



What is RAD Studio?



The fastest way to develop cross-platform native apps with flexible cloud services and broad IoT connectivity



embarcadero

Fast, Visual Development



RAD Studio provides powerful VCL controls for Windows 10 and enables FMX cross-platform development for Windows, Mac, iOS and Android



embarcadero

RAD Studio Platform Support



Target the latest versions of major operating systems with one codebase!



Officially Supporting

embarcadero

RAD Studio 10.2.x



Key features delivered in recent months:

- Mobile (iOS and Android) added to the Professional edition, now Win, Mac, iOS, & Android
- CMake command-line support for C++Builder
- New VCL GUI controls, layouts and FMX & VCL QuickEdit designers
- IDE Enhancements, including a Dark Theme, new component icons and layouts
- C++ compiler and linker improvements
- RAD Server enhancements for Sencha EXT JS
- Community Edition for Delphi and C++Builder



embarcadero

RAD Studio 10.2.3



Enhanced C++ support, IDE UI/UX improvements and quality enhancements

Visual Framework Enhancements
(Delphi & C++)

- VCL High DPI Quality Improvements
- FireMonkey Windows and Android quality
- FireMonkey platform quality
- FireMonkey User Interface Templates

C++Builder

- CMake Support for our toolchain
- C++ Rename Refactoring

embarcadero

RAD Studio (all IDEs)

- IDE UI/UX Improvements
- RAD Server ExtJS Integration for enhanced Javascript client support
- Platform Enhancements across Windows, macOS, iOS, Android and Linux
- Added Mobile support (iOS and Android) to Professional edition for new and Update Subscription customers
- 10.2.3 Community Edition released in July 2018

Released in March 2018

Delphi and C++Builder Community Edition!



Free, limited use commercial edition with Professional level features

As of 10.2.3, Professional Edition includes Mobile, which means Community does too

- For casual developers, startups, students, and open source developers (restrictions apply, see Eula)
- Develop Windows, macOS, Android and iOS applications with a single codebase
- Visual Design using the C++Builder and Delphi VCL and FireMonkey frameworks
- Hundreds of included components to enhance your app and reduce development cycles
- Community comes for our two personalities, Delphi and C++Builder



embarcadero

Released in July 2018

Coming soon... 10.3 Rio

RX



embarcadero

10.3 Rio Key Features

RX

- C++ 17 for Win32
- New Delphi Language Features
- FireMonkey Android zOrder and API Level 26
- Windows 10, VCL, and HighDPI improvements
- Extensive IDE UI modernization
- RAD Server architecture extension
- Quality and Performance improvements

embarcadero

** Features are not committed until completed and GA released

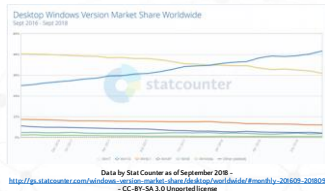
RX

10.3: RAD Studio Platforms

embarcadero

Windows 10 is King

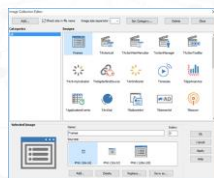
- The most important platform for our current customers
- Add Windows 10 features without breaking Windows 7 compatibility



** Features are not committed until completed and CA released

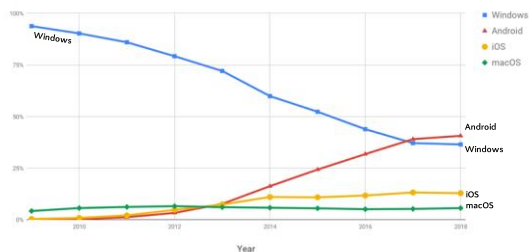
10.3 Windows 10 is King

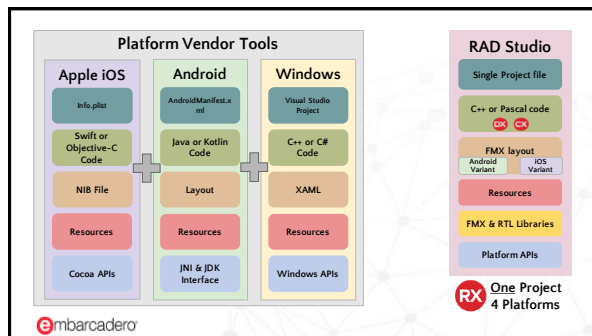
- Windows 10 = key focus for RAD Studio
- VCL extensions for Hi-DPI and 4K monitors support
 - PerMonitorv2, multi-resolutions image list, new APIs in 2018 flavor of Windows 10
- Additional WinRT API and Windows Store support

