

- [Home](#)
- Específicos
- Flex
- Getting Started
- Java
- Processo Datasul
- Progress
- Tools
- Trilhas
- User
- [Page Operations](#)
- [Browse Space](#)

SDK - Software Development Kit - v.1.0

Zoom Container

Added by [Rafael Miranda Moller](#), last edited by [Roberto Reinert](#) on Jan 22, 2010 ([view change](#))

Labels

[incubado](#), [flex](#), [metadados](#), [freeform](#), [revisão técnica](#), [zoomcontainer](#)

documento incubado

Visão Geral

O ZoomContainer é utilizado para possibilitar a abertura de telas de Zoom com simples ou múltipla seleção de elementos. Para utilização do componente ZoomContainer em um FreeForm são necessários alguns passos importantes:

1. [Definir o componente que fará a chamada para o ZoomContainer;](#)
2. [Definir os eventos iniciais para preenchimento de dados;](#)
3. [Definir os eventos utilizados pelo componente para chamada do ZoomContainer;](#)
4. [Atribuir as propriedades para abertura da tela de ZoomContainer;](#)

Definir o componente que fará a chamada para o ZoomContainer

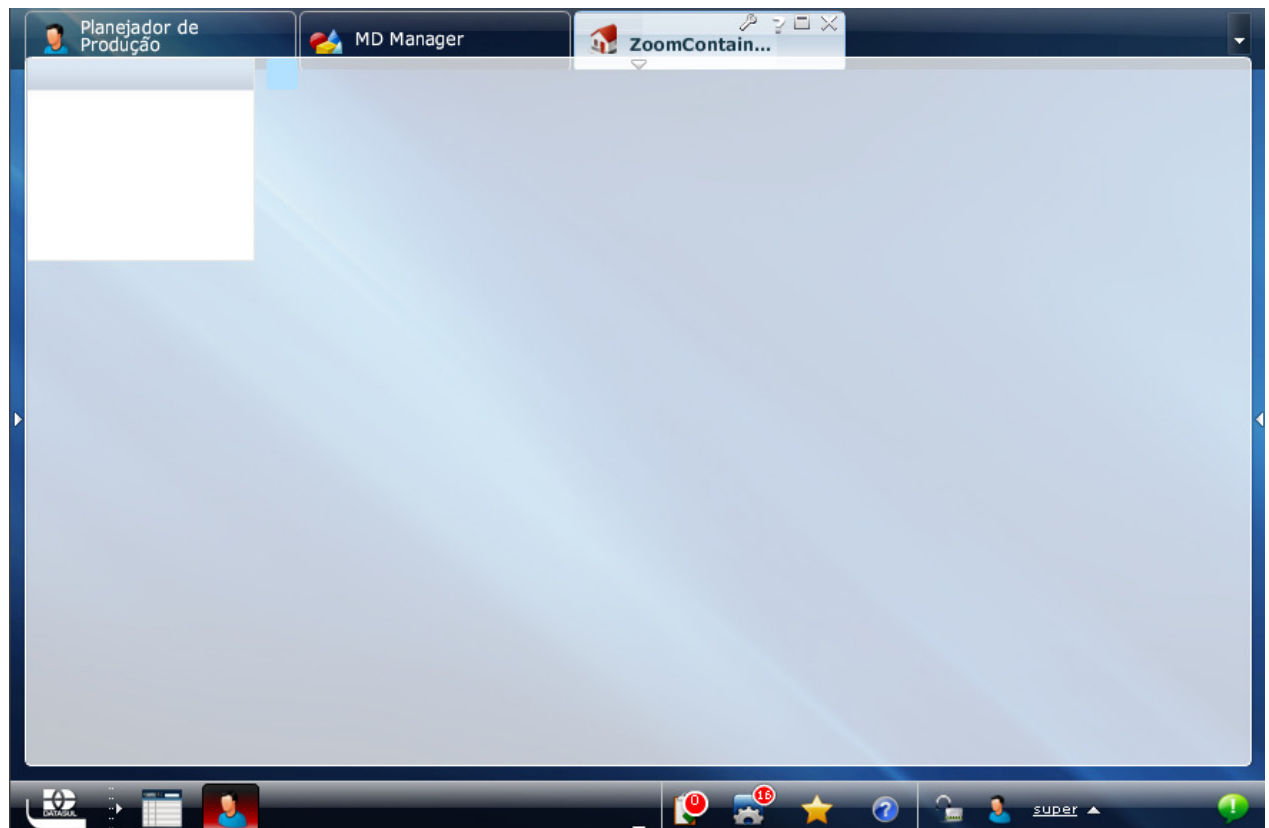
Para definir o componente que efetuará a chamada para o componente ZoomContainer, será utilizado o plugin para cadastro de FreeForm, conforme apresentado em documentação [Criando um Form](#), ou utilizando o [Importador de Protótipos em MXML](#).

