

Adobe Flex e ZendAMF

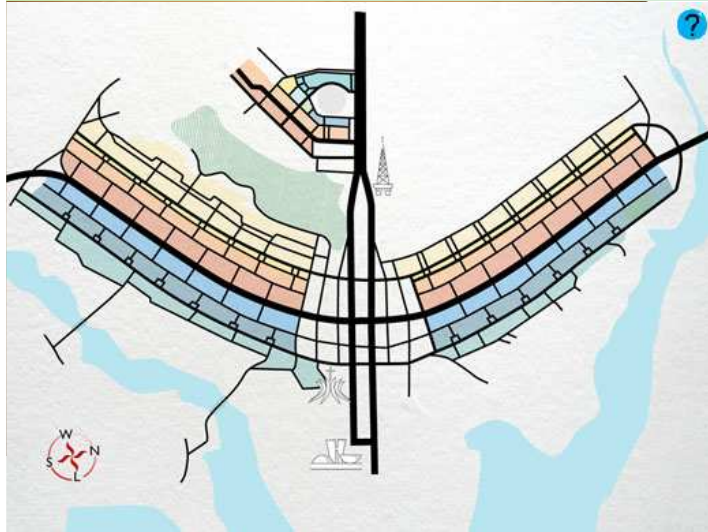
Leonardo França
Gerente AUGDF

Quem sou eu?

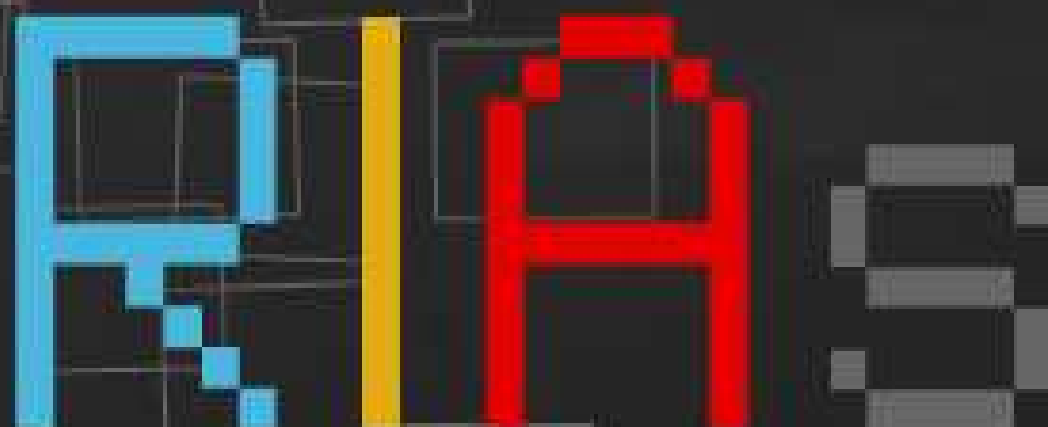


- 8 anos trabalhando com web
- Gerente do AUGDF – Adobe User Group de Brasília - www.augdf.com.br
- Adobe Certified Expert
- Desenvolvedor e Consultor RIA
- Atualmente no Sicoob
- www.leonardofranca.com.br
- twitter.com/leofranca5
- leo@osfederais.com