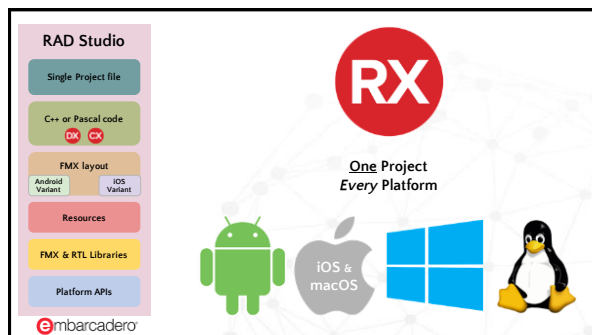


** Features are not committed until completed and CA released

Multi Device is the Future (& the future is now!)







10.3 Cross-platform is the Future

- Cross-platform = key focus for RAD Studio
 - FireUI multi device designer
 - FireUI preview on device
- Styled and native controls
 - Native on Windows, iOS, and now Android
- Support for
 - Android P with API level 26
 - iOS 12

RX

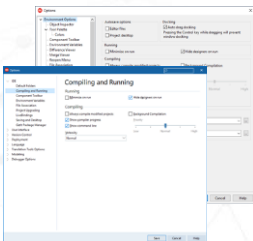
embarcadero

** Features are not committed until completed and CA released



UI Enhancements: IDE & Project Options

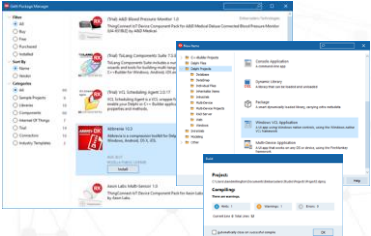
- UI cleanup: layout, alignment
 - Looks consistent
 - Spacing better for translation (eg German)
 - Dependent settings clear and obvious
- Option location cleanup: previously, hard to find items. Now,
 - Some categories renamed, some options moved
 - Dialog-wide option search in title bar



** Features are not committed until completed and GA released

IDE UI Enhancements: Other Dialogs

- GetIt!:
 - Scrolling list
 - Nicer look
- New Items:
 - More modern look
- Compile:
 - More modern look
 - Hint / warning / error icons light up, showing status at a glance

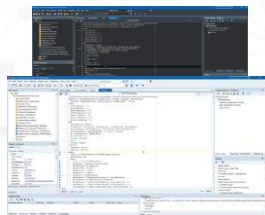


** Features are not committed until completed and GA released

IDE UI Enhancements: Main Window



- New Light and Dark styles, based on blue brand colours
- General cleanup: alignment, spacing
 - Everything where you expect, but neater
 - Readable (eg code editor tabs)
 - Search (IDE Insight) on title bar
- Better product first impression
- Better for long-term (eg 8 hours / day) use
- Docwiki help also has a new theme



** Features are not committed until completed and GA released

Code Completion



- For Delphi,
 - Focused on code completion correctness - giving good results
 - Testing with very large projects
 - Big quality fixes in 10.2.3 and 10.3
- For C++ with the C++17 compiler,
 - Code completion results are much more accurate
 - Completion results are significantly faster
 - Code completion is *asynchronous*. You can keep typing with no delays, code completion occurs in the background. This is a new feature for the IDE
 - It uses the Language Server Protocol and cquery



** Features are not committed until completed and GA released

.Net 4.x Upgrade



- RAD Studio now requires .Net 4.x as a prerequisite before installing
 - Replaces .Net 3.5
- This is most likely already on the target PC, given our Windows 7 minimum platform requirements