Exemplo de estrutura de um componente que será importado:

```

<?xml version="1.0" encoding="utf-8"?>
<mx:Application xmlns:mx="http://www.adobe.com/2006/mxml"
    xmlns:totvs="http://www.totvs.com/2009/framework"
    horizontalAlign="center" verticalAlign="middle">
    <mx:HBox>
        <totvs:DataGrid id="dgZoom">
        </totvs:DataGrid>
        <mx:LinkButton id="lkZoom"
            height="22"
            width="22"/>
    </mx:HBox>
</mx:Application>
    
```

A tela após importação dos dados e deploy do FreeForm:



i Informação

Pode-se observar que o componente DataGrid não possui as colunas atribuídas para o mesmo e o componente LinkButton não possui um ícone atribuído, estes valores serão adicionados no passo seguinte ao criar um EventHandler para manipular

os eventos.

Definir os eventos iniciais para preenchimento de dados

Após definir a estrutura da tela que efetuará a chamada do ZoomContainer, deve-se mapear os eventos iniciais para preenchimento de dados, conforme necessário.

Para definição dos eventos de um componente pode-se consultar a documentação de [Associação de eventos de interface](#).

Para este exemplo estaremos definindo os seguintes eventos:

Componente	Nome	Evento
DataGrid	dgZoom	creationComplete
LinkButton	lkZoom	creationComplete

