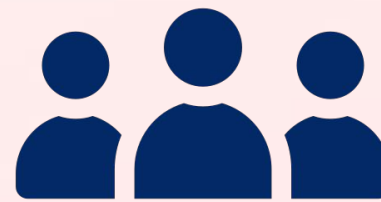
Developer Productivity

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



Fast Native Apps

Native compilers give your apps the speed they need.

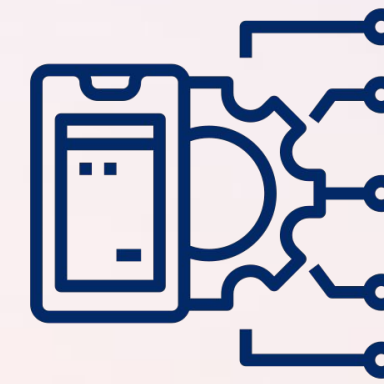
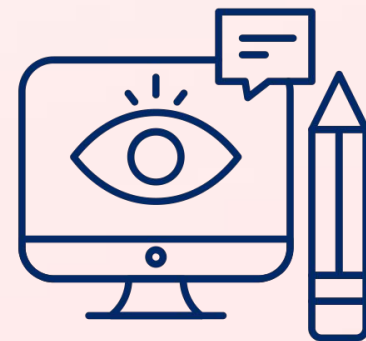


Strong Community

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.

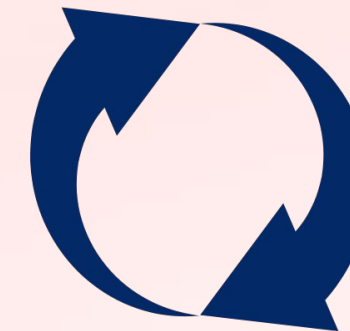
Database Access

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 Used Today?

Development with Delphi (and C++Builder) powers a large number of businesses **today**

Oil and Gas

Aerospace

Large Data Processing

Gis Applications

Accounting

Kiosk Applications

Communication Software

Measurements

Automotive

Flights Management

Remote Computer Access

Architecture

Healthcare

Business Analytics

Systems Administration

Music Editors

Video Games

Database Front Ends

Agricultural Management

IoT Management

E-Invoices

Trains and Railways

Mobile Apps Of All Types

Point of Sales

Government

Engineering Projects

Production Systems Monitoring

Animated Movies

Television

Complete ERP Systems

Real-Time Finances

Banking Systems

Tax Returns

Manufacturing

Warehouse Automation

Developer Tools

Advertising

Circuit Boards Design

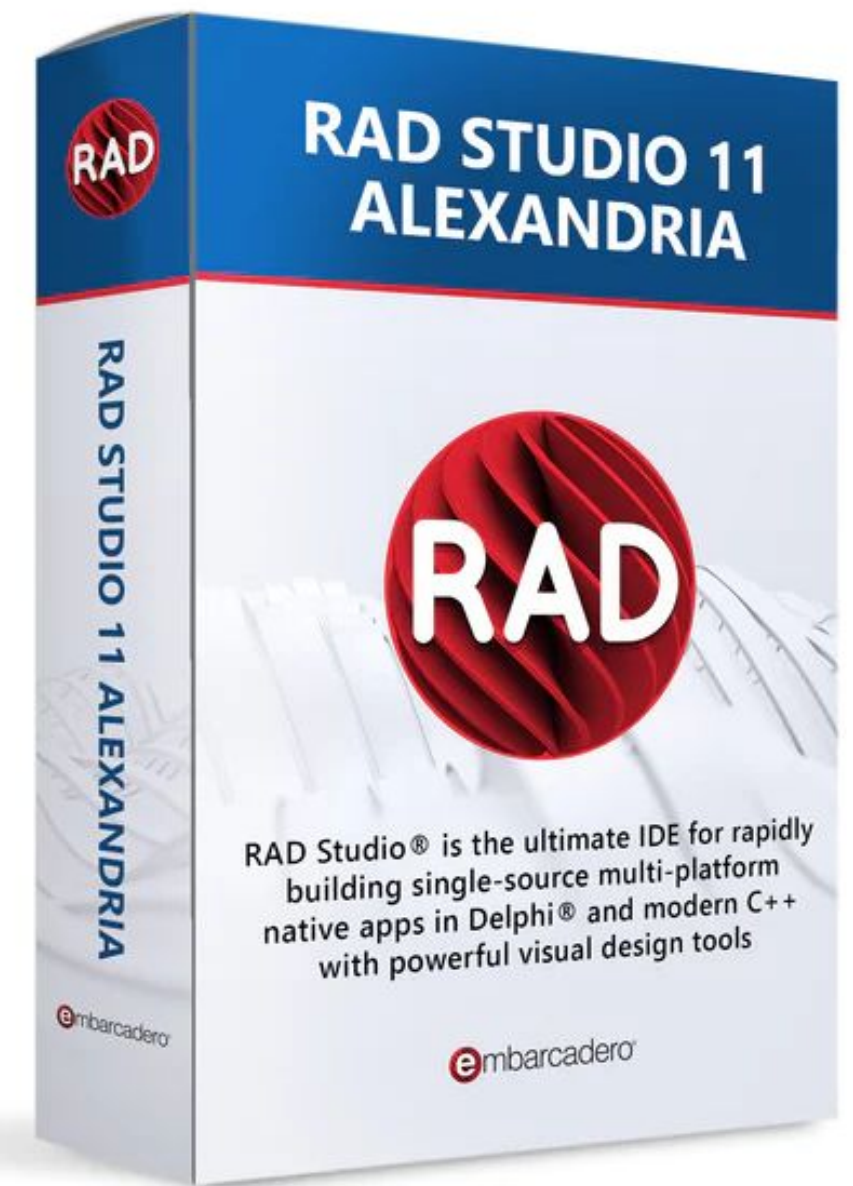
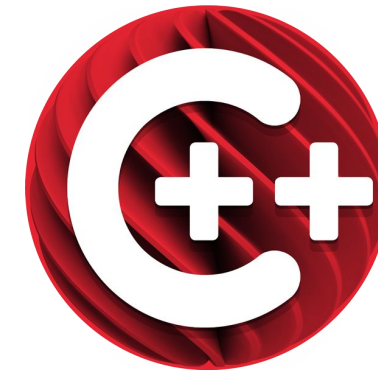
Shipping & Logistics