** Features are not committed until completed and GA released

Key IDE Quality Improvements



- Removed the RAD Studio Community Toolbar from the IDE
- The installer now distributes the Windows 10 April 18 version of the Windows SDK
- Significantly improved LiveBindings designer performance (aka. fixed the slow loading of large forms in the designer)
- We incorporated some of Andreas' "IDE Fix Pack" fixes/improvements
- Product demos are now shared on GitHub

embarcadero

** Features are not committed until completed and GA released

10.3: Languages



embarcadero

10.3 Languages



- Delphi has benefited from stability
 - Focus to improve the core language, keep it modern (new features in 10.3)
 - Improve compiler quality, stability and correctness – for large projects
- C++ 17 for Win32 in 10.3, and other platforms in 10.3.x
 - Most significant update of our C++ compiler in a few years
- Next focus will be on macOS 64-bit platform support for both languages



embarcadero

** Features are not committed until completed and GA released

New Delphi Language Features



- Inline variables and constants declarations, direct assignment of value at inline variable declaration point, block scope

```
begin
  var s: string := 20.ToString();
  for var I: Integer := 1 to 100 do ...
```

- Type inference for inline variables

```
var aDict := TDictionary<Integer, string>.Create;
```



** Features are not committed until completed and GA released

Delphi Language Changes



- Linux 64-bit compiler moves to non-ARC
 - Part of the new overall direction of removing ARC
 - Reasons: Performance, dual memory model complexity, cannot move VCL to ARC
 - ARC still on for mobile compilers in 10.3
- Side effect: AnsiString / AnsiChar available on Linux
- Improved C++ compatibility
 - Changes in the way records between 4 and 8 bytes in size are passed as parameters
- Improved compiler performance



** Features are not committed until completed and GA released

Delphi: Inline variables and constants



Pascal has always required you to define variables at the top of a method
Common reasons:

- Requires you to plan a method more carefully (think ahead)
- Results in smaller methods
- Etc

All these are true. But we see unplanned methods, and long methods, regardless.

Other languages have moved on. Local scope is useful.



Delphi: Inline variables and constants



In 10.3 Rio, variables can be declared in any block (begin..end.)

- Scope is limited to block
- Lifetime is limited to block - destroyed on block end
- Tightly scoped = cleaner code

```
for var I : Integer := Low(Items) to High(Items) do
begin
  const Item : Integer = Items[I];
  WriteLn(Format('Item: %d', [Item]));
end; // Item destroyed here (managed type like string, record)
// I not visible here
```



Delphi: Type Inference



What's wrong with the code we just saw?

```
for var I : Integer := Low(Items) to High(Items) do
begin
  const Item : Integer = Items[I];
  WriteLn(Format('Item: %d', [Item]));
end;
```

' : Integer' when the compiler can determine that itself.
Unnecessary typing. Hard to refactor.



Delphi: Type Inference



- Type inference is when the compiler knows what type a variable needs to be
- Why write something when you don't need to?
- A compiler's job is to assist you
 - If something is clear, you should not be forced to write it

In 10.3 Rio, inline variables and constants, without a type specified, will have the right type inferred.



Delphi: Type Inference

DX

- Type inferred from the type of the expression
 - Simple assignment of record or value


```
var s := 'Hello'; // string
var f := 3.14; // single
```
 - Method return type


```
const foo = TCar.Create(); // a TCar
```
 - Matches the apparent type - a class implementing an interface will be inferred as the class


```
TFoo = class(TInterfacedObject, IMyInterface) ...
const foo = TFoo.Create(); // TFoo, not IMyInterface
```

 - Better style is to use factory methods returning an interface instance
- Type is 'natural', eg `const I = 255` is an Integer, not a Byte.
- Leads to natural, concise code.