Após a definição dos eventos é necessária a [Criação do manipulador de eventos de interface](#) (EventHandler), o mesmo ficará da seguinte maneira:

```
<?xml version="1.0" encoding="utf-8"?>
<FormEventHandler xmlns="com.totvs.framework.md.events.*"
  xmlns:mx="http://www.adobe.com/2006/mxml">
  <mx:Script>
    <![CDATA[
      import mx.controls.LinkButton;
      import mx.controls.advancedDataGridClasses.AdvancedDataGridColumn;
      import com.datasul.framework.ui.datagrid.view.DataGrid;

      private var dgZoom:DataGrid;

      private var lkZoom:LinkButton;

      [Embed('/assets/icons/zoom_l6_hot.png')]
      private var zoomIcon:Class;

      public function onDgZoomCreationComplete(event:Event):void {
        var columns:Array;
        var columnCode:AdvancedDataGridColumn;
        var columnName:AdvancedDataGridColumn;

        columnCode = new AdvancedDataGridColumn();
        columnCode.headerText = "Código";
        columnCode.dataField = "primaryKey.codigoPais";

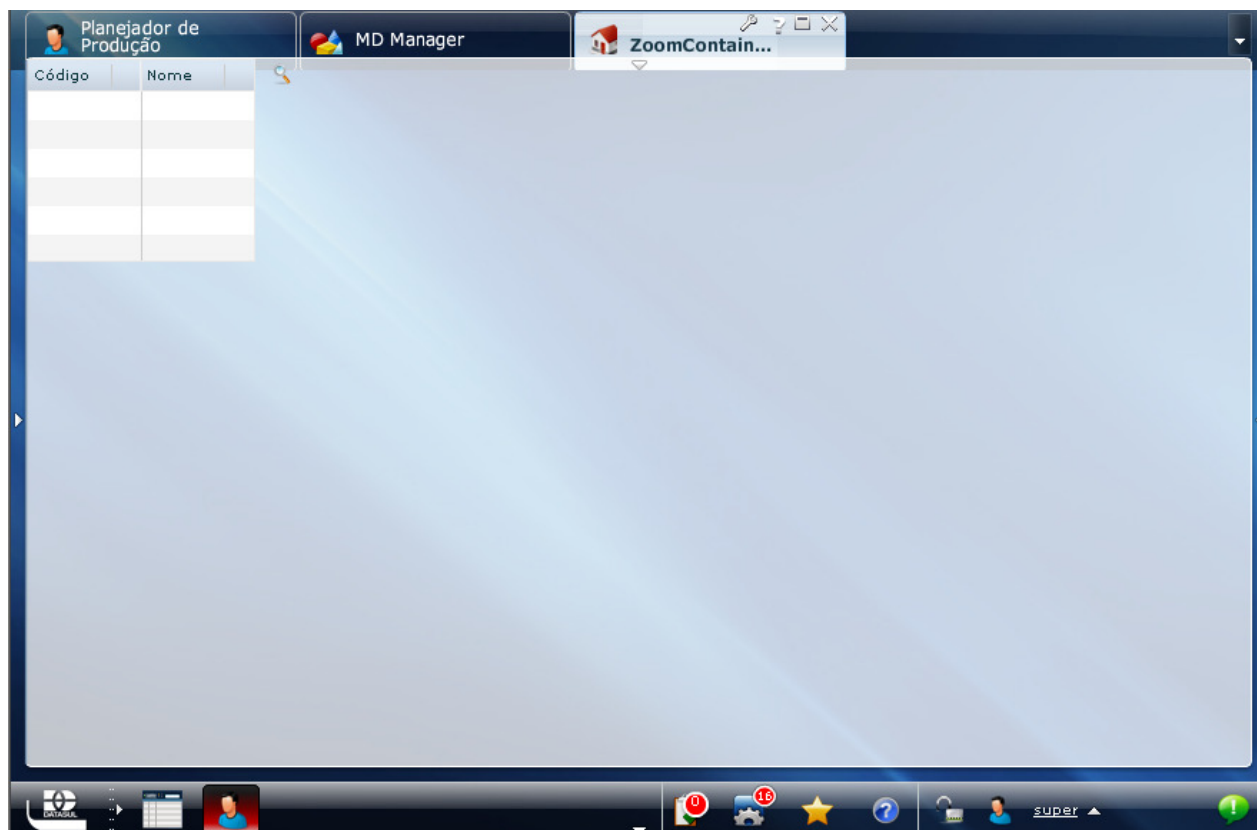
        columnName = new AdvancedDataGridColumn();
        columnName.headerText = "Nome";
        columnName.dataField = "nomePais";

        columns = new Array();
        columns.push(columnCode);
        columns.push(columnName);

        this.dgZoom = event.target as DataGrid;
        this.dgZoom.columns = columns;
      }

      public function onLkZoomCreationComplete(event:Event):void {
        this.lkZoom = event.target as LinkButton;
        this.lkZoom.setStyle("icon", zoomIcon);
      }
    ]]>
  </mx:Script>
</FormEventHandler>
```

Após a criação do EventHandler a tela será apresentada da seguinte maneira:



Definir os eventos utilizados pelo componente para chamada do ZoomContainer

Após definidos os eventos iniciais para preenchimento de dados, deve-se mapear os eventos utilizados para chamada do componente de ZoomContainer.