Trucks Tracking

There is no business segment RAD Studio isn't present

RAD 11.x Summary

A Summary of the Features in
RAD Studio 11.x Alexandria Series



RAD Studio IDE

High DPI IDE

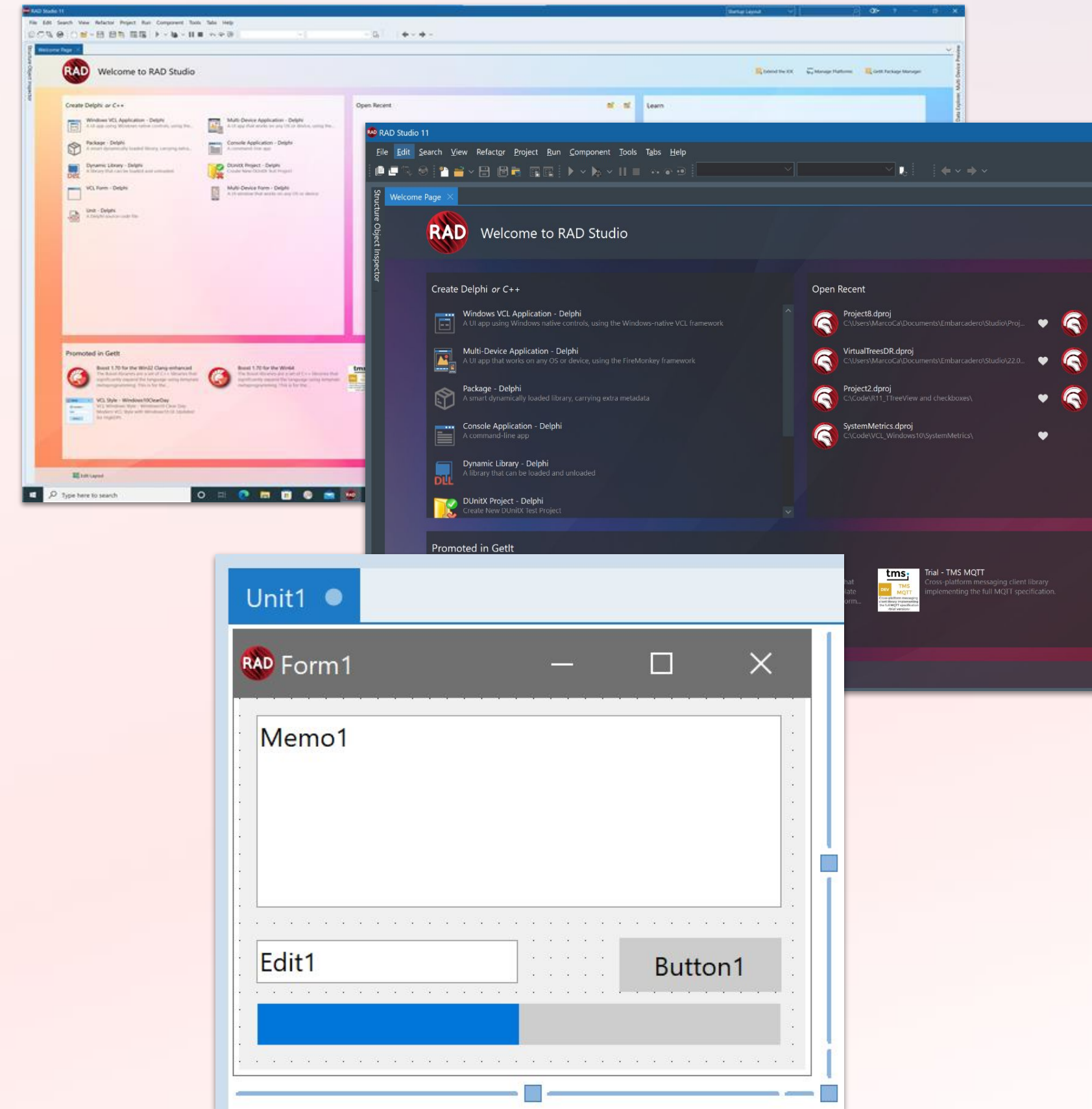
Crisp fonts in all windows and the editor

High DPI enabled form designer

Form and code visible at once in different windows, similar to the old undocked layout

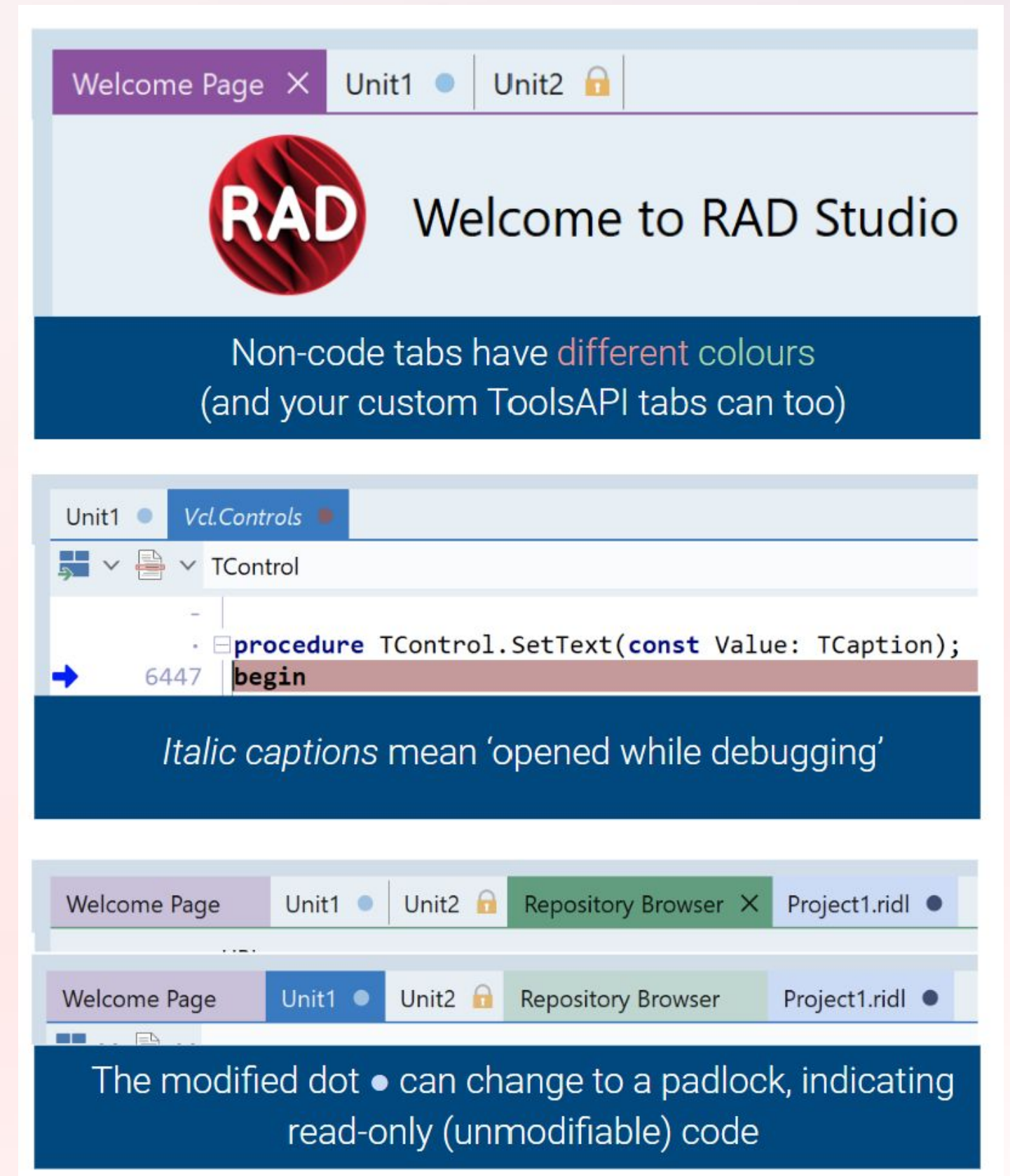
Multiple Active Styles in VCL Designer

New Welcome Page



IDE - tab state visualization

- Some tabs are different to other tabs:
 - non-code functions, eg the Welcome Page, Type Library editor, source control tabs
 - opened while debugging
 - modified contents
 - read-only
- Plus, the X to close is now configurable: off entirely, on the active tab, or on all tabs



Markdown and HTML

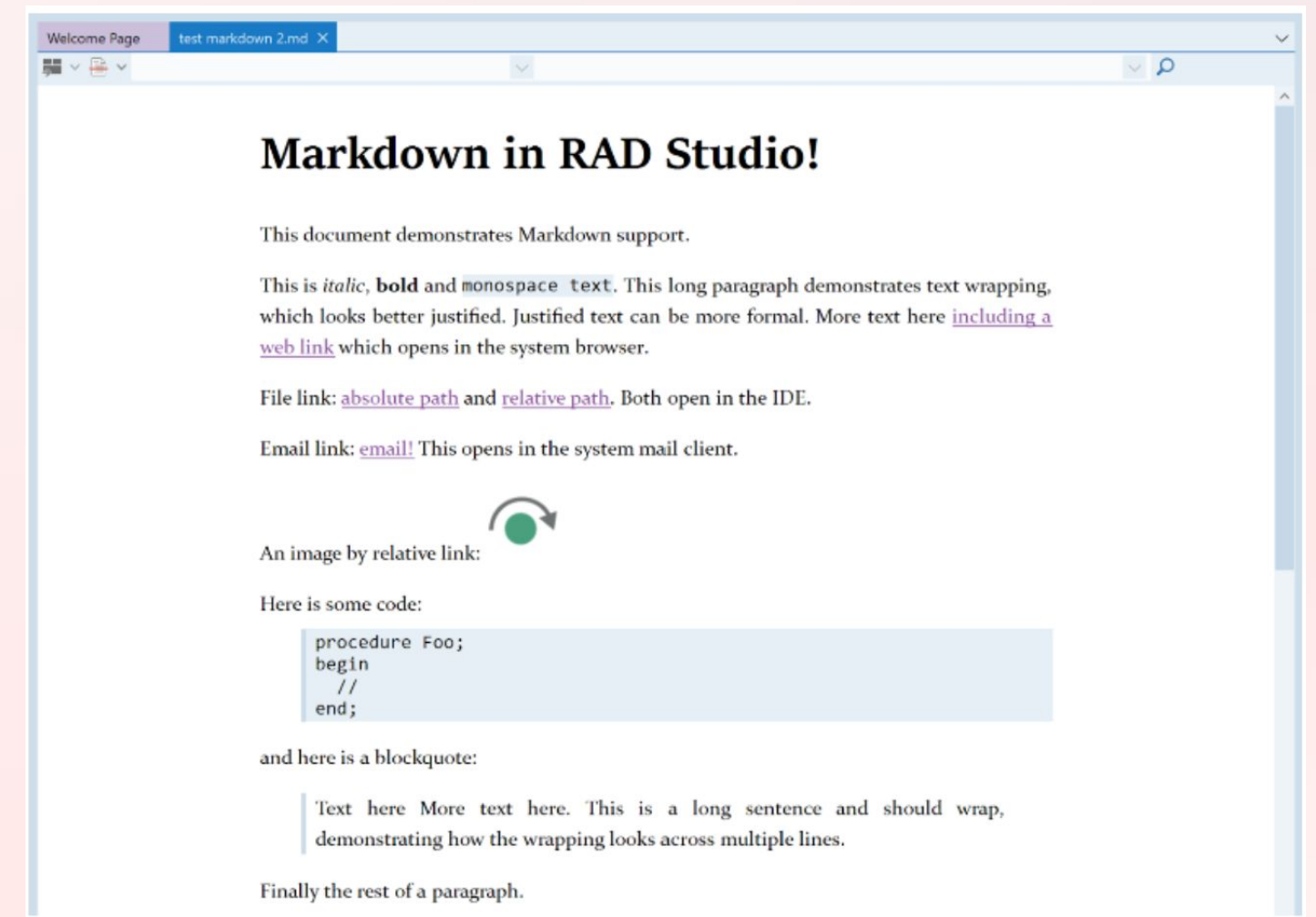
IDE supports Markdown documents, richly formatted: headings, links, images, code, etc

