



Embarcadero Conference

RAD Server Overview

Fabio Biasi Mello Rubim

Embarcadero

Conference



Fabio Rubim

- Desenvolvedor Delphi há mais de 7 anos
- Desenvolvedor sênior na Embarcadero Professional Services
- Graduado em Análise e Desenvolvimento de Sistemas pela Fatec Sorocaba
- Atualmente trabalho com aplicações mobile, web, DataSnap e contribuo com atualizações nos materiais didáticos dos treinamentos da Embarcadero.

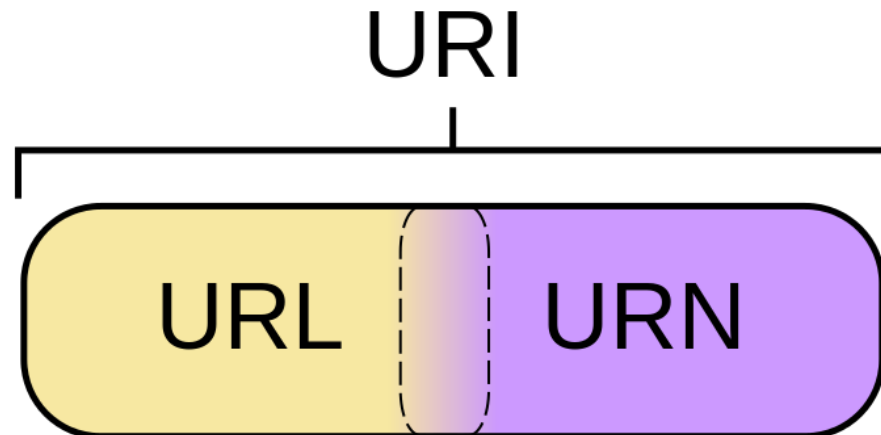


Agenda

- O que são URN, URL e URI?
- O que é uma API?
- WebServices ?
- O que é o SOAP?
- O que é um Resource?
- O que é REST, JSON e RESTful?
- O que é o RAD Server?
 - Características
 - E o EMS?
- Licenciamento
- Exemplos
- Beacon Fence

• O que são URN, URL e URI?

- URN: Uniform Resource Name, identifica um recurso, mas não informa como localizá-lo. Ex: [/contatos](#)
- URL: Uniform Resource Locator, identifica um recurso e diz como localizá-lo. Ex: <http://www.embarcadero.com.br/users>
- URI: Uniform Resource Identifier, é uma sequência de caracteres para identificar um documento/recurso de forma única.



O que é uma API?

- API: Application Programming Interface
- Uma DLL é um tipo de API
- Acessível através da programação
- Web API
- Utiliza um URI como identificador
- Permite integrar softwares diferentes
- Exemplos: Google Maps, Spotify, alguns bancos e fintechs (Open Banking)