embarcadero

C++17

CX

- C++17 Clang-enhanced compiler and toolchain for Win32 (bcc32x and bcc32c)
- C++ runtime library (RTL) built with the updated Clang-enhanced compiler for Win32, including 2018 edition of Dinkumware STL
- Also includes support for C++14
- Greatly improved code completion
 - Faster
 - Better results
 - Asynchronous, uses LSP

embarcadero

** Features are not committed until completed and GA released

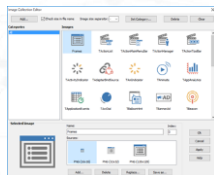
RX

10.3: VCL

embarcadero

VCL High DPI Image List

- New components: virtual image collection (holds images), image list (DPI-aware specific view of that collection)
- Multi-resolution, alpha channel, 32bpp bitmap / PNG (and can import old bitmaps)
- List is compatible with TImageList (including HIMAGELIST handle)
- Image list scales as parent form DPI scales (that means, share an image collection, but don't share an image list between two forms)
- Migration tools for old -> new lists

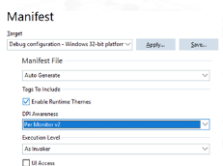


embarcadero

** Features are not committed until completed and GA released

Windows 10 "Per Monitor v2" Support

- PerMonitor v2 support in project options
 - Requires Windows 10 build 1703 "Creators Update" (from 5-April-2017)
- New GetSystemMetrics model
 - GetSystemMetricsForWindow
 - TControl.GetSystemMetrics
 - Highly compatible with existing source code
- Significant focus on VCL High-DPI Fixes



embarcadero

** Features are not committed until completed and GA released

Windows 10 API Enhancements

- New native Win32/Win64 APIs
 - High-DPI area
 - WM_POINTER area (pen support)
- Updated Object Pascal headers to the latest WinRT API
 - A dozen additional WinRT units
- New TWindowsStore component -- for Desktop Bridge applications
 - Manage paid apps
 - In-app purchase
 - Trial mode
- Additional WinRT controls improvements (Contracts)

embarcadero


** Features are not committed until completed and GA released



Android API Upgrade

RAD Studio 10.3 Rio includes support for Android API version 26 development:

- Required by Google for new Play Store applications starting August 2018 and for updates starting November 2018. Changes include:
 - New information in the manifest (with the proper API level)
 - The default configuration uses a recent version of the SDK/NDK
 - Requires installation of new SDK
 - Android permissions architecture redesigned
 - Might require changes to your code




** Features are not committed until completed and GA released

embarcadero

Android New Permission Model

Recent versions of the Android API have changed the mechanism to request permissions:

- The user can call `requestPermissions` on the activity and the callback triggers a RTL message subscription message
- New relevant/related types imported into the RTL Android import units include `Manifest.permission` and `Build.VERSION` in `Androidapi.JNI.Os.pas`, `ContextCompat`, `ActivityCompat`, and `PermissionChecker` in `Androidapi.JNI.Support.pas`.



** Features are not committed until completed and GA released

embarcadero

Agenda

- iOS 12 Support
- Designing around the notch
- Android API version 26
- Android New Permission Model
- Updated Android SDK & NDK
- Android Z-order and Native Controls
- FireDAC Driver Updates
- Android Things and Raspberry Pi 3



iOS 12

Android 9 "Pie"
API 28

embarcadero

RAD Studio

Single Project file

C++ or Delphi code

DX CX

FMX layout

Android Variant iOS Variant

Resources

FMX & RTL Libraries

Platform APIs

embarcadero

Multi-Device means
One Project
Every Platform

Apple iOS 12 Support

- Apple released it on 17-Sept-2018
 - Current version is iOS 12.0.1 with iOS 12.1 in beta
 - Introduced new devices iPhone XS, XR, & XS Max
- Coming support for iOS 10, 11, 12 (Device and App Store)
- Support for targeting new iPhone devices (32 & 64-bit)
- Support for the iOS 10 Simulator (*Delphi Only*)



iOS 12

embarcadero

IMPORTANT: Features are not committed until completed and GA released.

