

embarcadero®

Delphi Academy

Dicas rápidas, truques e técnicas



Desenvolvimento WEB Parte #1

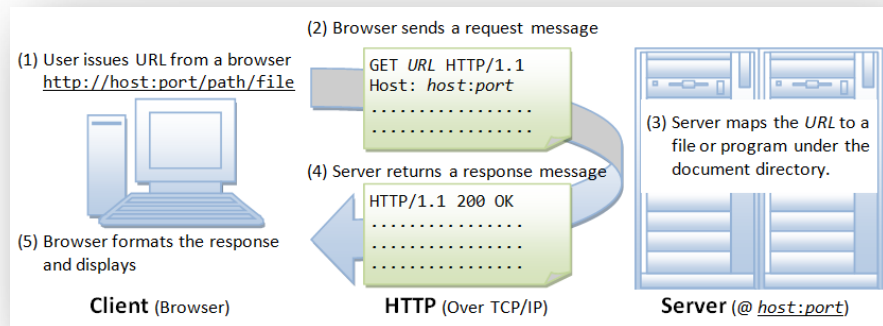
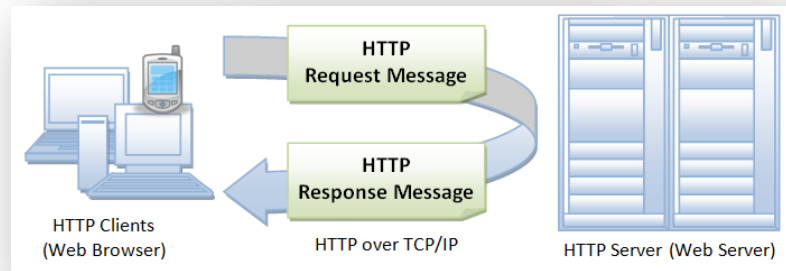
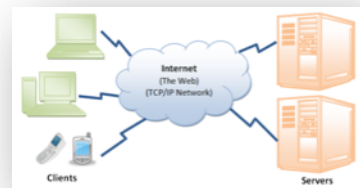
Fernando Rizzato
Lead Software Consultant, *Latin America*

AGENDA

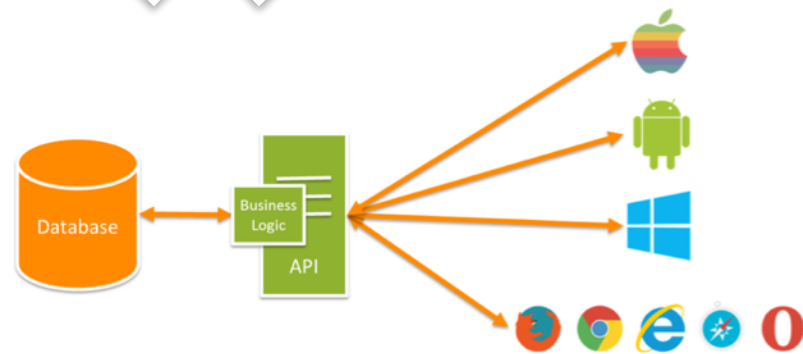
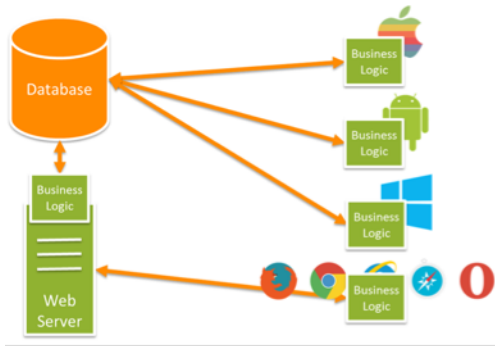
- Conceitos
 - Comunicação HTTP
 - Aplicações Desktop versus WEB
 - Browser: HTML, CSS, JavaScript
- Arquiteturas e Possibilidades
 - *Stateful* versus *Stateless*
 - Web Server e/ou App Server
- Demos
 - #1: WebBroker
 - #2: IW + BootStrap
 - #3: RAD Server + Ext JS

