SORTING DATA WITH A PROBINDINGSOURCE AND .NET CONTROLS

John Sadd Fellow and OpenEdge Evangelist Document Version 1.0 December 2009





DISCLAIMER

Certain portions of this document contain information about Progress Software Corporation's plans for future product development and overall business strategies. Such information is proprietary and confidential to Progress Software Corporation and may be used by you solely in accordance with the terms and conditions specified in the PSDN Online (http://www.psdn.com) Terms of Use (http://psdn.progress.com/terms/index.ssp). Progress Software Corporation reserves the right, in its sole discretion, to modify or abandon without notice any of the plans described herein pertaining to future development and/or business development strategies. Any reference to third party software and/or features is intended for illustration purposes only. Progress Software Corporation does not endorse or sponsor such third parties or software.

This document accompanies a series of presentations on sorting data in your .NET user interface using the ProBindingSource control together with Visual Designer and .NET controls. First I put together the supporting code to show some of the capabilities and also some of the issues around sorting data in the user interface as opposed to sorting in the data source. I create a new ABL Interface .cls file for a set of related classes, create a class that uses the Interface, and create a form with an Infragistics grid and a ProBindingSource to manage and display some data.

[This part of the document corresponds to the video presentation that is Part 1 of the Data Sorting series:]

Starting with the usual sequence of **File -> New**, I select **ABL Interface**. An interface is a special type of .cls file that defines a contract that other classes agree to adhere to by implementing the interface. In this case this is a very simple start on a Model to allow me to manage and provide access to different data sources in a consistent way. Typically interface names start with I, so this is IModel. I can enter a Description for what its role is, and when I click finish Architect generates the skeleton of a source file named IModel.cls.

You can see the generated ABL identifies this as an INTERFACE rather than a CLASS:

And what I enter in ABL is in effect prototypes of methods that any class that implements this interface must provide. The method definition just identifies the name and signature.

```
USING Progress.Lang.*.

INTERFACE IModel:

METHOD PUBLIC VOID FetchData (INPUT pcFilter AS CHARACTER ).

METHOD PUBLIC VOID SortData (INPUT pcSort AS CHARACTER ).

METHOD PUBLIC HANDLE GetQuery().

END INTERFACE.
```

The first one is **FetchData**, which tells the Model instance to populate itself with some set of data of the type the Model class manages. I just allow for an optional filtering string as an argument.

Because this session is about sorting data, I define a separate method **SortData** to communicate sort criteria to the model. This illustration shows how the FetchData and SortData methods in a Model class are expected to be referenced from a user interface View class:

– User	Interface "View" Class ————
moAny	Model:FetchData(<filter criteria="">).</filter>
moAny	Model:SortData(<sort criteria="">).</sort>
	,, ,, , ,, , ,, , ,, , ,, , ,, , ,, , ,, , ,, , ,, , ,, , ,,, ,
∏ ^{Any∧}	Nodel.cls
AnyA /* ds	Nodel.cls •DataSet definition */
⊢ Any∧ /* ds MET	Nodel.cls DataSet definition */ HOD FetchData(pcFilter):
- AnyA /* ds MET /* Ra	Nodel.cls
⊢ Any∧ /* ds MET /* Ra MET	Model.cls DataSet definition */ HOD FetchData(pcFilter): etrieve data into local ProDataSet */ HOD SortData(pcSortString):
- AnyA /* ds MET /* Rd MET	Nodel.cls DataSet definition */ HOD FetchData(pcFilter): etrieve data into local ProDataSet */ HOD SortData(pcSortString): port retrieved data */
- AnyA /* ds MET /* Ra MET /* Sa	Nodel.cls DataSet definition */ HOD FetchData(pcFilter): etrieve data into local ProDataSet */ HOD SortData(pcSortString): ort retrieved data */

And the class needs to be able to provide access to the data that's been retrieved, in this case the handle to a query on the data, so the method **GetQuery** has a return type of HANDLE:

User Interface "View" Class moMyBindingSource:Handle = moAnyModel:GetQuery(). - AnyModel.cls /* dsDataSet definition */ METHOD GetQuery(): RETURN hTempTable.

An interface can have definitions for methods, properties, and events. The only restriction is that they all need to be PUBLIC. The Interface represents the public contract that a class adheres to so that other classes can reference it consistently and reliably. I save and compile the interface, and then create a new **ABL Class** to use the interface. This is going to be the first class that adheres to the contract defined by the interface, in this case to provide access to Customer data. If I want to specify one or more interfaces that the class implements, I select the Add button next to the interface section:

DE OpenEdge Editor - Samples/IModel.cls - Op	🗊 New ABL Class		_ 🗆 🗵	_O×
File Edit Source Navigate Search Project (Create a user-defined	class		
📑 - 🔡 📥 🔏 💷 🦦 🏇 - 💽	Ontionally sper	ce Selection 📃 🗆 🗙	C	
CopenEdge Editor	implements. Type filter	text:		
	Package root: Matching in Package: Acces	terfaces: sibility.CAcCPropServices	Browse	
P DealerEmpXSD.p	Class name: 0 Acces	sibility.IAccessibleHandler		
dsCustomer.i	Modifiers: O Acces	sibility.IAccIdentity		
I IModel cls	O Acces	sibility.IAccPropServer	- During 1	
NavPanel.cls	Innerics: U Acces	sibility IAccPropServices	Browse	
TextDataModel.cls	Implements:	nControl - System.web.ut.webControls	Add	
IttDealer.i Image: Contine Image: Contimage: Contine Image: Contine Image: Contimage: Contimage:	Specify the cod	nControl - System Windows Forms Digital - System Runtime. InteropServices.ComT Digital - System. Windows.Forms Istics.Shared.ICreateItemCallback Intro: Shared.ICreateItemCallback	Remove	*/
	Specify the reta	Add OK Cancel		RACTER).
	Description			
	Description.			<u>_</u>
	Purpose:		A	
			<u>*</u>	h32.exe (Nov 19, 2009 2:52:50 PM)
↑ €	0	Finish	Cancel	

My new Interface **IModel** is in the list along with all the others that are part of the support for Microsoft .NET controls and Infragistics controls and so forth. In addition I check the boxes in the **Create a user-defined class** wizard that generate a default skeleton for a constructor to execute when the class is run, and a destructor where I could put code to clean up when the class instance is deleted. When I click Finish I can take a look at the code that's been generated for me.

```
/*-----
File : CustomerModel
Purpose : Retrieves Customer database data into a ProDataSet
------*/
USING Progress.Lang.*.
CLASS CustomerModel IMPLEMENTS IModel:
```

You can see that the CLASS statement tells the compiler that this class IMPLEMENTS the interface IModel.

The first element I add to the file is a reference to a ProDataSet that the Model uses to hold data. Now of course data could be retrieved and managed in a number of ways, which is one reason to keep the Model class that knows the details separate from the UI, but this is one specific example of how the interaction can be handled, using a temp-table and dataset definition such as this.

{dsCustomer.i}

The Model as the top of the data management stack on the client is responsible for interacting with wherever and whatever the actual data source is, and in this case getting the data into the DataSet that the definition in **dsCustomer.i** represents:

In a complete application there would be a separation between a Model class on the client that holds data locally for the user interface to use, and classes or procedures on the server where the database is connected, as illustrated here:



In this case I leave the basic code that does all the data management work right here in this one simplified class, as shown here:



I need a query definition for the actual data source, an ABL DATA-SOURCE definition for the DataSet to use, and a variable to hold the handle of a query on the temp-table that the data gets loaded into, to supply back to the user interface.



Next I need a few statements in the constructor to get the object set up when it's instantiated. So I need to attach the DATA-SOURCE to its temp-table, and then set up a dynamic query on that temp-table which is the connecting point to the binding source in the user interface.



This sketch shows the distinction between the query on the database table associated with the DataSource, used to populate the DataSet, and the query on the DataSet's temp-table of data after it's been retrieved.



Next you can see the effects of the default code generation I get from Architect by saying that this class implements **IModel**. Because I have to code an implementation of each of the methods defined in the interface, I'm provided with a starting point to remind me to fill each method in.



Without some skeleton implementation at least, I'd get a compiler error if I tried to save and compile **CustomerModel.cls**, because the compiler is cross-referencing this with the interface definition.

Here's the code that I write for FetchData:

```
METHOD PUBLIC VOID FetchData( INPUT pcFilter AS CHARACTER ):
DEFINE VARIABLE cPrepare AS CHARACTER NO-UNDO.
    cPrepare = "FOR EACH AutoEdge.Customer".
    IF pcFilter NE "" THEN
        cPrepare = cPrepare + " WHERE " + pcFilter.
    QUERY qCustomer:QUERY-PREPARE (cPrepare).
    DATASET dsCustomer:FILL().
    httCustQuery:QUERY-OPEN ().
END METHOD.
```

It prepares a query for the DataSet to use to FILL the temp-table, and for simplicity's sake, just expects any Filtering information passed in to be a WHERE clause for the database query. In real life this would require translation from the user interface understanding of a filter and what the database would require, but this will do for now.

And likewise I have a code for the **GetQuery** method, which just returns the temp-table query handle from the DataSet.



I haven't filled in code for **SortData** yet – that happens later – but because there's at least a skeleton implementation my syntax check is OK, and I can save the class and compile it.

Next I create a form to display the data in. This uses the Infragistics WinGrid control, which we call the **UltraGrid**. This advanced grid control has properties you can set to allow it to sort the data it displays on its own, so it gives me an opportunity to compare letting the user interface handle data sorting for display and having the data source handle it.

There are several alternatives for creating a binding source. In this case I select the **ProBindingSource** directly under the **OpenEdge Controls** in Visual Designer's Toolbox:

🕞 OpenEdge Editor - S	amples/CustomerUltraGri	d.cls - OpenEdç	e Architect - C:\Gui4DotNet				<u>_ ×</u>
File Edit Design Navi	gate Search Project Oper	nEdge Run Wi	ndow Help				
📬 🕶 🔡 📥 🔏	💷 🦦] 恭 • 🔘 • 🤇	🎍 • 🗍 🔗 •] 🖢 + 🖗 + 🏷 🔶	×			
😭 💕 OpenEdge Editor	•						
🗳 Resources 🛛 🦢	🔿 👰 📄 😫 🍸 🖓	IModel.c	s CustomerModel.cls	S *CustomerUltraGrid.cls (Design)	×		- 0
- CustomerUlt	raGrid.ds	_				Toolbox	ą
B DealerEmp.c	lgm	🔛 Cust	omerUltraGrid			70.000	
- 🕒 DealerEmp.×	sd						
DealerEmpFo	orm.cls					Microsoft Controls	
🔄 🖸 DealerEmpM	odel.cls					OpenEdge Controls	
DealerEmpX3	5D.p						
dsCustomer.	i					Rointer	
dsDealerEmp	b.i					ProBindingSource	
IModel.cls						WindowContainer	
NavPanel.ds	-	-			P	🔳 OpenEdge Ultra Cor	ntrols
	Broperties 😚 🖵 🛙						
		_					
	🗄 💈 😽 🐪						
CustomerUltraGrid : Prog	ress.Windows.Form	2					
Properties Events	1						
	0, 0, 0, 0	1					
previousState	Normal	7111					
RightToLeft	No			U	0		
RightToLeftLayout	False						
Showlcon	True						
ShowInTaskbar	True						
⊞ Size	490, 293						
SizeLinpStyle	Auto						
StartPosition	WindowsDefaultLocation						
Tout	Customort UltraGrid	📜 🗐 Console	🔀 🖹 Problems 🧔 Tasks		🗙 🍇 📑 🔓	1 🖉 🖉 🛃 🖬 🖬) - 🗆 🗖
T		<pre>cterminated</pre>	> CustomerUltraGrid [OpenEdge	Application] C:\Progress\OpenEdge102b\	bin\prowin32.exe (M	Nov 19, 2009 2:52:50 PM)	
Text							4
The text associated with	n the control.						
							v
		40					
户 🖹				Writable			

In the ProBindingSource Designer, I select Import schema from file:

)E OpenEdge Editor - S	amples/Custom	rUltraGrid.cls - OpenEdge Architect - C:\Gui4DotNet	
-ile Edit Design Navi	gate Search Pro	act OpenEdge Run Window Help	
📑 🕂 🔡 🖕 🔤	💷 🕲 🛛 🌮	· O • Q •]	
😭 鐣 OpenEdge Editor	r		
🐉 Resources 🖾 📛	🔿 🔞 🖪 🔮	🗢 🖓 🖓 🔁 🖸 🖸 Model.cls 🛛 🔄 CustomerModel.cls 🖏 *CustomerUltraGrid.cls (Design)	k 🛛 🖓 🗖
CustomerUlt	raGrid.cls		- Toolbox 9
🛛 🚯 DealerEmp.d	lgm	📴 ProBindingSource Designer	
🛛 🎯 DealerEmp.x	sd	1 1 X A J B 🙀	± CustomControls
🗌 💽 DealerEmpFo	orm.cls		Hicrosoft Controls
🗌 🔄 DealerEmpMi	odel.cls	Tables	= OpenE dge Controls
🗌 🖻 DealerEmpXS	5D.p	Click the Import schema from file button	- opene age control
 dsCustomer. 	i		Pointer
🗌 🚺 dsDealerEmp	o.i		💞 ProBindingSource
IModel.cls			WindowContainer
🗌 🔄 NavPanel.cls			🗉 OpenEdge Ultra Controls
	. Duranaukia		
	Propercie:		
	田		
CustomerUltraGrid : Progr	ress.Windows.Form		
Properties Events	1		
	0.0.0.0		
previousState	Normal		H
RightToLeft	No		
RightToLeftLayout	False		
Showlcon	True		
ShowInTaskbar	True		
	490, 293	# No root table defined.	
SizeGripStyle	Auto	- OK _ (amod I
StartPosition	WindowsDefault		
Taut	Customent	🔜 🖳 Console 🐹 🚺 Problems 🕢 Tasks	x 🛪 🗔 📮 🐖 🚽 📮 • 😁 • 🗆 🖬
TN	LustomerUltra		n)prowip32.exe (Nov 19, 2009 2:52:50 PM)
Text			
The text associated with	n the control.		
T* 🖹		Writable	
			1

This is prepared to use an XSD file, or an ABL source file containing a definition. I select the DataSet definition **dsCustomer.i** to use as a basis for the binding source.