Para este exemplo estaremos definindo os seguintes eventos:

Componente	Nome	Evento
LinkButton	lkZoom	click

Após a definição dos eventos é necessária a [Criação do manipulador de eventos de interface](#) (EventHandler), o mesmo ficará da seguinte maneira:

```
<?xml version="1.0" encoding="utf-8"?>
<FormEventHandler xmlns="com.totvs.framework.md.events.*"
  xmlns:mx="http://www.adobe.com/2006/mxml">
  <mx:Script>
    <![CDATA[

      import com.datasul.framework.ui.datagrid.view.DataGrid;
      import com.totvs.framework.zoom.control.ZoomEvent;
      import com.totvs.framework.zoom.view.ZoomContainer;

      import mx.containers.TitleWindow;
      import mx.controls.LinkButton;
      import mx.controls.advancedDataGridClasses.AdvancedDataGridColumn;
      import mx.core.Application;
      import mx.events.CloseEvent;
      import mx.managers.PopUpManager;

      private var dgZoom:DataGrid;

      private var lkZoom:LinkButton;

      private var _titleWindow:TitleWindow;

      [Embed('/assets/icons/zoom_l6_hot.png')]
      private var zoomIcon:Class;

      public function onDgZoomCreationComplete(event:Event):void {
        var columns:Array;
        var columnCode:AdvancedDataGridColumn;
        var columnName:AdvancedDataGridColumn;

        columnCode = new AdvancedDataGridColumn();
        columnCode.headerText = "Código";
        columnCode.dataField = "primaryKey.codigoPais";

        columnName = new AdvancedDataGridColumn();
        columnName.headerText = "Nome";
