Using Windmill Software with Other Applications: DDE

Windmill Software Limited

Manual Code: WM.DDE-6.0

Issue Date: February 2004

Information in this document is subject to change without notice.
For updates please refer to the Windmill website at
http://www.windmill.co.uk.

PO Box 58
North District Office
Manchester M8 8QR
UK
Tel: +44 (0)161–833 2782
E-mail: sales@windmill.co.uk
http://www.windmillsoft.com/
http://www.windmill.co.uk/

Table of Contents

Dynamic Data Exchange

1 Introduction

Windmill is a suite of data acquisition and control software. It runs on a Windows PC and takes readings from many different types of measuring device. You can download a free version of Windmill by subscribing to the *Monitor* newsletter at http://www.windmill.co.uk/newsletter.html.

This Manual explains how to use the Dynamic Data Exchange (DDE) facilities of Windmill.

Windmill can use DDE in 2 ways:

1. to exchange data in real-time with another Windows application;
2. to send and receive commands to and from another Windows application.

The DDE Panel, Logger, Chart and the optional Graphics and Test-Seq programs support DDE; the other Windmill applications have no DDE capabilities.

Both DDE applications must be loaded before a conversation can begin.

The rest of the chapter is divided into three sections.

Section 2	describes exchanging data; first generally then giving examples of DDE links with Excel, Visual Basic programs and networks.
Section 3	describes sending commands.
Section 4	summarises using DDE links with Graphics.

2 Exchanging Data over DDE links

Data is exchanged within the Windmill system automatically. This section is only needed if you wish to link a Windmill program to some other application, such as a spreadsheet or database. For this you can use either DDE Panel or Logger. (It's easiest with DDE Panel.)

The remote application may send or request data to or from any of the currently connected channels as and when required. Alternatively a "Hot link" or "Automatic Link" may be used, with the Windmill program sending new data to the other application as it becomes available. Additional information, such as channel names, units, alarm states and error codes, can also be obtained by remote applications.

DDE applications are often referred to as either Clients or Servers. A Server always responds to instructions from a Client. The DDE Panel, Logger and Chart programs act as Servers but not Clients. Graphics and Test-Seq can act as either Servers or Clients.

2.1 Enabling DDE Support

Windows programs cannot make DDE conversations unless they have a unique DDE Service Name. The first instance of the program loaded will be given a default service name, "Windmill" for DDE Panel and "Logger" for Logger for example. To change or allocate a Service Name select DDE from the File menu and fill in the DDE dialogue box.

2.2 The DDE Protocol for Requesting or Sending Data

Dynamic Data Exchange is a standard protocol. Windmill programs acting as DDE Servers support the following parts of the DDE protocol.

INITIATE	when an external application wants to start a conversation with a Windmill program.
REQUEST	when the external application wants to obtain some information from a Windmill program.
ADVISE	when the application wants to be kept informed of all changes in a data value.
POKE	when the application wants to send a data value to an output channel.
TERMINATE	when either program wants to end the conversation.

The conversation is generally initiated and controlled by the external application—the Windmill programs respond to instructions, or terminate the conversation when you close them. Graphics, however, can initiate and control conversations.

A three tier system is used to acquire information through DDE conversations: Service Name, Topic and Item.

SERVICE NAME	The Service Name identifies the application, "Windmill" or "Logger" for example. If more than one copy of an application is loaded you must give each a unique Service Name. See Section A.1.1 for details.
TOPIC	The Topic is the subject of conversation. Details of topics supported by Windmill are in the next Section.
ITEM	Each topic supports one or more items. The Item identifies the data being exchanged. For example, an item may be the name of a channel, or a spreadsheet cell reference.

2.3 DDE Topics

The Windmill programs support six topics: Data, Units, Range, Alarm, Error and System. The first six take the current set of active channel names, or “AllChannels”, as their items. The System topic has seven items giving status, data format and information about the other topics. Values are returned as text, terminated with a carriage return. When several values are returned they are separated by tab characters.

