



Delphi® 10 Seattle

The Fastest Application Development Platform for Windows 10, Mac, Mobile and IoT



Embarcadero® Delphi® 10 Seattle is the fastest way to build data-rich, hyper connected, visually engaging applications for Windows 10, Mac, Mobile, IoT and more. Quickly and easily update VCL and FMX applications to Windows 10 with the new Windows 10 VCL Controls, Styles and WinRT/UWP services components.

Ride the Windows 10 Wave

Windows 10 is being rapidly adopted. With Delphi 10 Seattle, get your VCL apps and users to Windows 10 now with the Windows 10 platform look and feel and new Windows 10 features and services. Use new VCL UI controls and Styles to create great looking Windows 10 apps and use new Windows 10 VCL components to access new platform features and services, including Notifications, Contracts and more.



Double the IDE Memory, Twice the Power

Delphi 10 Seattle delivers more developer productivity than ever around the daily activities of coding, building and debugging. In addition to enhanced multi-monitor support, the IDE can now access double the memory, so you can build and debug those large projects with ease. Delphi 10 Seattle has integrated and vastly improved performance with over 20 IDE productivity features. These are a set of features all developers benefit from every day.

Hyper Connected Apps for Windows, Mac, Mobile and IoT

Connected apps are distributed across platforms and multiple form factors like desktop, smart phone and tablets and also includes new IoT form factors like wearables, sensors, proximity awareness with beacons, smart light, smart sound, and gesture recognition devices for both physical and audible human input. The process of designing, building and deploying connected apps is radically simplified by the combination of new and improved features in Delphi 10 Seattle, including Wi-Fi, Bluetooth/LE components, AppTethering, EMS middleware and cloud integration through REST, like popular MBaaS services.



Here's what's new in Delphi 10 Seattle



Build and debug large projects with twice the available IDE memory



Extend existing Windows 10 applications with tethered mobile companion apps using Wi-Fi and Bluetooth connectivity



Get your apps and users to Windows 10 now!



Over 20 new IDE productivity features including the all new searchable Object Inspector



New VCL UI and Services Components for Windows 10



Build Android services that run in the background with Object Pascal

New Features

| | |
|--|---|
| Support for calling WinRT APIs, through over 40 specific Object Pascal interface units | Complete FireDAC support for the NoSQL MongoDB database, including a new FireDAC MongoDB driver |
| Support for Windows 10 Notifications using the NotificationCenter component | MongoDB specific datasets, including TFDMongoDataSet, TFDMongoQuery and TFDMongoPipeline |
| Support for contracts, the system mechanism for sharing information with other Windows 10 applications using the new SharingContract component | MongoDB API wrapping classes, including TMongoConnection, TMongoDatabase, TMongoCollection and more |
| New VCL Controls including ToggleSwitch, SplitView, SearchBox, ActivityIndicator and RelativePanel with Win 10 Styling and support. Can also be used on previous versions of Windows | Specialized JSON readers and writers, including the new TJsonTextReader and TJsonTextWriter classes, and support for Extended JSON |
| Windows 10 specific VCL styles to build applications matching Microsoft's Modern look and feel | MongoDB query, pipeline, update commands, and more with fluent methods builders |
| VCL Styling improvements, including support for styling common dialogs and the TWebBrowser component | JSON (JavaScript Object Notation) processing using a JSON.NET implementation for JSON streaming with new readers and writers (including base TJsonReader and TJsonWriter classes) |
| IDE built with large memory address model, to provide significantly more memory to the embedded compilers, integrated debuggers, and various tools executed in the IDE process | Binary JSON (BSON) readers and writers support, as part of the same JSON.NET architecture (including the new TBsonReader and TBsonWriter classes) |
| Form designer option to hide/show non-visual controls icon (reducing form design potential clutter) | JSON and BSON fluent method builders, including the TJSONArrayBuilder and TJSONObjectBuilder classes |
| Improved multi-monitor support in the IDE, with the ability to place most forms and panes on different secondary monitors | JSON and BSON fast forward-only iterator (TJSONIterator) |
| Object Inspector contents can be filtered to display specific elements | Example of the use of the FDSchemaAdapter component in DataSnap applications |
| Full customization of the Object Inspector layout, with the ability of hiding the description panel, the quick actions, and the new filter panel | Touch animation for Android platform |
| Unsaved file auto-recovery for the IDE – unsaved work is periodically saved to a temporary location | Modern looking SelectDirectory function for VCL applications and the IDE |
| Structure View Icons representing the corresponding component | New TBeaconDevice class for turning a device on one of the supported platforms into a "beacon" |
| Enhanced IDE Project Options to easily enable High-DPI Awareness in your applications, plus Windows 8.1/10 multi-monitor support for VCL applications | ScrollBar platform controls for iOS |
| DUnitX unit testing support for mobile platforms (iOS and Android) | DataSnap clients uses System.NET for HTTP and HTTPS, with no need to deploy the OpenSSL client library |
| Prototype synchronization – as you change the prototype for a function, you can use this feature to synchronize the interface and implementation sections to match | Shortcuts to increase/decrease the size of the font in the code editor |
| StyleViewer for Windows 10 Style in Bitmap Style Designer | FireMonkey apps can receive intents, regardless of the source (email, web link, other app). A new sample demonstrates this ability |
| Windows 10 specific FireMonkey styles to build applications matching Microsoft's Modern look and feel | FireMonkey controls zOrder support on Windows |
| FireMonkey native style presentation for Windows for Edit and Memo platform controls | Remote iOS 64-bit device debugging |
| Enhance Clipboard support to allow copy/paste of bitmaps in FireMonkey desktop applications | Support for Android Services in the IDE, including wizards to create Android Services and to add them to an existing Android app |
| Mouse-over Hints support for FireMonkey visual controls on desktop | Plus many other great features |