        columnName.dataField = "nomePais";

        columns = new Array();
        columns.push(columnCode);
        columns.push(columnName);

        this.dgZoom = event.target as DataGrid;
        this.dgZoom.columns = columns;
      }

      public function onLkZoomCreationComplete(event:Event):void {
        this.lkZoom = event.target as LinkButton;
      }
    ]]>
  </mx:Script>
</FormEventHandler>
```

```

        this.lkZoom.setStyle("icon", zoomIcon);
    }

    public function onLkZoomClick(event:Event):void {
        if(!this._titleWindow) {
            this._titleWindow = new TitleWindow();
            this._titleWindow.title = "ZoomContainer Pais";
            this._titleWindow.width = 800;
            this._titleWindow.height = 540;
            this._titleWindow.showCloseButton = true;

            this._titleWindow.addEventListener(CloseEvent.CLOSE, this.onClosePopUp);
        }

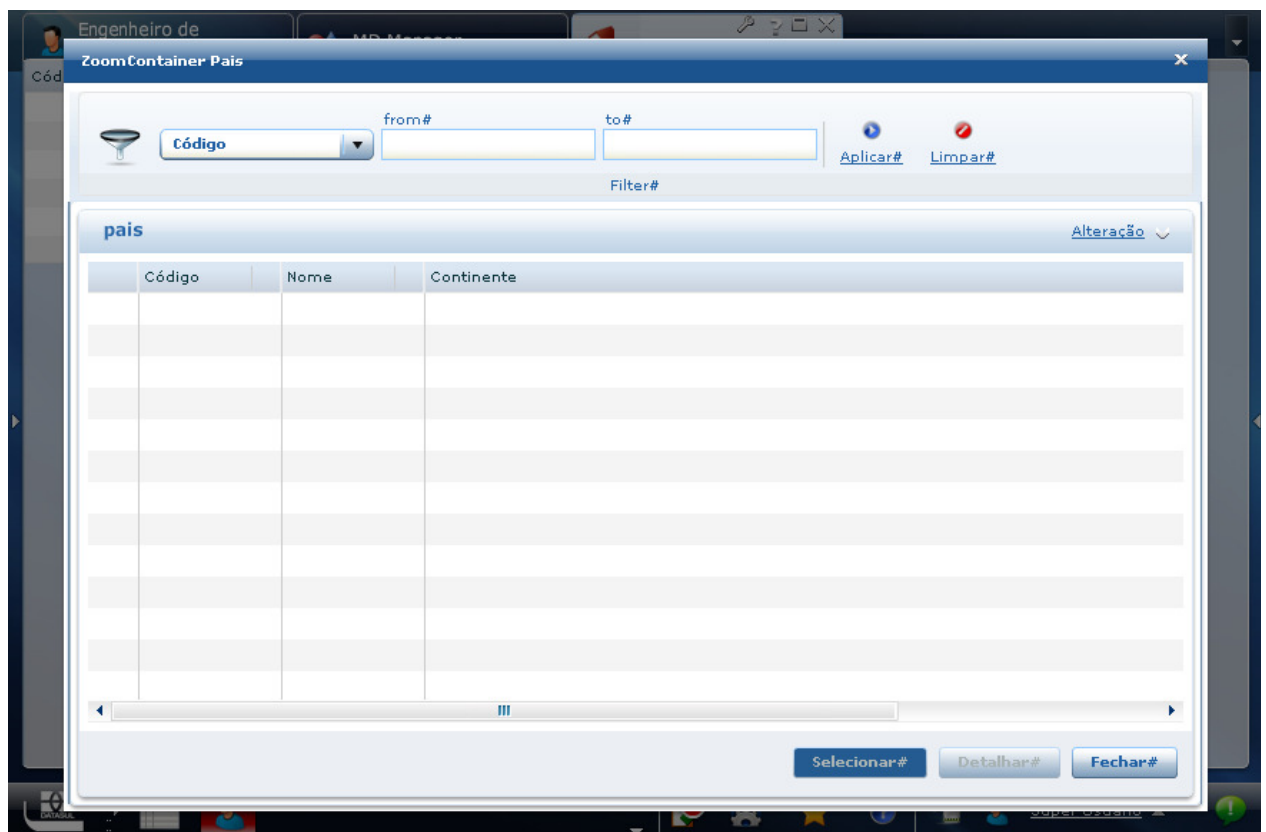
        PopUpManager.addPopUp(this._titleWindow, Application.application as DisplayObject, true);
        PopUpManager.centerPopUp(this._titleWindow);
    }

    private function onSelectZoom(event:ZoomEvent):void {
        if(event.selectedItems && event.selectedItems.length > 0) {
            this.dgZoom.dataProvider = event.selectedItems;
        }
        this.onClosePopUp(null);
    }

    private function onClosePopUp(event:Event):void {
        PopUpManager.removePopUp(this._titleWindow);
    }
}
]]>
</mx:Script>
</FormEventHandler>