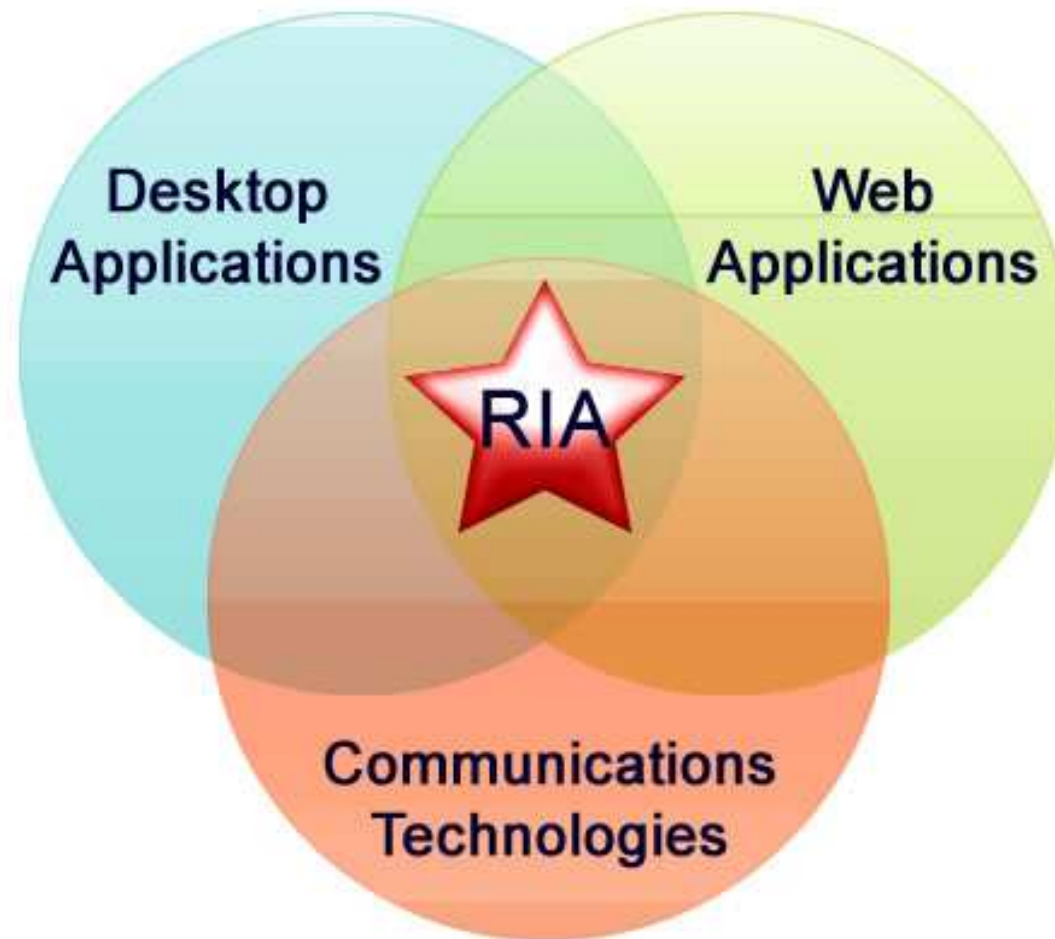


*Adobe
powers*

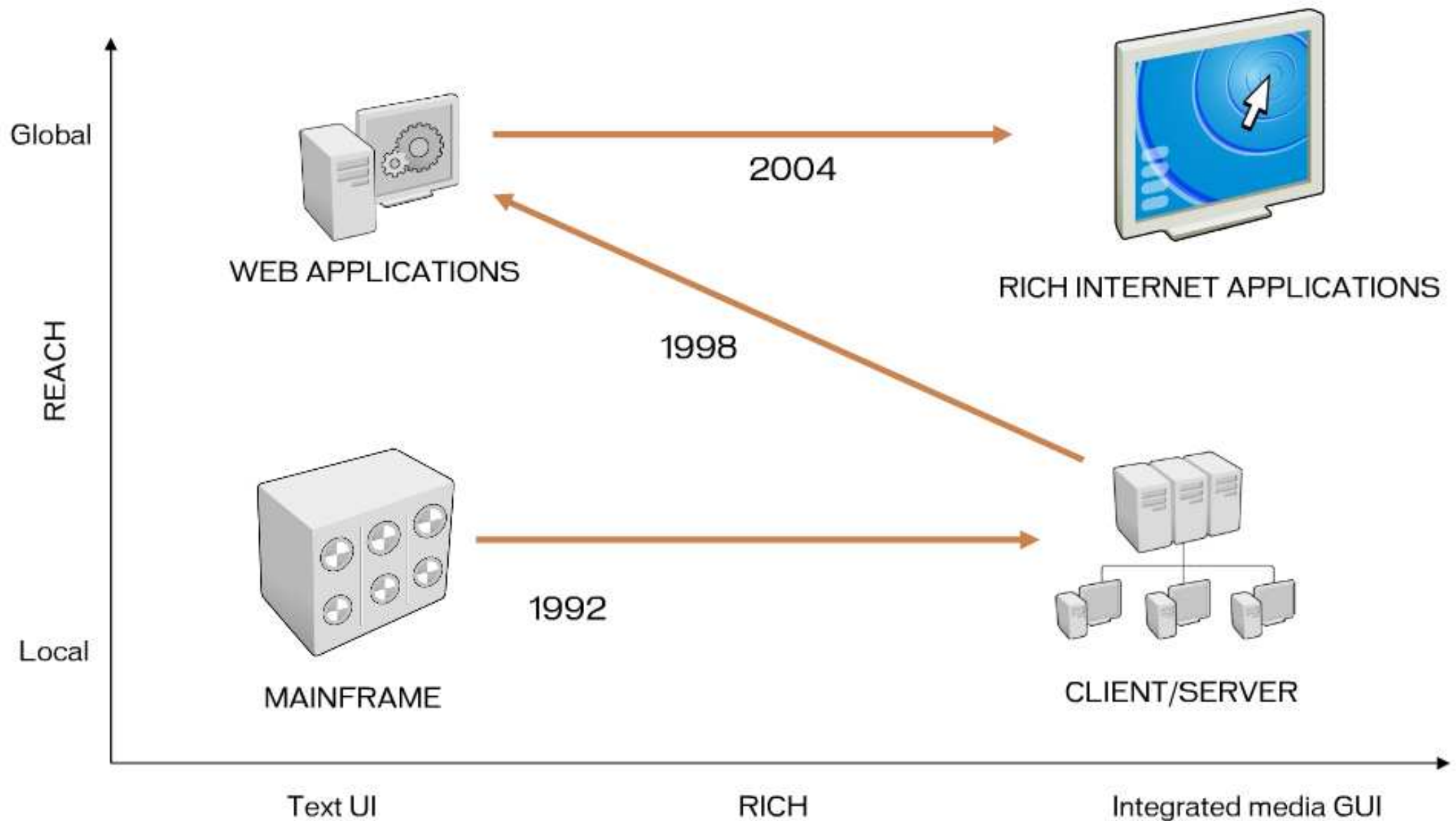


(rich Internet applications)