Back in the Design View I want to give the binding source a meaningful name. Since this a definition that will appear in the main block of the class, above any method definitions, it represents what we call a "data member" of the class, so name it **moBSCustomer**, using the naming convention "m" for member and "o" for object, since the variable will hold a reference to a binding source object.

Next I open up the list of **Ultra Controls**, and drop an **UltraGrid** onto the form. In the Quick Start wizard, I set the grid's data source to be the binding source I just created, and that's all I need to set to get started.

As with the binding source, I want to give the variable that holds the object reference for the grid a meaningful name in my source code. That's the object's **Name** property, which I set to **moUltraGridCustomer**.

The Text property is the title displayed at the top of the grid, and I set that to Customer Grid.

Here in the generated code are the variable definitions for the variables to hold my two object references:

	DEFINE	PRIVATE VARIABLE moBSCustomer AS Progress.Data.BindingSource NO-UNDO.	
	Int	fragistics.Win.UltraWinGrid.UltraGrid NO-UNDO.	
	DEFINE	PRIVATE VARIABLE moUltraGridCustomer AS	
	DEFINE	PRIVATE VARIABLE components AS System.ComponentModel.IContainer NO-UND	Э.
CLA	ASS Custo	omerUltraGrid INHERITS Form:	
USI USI USI	NG Prog NG Prog NG Infra	ress.Lang.*. ress.Windows.Form. agistics.Win.UltraWinGrid.*.	
-	File Purpose	: CustomerUltraGrid e : UltraGrid to display and sort Customer data */	
/*_			

I need another variable to hold a reference to the **CustomerModel** instance that I'll use to provide data to my "View", the user interface class:

DEFINE PRIVATE VARIABLE moCustomerModel AS CustomerModel NO-UNDO .

Next in the class's constructor, I need to create that **CustomerModel** instance, and then tell it to retrieve all the Customers – for starters I'm not specifying any filtering criteria. And finally, because the binding source needs to have its **Handle** property set to a buffer, a query, or a DataSet, I retrieve the query handle that I set up in the model class.

CONSTRUCTOR PUBLIC CustomerUltraGrid ():
<pre>SUPER(). InitializeComponent(). moCustomerModel = NEW CustomerModel(). moCustomerModel:FetchData(""). moBSCustomer:Handle = moCustomerModel:GetQuery().</pre>
CATCH e AS Progress.Lang.Error: UNDO, THROW e. END CATCH.
END CONSTRUCTOR.

Now I save this much so that we can see what happens when I run it. Here's the form with my grid, and you can see it's successfully attached the binding source to the Customer data held by the model.

🗊 OpenEdge Editor -	Samples/CustomerUltraG	rid.cl	s - OpenEdge Archi	tect - C:\Gui4DotNet				- U ×
File Edit Design Navigate Search Project OpenEdge Run Window Help								
📑 💕 OpenEdge Edit	or							
🍣 Resources 🛛 🤇			IModel.cls	CustomerModel.cls	S CustomerUltraGrid.cl	ls (Design) X 💽	CustomerUltraGrid.cls	
CustomerU	ltraGrid.cls						Toolbox	4
- 🔠 DealerEmp	.dgm		🖳 CustomerUl	traGrid				
🕑 DealerEmp	xsd							
DealerEmp	Form.cls			, (Customer grid		Microsoft Control	\$
DealerEmp	riodel.cis	👥 C	ustomerUltraGrid				🖃 OpenEdge Contro	ols
- i) dsCustome	r.i	_					🔄 🚽 📄 OpenEdge Ult	
dsDealerEn	10.1			Customer g	rid		- Pointer	
IModel.cls			CustomerFirstName	CustomerLastName	CustomerBirthCountry	CustomerE	A drive tion Countral	
NavPanel.o	ls	•	Robert	Kennedy	USA	07/13/1979	AnimationControl	
🖳 o star 🗖 oo ga			Peter	Wagner	USA	09/14/1987	State of the second sec	
Cutine St DB Str	JC E Propercies 23		Alice	Washington	Germany	05/19/1982		
	₩ 2↓ 🗔		Patrick	Lewis	Germany	03/19/1970	UltraCalcManager	
CustomerUltraGrid : Pro	gress.Windows.Form		Eric	Mvers	Germany	06/08/1950	UltraCalculator	
Properties Events	1		Laura	Torres	Germany	07/16/1984	UltraCalculatorDropE	own
Padding	0, 0, 0, 0		Carol	Murphy	Germany	12/01/1982 -	UltraLalendarLombo	
previousState	Normal	11				Þ	UltraCalendari ilo	
RightToLeft	No							
RightToLeftLayout	False						IlltraCheckEditor	
Showlcon	True						UltraColorPicker	
Showini askbar	Irue _			•			UltraCombo	
E Size CineCuieCtula	490, 293						UltraComboE ditor	-
StartPosition	WindowsDefaultLocation						•	ъC
Tag	ministrice of duite of duite							
Text	CustomerUltraGrid	-	Console 🔀 🚺	Problems ど Tasks		— × 4	× == 💩 🔄 🔄 🖉 • 🖸	•
	(r.)	-	CustomerUltraGrid [O	penEdge Application] C:\Pr	ogress\OpenEdge102b\bir	n\prowin32.exe (Nov	v 19, 2009 4:31:27 PM)	
The text associated w	th the control							<u>^</u>
The text desoundled w	arta sonao.							-
			<u>.</u>					
					Writable		1	

Now I know that somehow the **UltraGrid** is prepared to do data sorting for me, which seems like a nice feature, so I click on a column header, but nothing happens. So I've got to do some investigation to find out how to make this work for me.

[This next part of the document corresponds to the video presentation that is Part 2 of the Data Sorting series:]

Here's a quick tour of the **Infragistics Online Documentation** as an example of how it can help you understand what properties to set and what code to write for the controls in a form such as the one I'm building, and then use that information in my form to get the grid to show the data in a different sort order than I used to retrieve it from the database. Note that in addition to the online links shown in these examples, the Infragistics documentation is installed locally when you install OpenEdge 10.2B along with the Infragistics controls. You can access it from the Windows Start menu under **Programs->Infragistics->NetAdvantage for .NET->Windows Forms->Documentation->Documentation (local)**.

When I first run the form I discover that the sorting capability of the **UltraGrid** control isn't enabled by default, so I need to investigate what properties of the control have to be set to enable it to sort data for me. And just as I used MSDN in other presentations for information on Microsoft controls, I can go to the Infragistics website to get documentation on all the .NET controls in the **UltraControls** package that are available with OpenEdge 10.2.

Under the **Support** menu at <u>www.infragistics.com</u>, I can select **Online Documentation**. The controls that are available with OpenEdge 10.2B are identified as **Volume 2 of Windows Forms 2009**, so I select that version to get information on.



There's a lot of introductory material available that you can start with, but the two major sources of comprehensive information on all the controls are the **Developers Guide** and the **API Reference Guide**.

File Edit View Favorites Tools Help	File Edit View Favorites Tools Help					
🗑 Back + 🕥 + 💌 😰 🏠 🔎 See	irch 🤺 Favorites 🔗 🔗 - 🦭 - 🔜 🔟 - 🔜 🎇 🦄					
Address Address Address Address Address Address Address	e/WinForms/2009,2/CLR2.0/					
Google Search						
Infragistics	NetAdvantage® Windows Forms Online Help :: 2009.2					
	Copy URL to Clipboard Open Topic in New Window					
Contents	< 🗈 Language Filter: All					
🖃 📷 Getting Started	Infragistics(R) NetAdvantage(R) Windows Forms					
Welcome to NetAdvantage Windows Form	s Getting Started					
🗄 💽 Using the Help	In order to get you up and running quickly with the NetAdvantage Windows Forms product, we've 🗕					
🗉 😡 Installing NetAdvantage Windows Forms	provided you with this Getting Started section. Before you start using NetAdvantage Windows Forms, be					
Getting Support	,					
What's New	Wolcomo to NotAduantago Windows Forms					
B Revision History	If you are wondering "What exactly is "NetAdvantage Windows Forms" and what can it do for me?", then					
Known Issues and Breaking Changes	you should read this topic. It gives you a complete overview of the product.					
Breaking Changes in 2009 Volume 2	Using the Help					
Breaking Changes in 2009 Volume 1	In this section, you'll find useful topics on how to get the most out of the belo that is available. For					
i 📄 Breaking Changes in 2008 Volume 3	example, there is a topic that explains how to navigate and search through the information in the help.					
Breaking Changes in 2008 Volume 2	Installing NetAdvantage Windows Forms					
Breaking Changes in 2007 Volume 3	This section, guides you through the installation process and also provides you with some tins on					
Breaking Changes in 2007 Volume 1	installing NetAdvantage Windows Forms.					
Breaking Changes in 2006 Volume 3						
Rindwirtssdes	E-mail your feedback on this topic.					
API Reference Guide	Opinion about our help? Take our survey. Build Version: 9.2					
	Copyright © 1996-2009 Infragistics, Inc. All rights reserved.					

Let me see first what I can find out about the **WinGrid** control, which we call the **UltraGrid**, under the **Developers Guide**. Selecting the **Controls** section of the Guide, you see a list of all the available controls. There's introductory material on every control, but I go to the **Using WinGrid** section.



There's the section on **Sorting and Grouping** data with the grid, and within that, a **Sorting Columns** topic, which is what I want to enable on the control. And in that section I find exactly the question I want an answer to, how to enable sorting when the user clicks on a column:



There's information on setting the **HeaderClickAction** property to **SortSingle** or **SortMulti** to enable column sorting, along with some code samples in C# that are pretty close to what the equivalent ABL should look like.

De Ope		nerden Andrikens Cólona zo Mark	ĽЦ
File E	PetAdvantage® windows Forms Unline Help :: 2	JU9.2 - Microsoft Internet Explorer	
	G Back + 🕞 - 💌 😰 🎧 🔎 Search	💢 Favorites 🔣 🔯 - 🎲 - 💦 🔟 - 🛄 🕵 🥸	
🖧 Re	Address 🕘 http://help.infragistics.com/NetAdvantage/W	inForms/2009.2/CLR2.0/	
	Google Search 🔹	🛷 🖈 🐈 🔹 🖓 • 🗆 • 🏠 Bookmarks • 🎋 Check • 🚂 Translate • 🎦 AutoFill • 🌺 🔩 • 💮 Sign In •	7
	Infragistics	NetAdvantage \circledast Windows Forms Online Help :: 2009.2	
		Copy URL to Clipboard Open Topic in New Window	Is
Lusto Prot M M M M M M M M M M M M M M M M M M M		<pre>% Language Filter: All Infragistics(K) Meddawatage(R) Windows Forms Sorting Columns E-Hadyout.vveliue.meaueroitexection = meaueroitexection.sortsing(* In C#: private void ultra@rid1_InitializeLayout(object sender, Infragistics.Win.UltraWinGrid.InitializeLayoutEventArgs e) (</pre>	
Texi The	B WinGridBagLayoutManager	Opinion about our help? Take our survey. Build Version: 9.2 Intragistics Copyright © 1996-2009 Infragistics, Inc. All rights reserved.	*
] 🗣	🛃 Done	👔 👔 Internet	

I also find an example of setting the SortIndicator to Descending if I want to sort the data that way.



That's the kind of solution information on controls and their properties and events that the **Developers Guide** provides.