DATA TOPIC

The Data topic is used to get the actual reading from an input channel, or send the value to an output. The channel is specified by the item, and the “AllChannels” item can be used to read or set all the active channels simultaneously. Windmill responds to a request for data by taking a new reading and returning this value.

UNITS TOPIC

The Units topic is used to find out the unit of measurement for the specified channel. If “AllChannels” is used as the item then the units of all the active channels are returned as a list.

RANGE TOPIC

The Range topic is used to find the minimum and maximum data values that an input or output channel can handle. The returned text contains the minimum and maximum values in engineering units separated by a tab character. The “AllChannels” item can be used to obtain the minimum and maximum values for all the active channels.

ALARM TOPIC

The Alarm topic is used with the DDE Panel to find out which channels are in a state of alarm. The reply is 0, 1 or 2:

0 No Alarm,
1 Warning Alarm,
2 Critical Alarm.

ERROR TOPIC

The Error topic is used to display any IML error codes that occur during reading or writing. (Sections and give details of the error codes.) When a 0 is displayed no error has occurred.

SYSTEM TOPIC

The System topic has seven items which may be read by an external application:

Help	This details supported topics and items.
SysItems	This lists the items supported within the System topic. It gives the reply “Help SysItems Topics Formats Status Channels”
Topics	This lists topics on which conversations can be held. The reply is “System Data Range Units Alarm Error”
Formats	This is the data format used to transfer data. The reply is “CF_TEXT”, which means data is transferred as ordinary readable text. The text string is terminated by a carriage return. If several values are transferred then the values are separated by Tab characters.
Status or TopicItemList	This gives server status.
Channels	These are the names of the channels connected to the Windmill program. The names are returned in CF_TEXT format, and at the end of the list is a special channel called “AllChannels”. This special channel name can be used as the item with the Units, Data, Range, Alarm and Error topics to obtain a list of data for all the active channels—accessing all channels simultaneously.

Topics Supported for Requests and Pokes

Topic Name	DDE Panel	Logger
Alarm	R	R
Data	R P	R
Errors	R	R
Range	R	R
System	R	R
Units	R	R

Key: R – Requests supported
P – Pokes supported

A Request is when the external application wants to obtain some information from a Windmill program.

A Poke is when the external application wants to send a data value to an output channel.

(The Graphics and Test-Seq programs can poke data to other applications, but cannot accept Poke commands themselves.)

2.4 Putting it all Together: Acquiring Information

To acquire information over DDE you need to specify the Service Name, the Topic and Item. For example, to get the input value of a channel connected to the DDE Panel, the Service Name would be Windmill, the topic would be Data and the Item would be the name of the channel. The easiest way to set up your DDE link is to use the Copy to Clipboard button in DDE Panel. You can then simply paste the link into the client application. The client (Excel for example) will return the channel's value as a number written as ASCII text.

When any conversations are in progress some of the options on the Windmill program menus will be disabled. This is to prevent changes being made to the hardware setup, as this may affect the conversation. For example the channel name may be the Item of a conversation. Changing it would break the link between the programs and prevent any further data transfer, so this is prevented. The disabled options are enabled again when all the conversations have been terminated by the remote application.

2.5 Using Windmill with Excel

This section describes transferring information between Windmill and Excel. You may also be interested in the Excel page on our web site. This includes links to a series of tips on using Excel, http://www.windmill.co.uk/excel.html

Excel supports communication by Hot links and by macros. With Hot links, when Windmill takes a reading cells in Excel are automatically updated with the new data. Macros allow more flexibility. They let you, for instance, log each set of readings to a new line in the worksheet

