Pug Challenge Americas 2011

GUI for .Net Frameworks Inheritance and ABL Centric Controls

Presented by: Mike McMillan

Intui-Tech.com

Topics, eh!

- Inheritance
  - Things you should know
  - How to implement
- Custom Controls
  - Why Microsoft Native Controls?
  - ABL Centric Ideas

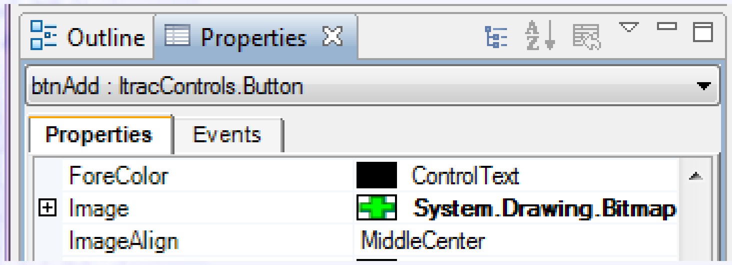
Inheritance

- Is it supported?
  - Not really – Design Time Only
    - Visual Designer = Visual Studio
    - Control Properties are replicated in Form or Control
    - Visual Studio has annotations - Designer does not
    - Annotations on roadmap for Open Edge Development

Inheritance – Things You Should Know

- Bloated Deployment
  - Images embedded redundantly in *.resx file

Button Example – btnAdd.cls



btnAdd : ItracControls.Button

ForeColor	ControlText
⊞ Image	**System.Drawing.Bitmap**
ImageAlign	MiddleCenter

```
METHOD PRIVATE VOID InitializeComponent(  ):
THIS-OBJECT:Image = CAST(resources:GetObject("$this.Image"), System.Drawing.Image).
```

Inheritance – Things You Should Know

- Bloated Deployment
  - Images embedded redundantly in *.resx file

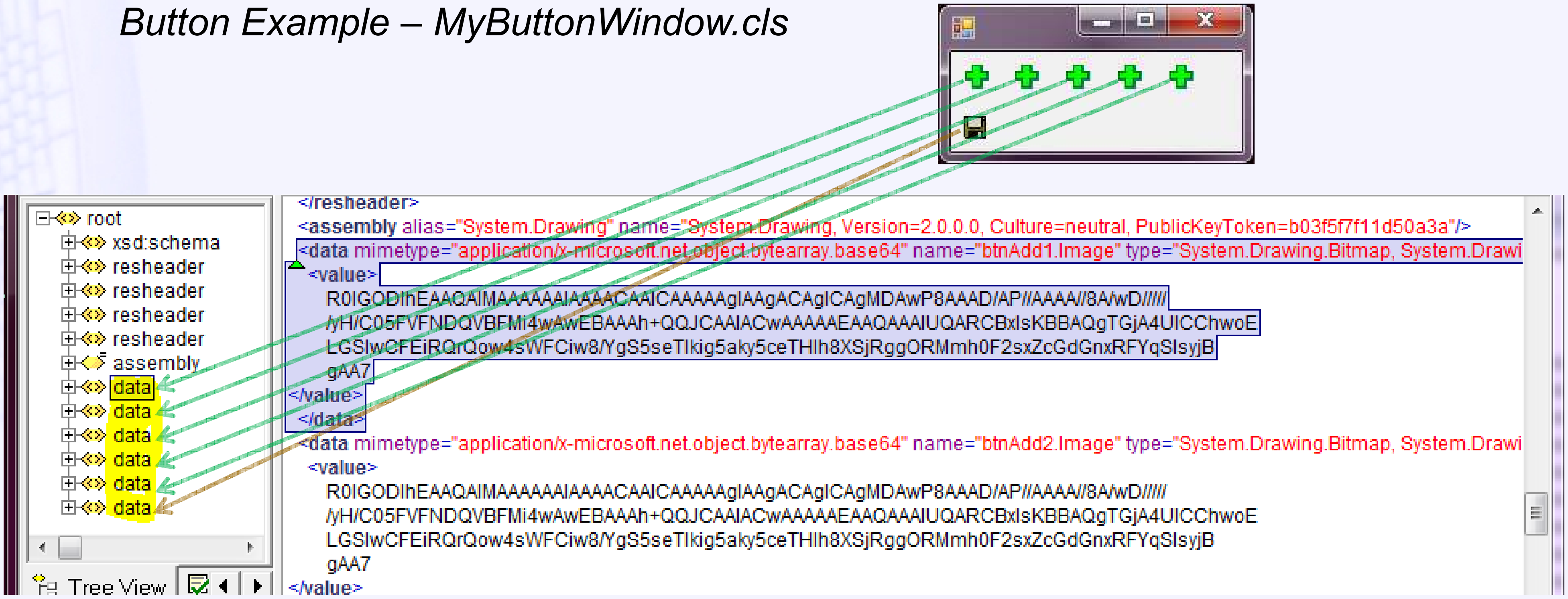
Button Example – MyButtonWindow.cls

```
    METHOD PRIVATE VOID InitializeComponent(  ):
THIS-OBJECT:btnAdd1:BackColor = System.Drawing.Color:Transparent.
THIS-OBJECT:btnAdd1:FlatAppearance:BorderColor = System.Drawing.Color:LightGray.
THIS-OBJECT:btnAdd1:FlatAppearance:BorderSize = 0.
THIS-OBJECT:btnAdd1:FlatStyle = System.Windows.Forms.FlatStyle:Flat.
THIS-OBJECT:btnAdd1:Image = CAST(resources:GetObject("btnAdd1.Image"),
                                System.Drawing.Image).
THIS-OBJECT:btnAdd1:Location = NEW System.Drawing.Point(2, 2).
THIS-OBJECT:btnAdd1:Name = "btnAdd1".
THIS-OBJECT:btnAdd1:Size = NEW System.Drawing.Size(25, 25).
THIS-OBJECT:btnAdd1:TabIndex = 0.
THIS-OBJECT:btnAdd1:UseCompatibleTextRendering = TRUE.
THIS-OBJECT:btnAdd1:UseVisualStyleBackColor = FALSE.
```