HTML and MD editable in code editor

Project readmes can be Markdown

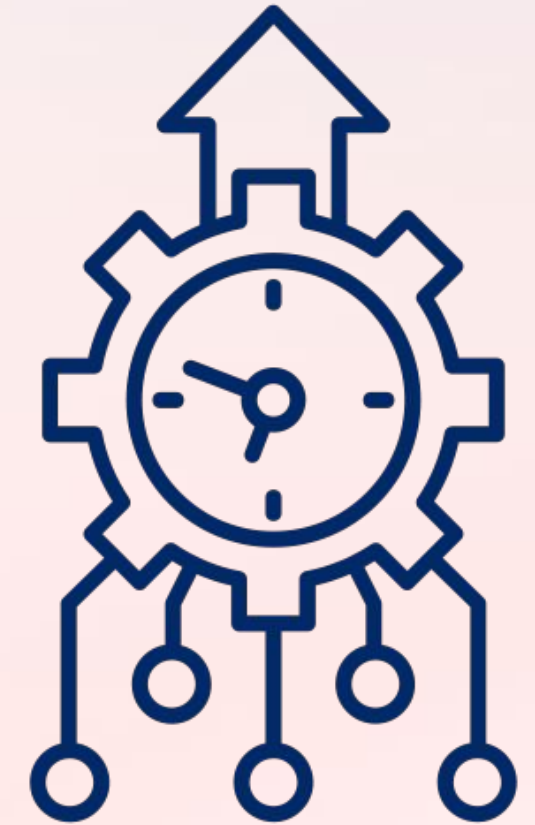
No Internet Explorer in the IDE any more!

Pure VCL implementation



Why DelphiLSP?

- LSP is a standardised protocol for code completion (and related features)
- Using LSP is a modern approach
 - We even ship a DelphiLSP addon for VSCode!
- Moving Code Insight out of process allows:
 - Greater stability
 - Handling big / giant projects: reduces memory usage in the IDE
 - Moves to a single parser: the compiler generates DelphiLSP results
 - Doing multiple things at once: we have multiple DelphiLSP processes producing different types of responses
 - Doing multiple things at once (again): see code insight while debugging
 - Keeps the IDE responsive: the main thread never waits



More Developer Help

The return of XSLT! Effectively, Help Insight as it was pre-11.0

uses

```
    FMX.Platform,  
    {$IF DEFINED(ANDROID) OR (DEFINED(IOS))}  
    FMX.DialogService.Async  
    {$ELSE}  
    FMX.DialogService.Sync  
    {$ENDIF}  
    ;
```

```
    if Addr = 0 then  
        Result := 'Nil'  
    else  
        {$IFDEF AUTOREFCOUNT}  
        Result := Format('%s@%8x(%d)', [ClassName, Addr, Instance.RefCount]);  
        {$ELSE}  
        Result := Format('%s@%8x', [ClassName, Addr]);  
        {$ENDIF}
```

```
if (GetWindowLong(Handle, GWL_EXSTYLE) and WS_EX_TOPMOST <> 0) and  
(  
    function Winapi.Windows.GetWindowLong(hWnd: HWND; nIndex: Integer): Integer - Winapi.Windows.pas \(42293\)  
    Parameters  
    hWnd  
        HWND  
    nIndex  
        Integer  
end;
```

See Inactive Code in the editor

Code that is not compiled (inactive) drawn in a faded semi-transparent color

Other RAD Studio IDE Enhancements

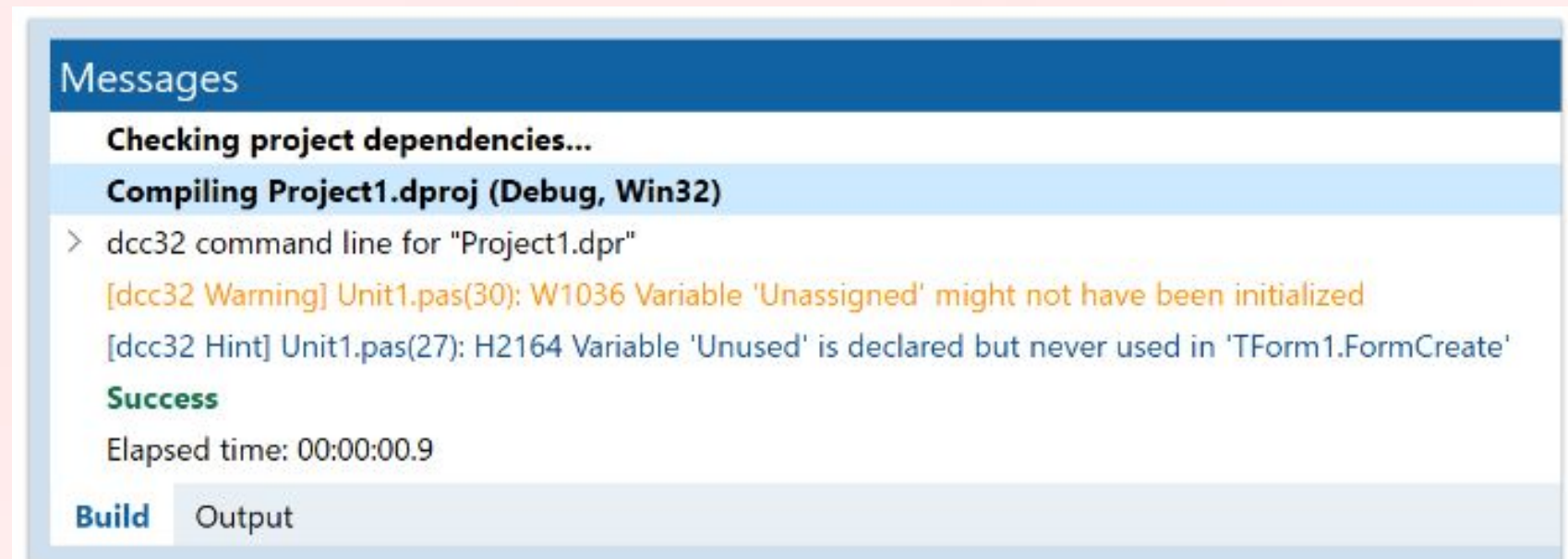
CPU Disassembly View added syntax highlighting

Getit Language filter (search for only Delphi or C++) and Local packages

Messages view: errors, warnings, and hints are differentiated by color (the colors can be customized)