WebServices

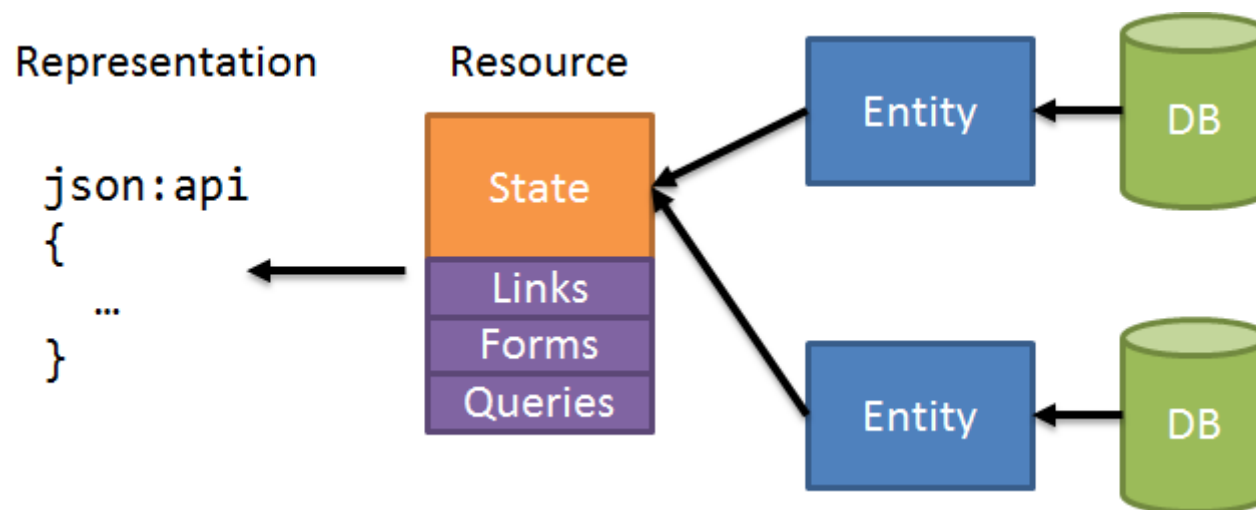
- Utiliza APIs
- Utilizado na integração de softwares
- Comunicação entre plataformas diferentes
- Muito utilizado na integração entre departamentos

O que é o SOAP?

- SOAP: Simple Object Access Protocol
- Empacota o XML para representar os dados e o protocolo HTTP para transportá-los
- Criam-se métodos para as funções de um sistema. Em um CRUD de *Clientes* teríamos basicamente funções para:
 - Criar um cliente Ex: <http://www.embarcadero.com.br/addCliente>
 - Pesquisar por clientes Ex: [/buscaCliente](#)
 - Atualizar os dados de um cliente Ex: [/atualizaCliente](#)
 - Deletar um cliente Ex: [/excluiCliente](#)

O que é um Resource?

- Resource: Qualquer informação que possa ser nomeada. Um documento, uma imagem, um objeto e etc. O cliente efetua uma requisição e obtém o recurso representando por alguma estrutura de dados. Por exemplo:
 - Um cliente é um recurso, poderíamos utilizar: /cliente



O que é REST, JSON e RESTful?

- REST: Representational State Transfer
 - É uma arquitetura, diferente do SOAP
 - *Representational?! Dados em JSON, XML, YAML e etc.*
 - Utiliza o protocolo HTTP
 - Utiliza-se dos métodos (ou verbos) HTTP para realizar as operações em um recurso. Os métodos mais utilizados são o GET, POST, PUT e DELETE.
 - Utiliza do conceito de API



O que é REST, JSON e RESTful?

- Dependendo do tipo de solicitação, podemos criar, atualizar, pesquisar ou deletar um cliente
- Exemplo: www.embarcadero.com.br/cliente
 - Listar os clientes : **GET** www.embarcadero.com.br/cliente
 - Exibir o cliente 1 : **GET** www.embarcadero.com.br/cliente/1
 - Criar um novo cliente : **POST** www.embarcadero.com.br/cliente
 - Atualizando o cliente 2 : **PUT** www.embarcadero.com.br/cliente/2
 - Excluindo o cliente 3 : **DELETE** www.embarcadero.com.br/cliente/3



O que é REST, JSON e RESTful?

- JSON: JavaScript Object Notation
- É um formato pra representar os dados
- É mais fácil de entender por humanos, menos “pesado” e verboso que o XML
- Muito utilizados em APIs REST
- REST suporta outros formatos como XML, HTML e YAML

O que é REST, JSON e RESTful?

XML

```
<empinfo>
  <employees>
    <employee>
      <name>James Kirk</name>
      <age>40</age>
    </employee>
    <employee>
      <name>Jean-Luc Picard</name>
      <age>45</age>
    </employee>
    <employee>
      <name>Wesley Crusher</name>
      <age>27</age>
    </employee>
  </employees>
</empinfo>
```

JSON

```
{ "empinfo" :
  {
    "employees" : [
      {
        "name" : "James Kirk",
        "age" : 40,
      },
      {
        "name" : "Jean-Luc Picard",
        "age" : 45,
      },
      {
        "name" : "Wesley Crusher",
        "age" : 27,
      }
    ]
  }
}
```

O que é REST, JSON e RESTful?