2.5.1 HOT LINKS

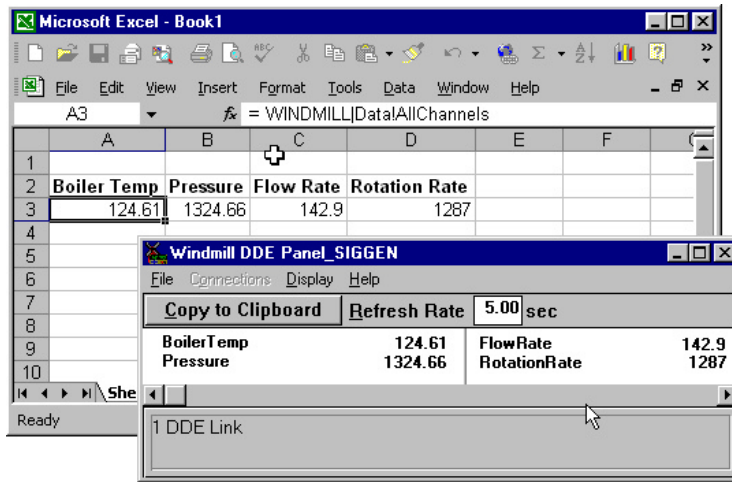
Hot links allow Windmill programs to automatically update Excel spreadsheets with current values. A hot link is set up between the Windmill program and a cell or group of cells in an Excel spreadsheet. As the data changes, the cells are updated with the new values. You can set up hot links with either Windmill DDE Panel or Logger: the easiest way is with DDE Panel.

A hot link can be used either to obtain the values of all channels handled by DDE Panel, or to obtain the values of individual channels.

There are 2 ways to establish hot links: use DDE Panel's Copy to Clipboard button; or enter the DDE references by hand into the Excel spreadsheet. To use the Copy to Clipboard button follow these steps.

1. Load DDE Panel.
2. Configure the hardware (File menu, Load Hardware Setup) and select the channels you wish to monitor.
3. Select the "Copy to clipboard" button.
4. Choose the type of information (topic) you want to copy, for example data or units.
5. Select the cell on your Excel spreadsheet where you want the data to appear.
6. Select Paste Link from Excel's Edit menu.

The DDE links are entered in the spreadsheet and when DDE Panel aquires new data, it automatically appears in Excel.



DDE Panel is automatically passing data to Excel via DDE.

Alternatively you could insert DDE links by entering the following formula by hand into the Excel cells:

= Service Name|Topic!Item

Example: = Windmill|Data!Chan1
or = Windmill|Data!AllChannels

If you are using AllChannels the links must be entered into an entire array: use Control + Shift + Enter therefore to enter the formula.

2.5.2 EXCEL MACROS

More flexible use of DDE can be obtained by writing Excel Macros. A brief guide to writing macros in Excel VBA (Visual Basic for Applications) is given below. You can copy and paste the macros from our web site at http://www.windmill.co.uk/excel.html

There are three stages when using DDE within a macro.

1. The first stage is the Initiate step where the conversation is set up. At this stage you specify the application (service name) and topic.

   ddechan = Excel.DDEInitiate (“Service Name”, “Topic”)

2. The second stage is the actual data transfer. Data can be requested from or sent to another program. The **Request** command is used to **get information** on any item, including the current input values from DDE Panel.

   wmdata = Excel.DDERequest (ddechan,"Item")

   Where ddechan is the value returned from DDEInitiate, Item is the channel name and wmdata(1) is the first value, wmdata(2) the second value and so on.

   The **Poke** command is used to **send data** to output channels via the Windmill DDE Panel. Only the Data topic can accept a Poke.

   Excel.DDEPoke ddechannel, “Item”, Range(“a1")

   Where Item is the channel name and Range is a reference to a single cell or an array of cells.