The Compile dialog shows the platform and build configuration

```
Unit1.pas.27: var i : Integer := Round(Random);
002FFA67 E8047DDFFF      call $000f7770
002FFA6C E85782DFFF      call $000f7cc8
002FFA71 50             push eax
002FFA72 C1F81F         sar eax,$1f
002FFA75 3BC2         cmp eax,edx
002FFA77 58             pop eax
002FFA78 7405         jz $002ffa7f
002FFA7A E87987DFFF      call $000f81f8
002FFA7F 8945F8         mov [ebp-$08],eax
Unit1.pas.28: i := i * 2;
002FFA82 6B45F802       imul eax,dword ptr [ebp-$08],$02
002FFA86 8945F8         mov [ebp-$08],eax
002FFA89 7105         jno $002ffa90
002FFA8B E87087DFFF      call $000f8200
```



Delphi Target Platforms and Compiler Changes

11.0: Delphi macOS 64-bit ARM target platform

11.2: iOS Simulator Support (on ARM macOS)

11.2: Android API level 32 support

Reinforced security Flags, including

- ASLR (Address Space Layout Randomization)
- DEP (Data Execution Prevention)
- HE-ASLR (High Entropy Address Space Layout Randomization) in Win64



Recent Changes in the Delphi Language

Recent notable Delphi language improvements

- Managed Records with automatic invocation of special operators
- Inline variables offering an additional scope/lifetime model and type inference

Smaller recent changes:

- Binary literals
- Digit separator support

```
10 begin
    try
        var I : Integer := 1_000_000; // Thousands Separator
        Writeln(I);
        I := %1001001; // Binary Literal
        Writeln(I);
16     Readln;
    except
        on E: Exception do
            Writeln(E.ClassName, ': ', E.Message);
20 end;
end.
```

Libraries Improvements

- **RTL**

- TZipFile improvements
- New classes: TNoRefCountObject, TURLStream, TPointerStream, TProxySubrangeStream
- Performance optimizations

- **VCL**

- “New” RichEdit implementation (RichEdit 4.1) and WinAPI modernization
- Many enhancements for scaling and High DPI
- WinUI3 demo
- TEdgeBrowser and TWebBrowser improvements

Libraries Improvements

- **FireMonkey**

- Desktop and Mobile Improvements
- WebView 2 (Chromium Edge) implementation of TWebBrowser
- Windows High-DPI improvements
- TPathData quality and performance

- **Data**

- FireDAC: Support for new database versions, reFind dbGo migration, FireDAC FDMonitor and SQL editor improvements
- RAD Server SysAdmin endpoints, RAD Server Lite
- Better HTTP support for MIME types and SSL certificates

11.3 Theme

A quality focused release with a handful of new features and some nice usability improvements



RAD Studio® 11.3

 embarcadero

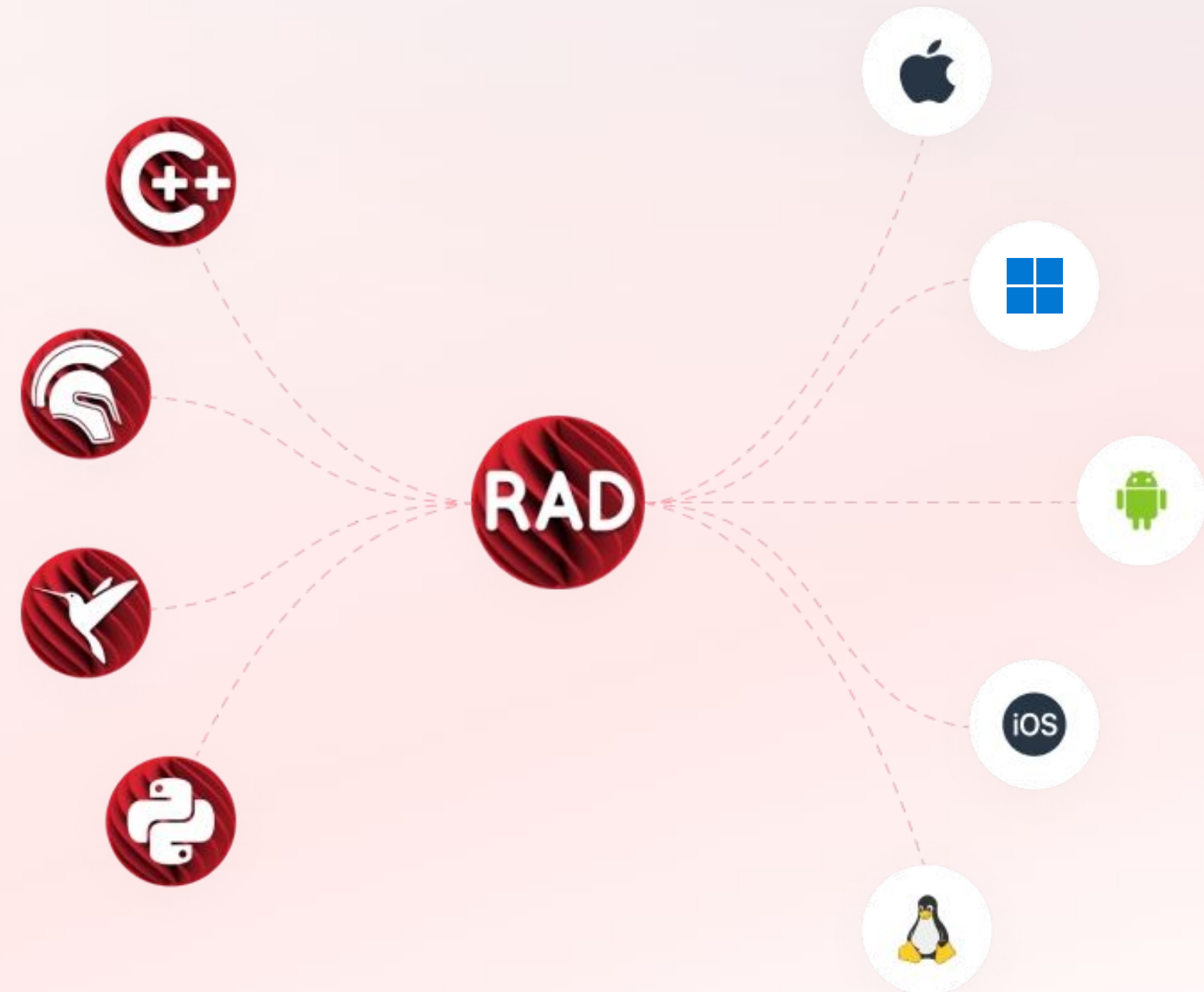
RAD 11.3 Features

What's new in
RAD Studio 11.3 Alexandria



Additional Supported Versions of Target Platforms

- New platforms since 11.2
 - iOS 16 (Delphi only)
 - *At this time C++ can only target the iOS 15 SDK*
 - Android 13
 - macOS Ventura
- Additional supported platforms
 - Ubuntu 22
 - Windows Server 2022