The **API Reference Guide** has comprehensive reference documentation on every aspect of every control. Expanding the **WinGrid** and its **Namespace**, there are categories of information on the **Classes** that support each control, the **Enumeration** values that it uses, its **Interfaces**, and related **Delegate** classes that define its event handlers. My **HeaderClickAction** and **SortIndicator** are enumerations, so expanding the **Enumerations** node, I see a description of **HeaderClickAction**:



There's information on the **SortSingle** value, which lets the user sort on a single column by clicking on it, and the **SortMulti** value, which lets the user sort on several columns in succession, as well as the **ExternalSortSingle** and **ExternalSortMulti** values that let you capture the sort request yourself, without the grid doing the sorting for you. You take advantage of the ExternalSort... properties by subscribing to the

AfterSortChange event, so I click on the link for that event get some information on it. There I learn that the event takes a subclass of the EventArgs class as input, called **BandEventArgs**.

CE Ope	MetAdvantage® Window	s Forms Online Help :: 2009.2	2 - Microsoft Internet Ex	plorer				
1 =9	🦣 File Edit View Favorites Tools Help 🥂 🥂							
	🕞 Back 🔹 💮 👻 💌	😰 🏠 🔎 Search 🤸	🕇 Favorites 🧐 🔗	• 🦫 • 🔜 🗷 • 🛄 🎎 🦓				
🚏 Re	Address 🙆 http://help.infrag	istics.com/NetAdvantage/WinFor	ms/2009.2/CLR2.0/		- D 🖓 🖓 🗖			
	Google	💌 🔧 Search 🔹 🕫 🧭	• 📲 · 🔯 • 🗆 • 🛠	🖁 Bookmarks 🔹 Kock 🔹 🏭 Translate 🔹 🎦 AutoFill 🔹 💥 🔧	• 🔵 Sign In • 📮			
	Infragistics		NetAdvar	ntage® Windows Forms Online Help ::	2009.2			
				Copy URL to Clipboard Open Topic in	New Window			
Custo Prop	Contents	TheckBoxAlignment	Collapse All Langu fragistics2. Win. UltraWind fterSortChange E ee Also Example	age Filter: All and-9-2 Event				
M	i Header i Header i Invalid∨a i LabelPor i LoadStyl	Nacement alueBehavior sition le	fragistics.Win.UltraWing E Event Data The event handler ri The following Bandle	rid Namespace > UltraGridBase Class : AfterSortChange Event eceives an argument of type <u>BandEventArgs</u> containing data related EventArgs properties provide information specific to this event.	d to this event			
E Pa	MergedC	CellContentArea	Property	Description				
pr D'	Merged	cells valuation i ype	Band	The associated band (read-only).				
Ri Sł Sł Si Si Si St	MultiCell MultiCell MultiCell MultipleB NewBan	Deration DerationErrorhfo.Errork SelectionMode BandSupport idLoadStyle	Example Following code show	s some of the information available in AfterSortChange event.	×			
	Nullable Perform4 PosChar http://help.infragistics.com/	utoSizeType yged	009.2/CLR2.0/html/Infragisti	cs2. Win. Ultra WinGrid. v9.2~Infragistics. Wi	et //			

I can open up layers of information on **BandEventArgs** and its sorted columns that I want to understand how to use, and find that **SortedColumns** is a collection of **UltraGridColumn** objects associated with one of the bands of the grid.

De Op	Anter State Constant State Sta	09.2 - Microsoft Internet Explorer	دلصل					
	File Edit View Favorites Tools Help							
	🚱 Back 🔹 🕥 👻 😰 🚮 🔎 Search	👷 Favorites 🚱 🔗 • 🎭 • 💦 🔟 • 🛄 🎉 🦓						
🐉 Re	Address 🍓 http://help.infragistics.com/NetAdvantage/Wii	nForms/2009.2/CLR2.0/	1-0					
	Google Search • •	🚿 • 🚽 • 🔕 • 🗆 • 🟠 Bookmarks • 🎋 Check • 🏚 Translate • 🍗 AutoFill • » 🛛 🔩 • 🔵 Sign In •	7					
	Infragistics	NetAdvantage® Windows Forms Online Help :: 2009.2	-					
		Copy URL to Clipboard Open Topic in New Window	pls					
Custo Prop M M M M M M M M M M M M M	HeaderCheckBox/sigment HeaderCheckBox/sigment HeaderCheckBox/sigment HeaderCheckBox/sigment HeaderCheckBox/sigment LoadStyle MergedCelEvaluation MergedCelEvaluation MultiCellOpration NewBandLoadStyle NewBandLoa	Integrated 2 Who Ultra Whoold 49 8 2 Sonce 2 Control						
	PerformAutoSizeType PosChanged	Copyright © 1996-2009 Infragistics, Inc. All rights reserved.						
] •	e e	🔰 🔰 Internet 🅢						

In this topic description there is information on the enumeration values and on using the sort events to capture user actions, and also how the **SortIndicator** is set to indicate whether the sort is descending or not.

I then take a look at the details of the **SortIndicator** enumeration, and see its values. If sorting is enabled for a column, I check for the value **Descending**, to see if the user has requested a descending sort.

I've learned a lot about the enumerations and the events that control sorting, so I return to Visual Designer to try out some of these values. The **DisplayLayout** property of the grid is really an object with a number of contained property values. So as the documentation examples showed me, I expand **DisplayLayout** and then **Override** in the **Properties View**, and here I find the **HeaderClickAction** property. If I drop down its list of values – remember that it's an enumeration with a fixed list of possible values – I can set the property to **SortMulti** to let the grid sort the data when the user clicks on one or more columns.

The OpenEdge Editor - Samples/Customer	JltraGrid.cls - OpenEdge Architect - C:\Gui4DotNet	- 0 >
File Edit Design Navigate Search Proje	t OpenEdge Run Window Help	
] 📬 • 🔛 👜 🛛 🔏 💷 🕲] 🏇 •	Q = Q ₄ = A = I ₂ = A = ← ← → =	
😭 💕 OpenEdge Editor		
🍣 Resources 🛛 🗖 🗖	CustomerModel.cls Statemer/UtraGrid.cls (Design) X Customer/UtraGrid.cls	
← → ⊚ 🚍 🔩 ▽		Toolbox P
DealerEmpForm.cls	EustomerUltraGrid	H CustomControlo
DealerEmpModel.cls	0Dp	
DealerEmpXSD.p	Customer grid	Microsoft Controls
- i dsCustomer.i -		OpenEdge Controls
	▶ Text Text _/_/ 🔽 _/_/	I OpenEdge Ultra Controle
📴 Ou 👷 DB 🗆 Pr 💥 🗖 🗆	Text Text _/_/ 🔽 _/_/_	Denicage onta controls
E ≜↓		
mol UtraBridDuctomer : Infracticios Win Litt		
mooraamacustomer - minagistics. wiit oto		
Properties Events		
GroupByRowPadd -1		
GroupByRowSpac -1		
GroupByRowSpac -1		
GroupBySummaryE Default		
GroupBySummary GroupBySummary GroupBySummary Summary GroupBySummary GroupBySumm		
GroupHeaderText0		
HeaderAppearance		
HeaderCheckBox4 Default	Start! Click 'Start!' to begin designing the UltraGrid	
HeaderCheckBoxS Default	Start. Ciek Start & Begin designing the One dand	
HeaderCheckBoxV Default		
HeaderClickAction SortMulti		
HeaderPlacement Default		
HeaderStyle Default		
HotTrackCellAppe		
HotTrackHeaderA	@ mothaLustomer	
HotTrackRowAppe	🔲 Canada 🕅 🔍 Backhara 🗍 🕅 Tanka	
About, UltraGrid Designer, Layout	cterminated > Customeri IltraGrid [OnenEdge Application] C:\Progress\OpenEdge102b\bin/mowin32 ever (Nov 23, 2000 1	1:43:13 AM)
HeaderClickAction	Commission and a service of the serv	
Determines what will occur when the us		7
	Writable	

When I run the form with this property setting, and click on the **CustomerLastName** column, the data is sorted by **CustomerLastName**. I can sort by **CustomerFirstName** within **CustomerLastName** by doing a shift-click on the **CustomerFirstName** column, and see the multiple levels of sorting supported by the grid when **HeaderClickAction** is set to **SortMulti**.

CopenEdge Editor - Samples/CustomerU	JltraGrid.cls - OpenEdge Ar	chitect - C:\Gui4DolA	iet			
Rie Edit Design Navigate Search Project	t OpenEdge Run Window	Help				
😁 - 🗉 🛆 🕹 🕹 - 🙆	🔉 - 💁 - 🛛 🛷 - 🛛 🗄	- 8 - 5 4 -				
🗈 🎳 OpenEdge Editor						
👫 Resources 🕮 🗖 🗖	Model.cls	merModel.cls		ogn) 🕺 💽 Custome	rUltraGrid.cls	-
← → ⊚ ⊟ 😵 ▽						Toolbox
C DealerEmpForm.ds	🛃 CustomerUltraGrid					+ CustomControls
C DealerEmpModel.cls	🛃 CustomerUltraGrid			-	OX 9	Customeoranos
DealerEmpXSD.p						Microsoft Controls
dsCustomer.i		Custo	omer grid			OpenEdge Controls
E Ou 📅 DB 🗖 Pr 🙁 🔍 🗖	CustomerFirstName 4	CustomerLastName 4	CustomerBirthCountry	CustomerBitthdate		OpenEdge Ultra Controls
t: A 103 V	Patrick	Alexander	Germany	02/08/1983		
d: X + 62	Robert	Baker	USA	04/29/1974		
oUltraGridCustomer : Infragistics Win Ultr	Susan	Barnes	Germany	07/17/1972		
Properties Events	Evelyn	Bing	USA	12/10/1965		
GroupByRowPadd -1	Helen	Brooks	Germany	07/31/1974		
GroupByRowSpac -1	Lany	Brooks	USA	02/29/1972	-	
GroupByHowSpac -1	Frank.	Burns	USA	04/10/1984	-	
E GroupBySummary	Mary	Burns	USA	09/02/1956	-	
GroupHeaderTextL	Melissa	Burns	Germany	10/25/1959	+	
HeaderAppearance	Bichard	Burne	Germany	01/05/1968	- 1	
HeaderCheckBox* Delaut	Helen	Carler	USA	05/25/1951	-	
HeaderCheckBox: Default	Heieri	Caller	Uam	03/23/1301	- <u>-</u>	
HeaderClickAction SortMulti				-		
HeaderPlacement Default						
HeaderStyle Default						
El HotTrackLeelAppe	Customer 3					
HotTrackRowApp	-	1 - 1				
About, Ultrafirid Designer, Lavout	📮 Console 🕄 🚺 Problem	is 🕗 Tasks		📕 X	🎬 🗟 🚮	P 🖉 🖻 🔍 💙 🖓
	CustomerUltraGrid [OpenEdge	Application] C:\Progres	s\OpenEdge102b\bin\prow	in32.exe (Nov 23, 2009	9 12:26:19 PM)	
HeaderClickAction						
Determines which was deter which the tot.	1					2
P° 🖪			Writable			

This example shows what seems like a useful feature built into the grid, without my having to do any coding at all. But now that I've seen what the grid can do on its own, I need to consider the fact that letting the user interface take over sorting of my data – in addition to filtering the data and other jobs that are really part of data management – is not necessarily a great idea. I'll next show you how to keep the data

management where it belongs and still let users use the grid control to see the data the way they want to see it.

[This next part of the document corresponds to the video presentation that is Part 3 of the Data Sorting series:]

Next this document shows how to intercept the event that fires when the user requests a sort, so that you can code the event handler yourself to apply that sort request to the query that you're managing rather than letting the grid do its own sorting.

The previous section of this document showed how to enable grid column sorting by setting the **HeaderClickAction** property. To illustrate what I meant by suggesting that letting the grid do the sorting is not necessarily a good idea, I'll add a button to the form and code a simple event handler for it. It's a Next button, to advance the query on the Customer data to the next row. Double-clicking on the control in Visual Designer automatically creates a subscription and a skeleton event handler for the control's default event, in the case of a Button, the **Click** event. I want to increment the **Position** property of the binding source when the button is clicked.



