

Extending to the Cloud





Agenda

- A changing world: Libraries vs. APIs
- Using REST APIs from Delphi
- ApiLayer and some of its APIs
- The AWS SDK for Delphi by Appercept
- Creating REST Servers
- Deploying to the Cloud (and Docker containers)



Libraries vs. APIs: Clarifying the Terms

Libraries

Traditionally, you'd add features to an application by writing code or using existing third-party libraries (including component libraries in Delphi).

- Compiling the library source code into the application
- Binding a library binary to your executable file
- Using a dynamic library loaded at runtime



Libraries vs. APIs: Clarifying the Terms

APIs

The Application Programming Interface term was for many years a reference to a local or platform API (like the Windows API, part of its SDK). Nowadays, “the term API is often used to refer to web APIs” <https://en.wikipedia.org/wiki/API>

- Web APIs are generally HTTP(S) based web services you invoke from a client or a server application to perform a very specific common task
- From the application developers perspective, they are just HTTP calls



Libraries Advantages

Using a **local library** has the advantage of:

- Generally faster execution
- No need of an Internet connection
- Possibly the ability of fine tuning and customizing the feature and tying it better with the code and UI
- At the security level, you don't need to worry about the communication being encrypted to avoid someone accessing to application data in transit



REST APIs Advantages

Using a **REST APIs** has the advantage of:

- Keeping the app smaller, as you don't need to distribute extra library code
- Always use the latest version of the library, as it can be updated separately from the client application
- Ability to access real time data, external storage, collaboration information and more
- At the security level, you don't need to worry about updating the library in case any issue is discovered, having to deploy a new version of the application



Libraries vs. APIs: The Economy Side

Finally, there is often a difference between the **costs of a library vs. a REST API**:

- Unless the library is free and open source, it would generally have an upfront fixed per-developer cost
- Most REST APIs have a pay per use model, often with a free level you can use for development and testing
- So the REST API might be cheaper for limited use, but become more expensive for extensive use
- Using REST APIs you can offload some of the processing power requirements out of your clients/servers



REST Clients in Simple Terms

- ❑ HTTP(S) as transport protocol
- ❑ XML or JSON for data communication
- ❑ Simple data passing
 - ❑ Plain document or URL with params
 - ❑ Response: plain text or mime



REST Clients in Delphi

- ❑ HTTP/S Connectivity
 - ❑ IdHTTP component (requires external OpenSSL)
 - ❑ HTTP Client Library (native platform APIs and SSL)
 - ❑ REST Client Library (uses HTTP Client Library)
 - ❑ Also SOAP clients support (uses HTTP Client Library)

- ❑ Data Processing
 - ❑ XMLDocument component
 - ❑ JSON processing classes (System.JSON)
 - ❑ REST Data Adapter



REST App Debugger

If you plan using Delphi's REST Client library, start from the REST Debugger





ApiLayer

- ❑ A rich set of APIs including a third party marketplace
 - ❑ <https://apilayer.com/>
 - ❑ Some easy examples:
 - ❑ <https://weatherstack.com/>
 - ❑ <https://vatlayer.com/>
 - ❑ <https://positionstack.com/>
 - ❑ <https://countrylayer.com/>
- ❑ Also from Idera Dev Tools
 - ❑ <https://www.filestack.com/>





Delphi Cloud Clients

- ❑ “Classic” Amazon support (S3 and more)
- ❑ “Classic” Azure support
- ❑ New Appercept AWS SDK for Delphi



Appercept's AWS SDK for Delphi (1/2)

- ❑ Long term distribution deal
- ❑ Included in Enterprise and Architect editions
- ❑ New releases coming at a constant pace

- ❑ Install via GetIt Package Manager
- ❑ Demos on GitHub
 - ❑ <https://github.com/appercept/aws-sdk-delphi-samples>





Appercept's AWS SDK for Delphi (2/2)

❑ APIs available in October 2022

- ❑ Polly
- ❑ Simple Email Service (SESV2)
- ❑ Simple Notification Service (SNS)
- ❑ Simple Queue Service (SQS)
- ❑ Simple Storage Service (S3)
- ❑ Textract
- ❑ Amazon Translate
- ❑ AWS Key Management Service (AWS KMS)
- ❑ AWS Secrets Manager





Some Blog Posts

Using Apilayer REST APIs from Delphi (January 2021)

<https://blog.marcocantu.com/blog/2021-january-apilayer.html>

A Delphi Demo with WeatherStack by API Layer (July 2022)

<https://blog.marcocantu.com/blog/2022-july-delphi-weatherstack.html>

Covers the content of this session, including some of the theory

Embarcadero Blogs Posts about AWS SDK (April 2022 to October 2022)

<https://blogs.embarcadero.com/author/appercept/>

Clients source code at

github.com/marcocantu/DelphiSessions/tree/master/Ekon26_RestClients



Creating REST Servers

Core idea is supporting HTTP(s) protocol, return JSON (or XML or Data in any format)

Delphi Web Server Technologies => Windows and Linux

- WebBroker (integrated in IIS, Apache, standalone, etc)
- DataSnap (old style)
- RAD Server
- (IdHTTPServer)
- Third party REST frameworks



RAD Server Objectives

REST web service engine, including

- Ready-to-use services

 - Includes users, group, permissions, push notifications...

- Plug-in architecture for modules

- Companion support database

- Web monitoring UI and usage analytics

- Local management console



What is RAD Server in One Slide

The engine for multi-tier applications

Deploy on premise or in the cloud

Comes with tons of ready-to-use features: users and groups management, multi-tenancy, detailed analytics, push notifications support, swagger documentation support, beacon fencing...