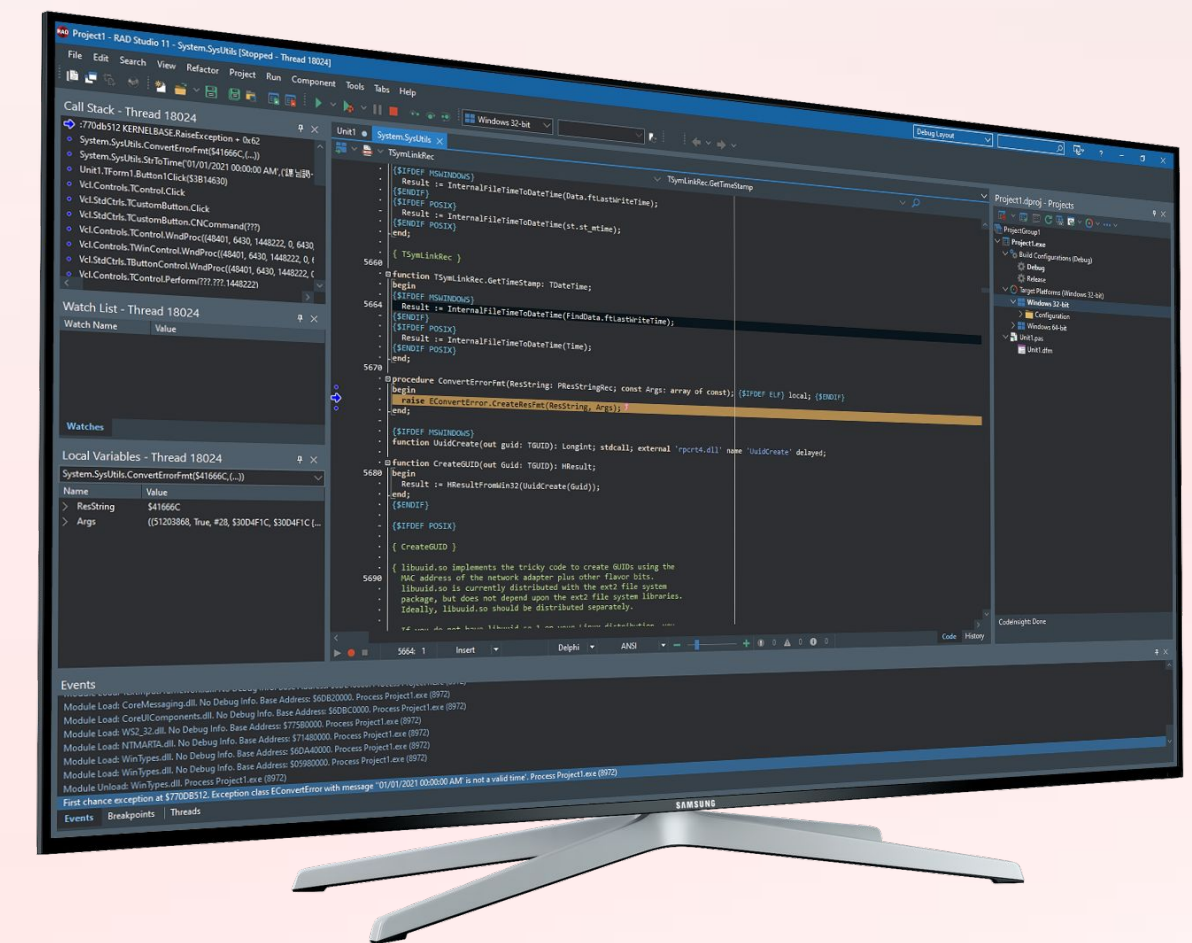
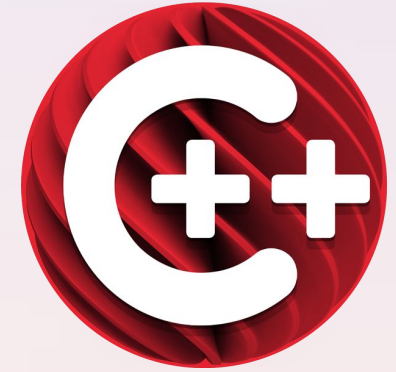
New in FireMonkey

- Biometric Authentication
 - For mobile applications, new Mobile Biometric Authentication component
 - TBiometricAuth provides a means of authorization to use an application or some specific features via Biometrics, like:
 - Face ID on iOS devices
 - Fingerprint on Android devices
- Android, better device integration with:
 - New IFMXPhoneDialerListenerService interface
 - DocumentFile Provider support



Toolchain Improvements

- New LLDB-based Delphi debuggers for macOS Intel and Android 32-bit
 - All non-Windows platform debuggers, across Delphi and C++ have now moved to the LLDB architecture
- New notarization process for macOS (introduced by Apple in XCode 14)
- Windows signing support in the IDE includes the time stamping of MSIX files
- Using a library suffix of `$ (Auto)` now works for C++ projects





LLDB in RAD Studio 11.3

LLDB is now used for all non-Windows platforms:

- Delphi macOS ARM 64
- Delphi macOS Intel 64
- Delphi Android 64
- Delphi iOS 64
- C++ iOS 64
- C++ Android 32
- Delphi Linux 64
- plus C++ Win64



New Features: Code Editor

Code Editor:
highlight matching words

SyncEdit and structure flow
high-DPI icons

```
procedure TFindDialog.CloseDialog;  
begin  
{$IF DEFINED(CLR)}  
    if Assigned(FResources) and (FResources.FFindHandle <> 0) then  
        with FResources do  
        {$ELSE}  
            if FFindHandle <> 0 then  
            {$ENDIF}  
                PostMessage(FFindHandle, WM_CLOSE, 0, 0);  
end;
```

Highlight words

By cursor



None

When selected

By cursor



Don't highlight keywords

Save

Double clicking:

Selects word



Selects word

Selects entire line

```
procedure TForm1.FormCreate(Sender: TObject);  
var  
    I : Integer;  
begin  
    if I > 5 then  
        Exit;↓  
  
        while True do  
            if False then  
                Continue↑  
            else  
                Break;↓  
  
            raise EProgrammerNotFound.Create('Eh, what?');↑  
end;
```

More (Smaller) IDE Changes

- Embarcadero no longer ships the Subversion DLLs, but recommends that you install and maintain your own Subversion client side DLLs
- Locate the current build configuration output folder in Explorer by right-clicking a build configuration (like Debug or Release) and choosing Show In Explorer
- When the IDE is run with elevated privileges, its title bar caption now indicates this with '[Administrator]' at the start of the text
- Markdown files can now be opened in multiple editor windows
- Inactive code opacity now controlled in the Options dialog

New Feature: ToolsAPI for the Code Editor

- Goal: IDE stability with third party plugins (no hacks required)
 - Encourage more third party addons via comprehensive API (we even moved some of our functionality to it)
- A complete, comprehensive, very detailed API
 - (a) painting in
 - (b) querying information about
 - (c) getting events for the code editor(s)
- The new API in is ToolsAPI.Editor.pas
 - Main interface is INTACodeEditorServices
- Demos available on GitHub's RAD 11 Demos Repository

```
• procedure TIDEWizard.PaintGutter(const Rect: TRect;  
•     const Stage: TPaintGutterStage; const BeforeEvent: Boolean;  
85 □ var AllowDefaultPainting: Boolean; const Context: INTACodeEditorPaintContext);  
•     const  
•     cColors: Array [Boolean] of TColor = (clLime, clRed);  
•     begin  
•         // replace colors for gutter line numbers.  
90     if BeforeEvent and (Stage = pgsLineNumber) then  
•         Context.Canvas.Font.Color := cColors[Odd(Context.EditorLineNum)];  
•     end;
```

New Feature: ToolsAPI for the Code Editor

- Demos on Github:
 - <https://github.com/Embarcadero/> -> RADStudio11Demos
 - Then: Object Pascal \ ToolsAPI \ Editor Demos
 - Demos show gutter, text, and background painting – but these just get you started!