- RESTful é um termo utilizado para dizer que uma aplicação servidora segue as *REST constraints*.
- Na prática uma aplica não RESTful teria um recurso para cada funcionalidade em um CRUD, mesmo utilizando, por exemplo, o método GET do HTTP. Por exemplo:
 - Listar os clientes : **GET** www.embarcadero.com.br/listarCliente
 - Exibir o cliente 1 : **GET** www.embarcadero.com.br/listarCliente/1
 - Criar um novo cliente : **GET** www.embarcadero.com.br/addCliente
 - Atualizando o cliente 2 : **GET** www.embarcadero.com.br/updateCliente/2
 - Excluindo o cliente 3 : **GET** www.embarcadero.com.br/deleteCliente/3

O que é o RAD Server?

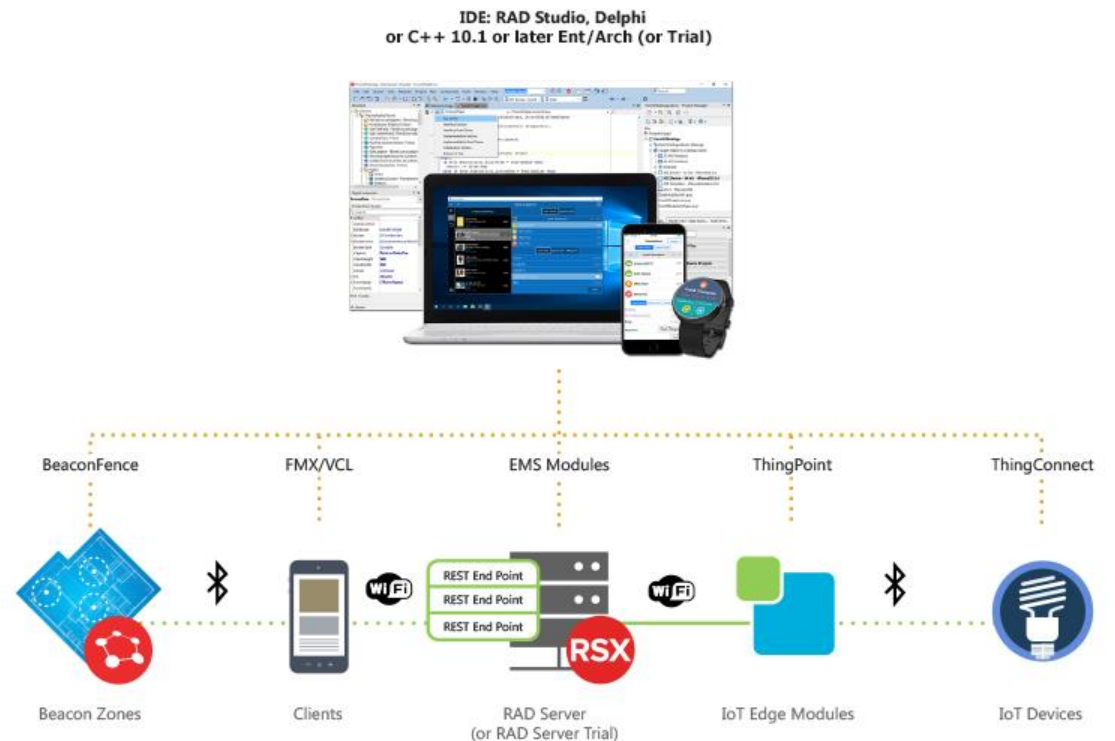
- Não é uma IDE à parte
- Aplicações backend
- Permite a publicação e gerenciamento de *endpoints*
- Aplicações orientadas a serviço (SOA)
- Infraestrutura pronta para criar serviços

O que é o RAD Server?