Designing Around the Notch



- RAD Studio Fire UI supports custom views
 - http://docwiki.embarcadero.com/RADStudio/en/Adding_a_Customized_View_to_the_View_Selector
 - New views coming soon to GetIt
- Optionally use fullscreen (*around the notch*) for hidden status bar

	iPhone X	iPhone Xr	iPhone Xs	iPhone Xs Max
Physical	1125 x 2436	828 x 1792	1125 x 2436	1242 x 2688
Logical	375 x 812	414 x 896	375 x 812	414 x 896
PPI	458 ppi	326 ppi	458 ppi	458 ppi
Multi	@3x	@2x	@3x	@3x

Source: iosres.com



embarcadero

Updated Android Support



- New permission model required by Google for new Play Store applications starting August 2018 and updates starting November 2018
- RAD Studio 10.3 Updates
 - New information in the manifest (with the proper API level)
 - Support for NDK 17b & newer SDK
 - Android permissions architecture redesigned
- Supports Android versions 9, 8, 7, 6, and 5



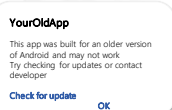
embarcadero

IMPORTANT: Features are not committed until completed and GA released.

Android Play Security & Performance Requirements



- August 2018: New apps required to target API level 26 (Android 8.0) or higher.
- November 2018: Updates to existing apps required to target API level 26 or higher.
- 2019 onwards: Each year the targetSdkVersion requirement will advance. Within one year following each Android dessert release, new apps and app updates will need to target the corresponding API level or higher.



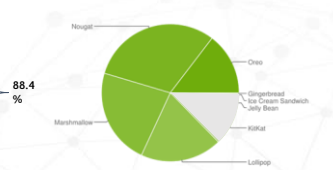
embarcadero

<https://android-developers.googleblog.com/2017/12/20/reviving-app-security-and-performance.html>

Understanding Android Versions & API Levels



Version	Name	API Level	Distribution
5.0	Lollipop	21	3.6%
5.1		22	14.7%
6.0	Marshmallow	23	21.6%
7.0	Nougat	24	19.0%
7.1		25	10.3%
8.0	Oreo	26	13.4%
8.1		27	5.8%
9.0	Pie	28	undefined



Data collected during a 7-day period ending on September 28, 2018

Any versions with less than 0.1% distribution are not shown. Source: <https://developer.android.com/distribute/market-share>

embarcadero

Installing Updated Android SDK & NDK



- You can install them with 10.3
- To manually install run the SDK Manager and select
 - platform-tools, build-tools-28.0.2, & android-27
- Or to update SDK from the command-line
 - `android.bat update sdk --no-ui --all --filter platform-tools,build-tools-28.0.2,android-27`
 - You need to make sure `AndroidSDK\Tools` is on your path for `Android.bat`
- To manually install NDK to r17b
 - The latest version r18b isn't supported yet <https://developer.android.com/ndk/>
 - Check older versions https://developer.android.com/ndk/downloads/older_releases
 - Download <https://dl.google.com/android/repository/android-ndk-r17b-windows-x86.zip>
- Run the RAD Studio SDK Manager to import the new SDK/NDK

embarcadero

New Android Runtime Permission Model



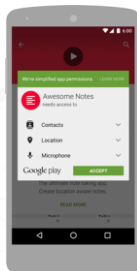
- Introduced in Android 6.0 (API 23) Marshmallow
- New model streamlines auto-update process
- Instead of approving all permissions at install
- Gives end users more control over permissions
- Managed through new Android API methods [checkSelfPermission\(\)](#) & [requestPermissions\(\)](#)



embarcadero

Old Permissions Approval Dialog

- During install all permissions are accepted
- User is unable to deny individual permissions
- All permissions publicized in the manifest
- If permissions change then must be approved before updating
- Note: *There are many apps in the Play Store that still use this model, even some by Google*