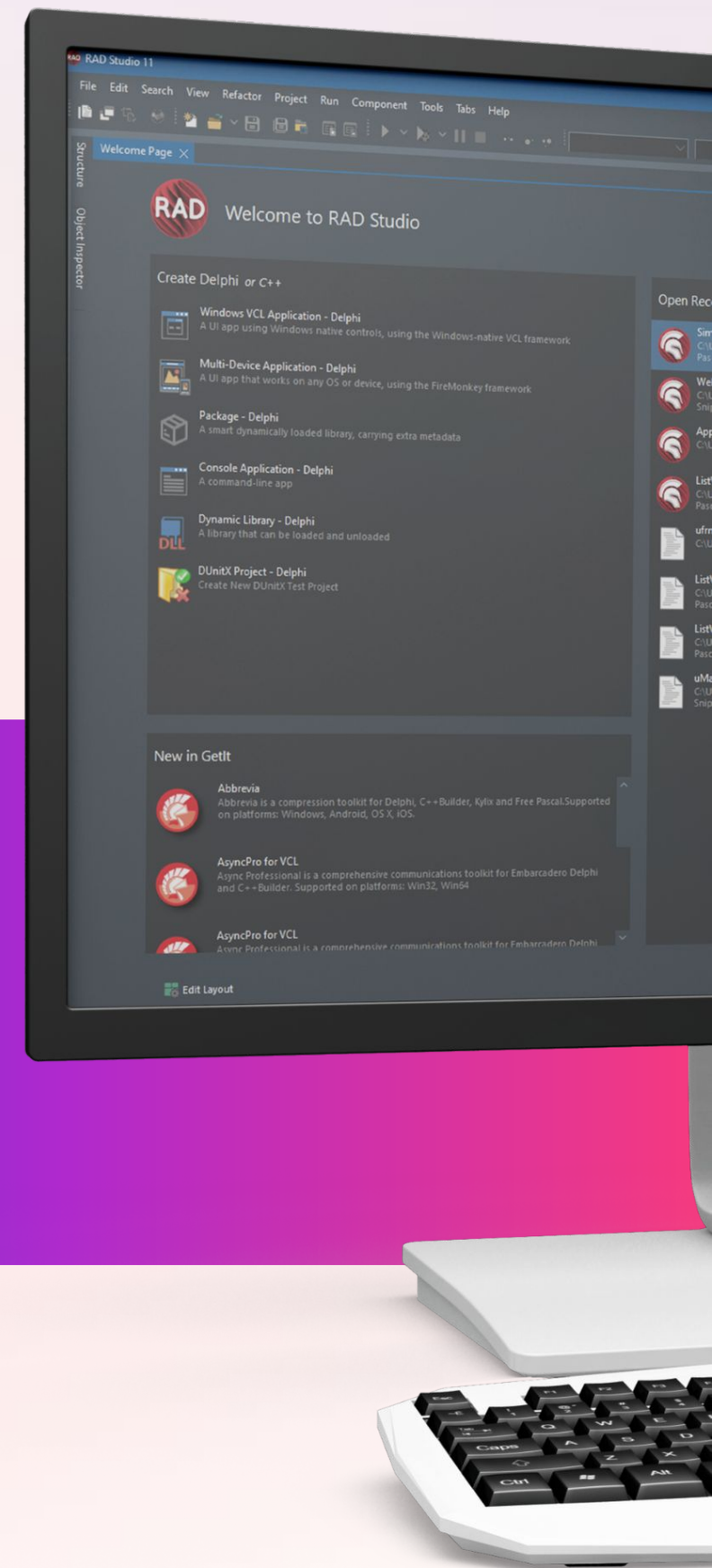
```
• procedure TIDEWizard.PaintGutter(const Rect: TRect;  
• const Stage: TPaintGutterStage; const BeforeEvent: Boolean;  
85 var AllowDefaultPainting: Boolean; const Context: INTACodeEditorPaintContext;  
• const  
• cColors: Array [Boolean] of TColor = (clLime, clRed);  
• begin  
• // replace colors for gutter line numbers.  
90 if BeforeEvent and (Stage = pgsLineNumber) then  
• Context.Canvas.Font.Color := cColors[Odd(Context.LineNumber)];  
• end;
```

```
CodeEditorBgDemo.Main X  
TIDEWizard.PaintLine(const Rect: TRect; const Stage: TPaintLineStage; const BeforeEvent: Boolean; var AllowDefaultPainting: Boolean; const Context: INTACodeEditorPaintContext);  
• function TIDEWizard.GetName: string;  
• begin  
• Result := 'CodeEditor.Background.Demo';  
• end;  
• function TIDEWizard.GetState: TWizardState;  
• begin  
• Result := [wsEnabled];  
• end;  
• procedure TIDEWizard.PaintLine(const Rect: TRect; const Stage: TPaintLineStage;  
• const BeforeEvent: Boolean; var AllowDefaultPainting: Boolean;  
• const Context: INTACodeEditorPaintContext);  
• begin  
• if BeforeEvent and (Stage = plsBackground) then  
• begin  
• var LColor: TColor := Context.Canvas.Brush.Color;  
• var LBitmap: TBitmap;  
• if not FBackgroundBuffer.TryGetValue(Context.EditControl, LBitmap) then  
• LBitmap := UpdateBackgroundBuffer(Context.EditControl);  
• BitBlt(Context.Canvas.Handle, Rect.Left, Rect.Top, Rect.Width, Rect.Height,  
• LBitmap.Canvas.Handle, Rect.Left, Rect.Top, SRCCOPY);  
• // Different background colour is for a selection, or current line highlight, etc  
• if LColor <> FBackgroundColor then  
• FillRectAlpha(Context.Canvas, Rect, LColor, 96);  
• AllowDefaultPainting := False;  
• end;  
• end;
```

```
erudecorp TIDEWizard.PaintText(const Rect: TRect; const ColNum: SmallInt;  
tsnoc Text: gnirts; tsnoc SyntaxCode: TOTASyntaxCode; tsnoc Hilight,  
BeforeEvent: Boolean; var AllowDefaultPainting: Boolean;  
tsnoc Context: INTACodeEditorPaintContext);  
nigeb  
// Paint comments using a different color  
fi BeforeEvent dna (SyntaxCode = atComment) neht  
Context.Canvas.Font.Color := clRed  
esle  
// Replace IDE drawing, reverse reserved words.  
fi BeforeEvent dna (SyntaxCode = atReservedWord) neht  
nigeb  
Context.Canvas.Brush.Color := TColor($64b5e8);  
Context.Canvas.FillRect(Rect);  
Context.Canvas.Font.Color := clBlack;  
Context.Canvas.TextOut(Rect.Left, Rect.Top, ReverseString(Text));  
AllowDefaultPainting := False;  
dne;  
dne;
```

11.3 Quality Highlights

*A quality focused release
with a handful of new features and
some nice usability improvements*

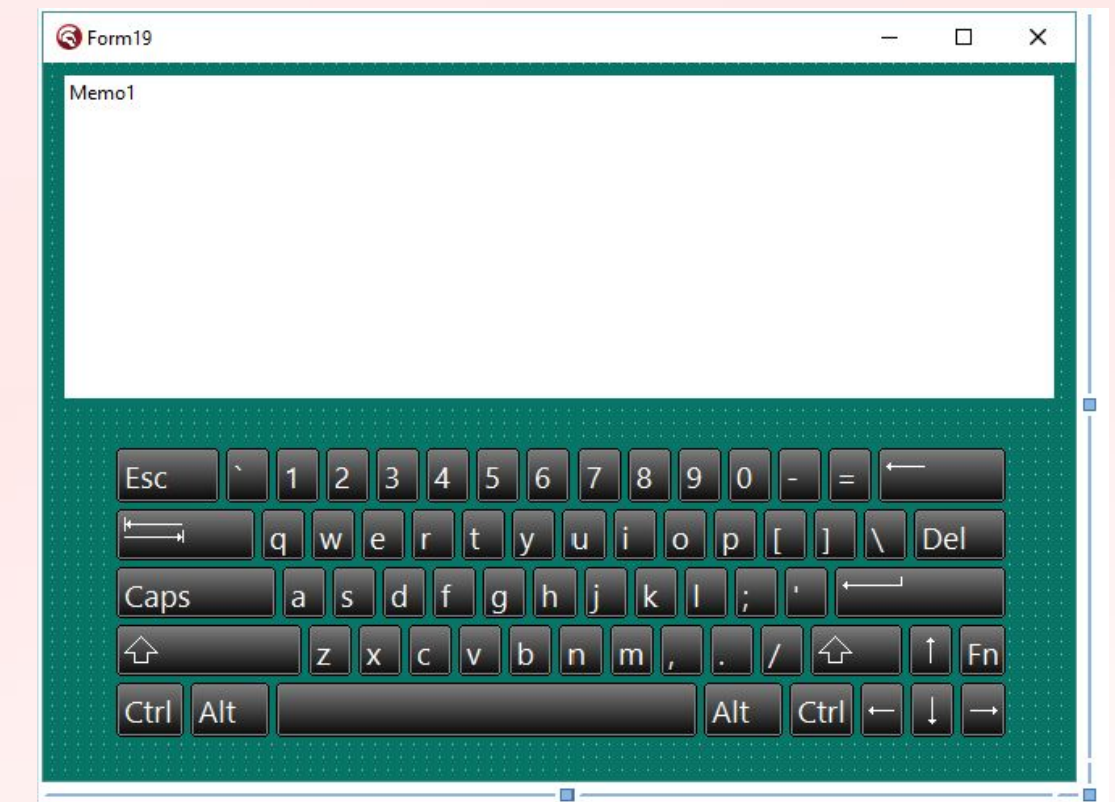
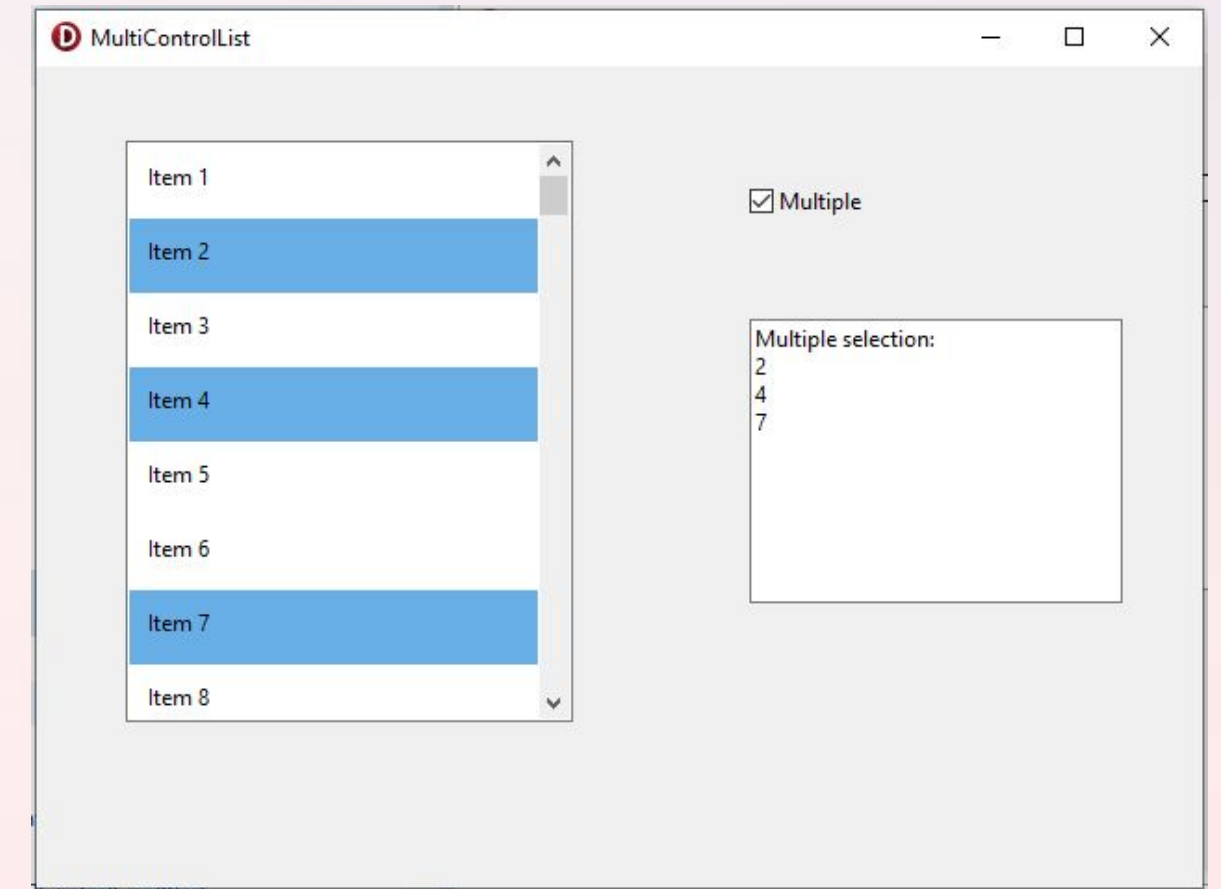


