01

What is RAD Server? Introduction

Today's computing landscape is no longer confined to a desktop, device, server or data center. Applications are being moved from the desktop to multiple devices, network edge connections, and to on-premises, public and hybrid cloud services. With RAD Server and RAD Studio you can build solutions that cover the wide spectrum of your company's (and customers) computing needs and business requirements.

This documentation will show you how to quickly design, build, debug and deploy service based multi-tier applications using RAD Server's REST based API hosting engine, components and technologies that are available in RAD Studio, Delphi and C++Builder Enterprise and Architect editions.

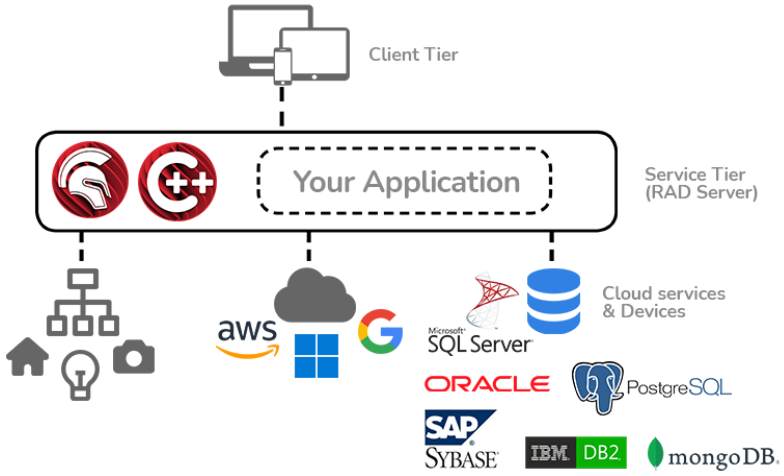
note

Throughout the RAD Server documentation and source code, you'll see references to EMS (Enterprise Mobility Services). EMS was the original name of what is now called the RAD Server product.

RAD Server Overview

Embarcadero's RAD Server provides a turn-key application foundation for rapidly building and deploying services-based applications using Delphi and C++Builder. RAD Server supports the REST (Representational State Transfer) protocol with JSON (or XML) parameter passing and return results. You can publish APIs, manage users and devices that are connected to the RAD Server, capture analytics about the use and users of applications, connect to local and enterprise databases using the

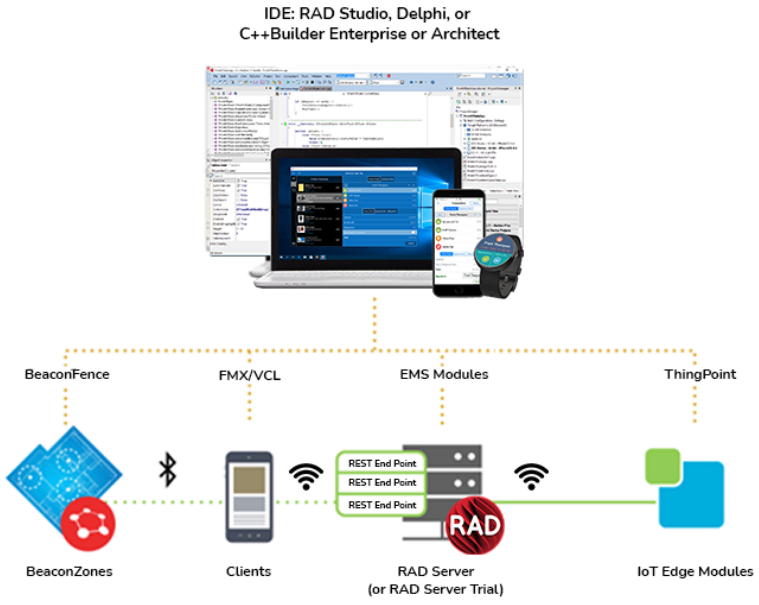
FireDAC components and much more. RAD Server also supports user authentication, push notifications, geolocation and data storage.