Rich Internet Application

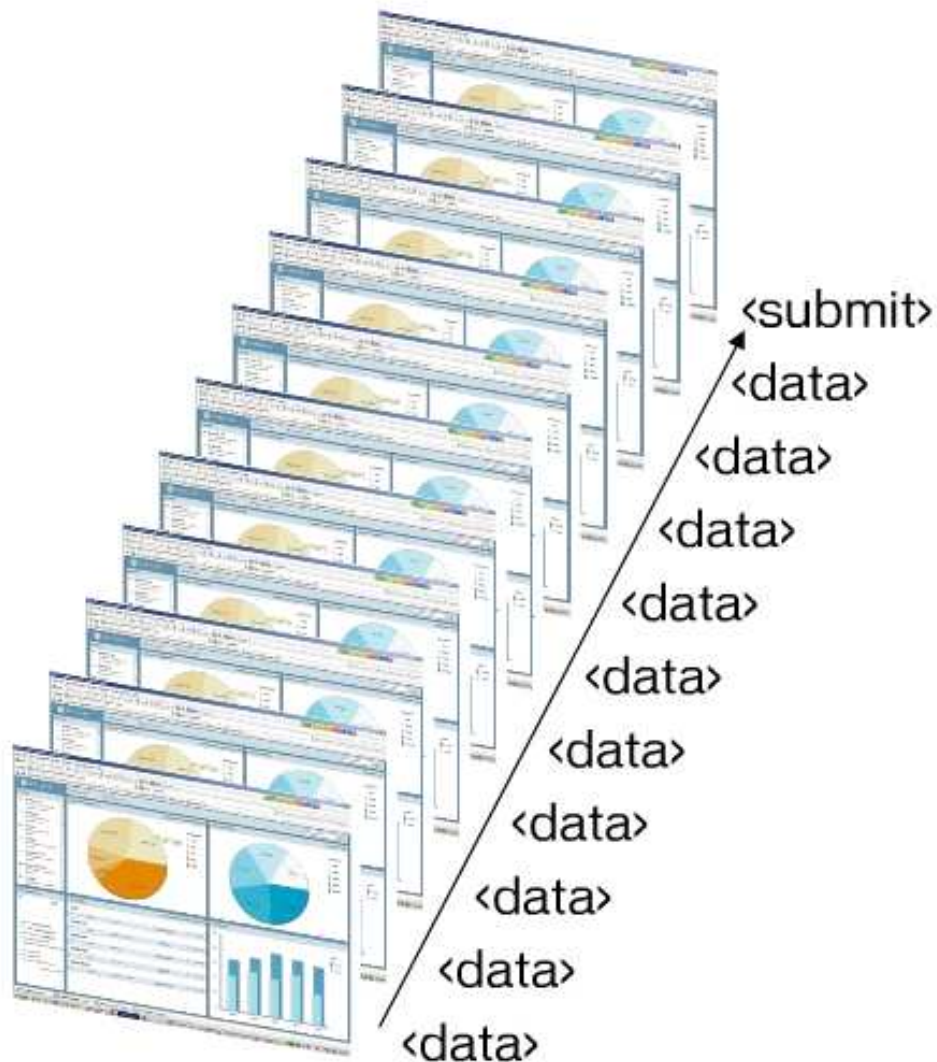


Evolução



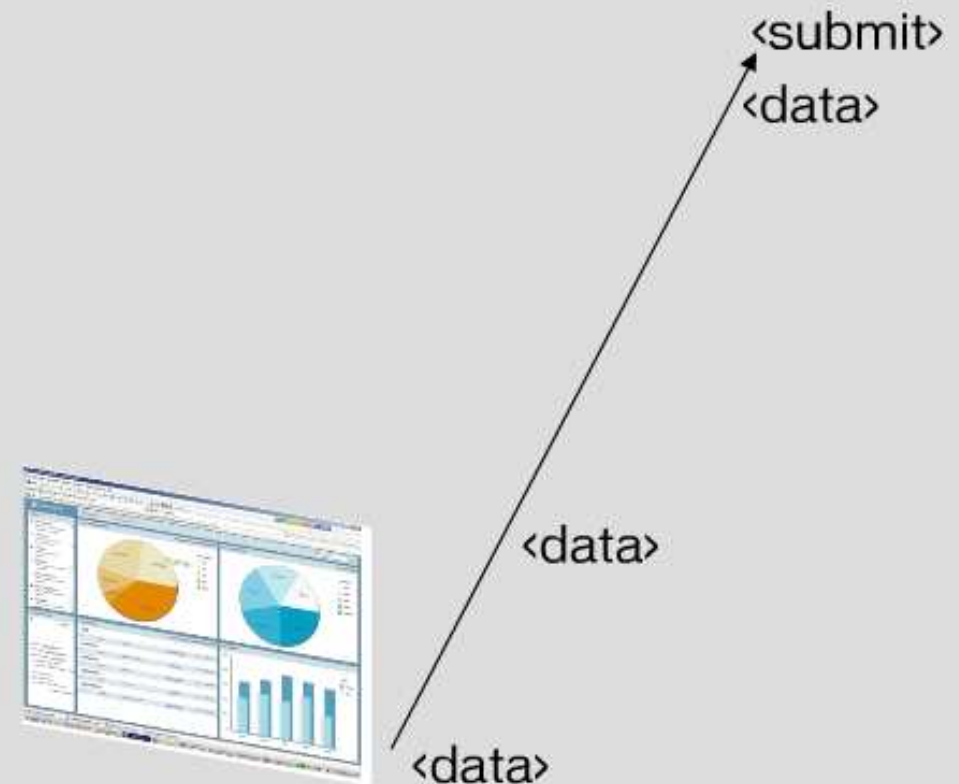
Por que RIA?

Aplicação Web Tradicional



Rich Internet Application

- pouca espera
- pouca banda usada
- pouco carregamento do servidor
- menos erros