IDE Quality

- Code Insight and DelphiLSP Quality improvements
 - Improvements include code completion, Help Insight, and navigation, with a strong focus on overall reliability, providing completion or navigation results in areas where they were not previously provided
 - The Delphi compiler now stores XmlDoc in a compiled DCU, so HelpInsight can display information in many areas where it previously did not
- Improvements to the High DPI IDE and in particular the VCL Form Designer quality under High DPI VCL
 - Better scaling of non-visual components, including in frames
 - Better HighDPI support for frames in the designer

VCL Quality (and New Features)

- New: TControlList component offers multiple selection via Shift/Ctrl + click, with new MultiSelect and Selected properties
- Quality: related to VCL styles, better support for the styling of the background of the TouchKeyboard component
- Many other improvements and fixes in the VCL styles support, improved StyleHook stability
- THotkey and TScrollbar work better on High DPI



FMX Multi-Platform Quality

- Windows: FireMonkey apps now properly handle terminate requests from Windows Restart Manager; also better accessibility, TWebBrowser integration, and improved support for multiple displays
- iOS: improvements in scrolling for TDateEdit, TTimeEdit, and TComboBox
- Android: improvements when moving the cursor in a TMemo with finger slide
- FireMonkey graphics improvements in the TBitmap and TCanvas classes.
- There are fixes for many controls, including Multiview (with platform control type), SpeedButton, Label's FocusControl, TTreeViewItem's IsChecked, TEdit when used inside a TVertScrollBar.
- ClipChildren now takes XRadius and YRadius into consideration
- Better Tab key cycling across user interface controls

RTL and Data Quality

- Delphi RTL library
 - Android memory allocation performance improvements
 - Some RTL performance improvements (`_InitializeArray`, `Move`, `TDirectory.GetFiles`)
 - An updated version of the ZLib library (1.2.13)
- Database and FireDAC
 - Improvements for PostgreSQL database performance (new connection setting `FastFetchMode`) and full compatibility with version 14 and version 15.1
 - For Oracle and other databases, better treatment of column names with square brackets
 - For Linux, the database layer offers preliminary support for UTF8 in the database RTL (like for `TStringField`) and in FireDAC's `DataSets`
 - Made `TBindingExpression` case insensitive
- Includes InterBase 2020 Update 4 Developer edition and IBLite/IBToGo edition

Web Related



- In the HTTP and web space
 - Improvements to TMultipartContentParser
 - Automatic token refresh to TOAuth2Authenticator and further extensions for OAuth2 clients
 - Better support for NestedElements in TRESTResponseDataSetAdapter
 - Fixes in RAD Server, SOAP and other related subsystems
- Newer versions of Appercept's AWS SDK for Delphi continuously available in GetIt

Quality Summary

- Overall, RAD Studio 11.3 Alexandria implements over
 - 23 feature requests from Quality Portal
 - Fixes over 400 bugs reported by customers on that site
 - Encompassing all areas of the product

- Summary of quality fixes across 11.x
(QP issues closed as fixed or cannot reproduce, 11.3 estimate)

- 11.0 924
- 11.1 843 (+ 37 in 11.1.5)
- 11.2 655
- 11.3 468 (*at release time*)

11.x Total: 2,927

** 3,059 as of June 1st, 2023*

Ecosystem

There is more than the product in the Delphi ecosystem



RAD Studio® 11.3

 embarcadero

Additional Product Initiatives

- Appercept's AWS SDK for Delphi
 - Long term distribution deal, included in Enterprise and Architect
 - New releases at a constant pace
- Python - *Based on Python4Delphi*
 - Delphi UI libraries (VCL + FMX) for Python
 - Free UI library for Python developers
 - Upgrade to a Delphi license for additional power and features
 - Also, you can use Python in Delphi applications



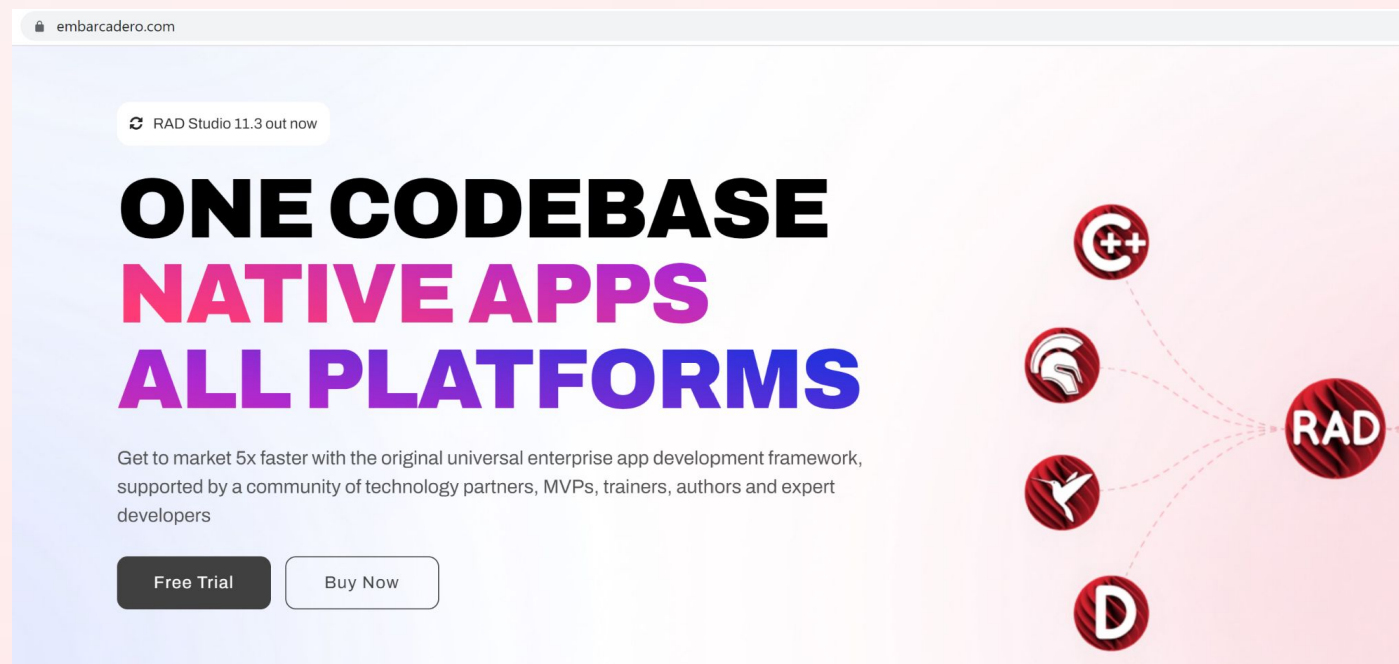
Embarcadero Sites for Delphi

delphi.embarcadero.com

Delphi 25th birthday website
Delphi 1 Client Server for free
Interview with Chuck J. & Anders H.