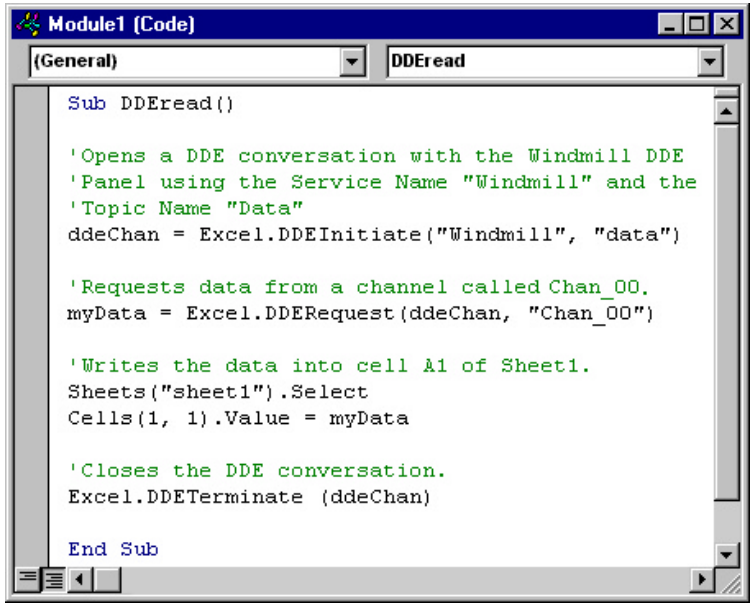
3. The third and final stage is Terminate. This closes the DDE conversation.

   Excel.DDETerminate (ddechannel)

   The following pages give examples of Excel macros.

Reading Data from a Channel and Writing it Into Excel

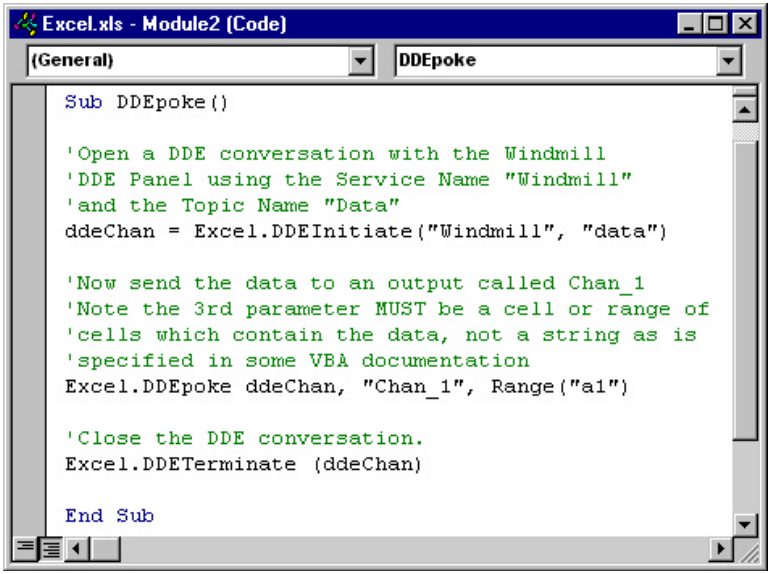
This macro reads data from a channel connected to the Windmill DDE Panel and places it into cell A1 in a worksheet called Sheet1.



Reading data from a channel and writing it into Excel.

Using Excel to Send Data to an Output Channel

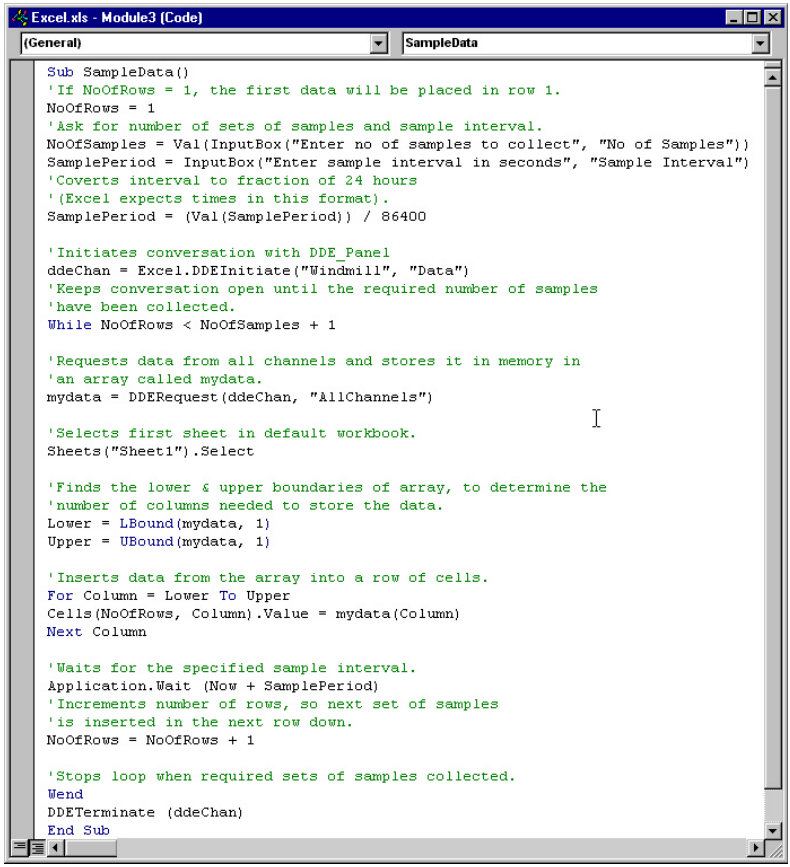
This macro sends data to an analogue or digital output channel called Chan_1.