Let's see what happens when I run the form containing my button. As before, I can take advantage of the grid's ability to sort the data itself by clicking on a column header such as CustomerLastName. But then I select the first row that's displayed in the grid after the sorting. Note the row selection marker at the beginning of the first row in the grid:

File Edit Design Navigate C + C A A A A A A A A A A A A A A A A A	e Search Proje J 🌚 J 🏇 • □	a a	penEdge Run Windo Q. •	w Help	• • •				
Contraction of the second seco	• # @ • = • = • =	0.	Model.cls	a - A - O - O -	• - • •				
CopenEdge Editor	• • • • •	0	Model.cls						
Resources 20	•• • • • •		Model.cls						
	🖻 🗞 🎽			tomerModel.cls		syn) 😫 💽 Customer	UltraGrid.cls		00
		1 N						Toolbox	4
dsCustomer.i	-		🛃 CustomerUltraGri	d			<u>-0×</u>	CustomControls	-
- dsDealerEmp.i		<u>8</u> 0	ustomerUltraGrid			-		Microsoft Controls	
Manufacture International Contraction				Durt	omar mid		- Ľ	A DeerEdee Controls	
			CustomerFirstName	Customed astName /	CustomeRithCountry	CustomerBithdate		OpenEdge Ulit	
25 OU 97 DB DP			Patrick.	Alexander	Germany	02/08/1983	-	G opencage ok	-1
	🗄 🎶 🖪 🍸		Robert	Baker	USA	04/23/1974		Pointer	
ultraButton1 : Infragistics.Win	Misc.UltraBu 💌		Sutan	Barnes	Germany	07/17/1972	-	HAmmaboriControl	
Properties Events			Evelyn	Bing	USA	12/10/1965		B-InboxControlStyler	
E Padding	0.0		Lany	Brooks	USA	02/29/1972	-	😨 UltraButton	
PressedAppearance			Helen	Brooks	Germany	07/31/1974	-	UltraCalcManager	
RightToLeft	No (none)		Mary	Burns	USA	09/02/1956	-	UltraCalculator	
ShowFocusRect	True		Richard	Burns	Germany	01/05/1968	-	UltraCalendarCombo	
ShowOutline	True		Melissa	Burns	Germany	10/25/1959		TUltraCalendarin/o	
E Size Stulei Aran Marre	75, 23		Frank	Burns	USA	04/10/1984		UltraCalendarLook	
StyleSetName	- 1		Helen	Cater	USA	05/25/1951		Charles Charles	
TabIndex	1	1	1					UbaColorPicker	- 11
TabStop	True	-	Net					odmoCartilU	- 11
Test	Next +1		riest					UltraComboEditor	
About. Custom Property Per	201	Γ	🐺 moBSCustomer					UltraCurrencyEditor	
			Tanania 😒 💽 Bushia	ms 🔊 Tasks			88 I 🗈 🗔	A A	
Test		Curt	omeri Bradicid (OnenEde	e Application C-IProgres	så OnenEdae 102hibiolorov	in 32. eve (Nov 23. 2001	1-05-11 PM)		-
The text associated with the	control.		concernants [opened]	a shine on the shirt of the	a construction of the second	a 102.1210 (201 203 200)	1.00.11110		10
		8							
T* 12					Writable				

Now I click the Next button.

🗜 OpenEdge Editor - Sa	mples/Custome	rUltra	Grid.cls - OpenEdge A	rchitect - C:\Gui4Dot!	Net			_0
File Edit Design Navig	ate Search Proj	ect C	OpenEdge Run Windo	// Help				
🗈 - 🔛 💩 🛛 🖌	📖 🗞 🏇 •	0	• 💁 • 🛛 🛷 • 🛛 🔮	a - A - 🍫 🔶 -	• ÷ •			
😭 💕 OpenEdge Editor								
🖑 Resources 🛛			IModel.cls	comerModel.cls 🛛 🆏		sign) 🗙 💽 Custome	rUltraGrid.cls	-
	ि 📄 😫 🏱							Toolbox
🗌 🚺 dsCustomer.i	-		📑 CustomerUltraGri	d				+ CustomControls
🗌 🗋 dsDealerEmp.i			CustomerUltraGrid			_		Ninese Controls
IModel.cls		1 -					P	Microsoft Controls
- SavPanel.ds	-			Cust	omer grid			OpenEdge Controls
🗟 Ou 💀 DB 🔲	Pr 23		CustomerFirstName	CustomerLastName /	CustomerBirthCountry	CustomerBirthdate		OpenEdge Ult
			Jack	Kelly	Germany	04/25/1978		Pointer
	ta: 2↓ tos *		Robert	Kennedy	USA	07/13/1979		AnimationControl
ultraButton1 : Infragistics.V	Vin. Misc. UltraBu 💌		Arthur	Lee	Germany	03/09/1974		AnnStylistBuntime
Properties Events]		Chris	Lee	Germany	06/10/1956		InboxControlStyler
Padding	0,0 🔺		Sarah	Lee	USA	06/21/1976		UltraButton
PressedAppearance			Patrick	Lewis	Germany	03/19/1970		IltraCalcManager
RightToLeft	No		Steve	Marshall	Germanu	03/10/1951	+	UltraCalculator
Shapelmage ShawEcourePoot	[[none]		Look	Marahall	LICA	05/16/1959	-	UltraCalculatorDropDown
ShowDutline	True		Jack	maisiidii	038	03/10/1300	+	UltraCalendarCombo
E Size	75.23		Sarah	Martin	USA	01703/1968	- II I	I UltraCalendari ook
StyleLibraryName			Robert	Martinez	USA	03/26/1975		IllraChart
StyleSetName			Larry	Martinez	Germany	06/22/1965	_	UltraCheckEditor
TabIndex	1	Ιī	4	1	•	'		UltraColorPicker
TabStop	True		[-		UltraCombo
Tag	Mant		Next					UltraComboEditor
Text	NEXI -	lr-						UltraCurrencyEditor
About, Custom Property P	Pages		🚏 moBSCustomer					1104-D-1-C
			Console 💥 💽 Deable	ros 🖉 Tasks		— 92	Xa 🗈 🗔	
Text		a	toward BunCrid FOrenEda	a Annication C. Dramor	of OpenEdge 102h his beau	ie 22 eue /Mau 22, 2001		
The text associated with	the control.	Cus	comercitragnia (OpenEdg	e Application J Criprogres	syopenbage102b(bin)prov	miaziexe (Nov 23, 200	9 1:05(11 MM)	
								Ľ
1° E					Writable			

Well, what happened? The grid's position marker jumped to a completely different row. If I click Next again, the same thing happens: the grid jumps to another row that is nowhere near the next row as displayed in the grid. What the Next button is doing has no relation to the data as the grid is displaying it, which is certainly very confusing. Let's take a look at what's happening here.

In this diagram I illustrate that in the Model, there's a database query to retrieve requested data into local storage in a DataSet, and then there's a query on that data for the Customer temp-table. The ProBindingSource is connected to that temp-table query handle, and the ProBindingSource in turn becomes the **DataSource** for the grid, so everything is connected together.

ſ	– CustomerUl	traGrid "View'	·		I	
	FirstName	LastName				
	Robert	Kennedy				
	Peter	Wagner				
	Alice	Washington	R			
	Patrick	Lewis		7		
	C	ustomer Bindin	g Sour	ce		
Ì						
ſ	– CustomerMa	odel		httCu	istQuery	
	/* DataSet	definition */	1	⊺emp-tab	le query handle	
1	METHOD Fe	:tchData():			T	l
	/* Locate	e data source	*/			
	/* Make	request of da	ita sou	rce */		
	/* Popula	ite local DataS	iet */	~		
l	Physic	al Data Sou	irce [
	$ \$	Query qCust	omer		H-DATA-SOURC	E

But if I let the grid sort its displayed data independently, then the connection with the data as held in the Model is effectively broken. The Model doesn't know anything about the sort sequence that the user is seeing, so when the Next button advances the **Position** of the binding source, which does a Get-Next on the temp-table query, this is totally out of sync with what the user interface is displaying to the user. Allowing this to happen is obviously not a good thing.



So as nice as this sorting in the grid seems, it's really not a good idea for an application that is in the serious data management business. It's better to think of this grid capability more as demo-ware.

Instead, it's time to write the code to add to the Model to let it control the sorting. Here's some simple code to manage sorting directly on the temp-table that holds the data I've retrieved.

```
METHOD PUBLIC VOID SortData( INPUT pcSort AS CHARACTER ):
DEFINE VARIABLE iSortField AS INTEGER NO-UNDO.
DEFINE VARIABLE cSortString AS CHARACTER NO-UNDO INIT "".
DO iSortField = 1 TO NUM-ENTRIES (pcSort) BY 2:
    cSortString = cSortString + " BY " + ENTRY (iSortField,pcSort) + " " +
    ENTRY (iSortField + 1, pcSort). /* Optional descending qualifier */
    END.
    httCustQuery:QUERY-CLOSE () NO-ERROR .
    httCustQuery:QUERY-PREPARE ("FOR EACH ttCustomer " + cSortString).
    MESSAGE cSortString VIEW-AS ALERT-BOX.
    httCustQuery:QUERY-OPEN ().
END METHOD.
```

Basically this bit of code is doing two things. It's taking a character string passed in as a parameter, which just alternates sort fields and an optional "DESCENDING" qualifier, and turns that into a BY clause for an ABL query. And then it re-prepares the temp-table query on the local data held by the model with that BY clause and re-opens it. This makes the new sort sequence available to the binding source in the form, and through the binding source to the grid. To help you see what's happening, I put in a message statement as well that displays the resulting sort clause in the query.

Now let's look at the other side of this call, which is an event handler in the form class that captures the sort request and constructs the list of columns to sort, passing this as a character string to the **SortData** method. I select the **Events** tab in the **Properties View** to get a list of all the events that the grid

supports. Remember that in the Infragistics documentation I learned that there's an **AfterSortChange** event that fires whenever the user requests a sort by clicking on a column header.

OpenEdge Editor - Samples/Custor	nertiltraf	Frid.cls - OpenEd	ae Architect - C:\Gui	4DotNet				- (DÍ X
File Edit Design Navigate Search F	roject Or	senEdge Run W	Indow Help					
] 🗂 • 🛛 🛆 🗠 🏷 🎄	- 0 -	Q. •] 🛷 •] (z) + (z) + (5	¢•÷ -				
📑 💐 OpenEdge Editor								
💞 Resources 😒 💭	🗆 💽 I	Model.cls	CustomerModel.cls	CustomerURraGri	d.cls (Design) 🔅	CustomerUltraGrid.cls		
e - + 10 📄 😵				-			Toolbox	4
dsCustomer.i	A	🛃 CustomerUltr	aGrid			_ O ×	Ti Custon Controls	-
dsDealerEmp.i		o		0			U CustomControis	
- C IModel.cls	-	L		Customer grid			Hicrosoft Controls	
NavPanel.cls	-						OpenEdge Controls	
95 ou 199 may 12 9		 Test 	Test	Text			OpenEdge Ult	
	-	Text	Test	Text	_/_/		Pointer	
tt: 2 + 155	_						Animation Control	
moUltraGridCustomer : Infragistics.Win.Ultr	-						ET AppStvistRuntime	
Properties Events							Sh InboxControlStyler	
AlterBowlindate	-1	9				9	C UltraButton	
AlterSelectChanc	-						UltraCalcManager	
AlterSortChange							UltraCalculator	
AlterSummayDia	-						UltraCalculatorDropDown	
BackColorChang							UltraCalendarCombo	
Backgroundimag							TUltraCalendarinio	
Backgroundimag		1				<u>}</u>	TUltraCalendarLook	
BeforeAutoSizeE		Starti	Click 'Starti' to b	eain desianina the l			UltraChart	
BeforeBandHidde							UltraCheckEditor	
BeforeCardComp		Neut	1	Ŭ,		Ÿ	UltraColorPicker	
BeforeCellActivat	-	14694	J				UltraCombo	
About, UltraGrid Designer, Lavout							UltraComboEditor	
Wizard, Reset Lavout							UltraCurrencyEditor	
		22 maRSC stores						-
		- moescustomer					•	2
	- 0	Console 23 💽 P	roblems 🕢 Tasks			👘 M 🖓 🔂	1 🖉 🖉 💌 🖻 • 📑 •	
Click	charr	vinated > Curtome	d BraGrid I OnenEdge Ar	olication C. Dromerski	menEdan (02b)b	nlocowie 32 eve (Nov 23, 200	0.1-05-11.0M3	
Occurs when the component is clicked.	s.ceri	in accord Copcome	owners fober code w	Surveyord 215 http://doi.org/		1991 STILL 1992 STILL 1997 STILL 199	Y HAVE THEY	100
	and the							1
L								2
C° 🗈								

If I just double-click that event name in the Events tab, I get the start of an event handler and a subscription. Here's the code I write to create the sort string to pass to the Model. The first part checks to see if the **HeaderClickAction** is set to either **SortSingle** or **SortMulti**.

```
@VisualDesigner.
METHOD PRIVATE VOID moUltraGridCustomer_AfterSortChange
  ( INPUT sender AS System.Object,
   INPUT e AS Infragistics.Win.UltraWinGrid.BandEventArgs ):
   DEFINE VARIABLE oSortColumn AS UltraGridColumn NO-UNDO.
                                            NO-UNDO
   DEFINE VARIABLE iColumn AS INTEGER
   DEFINE VARIABLE cSortString AS CHARACTER
                                                 NO-UNDO INIT "".
                              AS UltraGridBand NO-UNDO.
   DEFINE VARIABLE oBand
   IF Progress.Util.EnumHelper:AreEqual
         (moUltraGridCustomer:DisplayLayout:Override:HeaderClickAction,
          HeaderClickAction:SortMulti)
   OR Progress.Util.EnumHelper:AreEqual
        (moUltraGridCustomer:DisplayLayout:Override:HeaderClickAction,
          HeaderClickAction:SortSingle)
   THEN RETURN.
```

The control represents the **HeaderClickAction** value as an enumeration, a set of fixed coded values. ABL doesn't support enumerations directly, but it does provide an **EnumHelper** class with methods like **AreEqual** to let you *work* with enumeration values. Remember too that **SortSingle** and **SortMulti** are the values that tell the grid to do its own sorting, so if that's what it's set to, the event handler just returns and lets the grid do its thing.