- Preocupação em desenvolver as regras de negócio
- Middleware de integração
- Container de aplicações
- Sua aplicação é uma BPL
- Possui Analytics para as APIs

Características do RAD Server

- Coleção de serviços já prontos para a utilização
 - Gerenciamento de usuários e grupos
 - Autenticação e autorização
 - Push notifications
 - Beacon Fence
 - Rastreamento e localização
 - Integrar dispositivos IoT





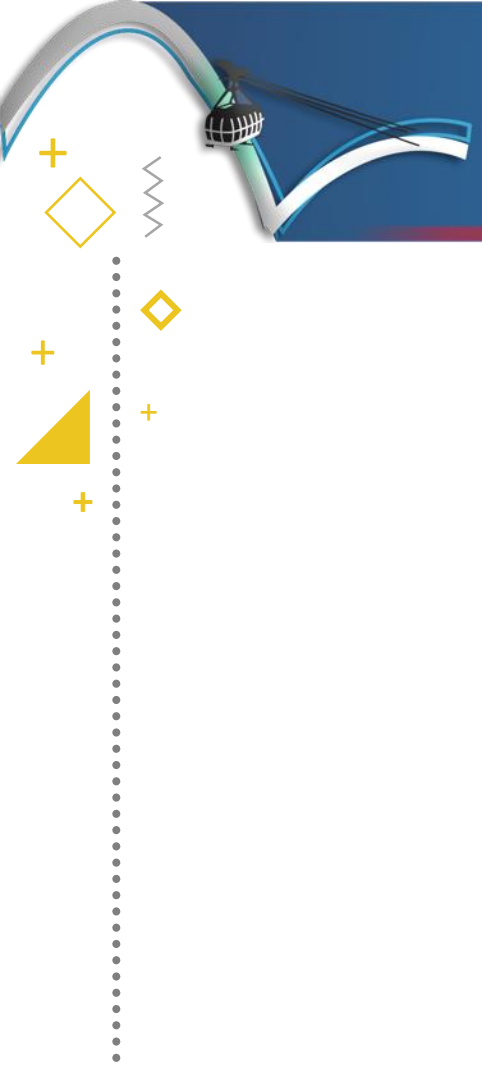
E o EMS?

- Enterprise Mobility Services, já possuía:
 - Publicação de REST *endpoints*
 - Gerenciamento de usuários
 - API Analytics
 - Push Notification
- O RAD server é o EMS com mais funcionalidades:
 - Beacon Fence: Localização por proximidade
 - IoT EdgeWare: Componente para trabalhar com produtos IoT.



Licenciamento

- O RAD Studio Tokyo inclui um trial de deploy para até 5 usuários
- Para desenvolver e testar REST *endpoints*, Location Tracking, ou IoT Edgware é necessário uma licença Enterprise ou Architect do Tokyo
- As edições Enterprise e Architect também possuem um trial para efetuar o deploy para até 5 usuários
- Deploy
 - Single site: Permite instalar a licença em um servidor e a conexão de quantos clientes o mesmo suportar
 - Multi site: Instalação em múltiplos servidores e cada servidor receber quantas conexões suportar