Inheritance – Things You Should Know

- Bloated Deployment
  - Images embedded redundantly in *.resx file

Button Example – MyButtonWindow.cls



```
</resheader>
<assembly alias="System.Drawing" name="System.Drawing, Version=2.0.0.0, Culture=neutral, PublicKeyToken=b03f5f7f11d50a3a"/>
<data mimetype="application/x-microsoft.net.object.bytearray.base64" name="btnAdd1.Image" type="System.Drawing.Bitmap, System.Drawi
  <value>
    R0lGODlhEAAQAIMAAAAAAIAAAACAAICAAAAAgIAAgACAgICAgMDAwP8AAAD/AP//AAAA//8A/wD/////
    /yH/C05FVFNDQVBFMi4wAwEBAAAh+QQJCAAIACwAAAAAEAAQAAAIUQARCBxlsKBBAQgTGjA4UICChwoE
    LGSlwCFEiRQrQow4sWFCiw8/YgS5seTlkig5aky5ceTHlh8XSjRggORMmh0F2sxZcGdGnxRFYqSlsyjB
    gAA7
</value>
 </data>
<data mimetype="application/x-microsoft.net.object.bytearray.base64" name="btnAdd2.Image" type="System.Drawing.Bitmap, System.Drawi
  <value>
    R0lGODlhEAAQAIMAAAAAAIAAAACAAICAAAAAgIAAgACAgICAgMDAwP8AAAD/AP//AAAA//8A/wD/////
    /yH/C05FVFNDQVBFMi4wAwEBAAAh+QQJCAAIACwAAAAAEAAQAAAIUQARCBxlsKBBAQgTGjA4UICChwoE
    LGSlwCFEiRQrQow4sWFCiw8/YgS5seTlkig5aky5ceTHlh8XSjRggORMmh0F2sxZcGdGnxRFYqSlsyjB
    gAA7
</value>
```

Inheritance – Things You Should Know

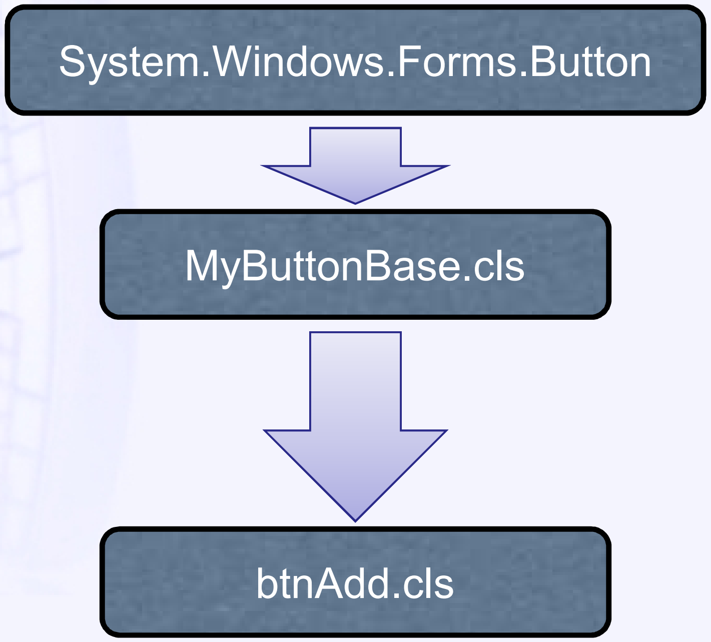
- Inheritance goes away

Button Example – MyButtonWindow.cls

```
   METHOD PRIVATE VOID InitializeComponent(  ):
THIS-OBJECT:btnAdd1:BackColor = System.Drawing.Color:Transparent.
THIS-OBJECT:btnAdd1:FlatAppearance:BorderColor = System.Drawing.Color:LightGray.
THIS-OBJECT:btnAdd1:FlatAppearance:BorderSize = 0.
THIS-OBJECT:btnAdd1:FlatStyle = System.Windows.Forms.FlatStyle:Flat.
THIS-OBJECT:btnAdd1:Image = CAST(resources:GetObject("btnAdd1.Image"),
                                                      System.Drawing.Image).
THIS-OBJECT:btnAdd1:Location = NEW System.Drawing.Point(2, 2).
THIS-OBJECT:btnAdd1:Name = "btnAdd1".
THIS-OBJECT:btnAdd1:Size = NEW System.Drawing.Size(25, 25).
THIS-OBJECT:btnAdd1:TabIndex = 0.
THIS-OBJECT:btnAdd1:UseCompatibleTextRendering = TRUE.
THIS-OBJECT:btnAdd1:UseVisualStyleBackColor = FALSE.
```

Inheritance – Things You Should Know

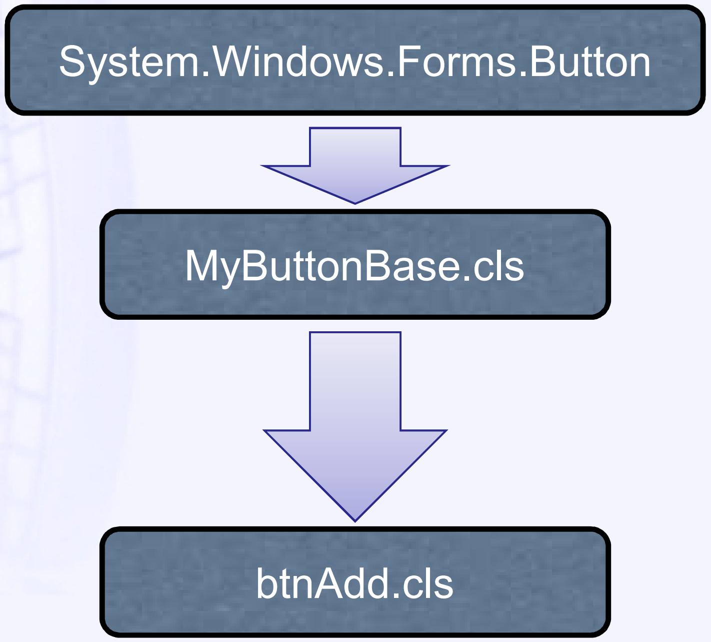
- Inheritance goes away
  - What if I need to change the class?



- Properties for all Buttons
  - Background Image, FlatStyle, Mouse Down, etc.
- Add Button Properties
  - Image, Tooltip, etc.

Inheritance – Things You Should Know

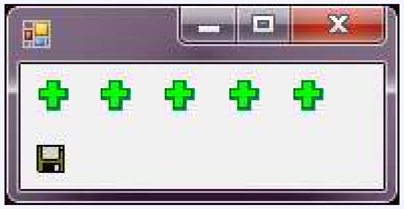
- Inheritance goes away
  - What if I need to change the class?