Develop and Test REST endpoints and Location Tracking

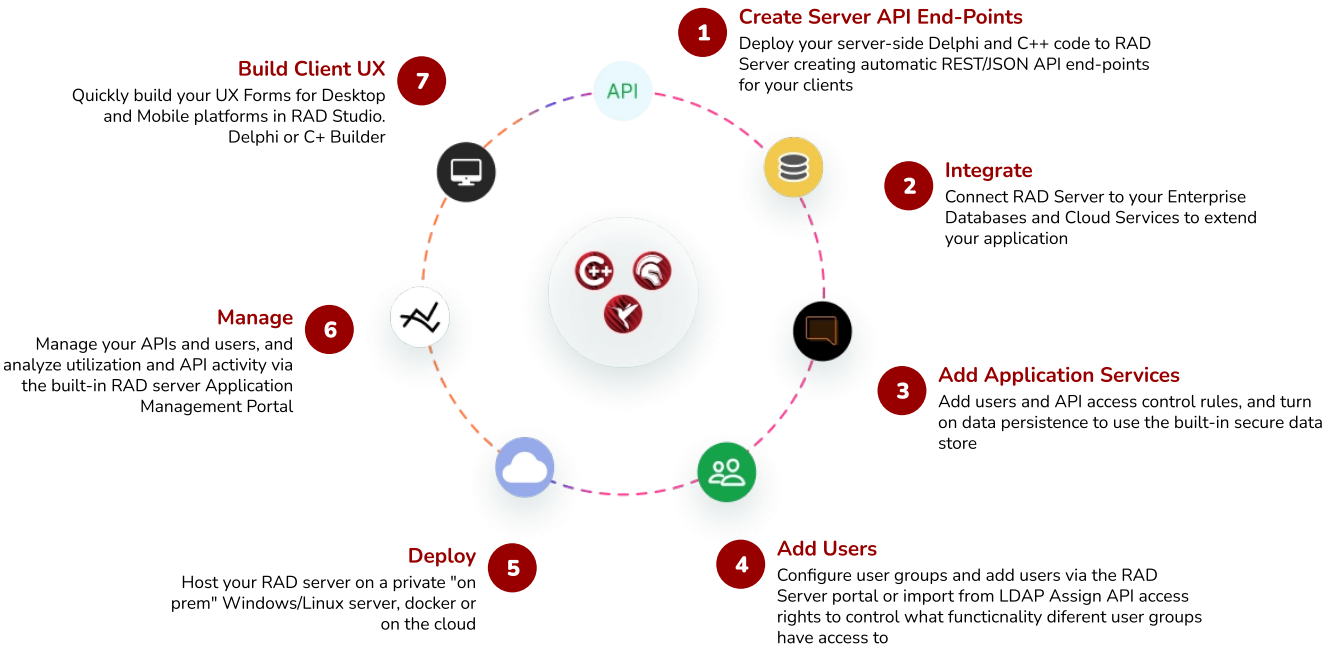
With RAD Server's wizards, components and tools you can quickly develop new middleware and back-end applications or migrate your existing Delphi and C++Builder client/server applications to a RAD Server based application to run on a server or in the cloud. You can publish your endpoints for REST calls from desktop, mobile, console, web and other types of applications. RAD Server comes with a full set of the tools, components, database connectivity and interfaces that you will rely upon in building your service applications.

RAD Server applications can be deployed on top of Microsoft Windows IIS and Apache web servers and you can deploy your Delphi based services to Linux Intel 64-bit servers. For C++Builder support for Linux stay tuned for updates to the Embarcadero RAD Studio blogs.



Building RAD Server based applications – Seven Key Aspects

To build RAD Server based applications, the diagram below guides developers through seven aspects and development phases.



Multi-Tier Development Made Easy

To start, create your server REST/JSON API-based endpoints (you can also use XML instead of JSON if required). Next you will extend your endpoints by integrating a wide range of databases, cloud services and other technologies.

You can add more application endpoints to users and create API access control rules. You can write code that leverages RAD Server's built-in secure data store to keep track of persistent data. You can create user groups and add users via console portal and import and authenticate users via LDAP-based API services.

After you have developed and debugged your applications you can host RAD Server applications on a private on-premises Windows and Linux servers. You can also migrate your applications to cloud systems like Amazon AWS, Microsoft Azure, Google and other cloud providers.

After your application is put into production, you can manage access to your APIs, control users access and analyze the utilization of your endpoint API activity with built-in application management interfaces. Finally, you can build desktop, mobile, Web, console and other application types supported by RAD Studio. You can also build modern Web client applications using the Sencha's Ext JS set of components and use other tools and programming languages to build client applications that support your RAD Server application's REST/JSON functionality.

Requirements for Building a RAD Server Applications

The following sections contain the product and technical requirements for building, testing and deploying RAD Server applications. Unless otherwise noted, “RAD Studio” and the IDE apply to the RAD Studio, Delphi and C++Builder products.

Using the RAD Studio IDE

A RAD Studio Enterprise or Architect Edition with a commercial license is required to build RAD Server applications. The trial edition of RAD Studio Enterprise can be used for 30 days for development and testing. The trial edition does not support deployment to a production server.

RAD Server Testing and Deployment Licenses

The free 30 day RAD Studio trial includes RAD Server 5-user development trial. RAD Server deployment licenses are included in Enterprise and Architect commercial editions of RAD Studio. RAD Studio Enterprise includes a single-site deployment license for RAD Server whereas RAD Studio Architect edition includes a multi-site deployment license for customers who are on active Update Subscription. Since RAD Studio Alexandria, Enterprise as well as Architect include the option of deploying RAD Server Lite in a multi-site environment.