Exemplos

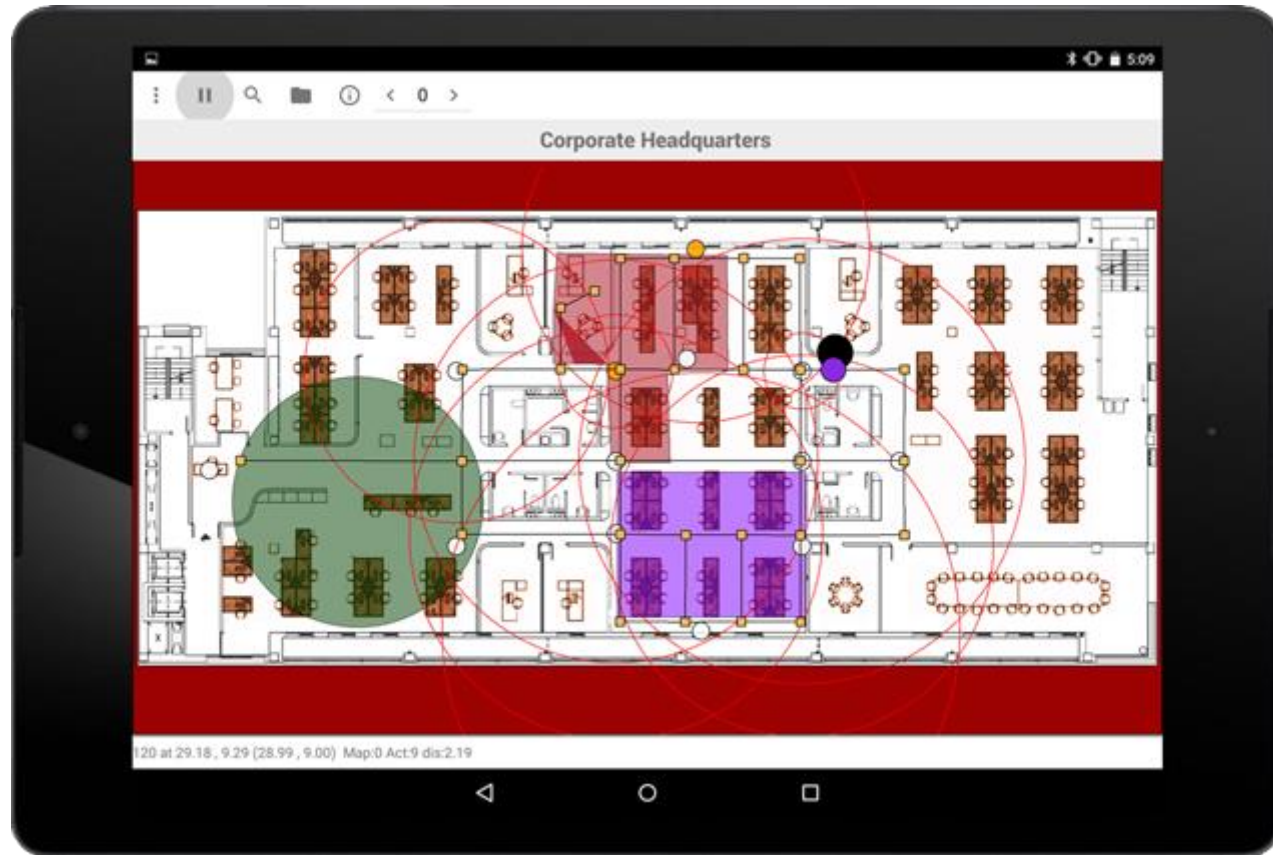
Beacon Fence

- GPS *Indoor*
- *Beacons: Utiliza BLE (Bluetooth Low Energy)*

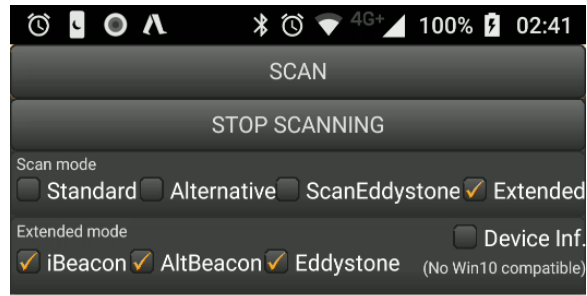


Beacon Fence

- Editar o mapa com o *Beacon Fence Map Editor*



Beacon Fence



OBRIGADO



Contato:

br.linkedin.com/in/fabiorubim

frubim@embarcadero.com.br

fabiorubim@hotmail.com

Embarcadero

Conference





Embarcadero Conference