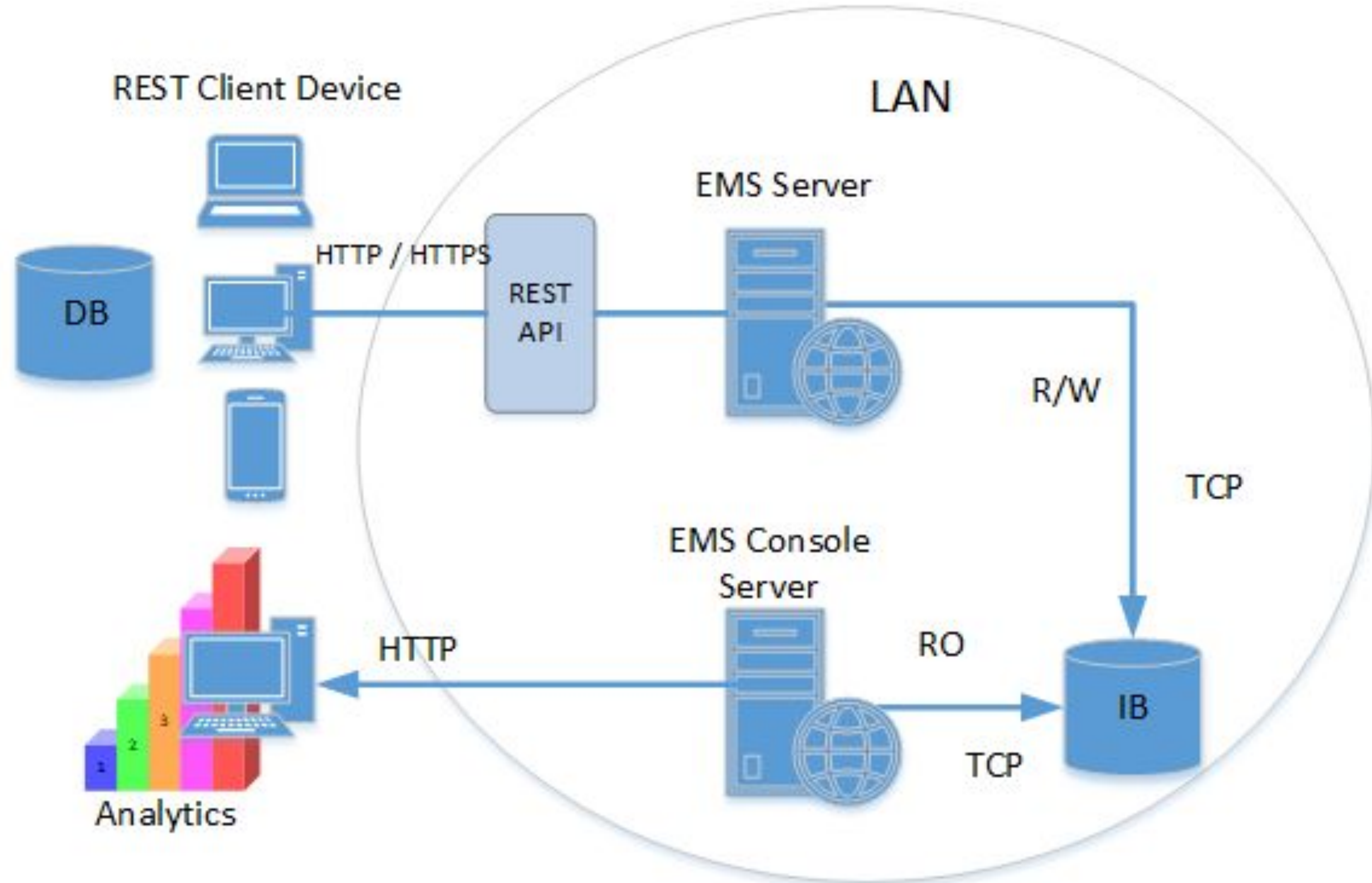
Migrate your data access logic to a web service

Build flexible and thin Windows VCL clients, multi-device clients for desktop and mobile or Web + JavaScript clients

Works on Windows and Linux, IIS and Apache



RAD Server Architecture





First Steps In Publishing APIs

- Creating an EMS resource package
 - Implementing the REST APIs
 - Executing the server
 - Calling the APIs form a browser or a client app
- Examining the console
 - And the user management app
- Configuration wizards
 - And the EMS.INI file key settings



Delphi, Docker, PAServer

Docker PAServer

Install support for Delphi IDE deployment

Linux PAServer Docker script

<https://github.com/Embarcadero/paserver-docker>

Ready-to-use DockerHub Image

<https://hub.docker.com/r/radstudio/paserver>

Commands:

```
docker pull radstudio/paserver
```



Delphi, Docker, RADServer

Docker RADServer with/out IB

Install support for RAD Server alone or with InterBase

Linux RADServer Docker scripts

<https://github.com/Embarcadero/pa-radserver-docker>

<https://github.com/Embarcadero/pa-radserver-ib-docker>

Ready-to-use DockerHub Images

<https://hub.docker.com/r/radstudio/pa-radserver>

<https://hub.docker.com/r/radstudio/pa-radserver-ib>



RAD Server Scalability (and Docker)

RAD Server is Stateless

- RAD Server REST calls are stateless

- Use a proxy to route different calls to different instances

- Requires unified IB DB backend (users management)

Deploy to Scale

- 1 server with IB (and License)

- N servers/containers with RAD Server engine only

Q&A



marco.cantu@embarcadero.com