- Properties for all Buttons
  - ~~Background Image, FlatStyle, Mouse Down, etc.~~
- Add Button Properties
  - ~~Image, Tooltip, etc.~~

Inheritance – Things You Should Know

- Inheritance goes away
  - It is all embedded into the form



Button Example – MyButtonWindow.cls

```
    METHOD PRIVATE VOID InitializeComponent(  ):
    @VisualDesigner.FormMember (NeedsInitialize="true").
      THIS-OBJECT:btnAdd1 = NEW ItracControls.Button.btnAdd().
      THIS-OBJECT:SuspendLayout().
THIS-OBJECT:btnAdd1:BackColor = System.Drawing.Color:Transparent.
THIS-OBJECT:btnAdd1:FlatAppearance:BorderColor =
System.Drawing.Color:LightGray.
THIS-OBJECT:btnAdd1:FlatAppearance:BorderSize = 0.
THIS-OBJECT:btnAdd1:FlatStyle = System.Windows.Forms.FlatStyle:Flat.
THIS-OBJECT:btnAdd1:Image = CAST(resources:GetObject("btnAdd1.Image"),
                                                     System.Drawing.Image).
THIS-OBJECT:btnAdd1:Location = NEW System.Drawing.Point(2, 2).
THIS-OBJECT:btnAdd1:Name = "btnAdd1".
THIS-OBJECT:btnAdd1:Size = NEW System.Drawing.Size(25, 25).
THIS-OBJECT:btnAdd1:TabIndex = 0.
THIS-OBJECT:btnAdd1:UseCompatibleTextRendering = TRUE.
THIS-OBJECT:btnAdd1:UseVisualStyleBackColor = FALSE.
```

Inheritance – How to implement

- The Progress Workaround

1. Set properties in method outside of InitializeComponent

Button Example – MyButtonBase.cls

```
METHOD PUBLIC VOID  InitializeCustom (  ):
      THIS-OBJECT:TEXT = ''.
      THIS-OBJECT:BackColor = System.Drawing.Color:Transparent.
      THIS-OBJECT:FlatAppearance:BorderColor = System.Drawing.Color:LightGray.
      THIS-OBJECT:FlatAppearance:BorderSize = 0.
      THIS-OBJECT:FlatStyle = System.Windows.Forms.FlatStyle:Flat.
      THIS-OBJECT:Size = NEW System.Drawing.Size(25, 25).
      THIS-OBJECT:UseVisualStyleBackColor = FALSE.
END METHOD.
```

Button Example – btnDelete.cls

```
METHOD OVERRIDE PUBLIC VOID InitializeCustom(  ):
        DEFINE VARIABLE vImage AS System.Drawing.Image NO-UNDO.

        SUPER:InitializeCustom().
      THIS-OBJECT:SetToolTip("Delete Me Baby").
        vImage = System.Drawing.Image:FromFile("img\Deleterec.gif").
      THIS-OBJECT:IMAGE = vImage.
END METHOD.
```

Inheritance – How to implement

- The Progress Workaround

  2. Call it at run time

Button Example – Run MyButtonWindow.cls

```
DEFINE VARIABLE wWindow AS CLASS ItracControls.Window.

wWindow = NEW MyButtonWindow().
wWindow:InitializeCustom().
WAIT-FOR System.Windows.Forms.Application:Run (wWindow).
```

Button Example – MyButtonWindow.cls

```
    METHOD OVERRIDE PUBLIC VOID InitializeCustom(  ):

        SUPER:InitializeCustom().
      THIS-OBJECT:btnDelete1:InitializeCustom().

    END METHOD.
```

Inheritance – How to implement

- The Progress Workaround

  2. Call it at run time

Button Example – Run MyButtonWindow.cls

Or better yet….

```
DEFINE VARIABLE wWindow AS CLASS ItracControls.Window.

wWindow = NEW MyButtonWindow().
wWindow:InitializeCustom().
WAIT-FOR System.Windows.Forms.Application:Run (wWindow).
```

Or better yet…. *MyWindowBase.cls*

```
METHOD PUBLIC VOID  InitializeCustom (  ):
  DEFINE VARIABLE intCount       AS INTEGER NO-UNDO.
  DEFINE VARIABLE intLoop        AS INTEGER NO-UNDO.
  DEFINE VARIABLE objControl     AS CLASS System.Windows.Forms.Control NO-UNDO.

  intCount = THIS-OBJECT:Controls:Count.
  DO intLoop = 0 TO intCount - 1:
    objControl    = THIS-OBJECT:Controls:Item[intLoop].
    DYNAMIC-INVOKE (objControl, 'InitializeCustom') NO-ERROR.
  END.
END METHOD.
```