```

Após a criação do EventHandler a tela será apresentada da seguinte maneira:



Atribuir as propriedades para abertura da tela de ZoomContainer

Após definir os eventos utilizados para chamada do componente ZoomContainer, deve-se verificar as propriedades que serão atribuídas para o componente [ZoomContainer](#) para sua abertura.

Para este exemplo estaremos atribuindo as seguintes propriedades para o componente ZoomContainer:

Propriedade	Valor
detailMode	false
fieldCode	primaryKey.codigoPais
fieldLabel	nomePais
xmlName	crm.dbgeo.CRUDPais
voClassName	com.totvs.crm.dbgeo.be.Pais
typeColumn	Long
queryColumnName	codigoPais
appName	crm
database	dbgeo
appVersion	1.0.0
queryColumnName	codigoPais
percentWidth	100

percentHeight	100
selectionMode	ZoomContainer.MULTIPLE_ROWS

O código o EventHandler ficará da seguinte maneira:

```

<?xml version="1.0" encoding="utf-8"?>
<FormEventHandler xmlns="com.totvs.framework.md.events.*"
  xmlns:mx="http://www.adobe.com/2006/mxml">
  <mx:Script>
    <![CDATA[

      import com.datasul.framework.ui.datagrid.view.DataGrid;
      import com.totvs.framework.zoom.control.ZoomEvent;
      import com.totvs.framework.zoom.view.ZoomContainer;

      import mx.containers.TitleWindow;
      import mx.controls.LinkButton;
      import mx.controls.advancedDataGridClasses.AdvancedDataGridColumn;
      import mx.core.Application;
      import mx.events.CloseEvent;
      import mx.managers.PopUpManager;

      private var dgZoom:DataGrid;

      private var lkZoom:LinkButton;

      private var _zoomContainer:ZoomContainer;
      private var _titleWindow:TitleWindow;

      [Embed('/assets/icons/zoom_l6_hot.png')]
      private var zoomIcon:Class;

      public function onDgZoomCreationComplete(event:Event):void {
        var columns:Array;
        var columnCode:AdvancedDataGridColumn;
        var columnName:AdvancedDataGridColumn;

        columnCode = new AdvancedDataGridColumn();
        columnCode.headerText = "Código";
        columnCode.dataField = "primaryKey.codigoPais";

        columnName = new AdvancedDataGridColumn();
        columnName.headerText = "Nome";
        columnName.dataField = "nomePais";

        columns = new Array();
        columns.push(columnCode);
        columns.push(columnName);

        this.dgZoom = event.target as DataGrid;
        this.dgZoom.columns = columns;
      }

      public function onLkZoomCreationComplete(event:Event):void {
        this.lkZoom = event.target as LinkButton;
        this.lkZoom.setStyle("icon", zoomIcon);
      }

      public function onLkZoomClick(event:Event):void {
        if(!this._titleWindow) {
          this._titleWindow = new TitleWindow();
          this._titleWindow.title = "ZoomContainer Pais";
          this._titleWindow.width = 800;
          this._titleWindow.height = 540;
          this._titleWindow.showCloseButton = true;

          this._titleWindow.addEventListener(CloseEvent.CLOSE, this.onClosePopUp);
        }
        if(!this._zoomContainer) {
          this._zoomContainer = new ZoomContainer();
          this._zoomContainer.detailMode = false;
          this._zoomContainer.fieldCode = "primaryKey.codigoPais";
          this._zoomContainer.fieldLabel = "nomePais";
          this._zoomContainer.xmlName = "crm.dbgeo.CRUDPais";
          this._zoomContainer.voClassName = "com.totvs.crm.dbgeo.be.Pais";
          this._zoomContainer.typeColumn = "Long";
          this._zoomContainer.queryColumnName = "codigoPais";
          this._zoomContainer.database = "dbgeo";
          this._zoomContainer.appName = "crm";
          this._zoomContainer.appVersion = "1.0.0";
          this._zoomContainer.percentWidth = 100;
          this._zoomContainer.percentHeight = 100;
          this._zoomContainer.selectionMode = ZoomContainer.MULTIPLE_ROWS;

          this._zoomContainer.addEventListener(ZoomEvent.ZOOM_SELECT, this.onSelectZoom);
          this._zoomContainer.addEventListener(ZoomEvent.ZOOM_BACK, this.onClosePopUp);

          this._titleWindow.addChild(this._zoomContainer);
        }
        PopUpManager.addPopUp(this._titleWindow, Application.application as DisplayObject, true);
        PopUpManager.centerPopUp(this._titleWindow);
      }

      private function onSelectZoom(event:ZoomEvent):void {
        if(event.selectedItems && event.selectedItems.length > 0) {
          this.dgZoom.dataProvider = event.selectedItems;
        }
        this.onClosePopUp(null);
      }

      private function onClosePopUp(event:Event):void {
        PopUpManager.removePopUp(this._titleWindow);
      }

    ]]>
  </mx:Script>
</FormEventHandler>

```

Ver Também

[ZoomContainer](#)

Informações sobre este Documento

Versão	Estado	Compatibilidade	Criador	Criação	Último Revisor	Última Revisão
11	incubado	3.0.0-SNAPSHOT	Rafael Miranda Moller	Out 14, 2009 12:27	Roberto Reinert	Jan 22, 2010 08:38



O que é RSS?

Your Rating:

Results:

0 rates

[Add Comment](#)