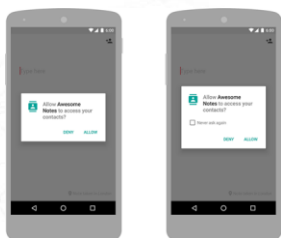


embarcadero

<https://developer.android.com/guide/topics/permissions/overview>

New Permission Approval Dialog

- First time you request the permission the user has the option to Deny or Allow
- If they Deny once, then they can prevent the app from asking again
- It is important when and how you ask



embarcadero

<https://developer.android.com/guide/topics/permissions/overview>

Dangerous vs. Normal Permissions

- The user isn't prompted for *Normal* permissions
 - Examples: Internet, Vibrate, Bluetooth, etc.
- The user must approve *Dangerous* permissions
 - Typically that involve privacy concerns
 - They are grouped into permission groups:
 - Calendar, Call_Log, Camera, Contacts, Location, Microphone, Phone, Sensors, SMS, & Storage
 - User approves an entire group
 - For example Sending and Receiving SMS
- The developer still must declare and request *all* permissions from the OS!



embarcadero

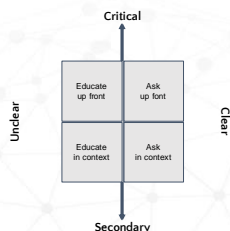
<https://developer.android.com/guide/topics/permissions/overview>

More information

When to Ask for Permissions

RX

- A user is more likely to grant permissions if you ask at the right time and in the right way
- For permissions that it isn't clear why your app needs them, then provide education before asking
- Or provide education if the user denied the first request
- Don't request all of them at once.

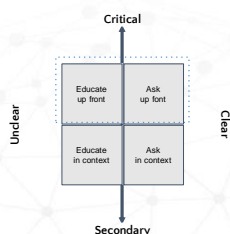


embarcadero

When to Ask Up - Critical

RX

- If it is critical to your apps function, ask when the *app first launches*
- If it is clear why you need it, just ask
 - Example: an SMS App
- If it isn't clear why your app might need the permission, then educate.
 - Example: a Note taking app that stores where the note was taken
 - Education in onboarding that greets the user when they first run the app
- If it isn't critical, wait to ask!

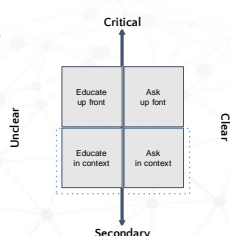


embarcadero

When to Ask Up - Secondary

RX

- Chances are you app needs permissions for features a user may or may not encounter right away.
 - For these, ask when user invokes action
 - Example: camera, share sheet, etc.
- If the reason for the permission isn't obvious, then educate before asking
 - Example: a dialog right before requesting that explains the feature and permission
- By delaying secondary requests it keeps from bombarding the user with too many permissions on first run



embarcadero

How to Ask for Permission (in the beta)



```

uses
  System.Permissions, (IFDEF ANDROID) Androidapi.JNI.Cs, Androidapi.JNI.JavaTypes, Androidapi.Helpers,
  (ENDIF) FPO.DialogService;
// in event handler
(ENTER ANDROID)
Request
  if sLocationSensorActive.IsChecked then
    procedure (const AGrantResult: TRequestResult) const AGrantResult: TRequestResult;
    begin
      if (Length(AGrantResult) = 1) and (AGrantResult[0] = TPermissionStatus.Granted) then
        | activate or deactivate the location sensor |
        LocationSensor1.Active := True
      else
        begin
          sLocationSensorActive.IsChecked := False;
          TDialogService.ShowMessage('Location permission not granted');
        end;
      end;
    end;
  else
    (ENDIF)
  end;

```

- You still need to define the permission in the project properties so it gets put in the app manifest!

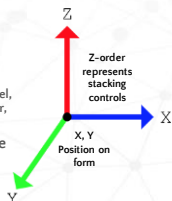
embarcadero

IMPORTANT: Features are not committed until completed and CA released.