Inheritance – How to implement

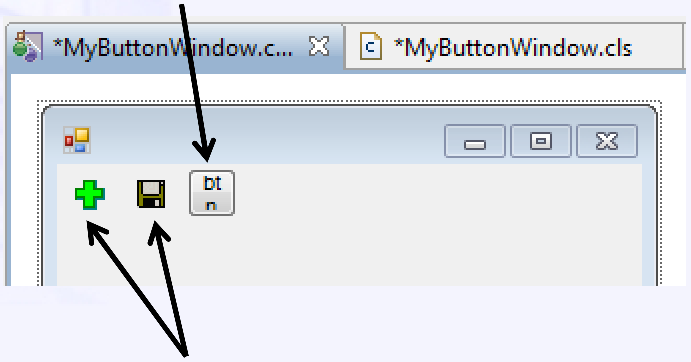
- The Progress Workaround

  3. What's the down side?

     - *You loose WYSIWYG functionality at development time*

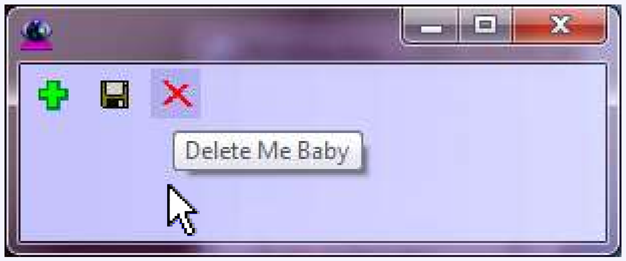
Development

with customization



without customization

Run Time



Demo

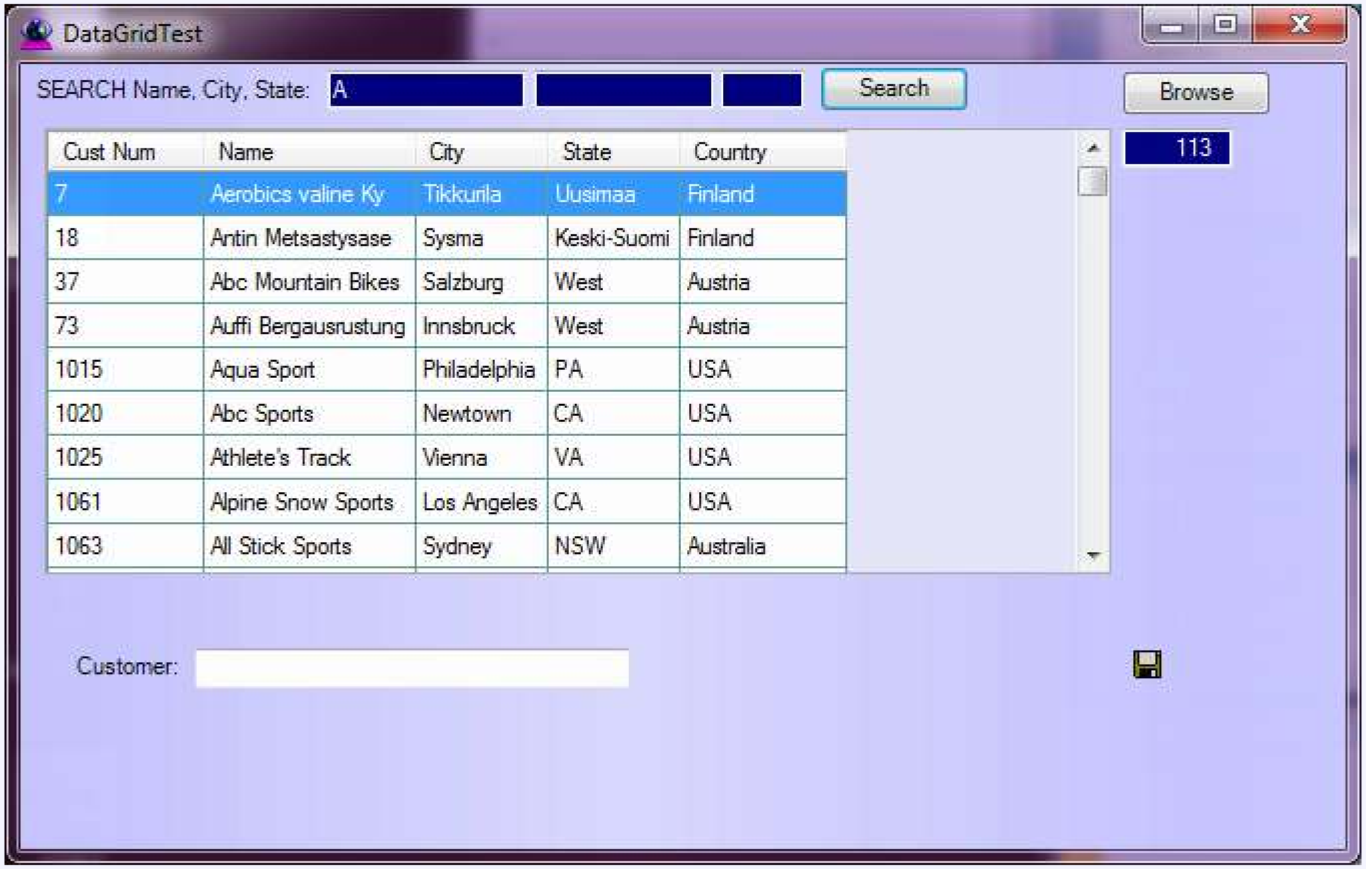
Custom Controls

- Why?
  - Protect your application
  - You don’t know what may change for your App

    Do it even if you think you don’t need to
- Microsoft Native vs. 3rd Party Controls?
  - Easier to work with the fundamentals
  - What is the 3rd party’s future
  - You say when and how to change the look/behavior
  - More Updates, More Often

Custom Controls - ABL Centric Ideas

- The Datagrid



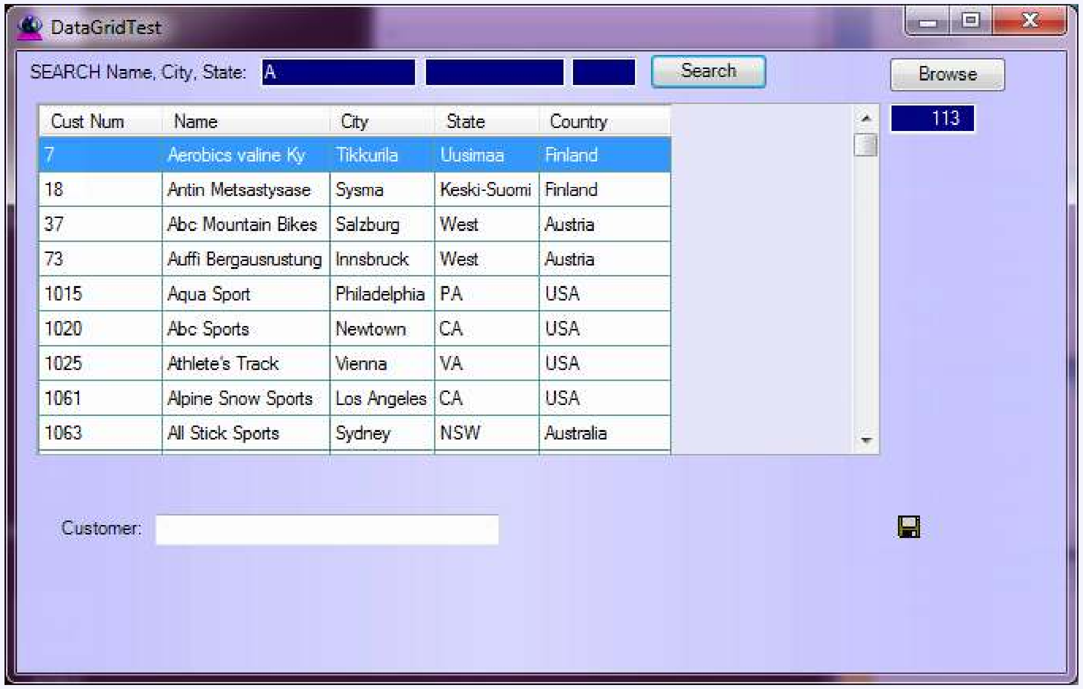
SEARCH Name, City, State: A

Cust Num	Name	City	State	Country
7	Aerobics valine Ky	Tikkurila	Uusimaa	Finland
18	Antin Metsastysase	Sysma	Keski-Suomi	Finland
37	Abc Mountain Bikes	Salzburg	West	Austria
73	Auffi Bergausrustung	Innsbruck	West	Austria
1015	Aqua Sport	Philadelphia	PA	USA
1020	Abc Sports	Newtown	CA	USA
1025	Athlete's Track	Vienna	VA	USA
1061	Alpine Snow Sports	Los Angeles	CA	USA
1063	All Stick Sports	Sydney	NSW	Australia

Customer:

Custom Controls - ABL Centric Ideas

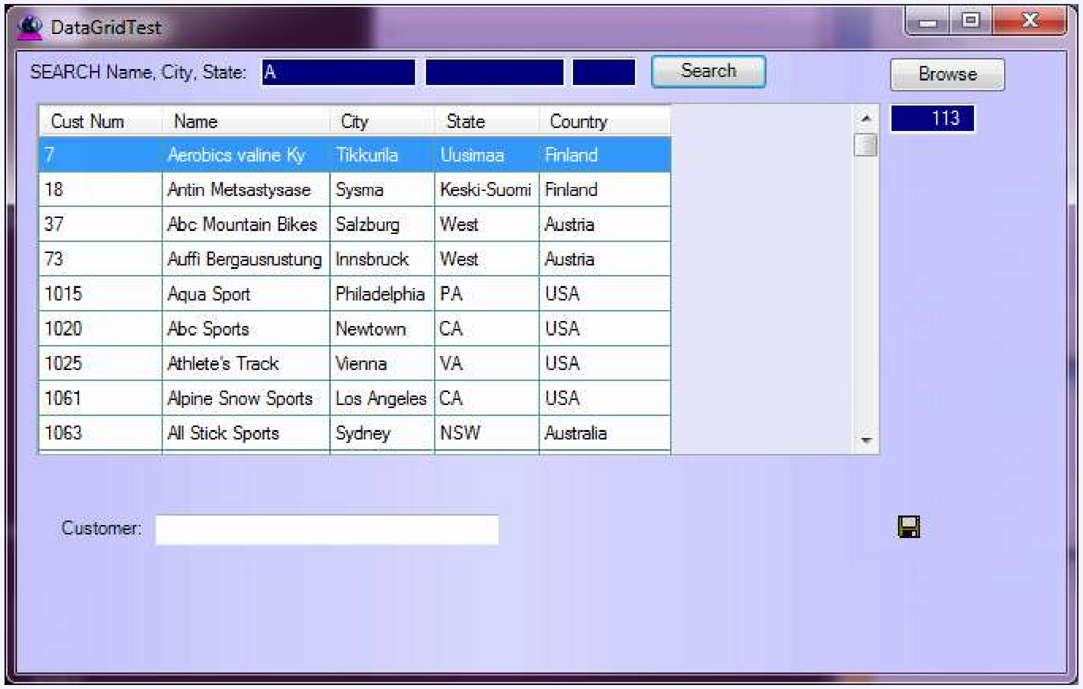
- The Datagrid
  - Hide and Display Columns
  - Change Column Order
  - Resize with container
  - Multi column sort
  - Query



Cust Num	Name	City	State	Country
7	Aerobics valine Ky	Tikkurila	Uusimaa	Finland
18	Antin Metsastysase	Sysma	Keski-Suomi	Finland
37	Abc Mountain Bikes	Salzburg	West	Austria
73	Auffi Bergausrustung	Innsbruck	West	Austria
1015	Aqua Sport	Philadelphia	PA	USA
1020	Abc Sports	Newtown	CA	USA
1025	Athlete's Track	Vienna	VA	USA
1061	Alpine Snow Sports	Los Angeles	CA	USA
1063	All Stick Sports	Sydney	NSW	Australia

Custom Controls - ABL Centric Ideas

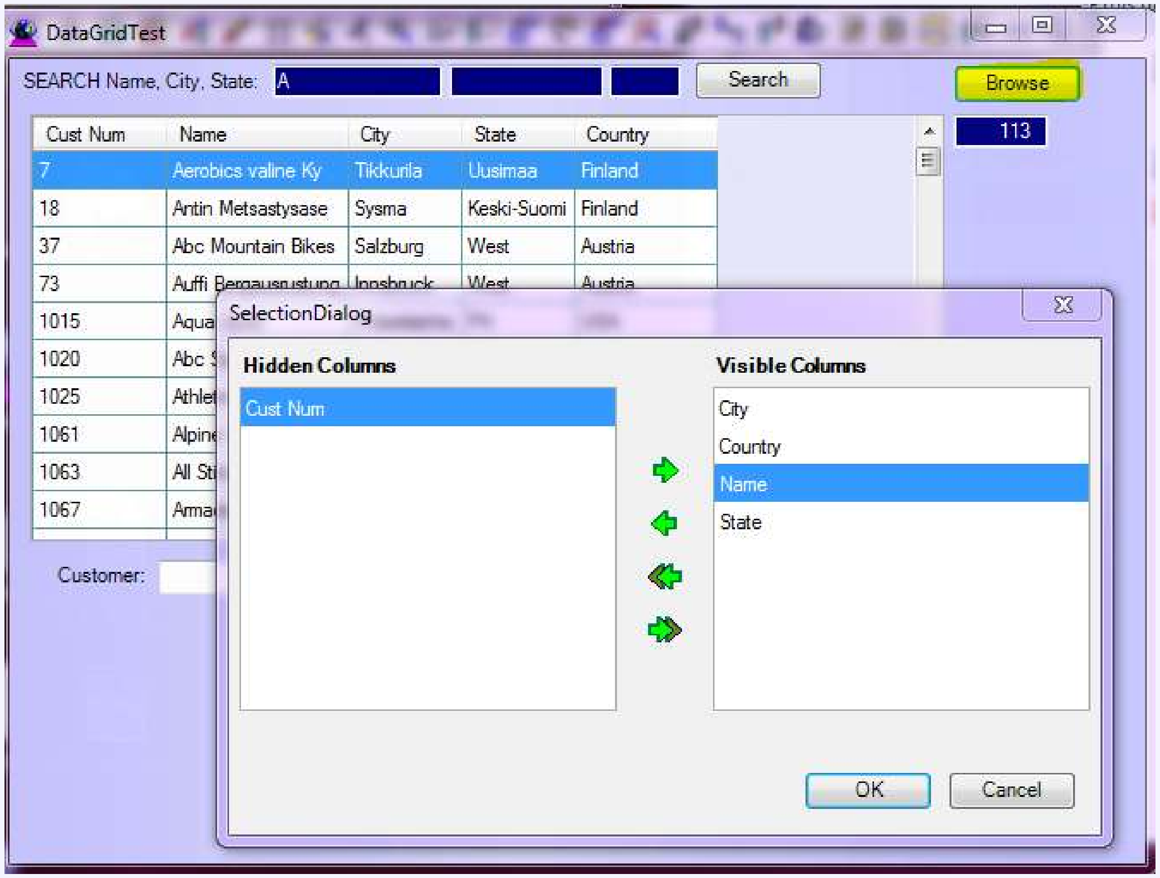
- The Datagrid
  - ✗ Hide and Display Columns
  - ✓ Change Column Order
  - ✓ Resize with container
  - ✗ Multi column sort
  - ✗ Query



Cust Num	Name	City	State	Country
7	Aerobics valine Ky	Tikkurila	Uusimaa	Finland
18	Antin Metsastysase	Sysma	Keski-Suomi	Finland
37	Abc Mountain Bikes	Salzburg	West	Austria
73	Auffi Bergausrustung	Innsbruck	West	Austria
1015	Aqua Sport	Philadelphia	PA	USA
1020	Abc Sports	Newtown	CA	USA
1025	Athlete's Track	Vienna	VA	USA
1061	Alpine Snow Sports	Los Angeles	CA	USA
1063	All Stick Sports	Sydney	NSW	Australia