Para os usuários



Para os desenvolvedores



Para os gerentes de TI







ADOBE® FLASH® PLATFORM

Adobe Flash Platform



Applications, Content and Video



Tools to Design and Develop



Flash CS4
Professional



Flash
Catalyst



Flash
Builder

Framework



Flex

Clients



AIR



Flash
Player

Servers/
Services

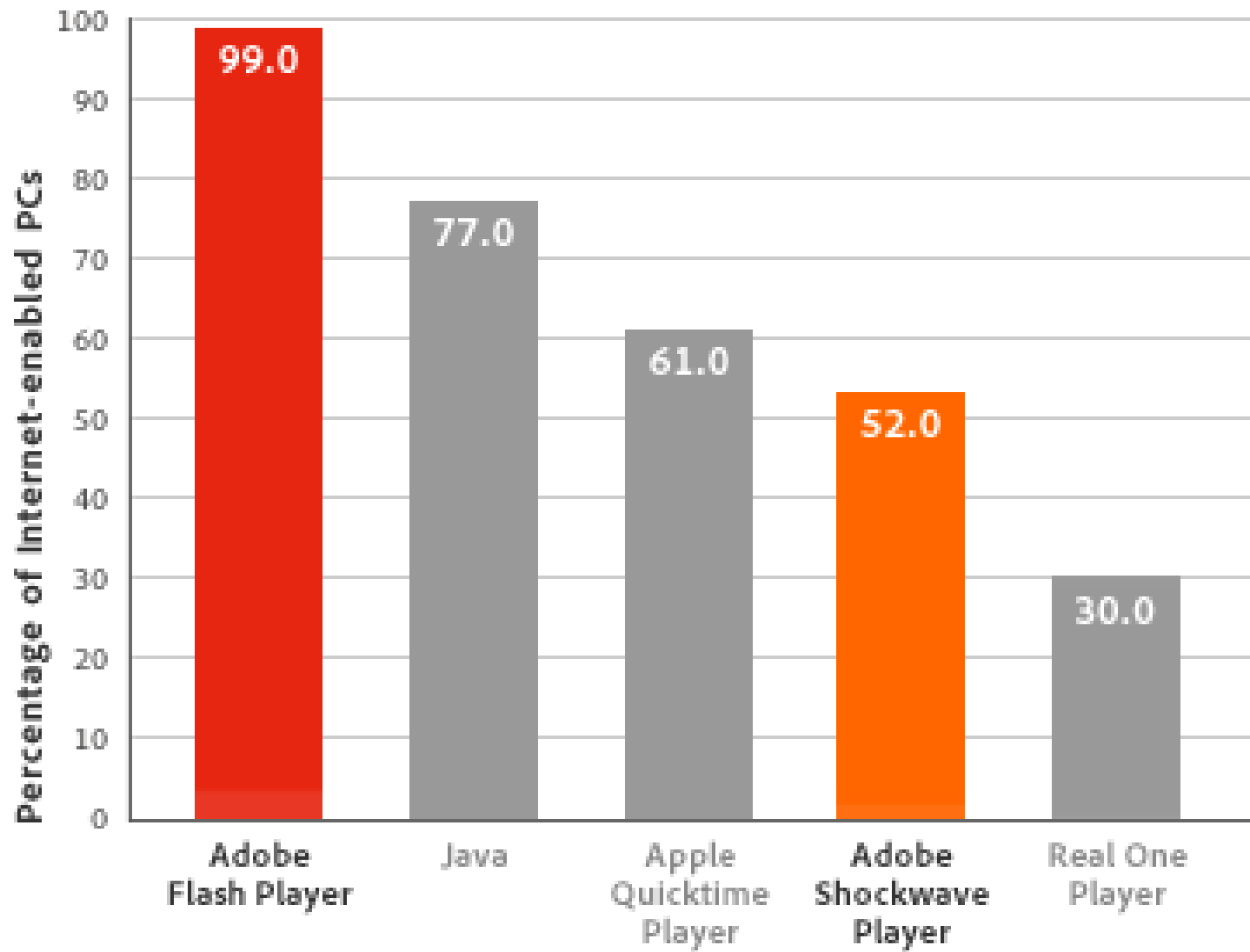


Flash Media
Server Family



BlazeDS
Data Services

No mundo



Open Screen Project



Um Conteúdo, vários dispositivos

Flash em todo lugar



symbian



TOSHIBA



SONY

CASIO



intel

ACCESS



NOKIA



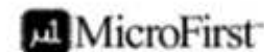
PocketPC



SONY



icebox



Flash Mobile

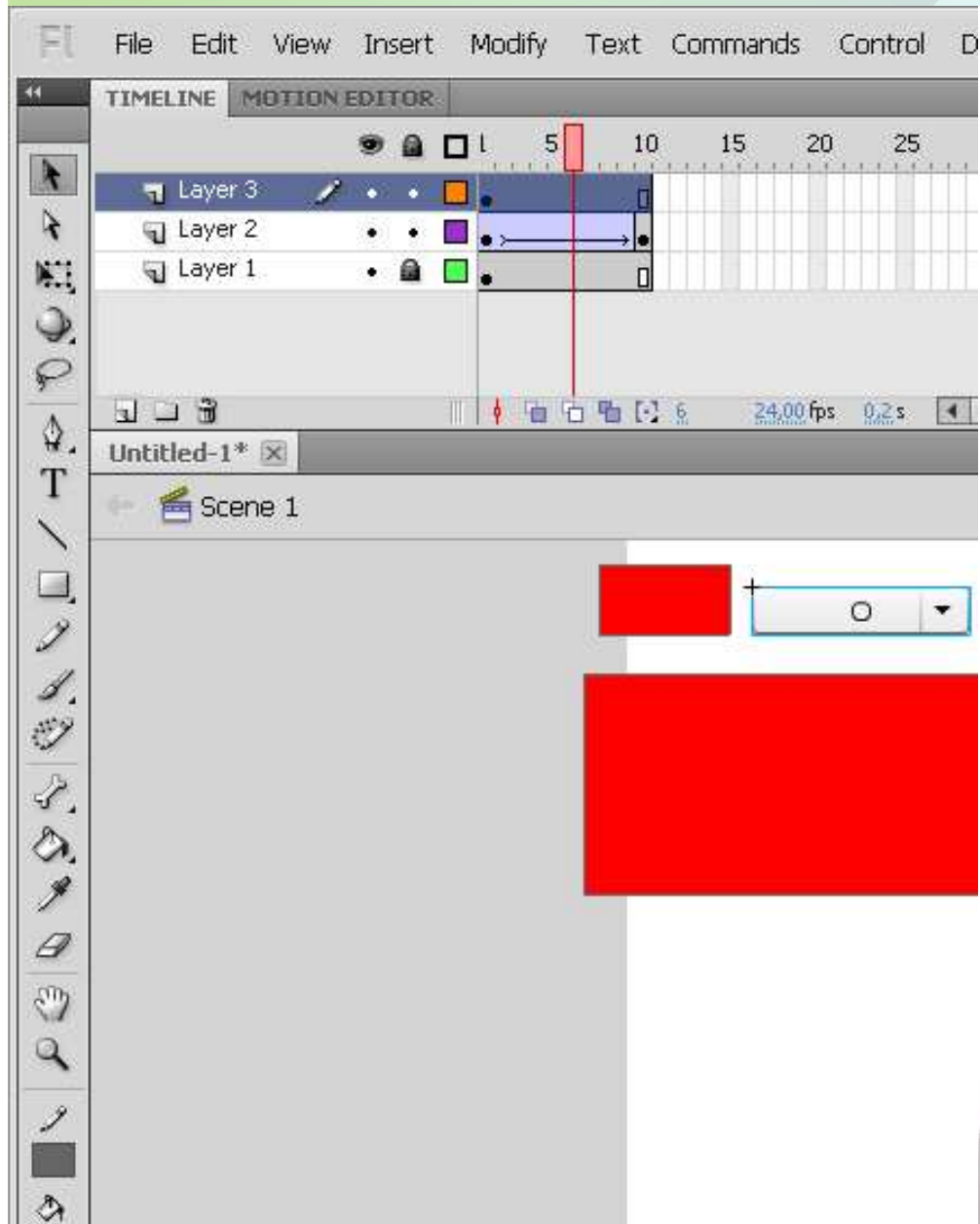


Flash Platform + Android





```
<?xml version="1.0" encoding="utf-  
<s:WindowedApplication xmlns:fx="h  
width="800" height="600">  
  <s:Group width="100%">  
    <s:layout>  
      <s:VerticalLayout padd  
</s:layout>  
    <s:Group width="100%">  
      <s:layout>  
        <s:HorizontalLayou  
</s:layout>  
        <mx:Label text="URL:"/  
        <mx:TextInput width="1  
        <mx:Button label="GO"  
      </s:Group>  
      <mx:HTML width="100%" heig  
    </s:Group>
```



Package Explorer

- (default package)
- admin
- assets
- br.com.lfsystems
 - business
 - controls
 - ButtonApps.mxml
 - ComboTipoUserController
 - DataGridSysController.as
 - DataGridUsersController.as
 - DefaultItemRenderer.mxml
 - MainRevendedorController