But otherwise the method uses some of what we saw in the Infragistics documentation about the **Band** object and its **SortedColumns** property, along with the **SortIndicator** enumeration, to extract the user's request from the event arguments object that is passed in, and turn that into a generic character string that the Model can deal with:



The essential element of this simple code is to make sure that the Model doesn't have to understand anything about the event that initiated the request or the structure of the control elements that provide the information about what columns were clicked. That's the form's job, to deal with the specifics of the controls in the user interface. At the same time, I don't want the form to have to understand how to put together an ABL query to re-sort data; that's the Model's job. So I just put together a list of sort columns and whether they're descending or not, and then run SortData to tell the Model to deal with it. This separation of responsibilities is illustrated by the following diagram:



One more thing I have to do up at the top of the form class is to add a **USING** statement for the **UltraWinGrid** so that the compiler will recognize my references to the grid components **UltraGridColumn** and **UltraGridBand** in the **AfterSortRequest** event handler.



Now I save what I've done, and go back to the design view, where I need to reset the **HeaderClickAction** property to one of the values that tells the grid *not* to do its own sorting. Once again, under **DisplayLayout** and **Override** in the **Properties** View, I find the **HeaderClickAction** property. And as I learned from the online documentation, the values **ExternalSortSingle** and **ExternalSortMulti** tell the grid just to invoke the **AfterSortChange** event and pass the selected columns into it without doing any sorting on its own. So I select **ExternalSortMulti** for multi-column sorting.

OpenEdge Editor - Samples/Custome Fle Edit Design Navigate Search Proj	RtraGrid.cls - OpenEdge Architect - C:\Gui4DotNet t OpenEdge Run, Window Heb	
🔁 • 🔛 🛆 🔏 🗆 🗞 • 🗂	Q • Q •] Ø •] ≥ • 8 • ⊕ ⇔ • ⊕ •	
📑 💐 OpenEdge Editor		
💞 Resources 💠 🗖 🗆	🗈 IModel.cls 🔹 CustomerModel.cls 🚳 *CustomerUtraGrid.cls (Costyn) 🗶 🗟 *CustomerUtraGrid.cls	- 0
e -> 1a 😑 🕸 🏹		Toolbox 7
dsCustomer.i	EustomerUltraGrid	CustomControls
- i dsDealerEmp.i	O Dustrees and	+ Microsoft Controls
C IModel.cls	Contrast Sec	al Once Edge Controls
Navyanei.os	Test Test Test // □ //	UpenLage Controls
🗄 Ou 📅 DB 🗆 Pr 🕴 🖓 🗖		 Upent dge Uit
े ह्या 10 स		Pointer
med DraGodCoutemar : Infrankting high 1Dr		AnimationControl
moultabalocustomer : Innagistics wirt Uits		8 AppStylistRuntime
Properties Events		InboxControlStyler
HeaderCheckBox! Default	Y	🕄 UltraButton
HeaderCheckBox\ Default		UltraCalcManager
HeaderClickAction ExternalSo		UltraCalculator
HeaderPlacement Default		UltraCalculatorDropDown
HeaderStyle Default		UltraCalendarCombo
⊞ HotTrackCellAppe		UltraCalendarinto
HolTrackHeaderA		IlitraCalendari ook
HolTrackRowApp	Charti Click Stratil to basic designing the UltraCold	T UbaChad
HotTrackBowCell4	cick start to begin designing the circatant	The Ubachards E direct
HotTrackBowSele	0	Ultra Cala Dialas
InvalidValueBehav Delault	Next	Unacoonnoker
About UltraGid Darisser Laurut		UltraCombo
Wigard Reset Lavout		UltraComboEditor
		UltraCurrencyEditor
	S moBSCustomer	- · · · · · · · · · · · · · · · · · · ·
Hander ClickAster	🛄 Console 🕮 🚺 Problems 🛃 Tasks 👘 👘 📊	rite i 🖬 🖬 🖓 🖓 🖓 🖓
Determines what will occur when the	<terminated> CustomerUltraGrid [OpenEdge Application] C:\Progress\OpenEdge102b\binlprowin32.exe (Nov 23, 2009 1</terminated>	05:11 PM)
user clicks on a header.		100
	x	2
-		
U	Wrkable	

I save and run the form with this new value. When I click on the **CustomerLastName** column, the MESSAGE statement in the Model's **SortData** method appears, to confirm that it was invoked to resort the data, rather than the grid doing the sorting. If I then Shift-Click on the **CustomerFirstName**, that column gets added to the sort. And if I click on the direction arrow in the grid column header, that action gets passed in the **SortIndicator** property as a **Descending** qualifier. The MESSAGE statement alert box now reflects all three of these combined clicks on the **UltraGrid** column headers:

🛱 OpenEdge Editor - Sa	mples/Customer	Ultra	Grid.cls - OpenEdge Ar	chitect - C:\Gui4Dot!	Net				_ (0(×
File Edit Design Navig	ste Search Proje	et C	penEdge Run Window	r Help					and its
	u % Kr -	0	• Q. • 1 - 1 - 1 - 1 - 1 - 1 - 1 - 1 - 1 - 1	- 3 - 6 0 -					
The second day Editor		×.	· · · · · · · · ·						
Copercode corce								-	0.8
S Resources 33		6	Model.ds Cush	omerModel.cls	CustomerUlbraGrid.ds (De	ayn) 🕺 💽 Custome	rUltraGrid.cls		
	🔬 🚍 😫 🎽	ι.,						Toolbox	9
dsCustomer.i			🛃 CustomerUltraGrid					CustomControls	-
- i dsDealerEmp.i			CustomerUltraGrid			-		Hiccoroft Controls	
Model.cls				6	and a later of the		P	Dave Edge Controls	_
Naviranei.cs	-		Contract Conditions of T	Custo	omer grid	Common California		UpenEdge Controls	_
😤 Ou 📅 DB 🗔	Pr 🐹 🗖 🗆		Customer intName V	CustomerLastName /	CustomerBirthCountry	LustometEinthdate	-	Upent dge Ult	
	田刻眼室		Fault	Alexander	uemany	02/06/1363		Pointer	
mol IbraGridContomer : Info	witing W/n Life V						-	AnimationControl	-
Incondunación de la ma	spones. Witcolo.						_	AppStylistBunkime	
Properties Events							0	InboxControlStyler	
AccessibleRole	Default 🔺							UltraButton	
AllowOrop	False		Message (Press HELI	⁹ to view stack trace) 🗙			UltraCalcManager	
AlphablendMode	Optimized		RV Contract within	- PV Color Post	ARCONOMIC .		+	UltraCalculator	
Anchor	Top, Left		BT Cuscomercasevan	e er customerrirsova	ne DESCENDUNA		-	UltraCalculatorDropDown	
Causes valoadon	(none)			W Halo	1		-	UltraCalendarCombo	
Curror	Default			mep			1	UltraCalendarinto	
DataMember								UltraLalendarLook	
DataSource	moBSCuste							M Ubachad	
DisplayLayout		1				1		UtraColorRick or	
Dock	None 💌	ш				-		- UltraCombo	
About, UltraGrid Designer	Lavout		Next					UltraComboEditor	
Wizard Reset Lavout		-						E Ultra CunerocuE dtor	
			221 and BCC, selectors						*
			- mob 3 Lostomer					4	<u> </u>
			Console 33 👔 Problem	ns 🧟 Tasks		😸 😹	🦮 🕞 🗸	🖉 🖉 😁 🕒 😁	00
DisplayLayout		Cust	tomerUltraGrid [OpenEdge	Application1 C:\Progres	s\OpenEdge102b\bin\prov	in32.exe (Nov 23, 2005	9 1:47:10 PMD		
The main layout of the gri	1	1		,					191
		X							
							1		
] U. G					Writable		1		

And you can see that re-opening the temp-table query down in the Model makes the data available to the grid through the binding source, with the new sort sequence. What we see initially looks the same as when the grid did its own sorting on its local copy of the data.

Constant data and the second		Liburi	C.:		d – la			
DE UpenEage Ealtor - S	amples/Customer	ultrat ct. 0	Grid.cis - OpenEage Ar	renitect - L:\Gui4Dotr	vet			. الالم
	100 90a #** -	Δ,	• 0 • 1 1 - 3	- N - *5 /5 -				
		·	. 🛥 . 1 🗸 . 1 %					
E S OpenEdge Editor								
💱 Resources 🛛		C :	IModel.cls 🛛 💽 Cust	omerModel.cls 🏻 🖏		sign) 🗙 💽 Custome	rUltraGrid.cls	
	💿 📄 🔄 🎽							Toolbox 7
dsCustomer.i	i 🔺		📴 CustomerUltraGric	1			- <u> </u>	+ CustomControls
- 🚺 dsDealerEmp	J.	🔡 C	EustomerUltraGrid			_		
IModel.cls	_							Microsoft Controls
NavPanel.cls	<u> </u>			Cust	omer grid			OpenEdge Controls
🗄 Ou 📅 DB 🔲	Pr 🛛 🗖 🗖		CustomerFirstName V	CustomerLastName 4	CustomerBirthCountry	CustomerBirthdate		OpenEdge Ult
		•	Patrick	Alexander	Germany	02/08/1983		Pointer
	C: Z + EC)		Robert	Baker	USA	04/29/1974		AnimationControl
moUltraGridCustomer : Inf	ragistics.Win.Ultr 🚬		Susan	Barnes	Germany	07/17/1972		AppStylistRuntime
Properties Events	1		Evelyn	Bing	USA	12/10/1965		InboxControlStyler
AccessibleRole	Default 🔺		Larry	Brooks	USA	02/29/1972		🖳 UltraButton
AllowDrop	False		Helen	Brooks	Germany	07/31/1974		D UltraCalcManager
AlphaBlendMode	Optimized Ten Left		Richard	Burns	Germany	01/05/1968	-	UltraCalculator
CausesValidation	True		Melissa	Burns	Germanu	10/25/1959	-	UltraCalendarCombo
ContextMenuStrip	(none)		Manu	Burne	LISA	09/02/1956	-	
Cursor	Default		Frank	Dums	UCA	03/02/1300	- 11	TultraCalendarLook
DataMember			Frank	burns	USA	04/10/1364	_	🕅 UltraChart
DataSource	moBSCust		Helen	Carter	USA	05/25/1951	- 0	🖾 UltraCheckEditor
DisplayLayour	None	l II:	•					UltraColorPicker
About UltraGrid Design			Next					UltraCombo
Wizard, Reset Layout.		_						UltraComboE ditor
								UltracurrencyEditor
			mobSCustomer					
			Console 🔀 🚼 Probler	ms 🧔 Tasks		= ×	🔌 📑 🗛	1 🖉 🖉 🛃 • 📬 • 🗖 🗖
DisplayLayout		Cust	omerUltraGrid [OpenEdge	Application] C:\Progres	s\OpenEdge102b\bin\prov	vin32.exe (Nov 23, 2009	9 1:47:10 PM)	
i ne main layout of the g	na.					. ,		
		4						Þ
					Watable		1	
					whicable		1	

But what's significant is that the sort you're seeing is now the same sort order managed by the query in the Model, because the grid is actually using the result of re-preparing and re-opening the query with the BY clause generated by **SortData**. So If I click on the Next button, the position indicator advances in the grid the way you would expect.