Custom Controls - ABL Centric Ideas

- The Datagrid
  - Hide and Display Columns
  - Multi column sort



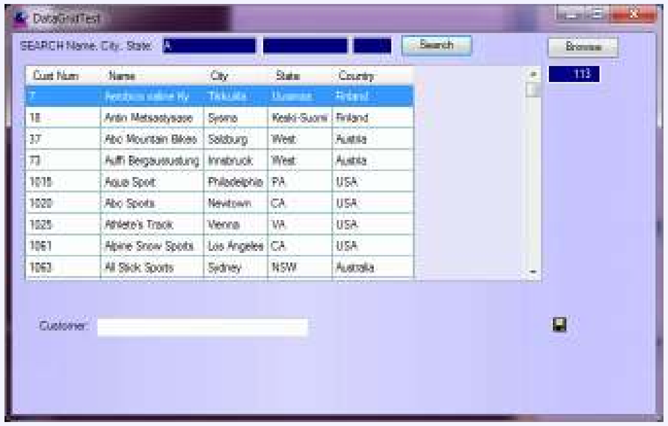
Cust Num	Name	City	State	Country
7	Aerobics valine Ky	Tikkurila	Uusimaa	Finland
18	Antin Metsastysase	Sysma	Keski-Suomi	Finland
37	Abc Mountain Bikes	Salzburg	West	Austria
73	Auffi Bergausrustung	Innsbruck	West	Austria
1015	Aqua			
1020	Abc S			
1025	Athlet			
1061	Alpine			
1063	All Sti			
1067	Arma			

Hidden Columns	Visible Columns
Cust Num	City
	Country
	Name
	State

Demo

Custom Controls - ABL Centric Ideas

- The Datagrid
  - Query



Cust Num	Name	City	State	Country
7	Aerobics valine Ky	Tikkurila	Uusimaa	Finland
18	Antin Metsastysase	Sysma	Keski-Suomi	Finland
37	Abc Mountain Bikes	Salzburg	West	Austria
73	Auffi Bergausrustung	Innsbruck	West	Austria
1015	Aqua Spot	Philadelphia	PA	USA
1020	Abc Sports	Newtown	CA	USA
1025	Athlete's Track	Vienna	VA	USA
1061	Alpine Snow Sports	Los Angeles	CA	USA
1063	All Stick Sports	Sydney	NSW	Australia

Required Properties/Methods

```
METHOD PRIVATE VOID CustDataGridTest_Load(INPUT sender AS System.Object,
                                          INPUT e AS System.EventArgs ):
  /* initialize grid */
  hdlBuffer = BUFFER ttbCustomerList:HANDLE.
  dataGrid1:KeyFieldName  = 'CustomerPK'.
  dataGrid1:InitializeGrid(hdlBuffer,'CustNum,Name,City,State,Country').
```

