When we use the ADO Data Environment Designer to make our connection, we don't need a Data Control, so let's pretend that we've created a new project, placed the DataGrid in our toolbox, and placed it on our form. Now we need to find and startup the Data Environment Designer. To do that, select Project-Add Data Designer from the Visual Basic Menu Bar ...

۴.	Add Eorm				
8	Add MD <u>I</u> Form				
緫	Add <u>M</u> odule				
۲	. Add <u>⊂</u> lass Module				
Ŧ	Add <u>U</u> ser Control				
b	Add Property Page				
1	Add User <u>D</u> ocument				
	Add WebClass				
	Add Data Report				
	Add DHTML Page				
	Add Data Environment 💦 📐				
	Add File Ctrl+D				
	Remove Form1.frm				
1	References				
	Components Ctrl+T				
	Project1 Prop <u>e</u> rties				

If you don't see Add Data Environment as an option, select Project-Components from the Visual Basic Menu Bar, click on the Designers Tab, and locate and select the Data Environment there...

 Addin Class Data Dynan ✓ Data Enviro ✓ Data Repor ✓ DHTML Page DTC Design Microsoft Feedback ✓ WebClass 	nics ActiveReports nment t e er orms 2.0 Form serConnection		Add MDTPo Add Module Add Class Module Add User Control Add Property Page Add User Document Add Data Report Add Data Report Add Data Environ More ActiveX Des
-Data Environi	nent		Delected Items Only
Location:	C:\\Common Files	\designer\MSDE.DLL	

Once you select Add Data Environment from the Project menu, you should see this window...



Right-click on the Connection object in the Data Environment Window and select Properties...

Content In the second section of the section	- DataEnvironment1 (DataEr	vironmer	nt)	_ 🗆 ×
🤹 🔁 🛍			1	
💼 DataEnvi	ronment1			
Conn	ection1 Expand All Collapse All			
	<u>D</u> elete			
	<u>R</u> ename Re <u>f</u> resh			
	Add C <u>o</u> mmand <u>I</u> nsert Stored Procedures			
	<u>V</u> iew Code			
Connection:	Properties			

This window will appear...

🗟 Data Link Properties 🛛 🗙
Provider Connection Advanced All
Select the data you want to connect to:
Microsoft Jet 3.51 OLE DB Provider
Microsoft Jet 4.0 OLE DB Provider
Microsoft OLE DB Provider for Internet Publishing
Microsoft OLE DB Provider for OLAP Services
Microsoft OLE DB Provider for Oracle
Microsoft OLE DB Simple Provider
MS Remote MSD at a Shape
OLE DB Provider for Microsoft Directory Services
<u>N</u> ext >>
OK Cancel Help

Look familiar? It should, it's the same wizard we saw when we created the connection using the Data Control. I'll repeat those same steps here (but in the interest of space I won't show you). Now we have a connection, but no recordset. Using the Data Environment, we must create a recordset via a Command Object. To create the Command Object, right-click on the Connection Object and select Add Command from the menu...



A Command object will appear in the Data Environment...

🏽 Project1 - DataEnvironment1 (Dat	taEnvironme	nt)	_ 🗆 ×
🍕 🛅 🛅 📉 🛃 🛃		1	
💼 DataEnvironment1			
🖻 🧐 Connection1			
Command1			
L			
Command: Command1 from Connection1			

Right-click on it, and select Properties. This window will appear...

Command1 Properties	×				
General Parameters Relation Grouping Aggregates Advanced					
Command <u>Name</u> : <u>Command1</u> <u>Connection</u> : <u>Connection1</u>					
Source of Data					
Database Object: Stored Procedure	•				
Object Name:	•				
O <u>S</u> QL Statement:	SQL <u>B</u> uilder				
OK Cancel	Apply Help				

We'll use a SQL Statement to create the underlying recordset (if you are not familiar with SQL, click on the SQL Builder button)...

Command1 Properties
General Parameters Relation Grouping Aggregates Advanced
Command Name: Command1 Connection: Connection1
Source of Data
○ <u>D</u> atabase Object: Stored Procedure
Object Name:
SELECT * FROM AUTHORS
UK Cancel <u>Apply</u> Help

and after clicking on the Apply Button, we now have a properly defined Data Environment...