 - SetActiveUserController
 - effects
 - events
 - model
 - dto
 - vo
 - LoggerVO.as
 - SistemasVO.as
 - TiposVO.as
 - UsersHasPermissoesV
 - UserVO.as
 - ComboStatesBr.mxml
 - Icons.as
 - view
 - form
 - FormApp.mxml
 - FormLogin.mxml
 - FormPermissoes.mxml
 - FormSys.mxml
 - FormTipos.mxml
 - FormUser.mxml
 - ComboTipoUserView.mxml
 - DataGridSysView.mxml
 - DataGridUsersView.mxml
 - MainRevendedorView.mxml

Outline

- br.com.lfsystems.controls
 - imports
 - SetActiveUserController
 - config : ServicesConfig
 - winPop : Canvas
 - SetActiveUserController()
 - childrenCreated() : void
 - setActiveUser(Event) : void
 - setActiveUserResult(ResultEver

Source Code

```
134 }
135
136 private function showLoginAgain(evt:CloseEvent):void
137 {
138     getFormLogin();
139 }
140
141 private function setScreenByPermission(type:int):void
142 {
143     PopUpManager.removePopUp(winPop);
144     winPop = null;
145     CursorManager.removeBusyCursor();
146     //trace(ObjectUtil.toString(evt.data));
147     winPop2 = Canvas(PopUpManager.createPopUp(this, br.com.lfsystems.core.utils.SplashBase, true));
148     PopUpManager.centerPopUp(winPop2);
149     winPop2["blSplash"].text = "Carregando Dados...";
150     winPop2["imgIcon"].source = Icons.ICON_LOAD_DATA_USER;
151     switch(type)
152     {
153     case 1:
154         trace("administrador");
155         screenAdministrador();
156     }
```

SetActiveUserController.as

```
13 import mx.containers.Canvas;