Refresh/Open the Grid's query

```
dataGrid1:OpenTheQuery().
```

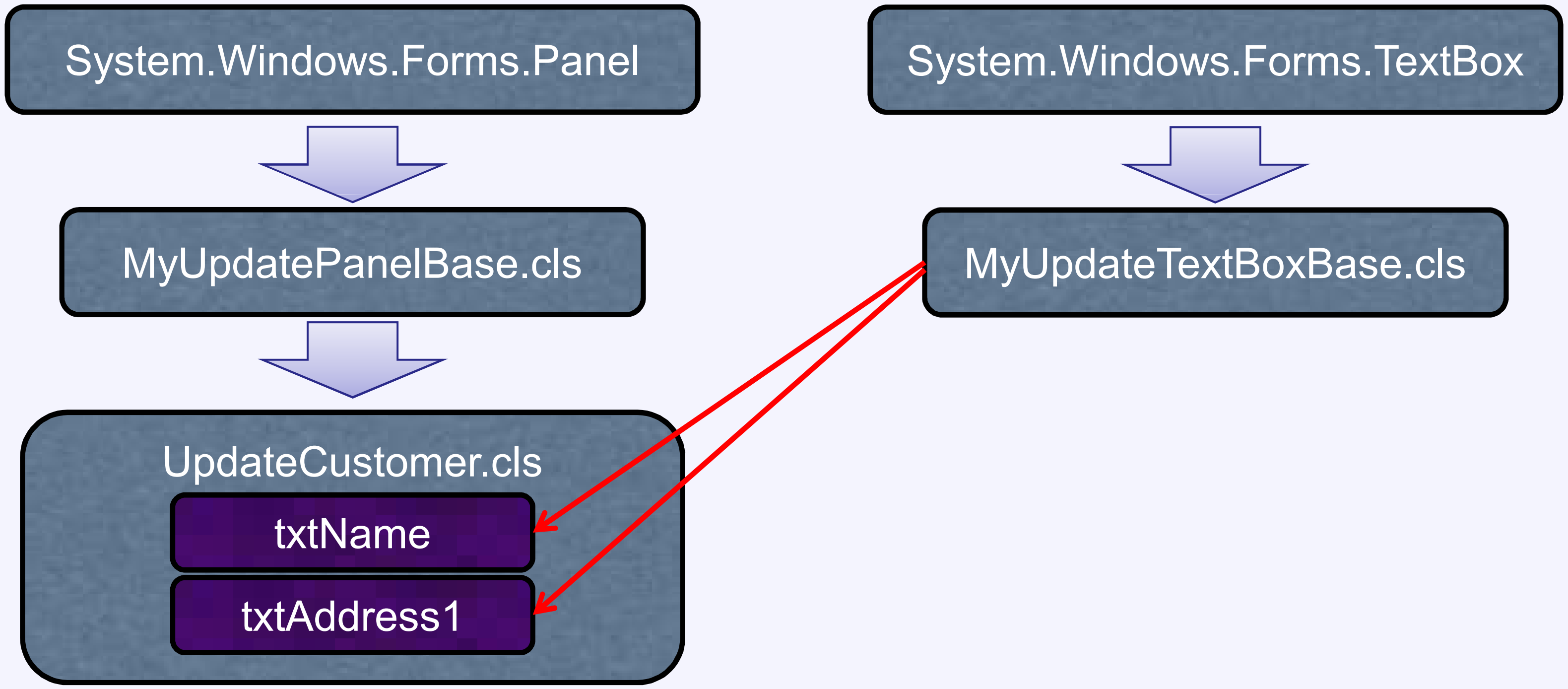
Custom Controls - ABL Centric Ideas

- The Datagrid
  - Query   Other Properties

```
/* HiddenFields MUST be set before InitializeGrid is called */
DEFINE PUBLIC PROPERTY HiddenFields AS CHARACTER NO-UNDO
/* HiddenFieldsUser MUST be set before InitializeGrid is called */
/* User Hidden fields                                           */
DEFINE PUBLIC PROPERTY HiddenFieldsUser AS CHARACTER NO-UNDO
/* HiddenFields is converted to HiddenFieldsRT at run time  */
/* This is what is hidden at run time System + User         */
DEFINE PRIVATE PROPERTY HiddenFieldsRT AS CHARACTER NO-UNDO
/* ColumnWidthUser MUST be set before InitializeGrid is called */
DEFINE PUBLIC PROPERTY ColumnWidthUser AS CHARACTER NO-UNDO
/* ColumnOrderUser MUST be set before InitializeGrid is called */
DEFINE PUBLIC PROPERTY ColumnOrderUser AS CHARACTER NO-UNDO
/* Query result count                                           */
DEFINE PUBLIC PROPERTY RecordCount AS INTEGER INITIAL 0 NO-UNDO
/* Current Sort                                                 */
DEFINE PUBLIC PROPERTY SortFieldList AS CHARACTER NO-UNDO
/* Delimited pairs When selecting pos 1 sort on pos 2           */
DEFINE PUBLIC PROPERTY SortXreferenceList AS CHARACTER NO-UNDO
/* WHERE clause criteria                                        */
DEFINE PUBLIC PROPERTY WhereClauseFilter AS CHARACTER NO-UNDO
```

Custom Controls - ABL Centric Ideas

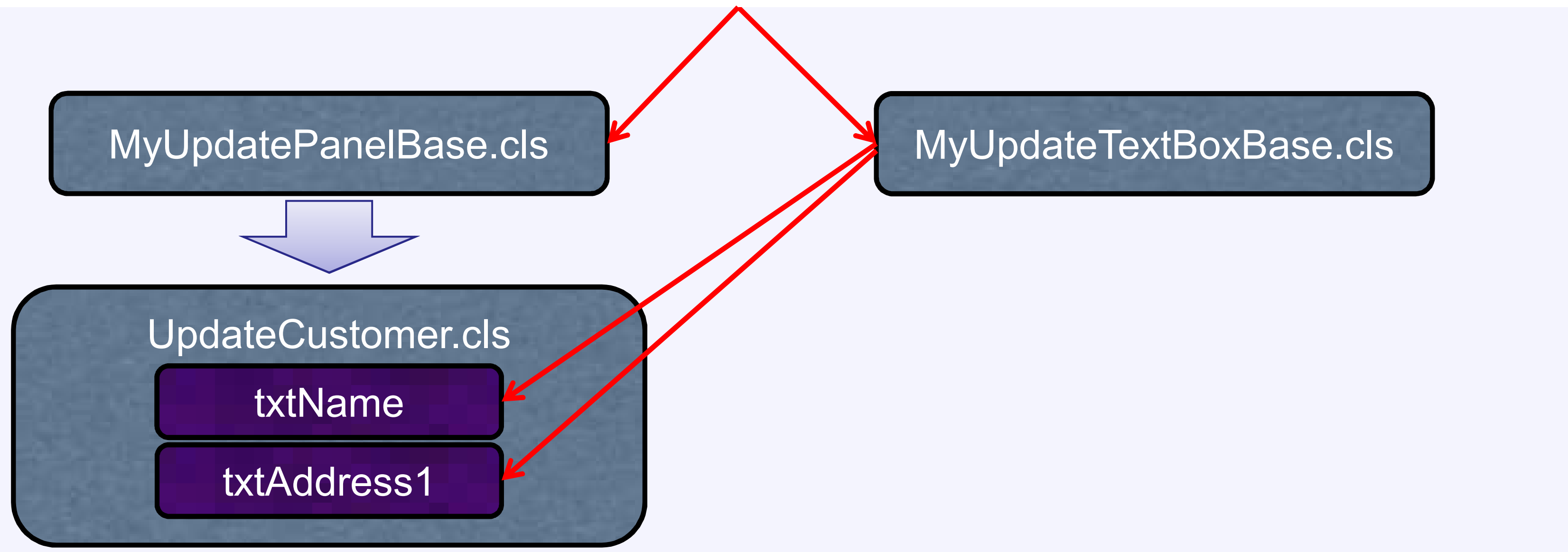
- CRUD
  - The Panel: The .Net “Data Viewer”
  - No data binding… No worries



Custom Controls - ABL Centric Ideas

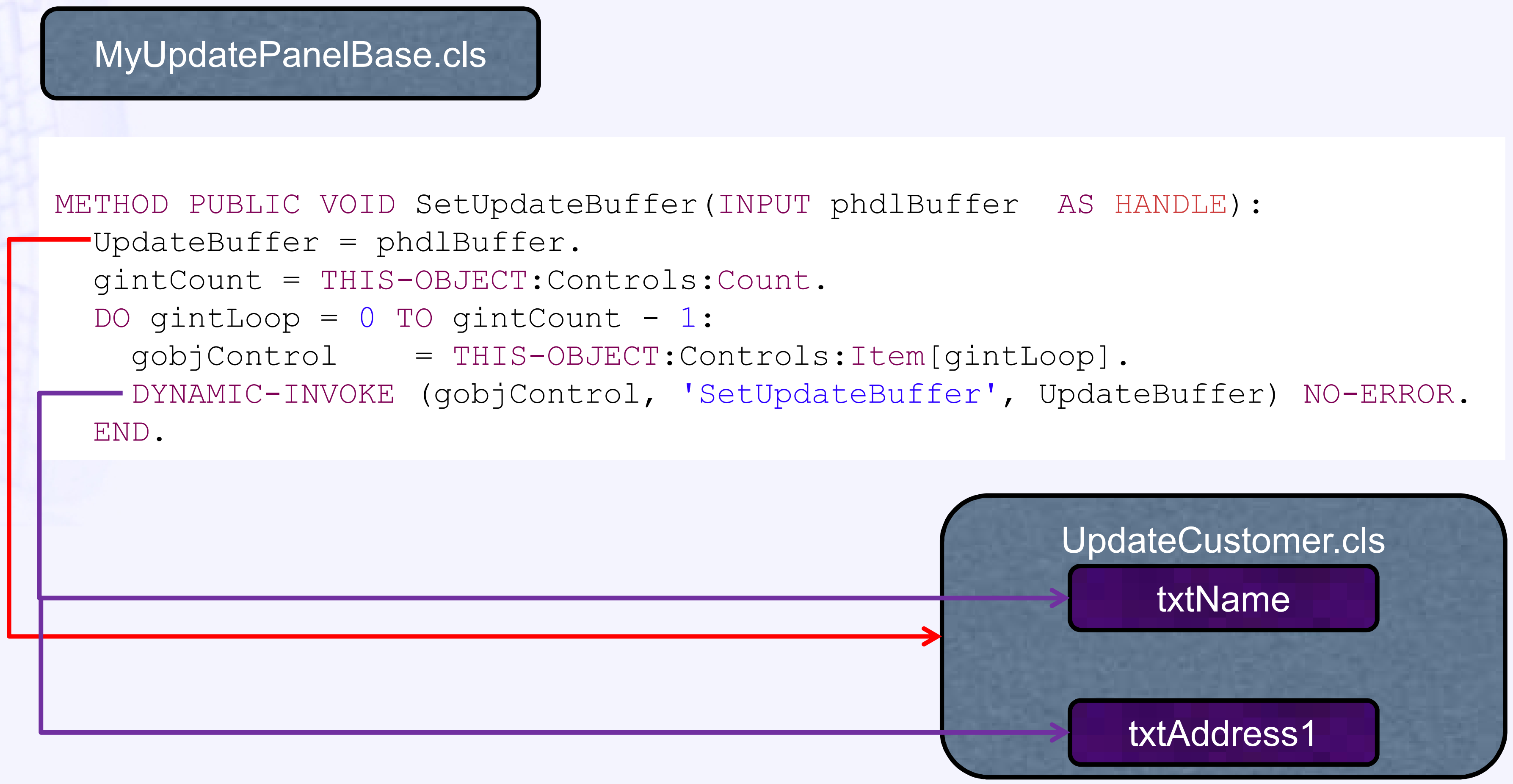
- CRUD
  - The Panel as the .Net “Data Viewer”
  - No data binding… No worries

```
DEFINE PUBLIC PROPERTY UpdateBuffer AS HANDLE NO-UNDO

METHOD PUBLIC VOID SetUpdateBuffer(INPUT phdlBuffer  AS HANDLE):
```