DenEdge Editor - Sar	mples/Custome	rUltr	aGrid.cls - OpenEd	ge Architect - C:\Gui4Dot	Net				_ 🗆 🗵
File Edit Design Naviga	ite Search Proje	sct	OpenEdge Run Vi	indow Help					
🗈 • 🔛 🚔 🔏 (山 🕲] 🏇 •	0	• 🏊 • 🛛 🔗 •] 🖢 = 🖓 = 🍫 🔶 •	• 🔿 •				
😭 🗳 OpenEdge Editor									
🗳 Resources 🔀	rUltraGrid.cls		- 0						
$\leftarrow \Rightarrow$	ā 🖻 😫 🏱	IC						Toolbox	7
🚺 dsCustomer.i			🖶 CustomerUltr	aGrid				+ CustomControls	
- 🗋 dsDealerEmp.i			CustomerUltraGri	id		_			
IModel.cls								Microsoft Controls	_
C NavPanel.cls	-			Cust	omer grid			🗄 OpenEdge Controls	
🖳 ou 🔤 de 🥅 d			CustomerFirstName	e 💎 CustomerLastName 4	CustomerBirthCountry	CustomerBirthdate		🖃 OpenEdge Ult	
		1 [Patrick	Alexander	Germany	02/08/1983		Pointer	
	ta: 2 ↓ 83 1	r [Robert 	Baker	USA	04/29/1974		AnimationControl	
moUltraGridCustomer : Infra	gistics.Win.Ultr 💌		Susan	Barnes	Germany	07/17/1972		AppStylistRuntime	
Properties Events		ΙΓ	Evelyn	Bing	USA	12/10/1965		InboxControlStyler	
AccessibleRole	Default 🔺		Larry	Brooks	USA	02/29/1972		UltraButton	
AllowDrop	False		Helen	Brooks	Germany	07/31/1974		∑ UltraCalcManager	
Anchor	Top. Left		Richard	Burns	Germany	01/05/1968		UltraCalculatorDropDow	m
CausesValidation	True		Melissa	Burns	Germany	10/25/1959		UltraCalendarCombo	
ContextMenuStrip	(none)		Mary	Burns	USA	09/02/1956		📅 UltraCalendarInfo	
DataMember	Default		Frank	Burns	USA	04/10/1984		UltraCalendarLook	
DataSource	moBSCuste		Helen	Carter	USA	05/25/1951		UltraChaokEditor	
DisplayLayout				1	1	1		UltraColorPicker	
Dock	None 🗾	۳ ا	[-		UltraCombo	
About, UltraGrid Designer,	Lavout		Next					📑 UltraComboEditor	
Wiedid, Hester Edybur								UltraCurrencyEditor	-
			🚏 moBSCustomer					I IIID.a.c.	
			Console 🔅 💌 P	roblems 🧖 Tasks		a ×	× 🗈 🗖		
DisplayLayout			choment IbraGrid IOne	pEdge Application] C/Progres	ci OpenEdge102bibioiorov	win32 ave (Nov 23, 2004			
The main layout of the grid	i.		somereicreane (ope	neago approacion y califitação	sysperiode to color in the ow	millerene (1909-20) 2005			-
		4							× ×
] 1° E			-		Writable]		

What the user is seeing is in sync with how the Model is actually managing the data, and this is what you want to do in your application to avoid confusion between the user interface and your data management logic. In addition, the external sort option is much faster than having the grid do the sorting, because it can take advantage of OpenEdge database and temp-table indexing, so that's another reason to keep the sorting in the Model where it belongs.

[The next part of the document corresponds to the video presentation that is Part 4 of the Data Sorting series:]

The previous sections of this document show how the binding source interacts with the Infragistics **UltraGrid** control. The Microsoft **DataGridView** control works quite differently, and uses a binding source property and an event that the UltraGrid doesn't. This part of the document shows you the differences.

To illustrate these differences, I create a new form that uses the same Model class for data retrieval that the UltraGrid form uses. I name the form class **CustomerDataGridView**.cls. Remember that one way to create a ProBindingSource is just to drag a source file with a ProDataSet or temp-table definition anywhere onto a form. I do this here by dragging the same definition file, **dsCustomer.i**, that I used in the other example, from the **Resources** View onto the new form. The **ProBindingSource Designer** appears to allow me to edit the definition if I need to:

Dpentidge Editor - Samples/Custor	nerDataGridView.cls - Op voiet: OpanEdaa Dup 3	entidge Architect - C:\Gui4DotNet		<u>_0×</u>
	• • • • •	- 5 - 5 - 6		
Contraction Editor				
Resources 23	CustomerDataGri	dView.ds (Design) 💠		
+ + * = *	2			Toolbox 7
C DealerEmpForm.ds	ProBindingSource D	lesigner		+ CustonControls
C DealerEmpModel.ds	1 1 X † 4	16 色		Microsoft Controls
DealerEmpXSD.p deCostomer.i	Tables	Fields	Table : ttCustomer	+ DeerFdee Controls
dsDealerEmp.i	- ItCustomer	CustomerFirstName	21 21 🔤	I DoenEdoe Ultra Controls
C IModel.cls		CustomerLastName	🗆 Design	G openeoge one control
🗄 Ou 📅 DB 🖾 Pr 🖄 🤗	1	Customer8inhCountry	Name ttCustomer	
E 🔅 🕫		CustomerBirthdate		
Property Value		CustomerGender		
		Cutomer, center are		
00.000 000		CustomerCover.phys.ae		
last modif November 19, 2009 12		g continue		
Inked False				
name dsCustomer.i			Harro	
path /Samples/dsCustomer.i			The name of the table	
size 963 bytes				
	1			
			OK Cancel	
	Console 22	Problems 20 Tasks		
	commind(ed> Deceluin	onew (operacoge application) Ct(Progress(Opera-	age1020(pergerowero2.exe (Dec 7, 200912	30.34 PM
	R			2 2
Samples/dsCustomer.i				
2			,	

The new ProBindingSource control gets placed in the non-visual tray below the form itself. I rename the control as I've done elsewhere to **moBSCustomer**. Next I select the Microsoft DataGridView from the **Microsoft Controls** group in the Toolbox and drop it onto the form. The Customer ProBindingSource is available to be chosen as the **DataSource** for the grid.

🗑 OpenEdge Editor - Samples/Customer	DataGridView.cls - OpenEdge Architect - C:\Gui4DotNet		
File Edit Design Navigate Search Proje	ct OpenEdge Run Window Help		
📑 • 🗟 🛆 🛛 🔏 💷 🗞 •	Q + Q + A + B + B + B + A + + +		
📑 🎳 OpenEdge Editor			
💐 Resources 🖸 🗖 🗖	S *CustomerDataGnifiteer.ds (Design) 🗙		
		Toolbox	9
DealerEmpForm.cls	EustomerDataGridView	The Contract Contracts	
DealerEmpModel.cls		Customeoratos	
DealerEmpl/SD.p	DatalandView Tasks	Microsoft Controls	_
dsCustomer.i	Choose Data Source (none)	Pointer	
dsDealerEmp.i	R None	Button	
Model.ds	- W moliSCustomer	CheckBox	
😤 Ou 📅 DB 🗔 Pr 🔀 🗖 🗖	A4	CheckedListBox	
an Al 100 V		ColorDialog	
E Z + BZ -		ComboBox	
dataGridView1 : System.Windows.Forms.D 💌	M	H ContextMenuStrip	
Properties Events		i→DataGridView	
The second second		DateTimePicker	
(Name) dataEriction1		1 DirectoryEntry	
AccessibleDesc	D	ChiectorySearcher	
AccessibleName		ErrorProvider	
AccessibleRole Default		EventLog	
AllowDrop False	Selecting a BindingSource binds directly to that	FileSystemWatcher	
AllowUserToAdk True	bindingbource.	FlowLayoutPanel	
AllowUserToDel True		FolderBrowserDialog	
AllowUserToOrc False		4 FontDialog	
AllowUserToRe: True	W moBSCustomer	GroupBox	
AllowUserToRe: True		[F1] HelpProvider	
Edit Column Add Column		ImageList	. I
KARL SCHREITER, COM SCHREITER,		4	
	🖾 Console 🖄 💽 Problems 🖉 Tasks	a 🖾 📧 💀 🖬 🗸 😁	
(DataBindings)	staminated a DataGriffiau (OperEdea Anderation) CillBoosers(OperEdea102b)bio(resuin22 ava (Der 7, 2000 12	20/E4 DM)	
The data bindings for the control.	commenced and a second s		10
	21		2
] 📭 🖪	Writable		

Now I need to go into the code for the form. I want to reuse the same **CustomerModel** class that I created for use with the other form class that displayed data in an UltraGrid. Even though the user interface controls work very differently, I can use the same data management class as before, because I separated the data management code cleanly from the form class that knows the details of how the UI works.

So in the same way as in CustomerUltraGrid.cls, I create an instance of the data management Model class using the **NEW** keyword, tell it to fill its DataSet with all the Customers, and retrieve the temp-table handle to use as the **Handle** property of the ProBindingSource.



Running the form shows us how this much of the job that we have to do works at this point. This screen capture shows that I've successfully gotten data into the grid, but there's no sorting when I click on a column header:

🕅 OpenEdge Edi	tor - Samples/Customer	DataG	GridViev	v.cls - OpenEdge A	rchitect - C:\Gui40	otNet				I
File Edit Source	Navigate Search Proje	ct Og	penEdge	Run Window H	elp					
] 📬 • 🔛 👜	14001**	0.	• 🏊 •	🛷 • 🖗 •	÷ 🗢 🗢 - 💡					
📑 🐉 OpenEdge	e Editor									
St Resources	- 0	\$ 30	Custome	rDataGridView.cls (De	isign) 💽 Custe		8		6	
	> 🔬 😑 😫 🏹	2	22	DEFINE PRIV	ATE VARIABLE	customerLind	eseExpiryDe	ateDataGri	dViewTextBoxColumn 🔉 Syste	<u>.</u>
C Deale	rEmpForm.cls	1	23	DEFINE PRIV	ATE VARIABLE	customerLice	nseDateDate	GridViewT	extBoxColumn AS System.Wind	-
C Deale	rEmpModel.cls	1	24	DEFINE PRIV	ATE VARIABLE	customerLast	NameDataGri	idViewText	BoxColumn &S System.Windows	
- Deale	rEmpXSD.p	- 4	25	DEFINE PRIV	ATE VARIABLE	customerIDDe	taGridView7	TextBoxCol	umn AS System. Windows. Forme	
dsCus	tomer.i		Custon	nerDataGrid¥iew				_ [] ×	BoxColumn 15 System, Windows.	:
di di Dice	seremp.i								BoxColumn 13 System, Window	21
S 14000	n.09			CustomerFirstName	CustomerLastName	CustomerBirthCoun	CustomerBirthdat	e Custon *	extBoxColumn 18 System.Wir	
2 OU 1 DB	i Pr 🛛 🗖 🗖		۰.	Robert	Kennedy[CustomerL	astName	7/13/1979			
	🗄 🏇 🖄 🎽			Peter	Wagner	USA	9/14/1987		ŀ	
Property	Value			Alice	Washington	Germany	5/19/1982			
E Info	false			Patrick.	Levis	Germany	3/19/1970			
editable	true			Eric	Myers	Germany	6/8/1950			
last modif	November 19, 2009 12			Laura	Torres	Germany	7/16/1984			
linked	false			Carol	Murphy	Germany	12/1/1982			
location	C:\Gui4DotNet\Sample dsCustomer.i			Thomas	Jenkins	Gemany	12/29/1954			
path	/Samples/dsCustomer.i			hutio	Vilalene	LICA	1/27/1992	U		
size	963 bytes		1	Jutan	watton	USA	1/2//1363			
			· _			_				
			13	END CAT	ua.				_	
		1	44	FUE CONSTRU	10700					
			46	END CONSTRU	C POR .					•1
			1						2	
		0	Console	23 E Problems	🖉 Tasks			📕 🗙 🦗	🕞 🔊 🔊 🔊 😁 - 😁 - *	
		Custo	omerDat	aGridView [OpenEdge	Application] C:\Prog	ress\OpenEdge102b	(bin/prowin32.exe	(Dec 7, 2009 1	:14:47 PM)	
										1
		4								2
] 🎌 🗈						Writable	Insert 4	0:58		

Like the UltraGrid, the Microsoft grid doesn't do any data sorting by default. So I need to find out what to set to enable sorting. If I select the Customer ProBindingSource in the Design view, I see all the properties that a ProBindingSource defines. The details of what these do are all documented in the book *OpenEdge Development: GUI for .NET Programming* in the OpenEdge doc set. Not surprisingly, the AutoSort property enables the binding source to handle sorting. By default it's False, so I reset it to True.