14 import mx.managers.PopUpManager;
15 import mx.rpc.events.ResultEvent;
16 import mx.utils.ObjectUtil;
17
18 public class SetActiveUserController extends SetActiveUserView
19 {
20     [Bindable]
21     public var config:ServicesConfig = new ServicesConfig();
22     private var winPop:Canvas;
23
24     public function SetActiveUserController()
25     {
26         super();
27     }
28
29     override protected function childrenCreated() : void
30     {
31     }
```



Flex Framework



Flex Framework

Flex Core Components

- UI Controls
 - Data Entry Controls
 - CheckBox**
 - ComboBox
 - DateChooser
 - DateField
 - Label
 - NumericStepper
 - RadioButton
 - RichTextEditor
 - Text
 - TextArea
 - TextInput
 - Buttons
 - List Controls
 - Tree and Grid Controls
 - Other Controls
- Containers
- Effects
- Formatters
- Validators
- Charting
- Coding Techniques
- Flex Data Access
- AIR Capabilities
- Cloud APIs
- Data Visualization
- Mapping
- Other Components

The CheckBox control consists of an optional label and a small box that can contain a check mark or not. You can place the optional text label to the left, right, top, or bottom of the CheckBox. When a user clicks a CheckBox control or its associated text, the CheckBox control changes its state from checked to unchecked or from unchecked to checked. CheckBox controls gather a set of true or false values that are not mutually exclusive.
Author: Adobe Systems, Inc.
Date Added: 2008-09-02