Custom Controls - ABL Centric Ideas

- CRUD
  - No data binding… No worries

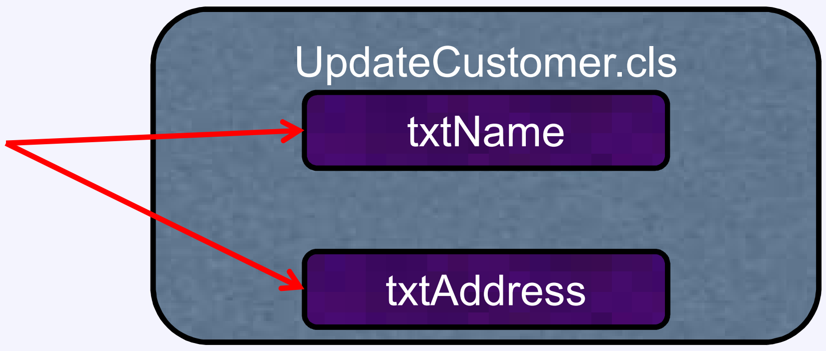


```
METHOD PUBLIC VOID SetUpdateBuffer(INPUT phdlBuffer  AS HANDLE):
  UpdateBuffer = phdlBuffer.
  gintCount = THIS-OBJECT:Controls:Count.
  DO gintLoop = 0 TO gintCount - 1:
    gobjControl    = THIS-OBJECT:Controls:Item[gintLoop].
    DYNAMIC-INVOKE (gobjControl, 'SetUpdateBuffer', UpdateBuffer) NO-ERROR.
  END.
```

Custom Controls - ABL Centric Ideas

- CRUD
  - No data binding… No worries

<Object-Name> = [txt] + <Buffer-Field-Name>



MyUpdateTextBoxBase.cls

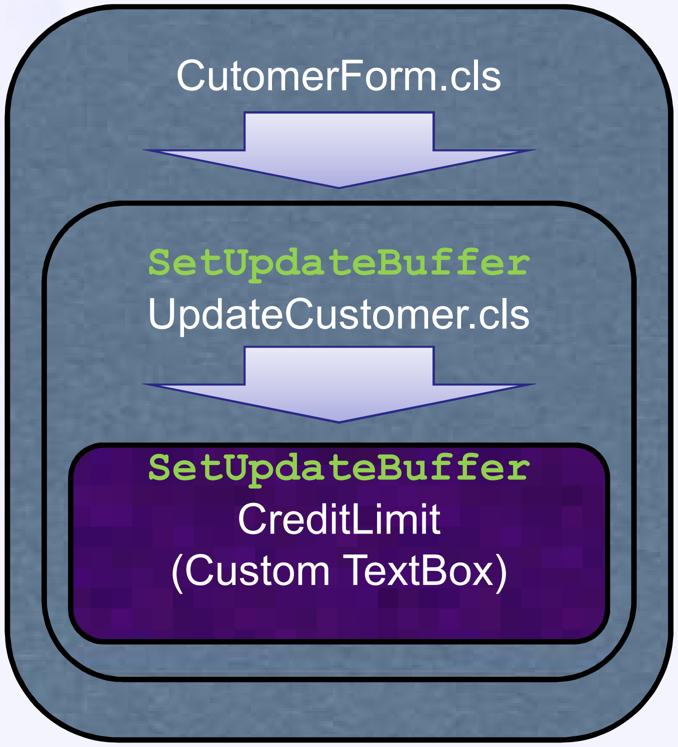
```
METHOD PUBLIC  VOID UpdateTextBox_TextChanged(INPUT sender AS System.Object,
                                              INPUT e AS System.EventArgs ):

  UpdateBuffer:BUFFER-FIELD(BufferFieldName()):BUFFER-VALUE = THIS-OBJECT:TEXT NO-ERROR.
```

Demo

Custom Controls - ABL Centric Ideas

- CRUD
  - Formats and Masking
    - Set Field Format & Set Mask

```
DEFINE PUBLIC  PROPERTY FORMAT       AS CHAR....
DEFINE PUBLIC  PROPERTY LegalChars   AS CHAR....
```



Custom Controls - ABL Centric Ideas

- CRUD
  - Formats and Masking
    - Use Format & Set Mask

```
METHOD PRIVATE VOID UpdateTextBox_KeyPress
  (INPUT sender AS System.Object, INPUT e AS System.Windows.Forms.KeyPressEventArgs ):

      IF      LegalChars > ''
         AND NOT CAN-DO (LegalChars,e:keychar) THEN DO:
        e:keychar = ''.
        RETURN.
      END.
  IF CAN-DO('DECIMAL,INTEGER',THIS-OBJECT:DATA-TYPE) THEN DO:
    /* manipulate the input and assign gdecNewDecimalValue */
   THIS-OBJECT:TEXT = STRING(gdecNewDecimalValue, THIS-OBJECT:FORMAT).
     e:keychar = ''.
  END.
```

Demo

?

Mike McMillan

Intui-Tech

PugChallenge@Intui-Tech.com

Thanks a lot, eh!

- Special thanks to….
  - Brian Maher
  - Shelley Chase
  - Peter Judge

Mike McMillan

Intui-Tech

PugChallenge@Intui-Tech.com