OpenEdge Editor - Samples/EustomerDataGridView.cls - OpenEdge Architect - C:\Gui4DotNet	: اما ـــ
ne cat besign havigate search project opencage kun window nep	
E Comeage Editor	
🥰 Resources 🕴 👘 🖓 "CustomerDataGrid Kew.cks (Design) 🕺 🙆 CustomerDataGrid Kew.cks	
	Toolbox 7
CustomerDataGridView	
C DealerEmpModeLcls	
E DealerEmp/SD.p CustomerFirstName CustomerLastName CustomerBirthCoun	IomerBithdate Customert
a dsCustomer.i	Pointer
- i dsDealerEmp.i	eb Button
C IModel.ds	CheckBox
	CheckedListBox
	ColorDialog
1 2 U U	E Combollos
meRSCurtamer - Promess Data BinderdSor	N ContextMenuShip
	DataGridView
Properties Events	DateTimePicker
(Name) moBSCustomer	Director Entry
AllowEdit True	DirectonSearcher
AllowNew True	EnvDouider
AllowRemove True	Enter formation
AutoSoft True	The Full Contend of Jakobar
AutoSync True	Paul and Paul
AutoUpdate Falce	FlowLayour and
Batching False	Cartholes
DataMember (F)	(³²) Course Day
DataSource W mo8SCustomer	Encuption (E) Hard Barriston
File	(F) Helphrovider
ProfiledingScience Designers	In ageList
	< >
Concelle 12 Problems 2 Tanks	
AutoSoft	
Indicates whether the BindingSource will customerbataondview [Openbage Application] C: (Progress)Openbage Application]	e102b(bin(prowin32.exe (Dec 7, 2009 1:14:47 PM)
automatically sort by specified column u_	2
	<u> </u>
1 LA 11 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1	

Now when I re-run the form and click on a column header, the data gets sorted on the field displayed in that column, as I would expect. Note that the Microsoft grid control doesn't coordinate with the binding source to do more than single column sorting. But because the binding source is doing the sorting in this case instead of the grid, the disadvantage seen in the UltraGrid example – where the sort order of the query and the visible sort order as shown in the grid are out of sync -- isn't a problem.

To demonstrate that the sort order shown in the Microsoft grid is the same as the sort order of the underlying query, I again add a **Next** button to the form to advance the query position by incrementing the binding source **Position** property.

🗑 OpenEdge Editor - Samples/Eustomei	DataGridView.cls - OpenEdge Architect - C:\Gui4DotNet	
File Edit Design Navigate Search Proje	ct OpenEdge Run Window Help	
🗈 • 🔛 🛆 🔏 💷 🕲 🏇 •	Q • Q ₄ • [A ² •] ½ • ½ • ½ • ⊕ • • • •	
📑 💐 OpenEdge Editor		
🥰 Resources 😫 🔷 🗖	🎝 *CustomerDataGridView.cls (Design) 💠 🕜 CustomerDataGridView.cls	° 8
(+ -+ ke) 😑 🍇 🏹		Toolbox 7
C DealerEmpForm.cls	EustomerDataGridView	CustomControls
DealerEmpModeLcls		- Nimmet Contrate
P DealerEmpVSD.p	CustomerFirstName CustomerLastName CustomerBithCoun CustomerBithdate CustomerE	Microsoft Controls
dsCustomer.i		Pointer
- i dsDealerEmp.i	*	ab Button
IModel.cls		CheckBox
8 Ou 📅 DB 🗆 Pr 🕮 🖳		CheckedListBox
te Al m 🗸		ColorDialog
a: X+ 02		ComboBox
button1 : System.Windows.Forms.Button		A ContextMenuStrip -
Properties Events		DataGridView
El Location 27 266		DateTimePicker
Locked False		In DirectoryEntry
E Margin 3.3.3.3		DirectorySearcher
E MaximumSize 0,0		U ErrorProvider
E MinimumSize 0.0	00	Even/Log
Modifiers Private	Ó Next Ď	25 FileSystemWatcher
Padding 0, 0, 0, 0	0-0-0	FlowLayoutPanel
RightToLeft No		FolderBrowserDialog
E Size 75, 23		- FontDialog
Tabindex 1	22 exp80 utomer	GroupBox
TabStop True	A monocontroller	(F1) HelpProvider
Tag		@PImageList
Text Next		A Label
Texplagn MiddleLenter	Consula 22 P. Bushimu A Tarka	
Test	deministration Contemporaria C	
The test associated with the control.	erenningen in concursion and and an adversarial claudice of the concursion of the co	0,000 11:00 01 May
	x	
	Writable	

I double-click on the button to get a **Click** event handler, and add code to increment the binding source **Position** value, which does a Get-Next on the associated ABL query.

@VisualDesigner.
METHOD PRIVATE VOID button1_Click(INPUT sender AS System.Object,
INPUT e AS System.EventArgs):
moBSCustomer:Position = moBSCustomer:Position + 1.
RETURN.
END METHOD.

When I re-run the form and click on the CustomerLastName column just as I did before, I see that the data is now sorted properly. And if I click the Next button, the row marker in the grid shows that the data sorting you see in the grid is in sync with the sort sequence of the ttCustomer temp-table guery.

File Edit Source Navigate Search Project OpenEdge Run Window Help • • • • • • • • • • • • • • •	
Image: Second secon	
Image: State Editor Image: State Editor Imag	
Resources 23 CustomerDataGridWew.ds (Design) CustomerDataGridWew.ds X	
	<u> </u>
C DealerEmpForm.ds S S S S S /*	
DealerEmpXSD.p	
····· D dsCustomer.i	*/
CustomerFirstName CustomerBirthCoun CustomerBirthAate Custon	
Patrick Alexander Germany 2/8/1983 Obj	ject, INPUT e AS System
Bebert Baker USA 4/29/1974	
Susan Barnes Germany 7/17/1972	
Property Value Evelyn Bing USA 12/10/1965	
derived False Helen Brooks Germany 7/31/1974	
editate true Larry Brooks USA 2/29/1972	
last modif November 19, 2009 12 Mary Burns USA 9/2/1956	
linked false Melissa Burns Bernanu 10/25/1959	
name dCustomer.i	
path /Samples/dsCustomer.i	y be modified using the
size 963 bytes na .	
Next	r will involidate onv c
	iner().
75 @VisualDesigner.FormMember (NeedsInitialize="true").	0 -
76 DEFINE VARIABLE tableDesc1 AS Progress.Data.TableDesc No-	-UNDO.
📮 Console 🖾 🦹 Problems 🖉 Tasks 🛛 🗮 🗮 🙀	51 🖉 🖉 🖻 • 📬 • 🗖 🗖
CustomerDataGridView [OpenEdge Application] C:\Progress\OpenEdge102b\bin\prowin32.exe (Dec 8, 2009 11:58:43	3 AM)
	×
<u> </u>	F

I can make this doubly clear by adding a statement to the button's **Click** event handler to display the **PREPARE-STRING** of the temp-table query handle after it has been automatically re-prepared by the effects of setting the **AutoSort** property to **True**:

	_
@VisualDesigner.	
METHOD PRIVATE VOID button1_Click(INPUT sender AS System.Object,	
INPUT e AS System.EventArgs):	
moBSCustomer:Position = moBSCustomer:Position + 1.	
MESSAGE "moBSCustomer:Handle:PREPARE-STRING is: " SKIP	
moBSCustomer:Handle:PREPARE-STRING VIEW-AS ALERT-BOX.	
RETURN.	
END METHOD.	

The MESSAGE statement alert box that appears when I run the form with this version of the event handler, and then click on the CustomerLastName column header, shows that the binding source has properly reprepared the guery for me with the appropriate BY clause.

Message (Press HELP to v	view stack trace)	×
moBSCustomer:Handle:PRI PRESELECT EACH ttCustor	EPARE-STRING is: ner BY CustomerLastNam	ie
OK	Help	

The ProBindingSource, being an OpenEdge-specific .NET control, is able to generate the right BY clause itself, and then re-prepare and re-open the query that its **Handle** property points to, in effect doing the work that I did myself in the UltraGrid example using the **SortData** method. This example reinforces two important facts about the OpenEdge GUI for .NET support.

First, it makes it clear that different controls doing basically the same job, such as the Infragistics UltraGrid and Microsoft DataGridView, can work very differently, including in how they interact with the ProBindingSource.

Secondly it shows one of the advantages of having the ProBindingSource as an OpenEdge-specific control, serving as the intermediary between the .NET UI and ABL data management. Just setting **AutoSort** on the ProBindingSource to True enables the ProBindingSource to reopen the query it's attached to with the correct BY clause, as this diagram illustrates:



Beyond the automated support for sorting, there may be cases where you want to intercept the sort request yourself much as I did in the UltraGrid sample to do something that goes beyond what AutoSort does, for instance to enable multi-column sorting that the Microsoft grid and the AutoSort property of the binding source don't provide, or to support sorting of data that's being retrieved in multiple batches. The remainder of this document shows how to use that alternative.

If I look at the events that the ProBindingSource supports, I see a **SortRequest** event:

File Edit Design Navigate South Project Coarding Num Workdow Help Image: South Coar	@ OpenEdge Editor - Samples/Eustomer	DataGridView.cls - OpenEdge Architect - C:\Gui4DotNet		
Image: State of the state	File Edit Design Navigate Search Proje	rct OpenEdge Run Window Help		
E Concerdance State of the stat	🗈 • 🛛 🛆 🔏 💷 🦦 🏇 •	Q + Q ₄ + [A ² +] [[] + [
Resources 33 Resources 34 R	📑 💕 OpenEdge Editor			
Image: Section of the section of th	🦓 Resources 🕴 🔅 🗖	SustomerDataGridview.cls (Design) 🗱 🕜 CustomerDataGridview.cls		•
Control Control Contro Control Control Control Contro	+ + & 🖨 😫 🏹		Toolbox	9
Description CutomeEntName Datamed.atName Datamed.atName Datamed.atName Description Socialization Image: Socialicitation	C DealerEmpForm.cls	CustomerDataGridView	CustomControls	-
discustomer:///www.setters.org/line/incomerce/indoneses/opent/dept/2003/informer/indoneses/opent/indoneses/opent/dept/2003/informer/indoneses/opent/dep	DealerEmpModeLos DealerEmp/SD.p	Custome/FistName CustomerLastName Custome/BithCoun Custome/Bithdate CustomerE	Microsoft Controls	
determined and the Balance	dsCustomer.i		R Pointer	
Deckdos D	dsDealerEmp.i	*	Button	
Between Structure: Program Structure: Decked.wf80x Program Structure: Program Structure: Decked.wf80x Program Structure: Decked.wf80x Decked.wf80x Decked.wf80x	IModel.cls		CheckBox	
Bill Charlowing Bill Charlowing Properties Descrites Descrites <td>🗄 Ou 📅 DB 🗆 Pr 23 🔍 🗆</td> <td></td> <td>CheckedListBox</td> <td></td>	🗄 Ou 📅 DB 🗆 Pr 23 🔍 🗆		CheckedListBox	
Bit Controlling Properties Data Standy Properties Data Standy Data Standy Discondy Standy Discondy Standy Discondy Standy Discondy Standy Discondy Standy Discondy Standy Discondy Standy	1 A 1 10 7		ColorDialog	
Inde Scutterer: Progens Date Andrego V Properties Bending Conset New W Determinations Dete	G: X+ 03		ComboBox	
Prosente Eventa Product Constantion Construction Data Manager Data M	moBSCustomer : Progress.Data.BindingSor		ContextMenuStrip	_
Participantial Date TrainPicket Constribution Declargional Constribution Declargional Declargional Declargional	Properties Evente		DataGridView	
Image: comparison comparison Image: comparison comparison Comerc/Comparison Image: comparison Data Membra Cha Data Membra Cha	Disf. Consta		DateTimePicker	
Construint of the solution of the solutio	Cased C		5 DirectoryEntry	
Concention-popel Implementation DataAmention-David Implementation Definition-David Implementat	CranteRow		S. DirectorySearcher	
Image: Settlespeet Settlespeet Dockning Consol Settlespeet Settlespeet Settlespeet Settlespeet Settlespeet	CurrentChargert	1	ErrorProvider	
Dadkers Net Dadkers Pickoverskip Dadkers Pickoverskip Diskers Pickoverskip	Durenillen/Dara		territ.og	
Diadoutochow Implementation Diadoutochow Implementation Dittod Implementation Pailindinutocure Implementation Dittod Implementation Implementation	DataEnor	Next	2 FileSystemWatcher	
Deta Source/Corr UniCharged Office Deta Concept Source So	DataMemberCha		FlowLayoutPanel	
Lucio Anged Offind Padiation Changed Offind Padiation Changed Settlequest Settlequest Document Man Schwarz (Comme Data Schwarz Agekation) C:/Progress/Cyent Edge 102b/siniprover02: ene (Dec 8, 2009 11:58-13 Ad9 Lucio III - Control IIII - Control IIII - Control IIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIII	DataSourceChar		FolderBrowserDialog	
Offind Ciscuption Problech/Congred Image: Ciscuption Concurs left in the sure portions on sotion on a control, like clocking on the ciscuption Image: Ciscuption Image: Ciscuption Image: Ciscuption <	ListChanged		FontDialog	
Pedificiant/Carryond Image: Charged in the image: Charge in the	OffEnd		GroupBox	
Settlegent Image: Consume Data Settingent Consume the susce performs on action on a control, like clocking on the c. Consume Data Settingent Image: Consume the susce performs on action on a control, like clocking on the c. Consume Data Settingent Image: Consume the susce performs on action on a control, like clocking on the c. Consume Data Settingent	PositionChanged		F1 HelpProvider	
Endirichtung Designer A Labei x Settlegenet Console 20 1 moderne 20 migner Console 20 1 moderne 20 migner Console 20 1 moderne 20 migner Settlegenet Console 20 1 moderne 20 migner Settlegenet Console 20 1 moderne 20 migner Console 20 1 moderne 20 migner Console 20 migner <td< td=""><td>SotRequest</td><td></td><td>@ImageList</td><td></td></td<>	SotRequest		@ImageList	
Exercise date Uneque Console 53 1, Prodems 7 Tasks # 100 console 100 co	-		A Label	-
Settlegent Locus when he user performs an action on a control, like clicking on the c.	Profindingsource Designer	🖸 Console 🕮 🚼 Problems 🖉 Tasks 💿 🙀 🦉	a 🖉 🖉 🚽 🖬 - 😁	
Coccus when the sure performs an an active like cloking on the c.	SortRequest	sterminated> CustomerDataGridNew [CoenEdge Apple ation] C:(Progress)OperEdge102b)bin(provin32.eye (Dec 8	2009 11:58:43 AM)	
	Occurs when the user performs an	and a standard and a standard and and and and and and and and and an	, construction of the system of	10
	action on a convol, like clicking on the c	x1		1
	-			
) 📑 🖻			

Opening the ClassBrowser to see what I can learn about this event, I enter **SortRequest** as a search string, and select **SortRequestEventArgs**, the EventArgs object that is passed into a SortRequest event handler.