RAD Server requires an InterBase encrypted database as part of deploying your applications in a Production Environment. You will need to use a valid RAD Server license to install this version of InterBase.

In case you want to deploy your application using InterBase as well, you will need 2 instances of InterBase running: one for your application and one for RAD Server.

Roundup of Core RAD Server Features

RAD Server provides developers with a wide spectrum of features for building REST-based service applications. RAD Server (formerly known as EMS) was first introduced in RAD Studio version XE7. Since that first release enhancements and new capabilities have been added to address the needs of developers and add support for new platforms, architectures and techniques.

Core Features

Here is a list of some of RAD Server’s core features that you’ll want to leverage in building your services-based applications.

- **REST End Point Publishing** – RAD Server implements a turnkey foundation for your application back end APIs and Services. RAD Server provides an easy to use API for publishing your business logic. Delphi or C++ code can be hosted as an API and auto-published as REST/JSON endpoints which are measured and managed by RAD Server. Endpoint publishing features include:

  - Access Control – You can set up group and user level access, with authentication, to all application APIs and control who has access to your application's API functionality. Create your own users and groups or import them automatically from your LDAP infrastructure.

  - API Analytics – All REST API end-point activity is recorded and measured for robust statistics tracking and analytics. You can analyze user, API, and services activity daily, monthly and yearly to gain insight into how your application is being utilized. You can also filter activity for all resources or by specific groups, users, device installations, and more. You can also export analytics to a CSV file for additional analysis with other tools.

  - Desktop, Mobile & Web Client Applications – All C++ and Delphi code hosted on RAD Server is published as REST/JSON end points consumable by any type of client application on multiple platforms for extreme flexibility and future-proofing.

- **Integration Middleware** – RAD Server provides multiple integrations out of the box with connectivity to external servers, applications, databases, smart devices, cloud services and other platforms. Integration capabilities include:

  - Enterprise Data – RAD Server delivers high performance built-in connectivity to all popular Enterprise RDBMS servers. Database connectivity uses the FireDAC set of components and libraries for easy connectivity with data from a variety of sources.

  - Cloud Services – With RAD Server you can easily integrate REST cloud services from a variety of cloud, social, and BaaS platforms such as Google, Amazon, Facebook, Kinvey, and more.

- **Application Services** – RAD Server includes a collection of ready to use built-in services to power your application. RAD Server includes core functions such as user directory services and user management, push notifications, user location tracking, and built-in data storage. Some of these application and device services include:

  - Push Notifications – using RAD Server you can send programmatic or on-demand notifications to your application users and their devices. RAD Server currently supports push notification systems including Apple Push Notification service (APNs) and Google FireBase Cloud Messaging (FCM). You can also write custom code to connect with other push notification systems.

  - Built-in Secure Datastore – With RAD Server's support for securing an InterBase server's encrypted datastore you can use built-in APIS TO store and retrieve JSON data without requiring a separate database server.

  - User/Group Management –Using RAD Server APIs you can create and manage your users, user groups, and control access via the RAD Server console (RSConsole.exe). Integrate your ActiveDirectory (LDAP) or develop your own custom authentication middleware.

  - User Location/Proximity – Your RAD Server applications can leverage RAD Studio's support for GPS, beacon and beacon fence technology. RAD Server applications can track user

movement, both indoors and outdoors, and respond to proximity events when users enter and exit custom beacon zones or approach designated beacon points.

    - Static Files Provider – Map URLs to folders and return the content of files like HTML, JS, CSS, images and more. This is extremely handy in small deployments (IE: using RAD Server Lite) or in development environments.

- **API Documentation** – Create easily documentation of your API using attributes and the built-in Swagger OpenAPI integration. Embed Swagger UI into RAD Server itself or configure it in remote instances through the auto-generated YAML and JSON files.
- **Easy to Deploy** – RAD Server is easy to develop, deploy and operate making it ideally suited for ISVs and OEMs building re-deployable solutions. Deploy it on Windows, Linux or Docker.

See Also

For the latest updated information about installation of RAD Studio and deployment of RAD Server based applications please refer to the following Embarcadero online links.

RAD Server Product Overview

RAD Studio Installation Notes

RAD Studio and RAD Server Supported target platforms

RAD Server Database Requirements for a Production Environment

RAD Studio's Platform Status Page

InterBase

FireDAC

FireDAC Supported Databases

RAD Studio Enterprise Mobility Services

RAD Studio Product Feature Matrix (PDF - Check RAD Server section)

Swagger Open API

EMS Push Notifications

Apple Push Notification service (APNs)

Firebase Cloud Messaging (FCM)