COMUNICAÇÃO HTTP

- *Hypertext Transfer Protocol* (HTTP) é um protocolo de aplicação para sistemas de informação distribuídos
- O HTTP é a base da comunicação de dados para sistemas disponíveis na Internet



APLICAÇÕES DESKTOP VERSUS WEB



BROWSER: HTML, CSS, JAVASCRIPT

- **HTML** significa *Hyper Text Markup Language*
 - O HTML descreve a estrutura das páginas da Web usando marcadores
 - Os elementos HTML são os blocos de construção de páginas HTML
 - Os elementos HTML são representados por tags
 - As tag HTML rotulam partes de conteúdo, como "cabeçalho", "parágrafo", "tabela", e assim por diante
 - Os navegadores não exibem as tags HTML, mas usam-nas para renderizar o conteúdo da página

```
<!DOCTYPE html>
<html>
<head>
<title>Page Title</title>
</head>
<body>

<h1>My First Heading</h1>
<p>My first paragraph.</p>

</body>
</html>
```

BROWSER: HTML, CSS, JAVASCRIPT

- **CSS** significa *Cascading Style Sheets*
 - O CSS descreve como os elementos HTML devem ser exibidos na tela, papel ou em outros meios
 - O CSS economiza muito trabalho. Ele pode controlar o layout de várias páginas da Web de uma só vez
 - Conjuntos de estilo externos são armazenadas em arquivos CSS

```
body {  
    background-color: lightblue;  
}  
  
h1 {  
    color: white;  
    text-align: center;  
}  
  
p {  
    font-family: verdana;  
    font-size: 20px;  
}
```

BROWSER: HTML, CSS, JAVASCRIPT

- **JavaScript** (ou simplesmente JS) é uma linguagem de programação de alto nível, dinâmico, fracamente tipado, multi-paradigma e interpretado.
 - Ao lado de **HTML** e **CSS**, o **JavaScript** é uma das três principais tecnologias da produção de conteúdo da *World Wide Web*.
 - Ele é usado para tornar as páginas interativas. A maioria dos sites o emprega, e todos os navegadores da web modernos o suportam sem, a necessidade de plug-ins por meio de um mecanismo de JavaScript incorporado.

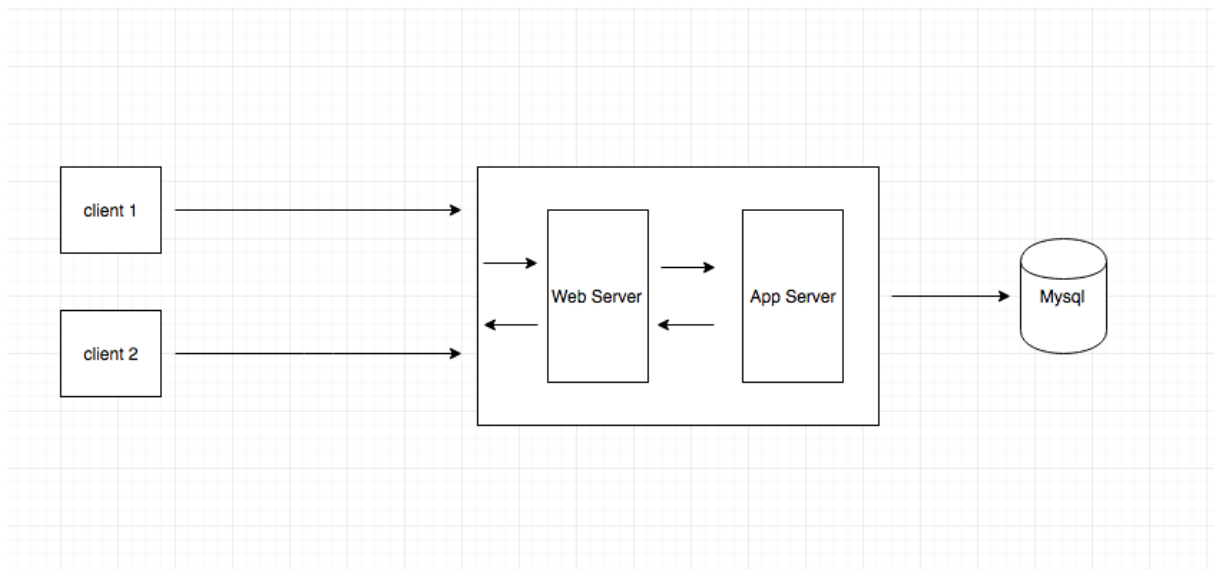
```
<!DOCTYPE html>
<html><head>
<script>
function myFunction() {
    document.getElementById("demo").innerHTML =
    "Paragraph changed.";
}
</script>
</head>
<body>
<h1>A Web Page</h1>
<p id="demo">A Paragraph</p>
<button type="button" onclick="myFunction()">Try
it</button>
</body>
</html>
```