Delphi 10 Seattle Editions

| | Professional | Enterprise | Ultimate | Architect |
|-------------------------------------|--|--|---|---|
| | Delphi 10 Seattle Professional Edition is designed for building rich stand-alone client applications for Windows and OS X with local data persistence. | Delphi 10 Seattle Enterprise adds iOS and Android targeting plus native Client/Server connectivity with all major Enterprise Databases and flexible Middleware for building powerful n-tier solutions. | Delphi 10 Seattle Ultimate Edition includes all of the capabilities of Enterprise plus a suite of powerful database tools to help develop and manage your data. | Delphi 10 Seattle Architect includes all of the capabilities of Enterprise plus Data Modeling to help you reverse and forward engineer your data. |
| Windows, OSX, iOS, Android Apps | Windows and OS X Only | All | All | All |
| Local Application Data Storage | X | X | X | X |
| Client/Server Database Connectivity | | X | X | X |
| DataSnap & EMS n-Tier Middleware | | X | X | X |
| DBPowerStudio Developer Edition | | | X | |

Delphi 10 Seattle System Requirements

| | |
|--|--|
| <ul style="list-style-type: none"> • 1 GB RAM (2 GB+ recommended) • 7-41 GB free hard disk space depending on edition and configuration, including space required for temporary files • DVD-ROM drive (if installing from a Media Kit DVD) • Basic GPU – Any vendor DirectX 9.0 class or better (Pixel Shader Level 2) • Intel® Pentium® or compatible, 1.6 GHz minimum (2GHz+ recommended) | <ul style="list-style-type: none"> • Mouse or other pointing device • Microsoft® Windows 10 (32-bit and 64-bit) • Microsoft® Windows 8 or 8.1 (32-bit and 64-bit) • Microsoft® Windows 7 SP1 (32-bit and 64-bit) • Delphi can also be run on OS X by using a virtual machine (VM) such as VMware Fusion or Parallels hosting Windows 7, 8 or 10 |
|--|--|

| For developing 64-bit Windows applications | For developing Mac OS X applications | For developing iOS applications | Supported Deployment Platforms |
|--|---|---|---|
| PC running a 64-bit version of Windows or a 32-bit development PC connected with a PC running a 64-bit version of Windows. | PC running Windows connected with an Intel-based Mac or a Mac running Windows in a VM, with 2GB RAM or more, running OS X 10.11 (El Capitan), 10.10 (Yosemite) or 10.9 (Mavericks). | PC running Windows connected with an Intel-based Mac or a Mac running Windows in a VM, with 2 GB RAM or more, running OS X 10.10 or 10.9 with Xcode 6. An Apple Developer account is required to deploy iOS apps to physical devices. | PCs and tablets with Intel/AMD processors running Windows 7, 8, 8.1, 10, Server 2008 or Server 2012. Macs running OS X 10.9, 10.10 or 10.11. iPhone, iPad or iPod Touch running iOS 7 through iOS 9.0. Android phones and tablets: ARMv7 devices with NEON support, running Ice Cream Sandwich (4.0.3-4.0.4), Jelly Bean (4.1.x, 4.2.x, 4.3.x) or Kit Kat (4.4.x) and Lollipop (5.x). |

Download a Free Trial Now! Visit www.embarcadero.com/trial

sales@embarcadero.com | www.embarcadero.com