🗱 Project1 - DataEnvironment1 (DataEnvironment)			
🍕 👝 🛅 🛬 🗶 🛃 🛃		1	
🛅 DataEnvironment1			
🖻 🥮 Connection1			
⊡			
Command: Command1 User-defined SQL fr	om Connection	1	li.

At this point, to 'bind' our DataGrid to the Data Environment Object that we've

just created, all we need to do open up the Properties Window for the Data Grid and specify the DataEnvironment as our Data Source and the Command Object in the DataMember property of the Data Grid...

🖀 Properties - I	DataGrid1 📃 🗖 🔉
DataGrid1 Data	Grid
Alphabetic Cate	gorized
(About)	
(Custom)	
(Name)	DataGrid1
Align	0 - vbAlignNone
AllowAddNew	False
AllowArrows	True
AllowDelete	False
AllowUpdate	True
Appearance	1 - dbg3D
BackColor	8H80000005&
BorderStyle	1 - dbgFixedSingle
Caption	
CausesValidation	True
ColumnHeaders	True
DataMember	Command1
DataSource	DataEnvironment1
DefColWidth	DataEnvironment1 💦 🗸 🗸
DataSource Specifies source of	f grid data

Now if we run the program, we'll find that the DataGrid is now 'bound' to our DataEnvironment...

🐃 Form1				_ 🗆 🗵
		Author	Year Born	
	▶ 1	Jacobs, Russell	I COLDON	
	2	Metzger, Philip W.		
	3	Boddie, John		
	4	Sydow, Dan Parks		
	6	Lloyd, John		
	8	Thiel, James R.		
	10	, John		
	12	Wellin, Paul		
	13	Kamin, Sam		
	14	Gaylord, Richard		
	15	Curry, Dave		
	17	Gardner Juanita Merc	a	▼

At this point, you may be thinking that the DataEnvironment really hasn't bought you all that much—after all, we can achieve the same functionality with the Data Control. That's true, but here's something the Data Control can never do for us. Suppose, instead of using a Data Grid to display the records in our Recordset, we'd like to display them in Textboxes?

It's pretty easy to display data in Textboxes using a Data Control—you just place labels and Textboxes on a form, and individually set the DataSource and DataField Properties of each Textbox to point to the Data Control. The DataEnvironment makes that even easier.

For now, let's delete the Data Grid from this form, and then click and drag the Command Object from the DataEnvironment Designer to our form---watch what happens!

🛎 Form1		
· · · · · · · · · · · · · · · · · · ·		
🕴 🛛 Au 🕼 🛱 📅 🖬 🖓 🖓 🖓		
] Authorn	Ц	
) AddudeD		
1		
Year Bort Hand		
· · · · · · · · · · · · · · · · · · ·		
· · · · · · · · · · · · · · · · · · ·	· · · · · · · · · · · · · · · · · · ·	

As you can see, Visual Basic automatically creates Label and Textbox controls for each 'field' in the Recordset that is produced by the Command Object. This can be a tremendous time saver, especially if the Recordset contains a lot of fields. Let's take some time to make the form a bit 'tidier'...

💐 Form1		_ 🗆 ×
Au_ID:	1	
Author:	Jacobs, Russell	
Year Born:		

http://www.johnsmiley.com/cis18.notfree/Smiley018/Smiley018.htm (10 of 12)3/28/2004 11:57:32 AM

...and then run the program. Voila, Data!

🐂 Form1		
Au_ID:	1	
Author:	Jacobs, Russell	
Year Born:		

Not bad for the little work we had to do to get this—but there's a big problem there's no way to move from record to record, as we have when we use a Data Control.

Let's place a command button on the form, and place this code in its Click Event Procedure—this will enable us to move to the next record in the Recordset...

Private Sub Command1_Click()

DataEnvironment1.rsCommand1.MoveNext

End Sub

Now if we run the program, and click on the Command Button, we'll easily be able to move to the next record in the Recordset...

💐 Form1		_ 🗆 ×
Au_ID:	2	
Author:	Metzger, Philip W.	
Year Born:		
	Command1	

Summary

I hope you enjoyed this article on using the Data Environment.