Expanding the list of its properties, I see that there's a **FieldName** property, which holds the unqualified name of the column that the user clicked -- remember that only single-column sorting is supported. There's also a **FieldIndex** property, which holds the position of the field within the list of fields in the binding source, if you want to use that instead of the field name, perhaps if the unqualified **FieldName** is not unique. There's also a logical property named **Ascending** that signals whether the sort is ascending or descending.

🗱 OpenEdge Editor - Samples/CustomerDataGridView.cls	s - OpenEdge Architect - C:\Gui4DotNet		.OX
File Edit Design Navigate Search Project OpenEdge R	un Window Help		
📬 • 🛛 🛆 🔏 📖 🕲 🎄 • 💽 • 🏊 • .	A •] 1 • 1 • 1 • 1 • 1 • 1		
📑 💱 OpenEdge Editor			
Class Browser 23		- 🗢 🜆 🖉 🖊 🖯 🖗 🍈 🗖	
Search Jostrequet	Getype 0 Getype 0	2 	÷
	Summary:		<u> </u>
		j	20

Again, you can learn all the details about these properties in the material on data binding in the *OpenEdge Development: GUI for .NET Programming* book, which you will find in the Product Documentation available in the OpenEdge section of PSDN on the Progress Communities website.



The list of **SortRequestEventArgs** properties in the **Class Browser** gives me the basic information I need to be able to use the **SortRequest** option in my example form. Since this is an alternative to asking the binding source to **AutoSort** the data itself, I set the **AutoSort** property back to False. Then in the **Events** tab for the binding source control, I double-click the **SortRequest** event to get code generated to subscribe to the event and to provide an event handler.

CopenEdge Editor - Samples/Customer	DutaGridView.cls - OpenEdge Architect - C:\Gui4DotNet	
File Edit Design Navigate Search Proje	sct OpenEdge Run Window Help	
] 🗈 • 🔛 🍐] 🔏 💷 🦦 🏇 •	Q + Q ₄ + [A +] ≥ + ∅ + ψ + ↔ +	
📑 🐉 OpenEdge Editor		
🦓 Resources 💠 🖳 🗖	🖏 *CustomerDataGridview.ds (Design) 💠 😢 CustomerDataGridview.ds	- 0
	Toolbox	4
C DealerEmpForm.cls	E CustomerDataGridView	5 .
C DealerEmpModel.cls	Custome Certification Customed autitianal Custome Office Counter Officiates Customed	rols
dsCustomer.i	Pointer	
dsDealerEmp.i	* Bulton	
C IModel.cls	CheckBox	
🐮 Ou 📅 DS 🗆 Pr 🗶 🖓 🗖	Checked.inflox	
〒 兌 - 巴 [▽]	ColorDialog	
moRSCustomer : Promote Data RischooSco	Controller Size	_
nobacusuler. Progess bala brungso	DataGrid/ow	
Properties Events	Date TimePicker	
BindingComplete A	I DirectoryEntry	
CasteRey	S. DirectorySearche	
CurrentChanged	ErrorProvider	
CurrentitemChan	EventLog	
DataEnor	Next 200 FieSystemWatch	or
DataMemberCha		-
DataSourceChar	a) receptions	log
Diffund	D C C C C C C C C C C C C C C C C C C C	
PositionChanged	🖓 moBSCustomer (F) HelpProvider	
SotRequest	@ ImageList	
	A Label	-
ProtindingSource Designer	Conscie 22 C Problems & Tasks	
SortRequest	cheminated > OustomerDataGidNew [OpenEdge Application] C1ProgressiOpenEdge102bibiniprowin32.eve (Dec 8, 2009 11:58:43 AM)	
Occurs when the user performs an action on a control like dicking on the c		101
activity of a called and a called a cal	x	2
1.00		
10 3		

The skeleton event handler that's generated for me shows the **SortRequestEventArgs** parameter being passed in, which in this case is named **args**. Using the properties of the args object, I add code to put together a generic sort string just as I did in the UltraGrid example. There can only be one field to sort on, so I get its name from the **FieldName** property, and append to that the word **DESCENDING** if the value of the **Ascending** property is **False**. And then I pass the string to **SortData**:



Note that this very different user interface form class, containing controls with different properties and events, is using exactly the same sort method in the Model as my other example used, because the code responsibilities are properly separated. When I run the form again and click on the CustomerLastName column, I see the MESSAGE statement that signals that **SortData** was invoked, though in this case the resulting sort is the same as **AutoSort** would have given me automatically.

I click on the column again, which in this control type signals that I want to reverse the sort order, and I see the BY clause MESSAGE statement displayed by **SortData**.

🕄 OpenEdge Editor - Samples/Customert	ataGridVie			otNet				- I X
File Edit Source Navigate Search Proje-	t OpenEdg	e Run Window H	ыþ					
] 🗂 • 🗟 🖮 🖌 📖 🕲] 🀲 •	0 • 9	• 🛷 • 🔄 •	🤯 • 🗢 🗢 •	÷ *				
🞦 🎳 OpenEdge Editor								
👫 Resources 💠 🗖 🗖	🆏 Custom	erDataGridView.cls (De	esign) 💽 Cust		8			° 🛛
	211	1.0						-
C DealerEmpForm.ds	212	/*Pur	pose:					
C DealerEmpModel.ds			,			1=1-4	1	
dsCustomer.i	Custor	nerDataGridView	_				*/	
dsDealerEmp.i		CustomerFirstName	Customed artiklame	CurtomeRithCoun	CustomeRithdat	le Curtos è	r 16 Suntan Object	
C DModel.cls		Patrick	Alexander	Germany	2/8/1983		, and appreaments jeece,	
🗄 Ou 📅 DB 🗆 Pr 🕺 📍 🗆		Robert	Baker	USA.	4/29/1974			
🗄 🍺 🕾 🎽		Susan	Barnes	Germany	7/17/1972		L.	
Property Value		Evelyn	Bing	USA	12/10/1965		r.	
derived false		Helen	Brooks	Germany	7/31/1974			
editable true		Moss and (Bress M	I R to siem stack I	race) IXI	2/29/1972			
last modif November 19, 2009 12		and a state of the			9/2/1956			
location C:\Gui4DotNet\Sample		BY CustomerLastN	iame DESCENDING		10/25/1959			_
name dsCustomer.i		OK	Help	1	1/5/1968	-		
size 963 bytes	1			-		<u> </u>		
		Next						
	-							
	234	END.						
	235							<u> </u>
	-		A					
	Console	Si Problems	Tasks	and Care Edge 100k	United assessed at 22 million		UN 201 501 501 101 101 101 101	
	cuscomercia	raannees [Obeuroide	e whhereangul critistob	ressyuperic dge1020	quergerowin32.exe	r (Dec o, 2009 12	10.51 PHy	
	R.							1
J 😷 🖪				Writable	Insert	220:1		_

As I noted before, you might want to intercept the **SortRequest** if you wanted the user interface to allow the user to specify multi-column sorting, or some other special treatment of the sort request that **AutoSort** doesn't provide, perhaps checking whether there's an index on the requested sort field. Clicking the Next button confirms that what I see in the grid and the sort order in the query are in sync with each other, whichever sorting option I choose to use.

🔀 OpenEdge Edit	tor - Samples/Eustomer	DataG	GridVie	w.cls - OpenEdge A	rchitect - C:\Gui40	botNet				
File Edit Source	Navigate Search Proje	ct O	perEdg	e Run Window H	ielp					
] 📬 • 🔛 🖆	1400 0.	0.	Q. •	·] 🖉 •] 👰 •	§ - 🗢 🗢 -					
📑 💱 OpenEdge	e Editor									
🗳 Resources 🛙	- 0	5	Custome	erDataGridNew.cls (De	esign) 💽 Custe		. H			
	🗧 😫 🍸	23	11							-
C Deale	rEmpForm.ds	21	12	Pur	nose:					
E Deale	rEmpModel.cls			7.01	poser				a	
P Deale	rEmp/(SD.p		Custon	nerDataGridNew				_ [] X	•/	
- dsDea	slerEng /									
-C Mode	il.ds 💌		-	Customer-sulfilame	LustomerLastName	CustomerBirthCoun	Lutionethinda	te Luston.*	r AS System.Object,	
8E ou 🕅 ce	🗆 Pr 😒 🗖 🗖		-	3001	Woods	UCA.	1.47.6005			
	1 1 1		· .	Court	woods	1054	0.070000			
Property	Value			Red	Woods	USA .	7404065	II	P-	
🖃 Info				reu	webos	Community of the second	0.07.0000	- U		
derived	false			Amanda	warson	uemany UCA	8/25/1353	- II		
last modif	November 19, 2009 12			Jushn	Watson	USA	1/2//1983			
linked	false			Alice	Washington	Gemany	5/19/1902			
location	C:\Gu4DotNet'(Sample			Margaret	Ward	USA	1/20/1962	II		
name	dscustomer.i			Scott	Wallace	Germany	7/28/1957	-		
size	963 bytes		•			_		<u> </u>		
				Next						
		23	14	END.						
		23	35							1
			-							<u> </u>
		0	Console	23 Problems	Tasks			📕 X 🛞	<u>ina a</u> (⇔ (⇔ (⇔ ⊂ ⊂ ⊂ ⊂ ⊂ ⊂	
		Oust	omerDal	taGridView [OpenEdge	e Application] C:\Prog	ress)OpenEdge102t	ijbinjprowin32.exe	e (Dec 8, 2009 12	2:18:51 PM)	100
		π.								- 20
1 📷 🕲		لكر				Writable	Insert	228 : 1	1	
10 10										

To review, in this document and the series of video sessions that it accompanies, I showed you how to divide responsibilities between what you can think of as the **View**, your form classes that know about the user interface definition, and the **Model**, the part of the application that manages data access, so that the user interface and the data management don't get out of sync with each other.

I showed how to create an **ABL Interface** that allows you to create a number of classes that consistently implement that interface.

I then created a **Model** data management class that implements the Interface, and added code to fill a local **ProDataSet** with data, and to make the handle of the DataSet's temp-table query available through a method call.

I created a new form class with a **ProBindingSource** -- derived from the same ProDataSet definition used by the Model class -- and added an **UltraGrid** to it, setting its **DataSource** to be the ProBindingSource, then added a Next button to increment the ProBindingSource **Position** property.

I then used the Infragistics documentation to learn about the UltraGrid properties and events, and showed how setting the grid's **HeaderClickAction** property to **SortSingle** or **SortMulti** enables the grid to sort its copy of the data locally.

I used the effects of a Next button to illustrate how the redisplayed data in the grid is not in sync with the sort order of the underlying query that originally populated it. I coded a **SortData** method in the Model class to take a generic sort request and turn it into a BY clause for a re-prepared temp-table query, and coded an **AfterSortChange** event handler in the View to construct the sort request from the members of the **SortRequestEventArgs** parameter.

I reset the grid's **HeaderClickAction** property to **ExternalSortMulti**, and showed how this allows my event handler to interact with the Model class to re-prepare and re-open the Model's query with the new BY clause and use this as the basis for the data displayed in the grid, so that the user interface and the underlying data management class are in sync.

Finally, I showed you how the Microsoft **DataGridView** control works very differently from the **UltraGrid** in how it interacts with the **ProBindingSource**. However, the DataGridView has the advantage that if you ask for automatic sorting support, it's the binding source that's in control, so the sort done for the UI is accomplished by means of the binding source re-preparing and opening the query it's pointing to in the Model, the simple data management class.