CheckBox

Sample 1

CheckBox Control Example

☒ milk Items in my cart
☐ eggs milk
☐ bread
☒ Send me coupons for items in my cart

sample1.mxml **Lang Ref**

```
<?xml version="1.0"?>
<!-- Simple example to demonstrate the CheckBox control -->
<mx:Application xmlns:mx="http://www.adobe.com/2006/mxml" backgroundColors="[0x000000,0x323232]" viewSourceURL="srcview/index.html">

    <mx:Script>
        <![CDATA[

            import mx.controls.Alert;

            // This event handler adds and deletes items from the shopping cart.
            private function modifyCart():void
            {
                cartItems.text = "";

                if(milkCB.selected == true){
                    cartItems.text += "milk" + "\n";
                }

                if(eggsCB.selected == true){
                    cartItems.text += "eggs" + "\n";
                }

                if(breadCB.selected == true){
                    cartItems.text += "bread" + "\n";
                }
            }

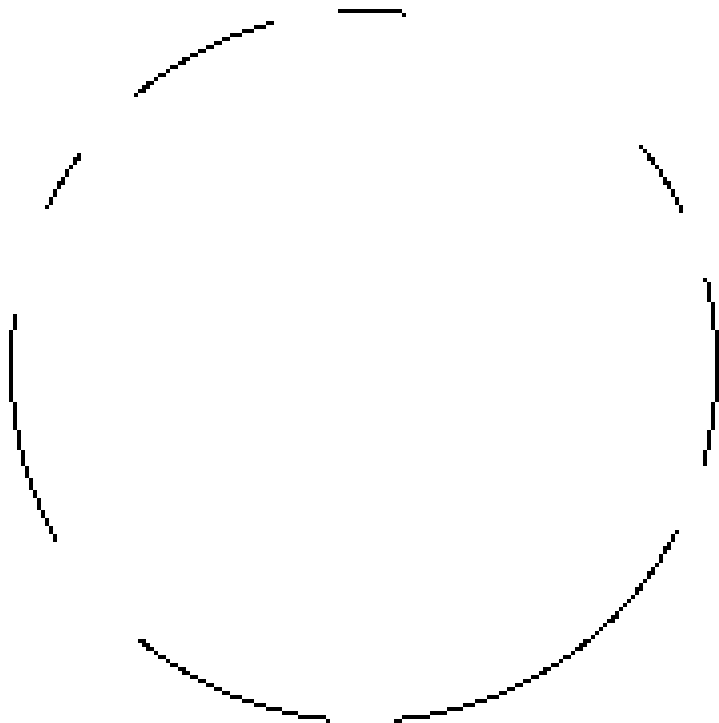
            // This event handler opens the Alert control.
            private function sendMessage():void
            {
                Alert.show("Hello World!");
            }
        ]]>
    </mx:Script>

    <mx:CheckBox id="milkCB" label="milk" x="100" y="100" />
    <mx:CheckBox id="eggsCB" label="eggs" x="100" y="150" />
    <mx:CheckBox id="breadCB" label="bread" x="100" y="200" />
    <mx:Text id="cartItems" x="100" y="250" />
    <mx:CheckBox id="couponsCB" label="Send me coupons for items in my cart" x="100" y="300" />

    <mx:Script>
        <![CDATA[
            milkCB.addEventListener(MouseEvent.CLICK, modifyCart);
            eggsCB.addEventListener(MouseEvent.CLICK, modifyCart);
            breadCB.addEventListener(MouseEvent.CLICK, modifyCart);
            couponsCB.addEventListener(MouseEvent.CLICK, sendMessage);
        ]]>
    </mx:Script>

</mx:Application>
```

About ©2009 Adobe Inc. All Rights Reserved **Version:** 1.2.3 **List Version:** 2009-09-04 **Samples:** 289



Ergonomia



Flex Components



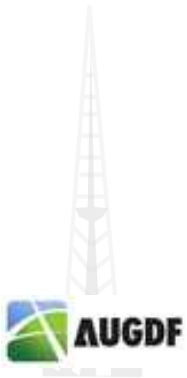
CSS e Skins



CSS e Skins



- Demo – Flex Style Explorer
- Demo – Windows Classic



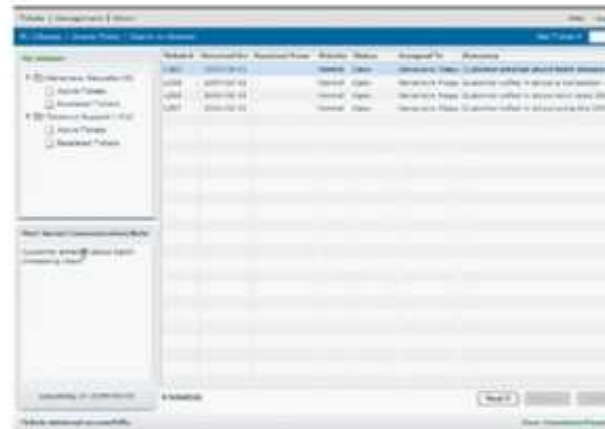
Onde é usado!