Android Z-order and Platform Native Controls



- Previously when mixing native and styled controls, the platform native was always-on-top
- Platform native controls:
 - TWebBrowser, TMediaPlayer, TBannerAd, TMapView
 - Now use material design theme on Android 5.0 or above
- These now respect Z-order with native controls
 - TMultiView, TButton, TSpeedButton, TColorButton, TCornerButton, TPopupBox, TMagnifierGlass, TPanel, TLabel, TCheckBox, TRadioButton, TGroupBox, TStatusBar, TToolBar, TProgressBar, TTrackBar
- And these controls can *optionally* be platform native
 - TCalendar, TEdit, TSwitch, TMultiView



embarcadero

Android Z-order and Native Controls



- Easily mix styled and natively rendered controls on the same form using the new Z-Order Support
 - Very similar to what we provide for Windows and iOS today
- Natively rendered components:
 - TSwitch
 - TEdit
 - TCalendar
 - TMultiView
- TWebBrowser, TMapView etc. will use material design theme on Android 5.0 or above

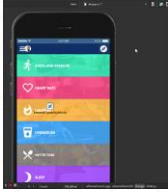


embarcadero

** Features are not committed until completed and CA released

Enhanced iOS Support

- Support for iOS 10, 11, 12 (Device and App Store)
- Support for the iOS 10 Simulator
- Support for targeting new iPhone devices



embarcadero

** Features are not committed until completed and GA released

Z-Order



embarcadero



10.3: RTL and Data

embarcadero

Delphi RTL Changes and Optimization



- Data structures growth strategy modified (slower growth) and configurable (can be replaced)
- TStringBuilder improvements and performance
- JSON improvements (and correctness)
- Generics lists related improvements and performance
 - For example, TList<T>.Add is 3x faster
- TMemIniFile performance improvement
- Regular Expressions: PCRE with UTF-16 on Windows
- The Zlib library has been updated to a newer version

embarcadero

** Features are not committed until completed and GA released

New in FireDAC and Database



- FireDAC drivers updates and improvements for
 - MySQL, MariaDB, SQL Server, PostgreSQL
 - Firebird, MongoDB, InterBase, SQLite, SQL Anywhere
- TFDBatchMoveJSONWriter and optimized Batch Move (10.2.x)
- New TClientDataSet.IncludeBlobsInDelta
- DataSnap
 - TDSRestConnection.SecureProtocols property
 - DataSnap REST uses "Content-Type=application/json"

embarcadero

** Features are not committed until completed and GA released

HTTP and REST and Cloud



- HTTP Client Library
 - Optimizations and improvements in multipart form data, MIME types (TMimeType), Accept headers
- REST Client Library
 - New events and properties in TRESTClient
- SOAP clients use the HTTP client library
 - Better HTTPS support on mobile(Soap.SOAPHTTPTrans.pas)
- Cloud
 - Azure storage emulator
 - AWS S3 support for recently added regions

embarcadero

** Features are not committed until completed and GA released

RAD Server Enhancements



- Performance enhancements and better memory management
- New EndPoint Attributes for Content-Type and Accept based URL mapping
- HTTP Verb to Custom Method Name Mapping
- Ability to Delegate Processing of a Request to a Custom Class or Component
- Ready-to-use components for JSON processing datasets and files/folders

```
type
  [ResourceName('rio')]
  T RIOResource1 = class(TDataModule)
  |
  | [ResourceSuffix('/', '')]
  | EMDataGetResource: TFMDataGetResource;
  | EmployeeConnection: TFDConnection;
  | EmployeeTable: TFDQuery;
```



** Features are not committed until completed and GA released

RAD Server Licensing



- RAD Server development features are included in Enterprise edition
 - Development / debug version of RAD Server with limited development license is built into RAD Studio
- Licenses for Apache and IIS deployment
 - Enterprise edition includes one free deployment single server license
 - Architect edition includes a free deployment multi-site license



** Features are not committed until completed and GA released

RAD Studio 10.3 SKUs



- Delphi Community Edition and C++Builder Community Edition
- Professional Editions, including Mobile platforms
 - Include IBLite (no hassle deployment)
- Enterprise Editions
 - Include DataSnap, Full FireDAC support, RAD Server
 - Single Site RAD Server deployment
 - Now with IBToGo free deployment for mobile
- Architect Editions — see next slide



** Features are not committed until completed and GA released

10.3 Architect SKU Reinvented



- Adding AquaFold Aqua Data Studio
 - General database management
- Adding Sencha Ext JS Professional Edition
 - For building Web clients
- Including multi-site deployment for RAD Server
 - Offer active since August 2018
- Removing ER/Studio Developer Edition and DB PowerStudio developer edition (existing customers retain their license)

embarcadero

** Features are not committed until completed and GA released



provides **everything** a developer needs
to **rapidly build data intensive**
cross-platform enterprise **web** applications

embarcadero


Used technologies



embarcadero

Robust components

- Grid
- Tree
- TreeGrid
- PivotGrid
- Chart
- Calendar
- Dashboard
- Form
- Form fields
- Window
- Panel
- Toolbar
- Menu
- d3 Visualizations
- etc.