Sending data from an Excel cell to an output channel connected to DDE Panel.

Repeatedly Sampling Data with Excel

This macro repeatedly samples data from all channels connected to the DDE Panel and progressively stores the readings in Excel. When you run the macro you choose how many times to sample all the channels, and the interval between taking sets of samples. The macro loops until the required amount of samples have been collected.




```
Sub SampleData()
'If NoOfRows = 1, the first data will be placed in row 1.
NoOfRows = 1
'Ask for number of sets of samples and sample interval.
NoOfSamples = Val(InputBox("Enter no of samples to collect", "No of Samples"))
SamplePeriod = InputBox("Enter sample interval in seconds", "Sample Interval")
'Coverts interval to fraction of 24 hours
'(Excel expects times in this format).
SamplePeriod = (Val(SamplePeriod)) / 86400

'Initiates conversation with DDE_Panel
ddeChan = Excel.DDEInitiate("Windmill", "Data")
'Keeps conversation open until the required number of samples
'have been collected.
While NoOfRows < NoOfSamples + 1

'Requests data from all channels and stores it in memory in
'an array called mydata.
mydata = DDERequest(ddeChan, "AllChannels")

'Selects first sheet in default workbook.
Sheets("Sheet1").Select

'Finds the lower & upper boundaries of array, to determine the
'number of columns needed to store the data.
Lower = LBound(mydata, 1)
Upper = UBound(mydata, 1)

'Inserts data from the array into a row of cells.
For Column = Lower To Upper
Cells(NoOfRows, Column).Value = mydata(Column)
Next Column

'Waits for the specified sample interval.
Application.Wait (Now + SamplePeriod)
'Increments number of rows, so next set of samples
'is inserted in the next row down.
NoOfRows = NoOfRows + 1

'Stops loop when required sets of samples collected.
Wend
DDETerminate (ddeChan)
End Sub
```

Repeatedly reading data from all input channes, and storing the readings in Excel

2.6 Using Windmill with your own Visual Basic Programs

The Excel VBA (Visual Basic for Applications) language covered in the previous section is the common macro language in Microsoft Office applications. When you are writing your own programs in Visual Basic, the DDE commands are different.

Visual Basic objects, such as label and text box controls, have inherent DDE capabilities. You can define the DDE conversation by changing the object's methods and properties. For example, for a label object called "Label1"

Label1.Linktopic	Defines the Service and Topic names of DDE Server applications
Label1.LinkItem	Defines the DDE Item name
Label1.LinkMode	Defines the type of DDE link
Label1.LinkRequest	Updates the label's text using data from the DDE Server

Other methods are available for sending data to a DDE Server, using DDE Execute commands for example. Full details of these methods and properties are contained in the *Microsoft Visual Basic Language Reference Manual.*

The following two examples demonstrate how you would use this technique to aquire data from, and to send data to, the Windmill DDE Panel.