Product Configuration



Line of Business Applications



Data Visualization



Rich Content Applications



Complex, Multi-Step Processes, Forms

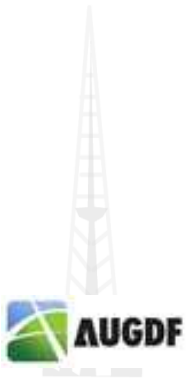


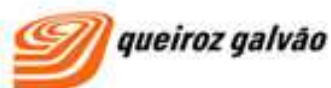
Product Selection

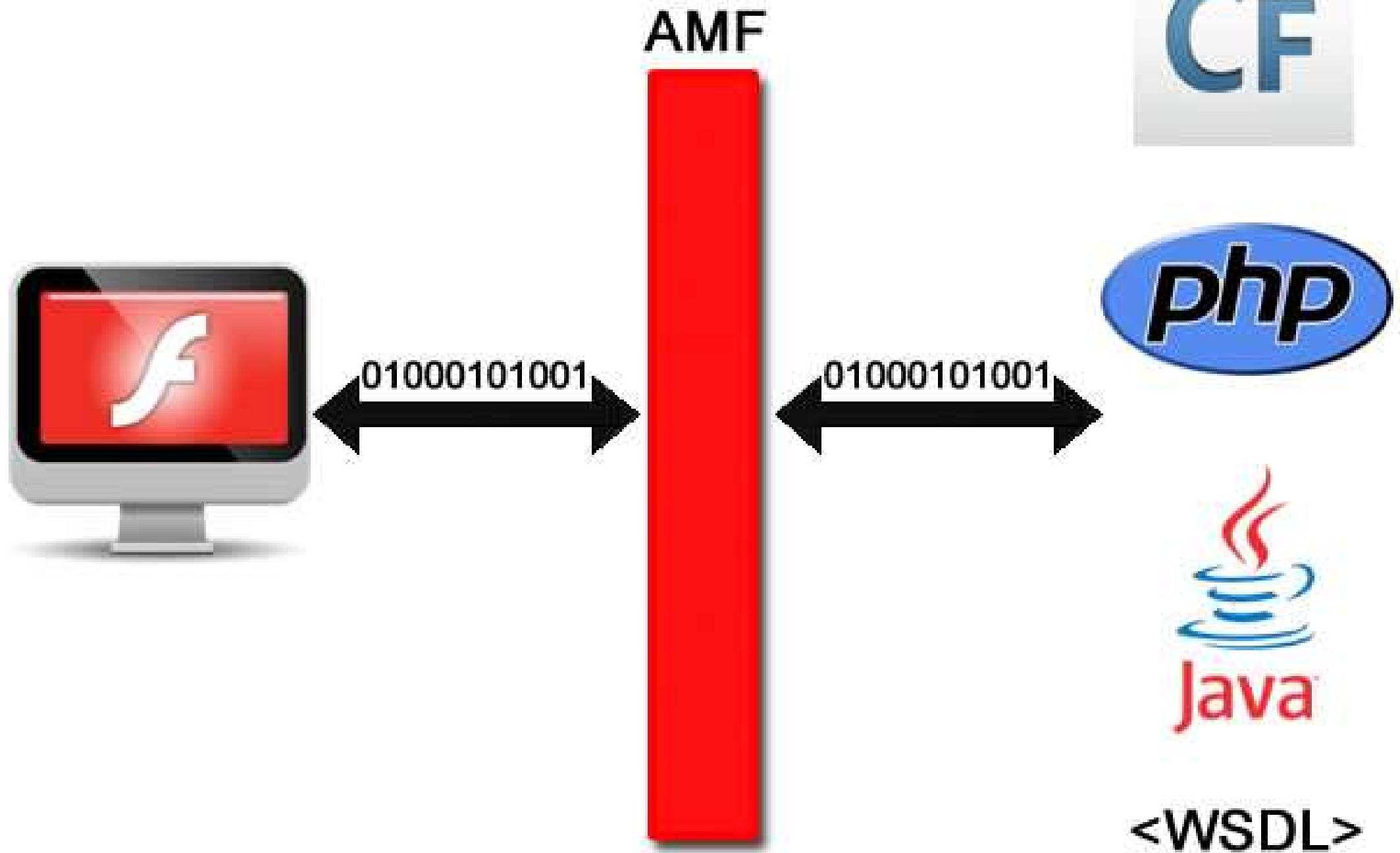


Onde é usado!

- Demo – Dashboard
- Demo – Flexstore
- Demo – Memory







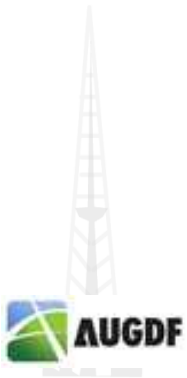
Zend AMF



Action Message Format



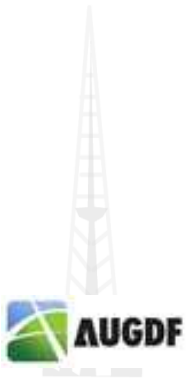
- Demo – Contatos
- Demo – Data Centric Development



Como começar?



- SDK Free
- Free para estudantes, desempregados e professores
- <https://freeriatools.adobe.com/>



Perguntas?