Robust components

- all components are perfectly compatible
- optimized to work with huge datasets
- easily customizable



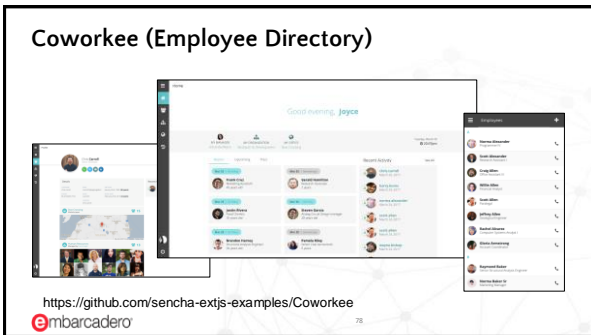
Standard Components in Ext JS - page 1

Grid	Pivot Grid	Windows	Tree Grid
Menu Buttons	Multi Column Form	Message Box	Check Tree
Toggle Buttons	Calendar	Gauges	Filtered Tree
Data Panel	Panels	DataView	Heterogeneous Tree
Link Buttons	Dialogs	Carousel	Linear data Geographical Tree
List	Tabs	Toolbars	Tree List
Forms	Navigation Tabs	Tool Tips	Audio
Chart Navigator	Sheet	Trees	Video

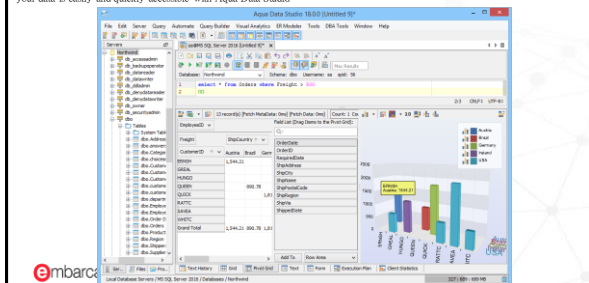


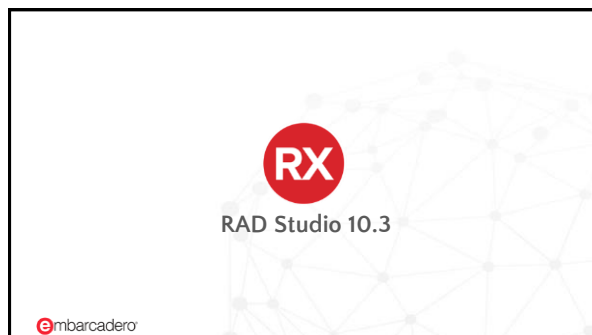






Aqua Data Studio is the universal database integrated development environment (IDE) for Database Developers, DBAs, and Analysts. It allows you to develop, access, manage, and visually analyze data. Whether you are working with relational, NoSQL, or cloud databases, your data is easily and quickly accessible with Aqua Data Studio.





RAD Studio 10.3 Rio — Highlights



- New language features!
 - C++ 17 for Win32
 - Delphi inline variables, type inference
- FireMonkey Android z-Order, Android native controls and API Level 26
- Windows 10, Windows Store, WinRT, VCL, and HighDPI improvements
- Extensive IDE UI modernization: cleaner, great to use for long periods of time
- Database: JSON and ExtJS integration, many updated FireDAC drivers
- RTL improvements: JSON, data structures, regex, REST and SOAP, and more
- RAD Server architecture extension and performance
- Large Quality and Performance improvements

10.3 Rio is one of our biggest ever releases!



** Features are not committed until completed and GA released