STATEFUL VERSUS STATELESS

- **Stateful** - mantém o controle do estado das interações, geralmente armazenando valores em um local designado
 - A favor: mais fácil pra desenvolver
 - Contra: menos fôlego para o servidor
- **Stateless** – quando não há registro das interações anteriores, e cada pedido deve ser tratado inteiramente baseado em informação que vem com o mesmo
 - A favor: mais fácil de escalar
 - Contra: mais difícil para desenvolver

WEB SERVER E/OU APP SERVER

- **Web Server** é projetado para servir conteúdo HTTP
- **App Server** também pode servir conteúdo HTTP, mas não está limitado a este, suportando outros protocolos e padrões



ARQUITETURAS E POSSIBILIDADES

- WebBroker² + Bootstrap e/ou JS¹ (+ DataSnap²)
- IntraWeb³ + Bootstrap e/ou JS¹ (+ DataSnap²)
- uniGui/Ext JS³ (+ DataSnap²)
- JS¹ + DataSnap² (REST)
- JS¹ + RAD Server² (REST)
- JS¹ + Delphi MVC Framework² (REST)

(1) JQuery, AngularJS, Ext JS, ...

(2) Stand-alone, Windows service, IIS ou Apache (Windows e Linux)

(3) Stand-alone, Windows service, IIS (Windows, Linux no roadmap)

DEMOS

RECURSOS ADICIONAIS

■ Documentação

- [http://docwiki.embarcadero.com/RADStudio/Tokyo/en/Creating Internet server applications Index](http://docwiki.embarcadero.com/RADStudio/Tokyo/en/Creating_Internet_server_applications_Index)
- [http://docwiki.embarcadero.com/RADStudio/Tokyo/en/Creating WebBroker Applications](http://docwiki.embarcadero.com/RADStudio/Tokyo/en/Creating_WebBroker_Applications)
- [http://docwiki.embarcadero.com/RADStudio/Tokyo/en/Using Web Broker Index](http://docwiki.embarcadero.com/RADStudio/Tokyo/en/Using_Web_Broker_Index)
- <https://www.w3schools.com/html/default.asp>
- <https://www.w3schools.com/css/default.asp>
- <https://www.w3schools.com/js/default.asp>

■ Blogs

- <https://spring.io/guides/gs/consuming-rest-jquery/>
- <https://www.nginx.com/resources/glossary/application-server-vs-web-server/>
- <https://www.oyewiki.com/tech/web-server-vs-application-server-difference-explained>
- <http://etutorials.org/Programming/mastering+delphi+7/Part+IV+Delphi+the+Internet+and+a+.NET+Preview/Chapter+20+Web+Programming+with+WebBroker+and+WebSnap/Delphi+s+WebBroker+Technology/>
- <https://community.embarcadero.com/blogs/entry/join-marco-cantu-for-web-razor-scripting-with-webbroker>
- <https://community.embarcadero.com/blogs/entry/how-to-make-restful-webbroker-using-c-builder>

OBRIGADO!

Perguntas?

Você pode me encontrar em:
@FernandoRizzato
fernando.rizzato@embarcadero.com

Siga-nos em
fb.com/DelphiBrasil
fb.com/EmbarcaderoBR