my.embarcadero.com

Customers Portal
Licences and downloads



LearnDelphi.org

Learning resources
Community Edition

blogs.embarcadero.com

Blogging platform
No internal forums

getitnow.embarcadero.com

lists all available GetIt packages

Open Source Initiatives

[**https://github.com/Embarcadero**](https://github.com/Embarcadero)

- Bold for Delphi - Model Driven Architecture
- Sponsored Open Source Projects
 - Dev C++
 - HeidiSQL
 - Python4Delphi + PyScripter + and more
 - SonarDelphi
- Octoid*
- MVP Collaboration Projects: XML Mapper
- InterBase drivers for .NET and Python*

** Coming soon*

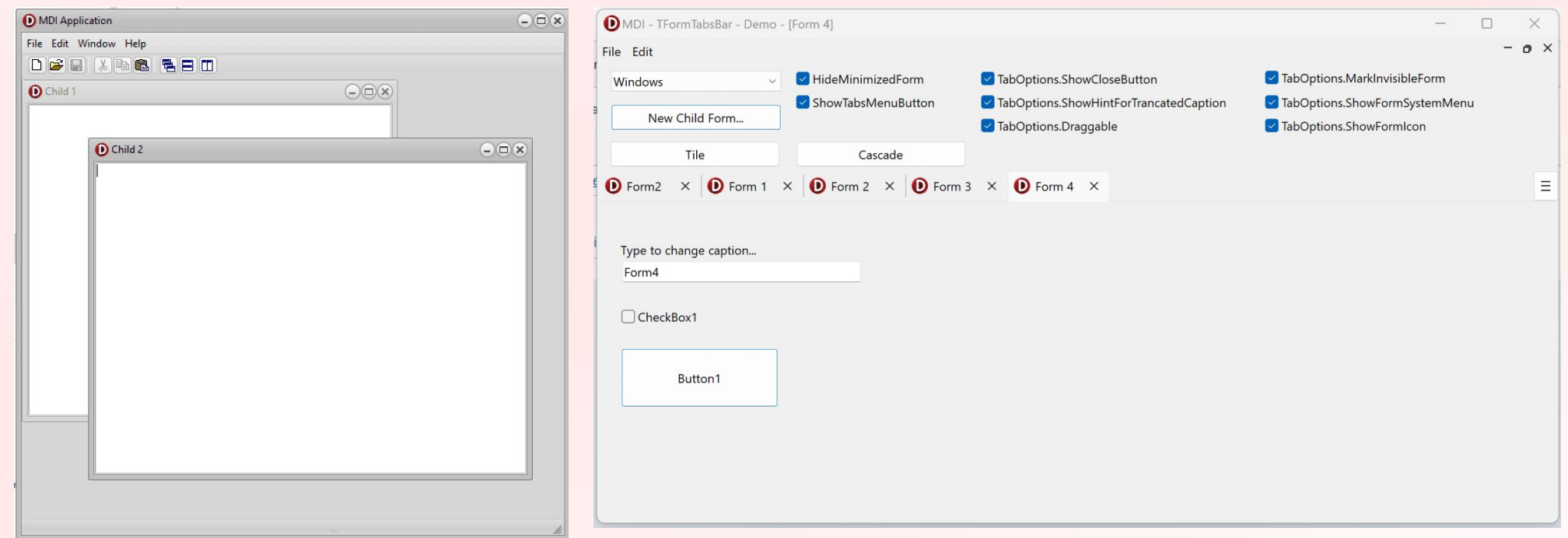
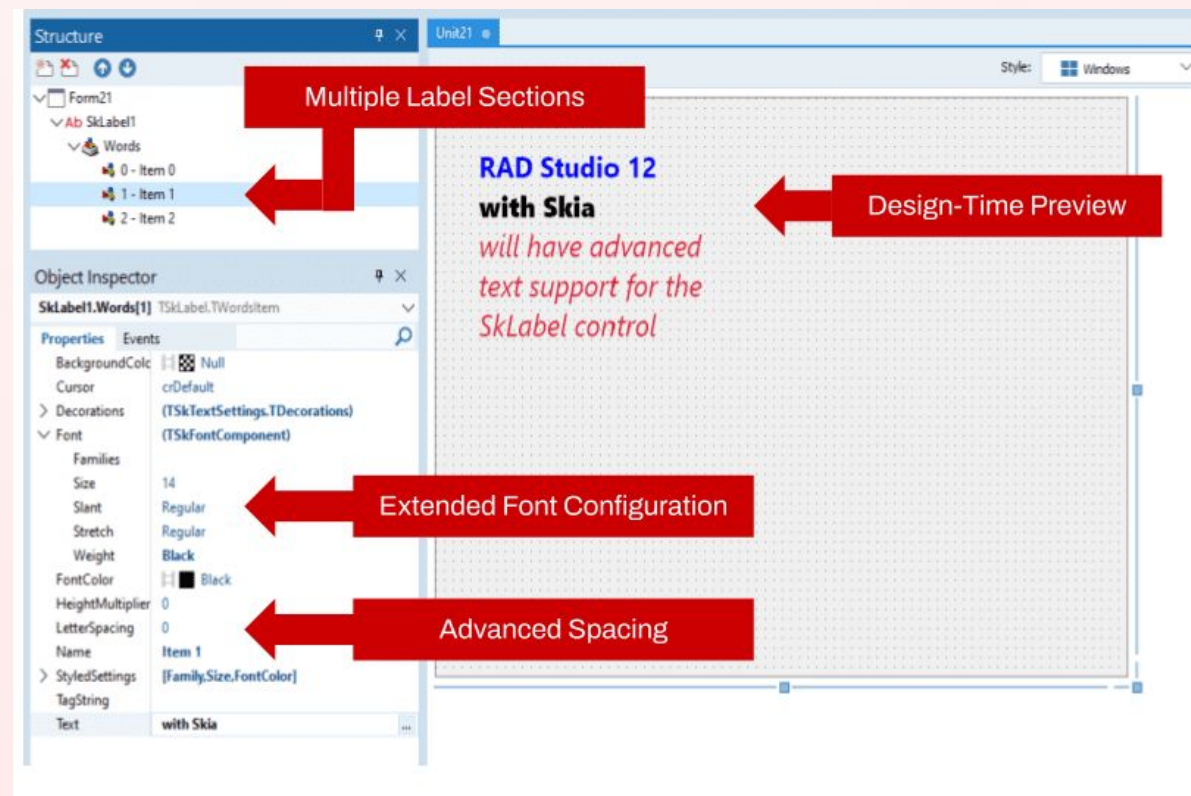


What's Coming

*Next will be the 12.x series
Work has started before 11.3 release*

RAD Studio 12.x: What's Coming**

- Work in progress for the next major release on several fronts
- C++ blog post covering the most relevant focus areas, Clang and Visual Assist:
<https://blogs.embarcadero.com/whats-coming-for-cbuilder-an-amazing-preview/>
- Library focus blog post on Skia and more VCL core work (including MDI):
<https://blogs.embarcadero.com/whats-coming-in-delphi-and-cbuilder-libraries/>



**** Features are not committed until completed and GA released**

Q&A

Marco Cantu, Product Lead, IDE & Tooling

marco.cantu@embarcadero.com