VISUAL BASIC EXAMPLE 1

This timer control is used to update a text control Label1. A DDE link is established between Windmill and the text control and a new data value is acquired for the control, found in Label1.Text.

```
Sub Timer1_Timer()
‘ DDE Service Name and Topic
Label1.LinkTopic = "Windmill|Data"
Label1.LinkItem = "OvenTemp" ‘ DDE Item Name
Label1.LinkMode = 2 ‘ 0=no link
                    ‘ 1=warm link
                    ‘ 2=cold link
Label1.LinkRequest  ‘ Request Data from
                    ‘ Windmill DDE Panel
Label1.LinkMode = 0 ‘ Close the link

End Sub
```

VISUAL BASIC EXAMPLE 2

Sending data to an output channel.

```
Label1.LinkTopic = "Windmill|Data"
Label1.LinkItem = "MyOutput" ‘ DDE Item Name
Label1.LinkMode = 2          ‘ 0=no link
                             ‘ 1=warm link
                             ‘ 2=cold link
Label1.Caption = "1.23"      ‘ Set up data string
Label1.LinkPoke              ‘  Send data
Label1.LinkMode = 0          ‘ Close the link
```

2.7 Network DDE Links

Network DDE means that real-time data from Windmill can be pasted into applications on other computers anywhere on the network. The mechanism for this is provided by a DDE Share. You must create a DDE Share on the computer running DDE Panel; remote computers can then access this share to pass data.

The share specifies the DDE Service Name and the topic on which you wish to pass data. It also provides security features. To create the share use the DDEShare utility supplied by Microsoft. In non-NT systems you are prompted for:

Share Name	Any name you choose to create. It should end in a $.
Application Name in	This is the service name you entered DDE Panel.
Topic Name	Any of the available topics but normally Data.
Item Name	Leave Blank.
Start Application on Connect	Leave UnChecked.
Type of Access	Full.
Full Access Password	You can enter a Password here if you desire or leave it blank.

You have now created the DDE Share. (For Windows NT systems please see the DDE Panel Help.)

Accessing the Data

In local DDE data is accessed by a 3 item address :

	Service Name	Topic Name	Item
eg	WINDMILL	Data	00000

In the syntax of Excel this would be

='WINDMILL'|'Data'!'00000'

It would access channel 00000 of the Data topic at the WINDMILL service name.

In Network DDE the same structure is used in a slightly different way:

	Service Name	Topic Name	Item
eg	\\Production\NDDE$	OVEN$	00000

In the syntax of Excel this would be

='\\Production\NDDE$'|'OVEN$'!'00000'

It would access channel 00000 of the DDE share OVEN$ located on the computer named Production. The \s and the NDDE$ show that this is a DDE share on a distant computer.

Linking DDE Panel to Excel over a Network

Open DDE Panel on the machine where you have created a share, and start reading data. Open Excel on the distant machine. Type the DDE address as above into the Excel formula window. If you have not specified a password then communication is established when you press return. If you have specified a password then you will be prompted for it.

2.8 Saving DDE Links

Most client programs save DDE links and so you should remember to load the server Windmill program with same hardware setup, before reloading the client program. An easy way to do this is to include the setup filename as a property of the Windmill program. See Sections , and for details.

3 Sending Commands over DDE Links

You can send Windmill simple instructions over DDE links, using the DDE Execute facility. This allows Windmill to be controlled remotely from DDE Client applications, such as Excel, Visual Basic or Windmill Graphics. DDE Panel, Logger, Chart and Graphics can all accept commands.

With a Visual Basic program or Excel macro, say, you can sequence a series of actions—perhaps for the automation of a test. For instance, an Excel macro may be used to read several input channels and perform a calculation on the data. The outcome of this may cause Logger to be started and data to be recorded to a disk file.

The controlling application may have other facilities for loading and controlling applications. These may be combined with DDE to provide quite complex sequences.

Windmill Graphics may also be used as a DDE Client to send a single Execute command to either itself or other Windmill applications. You could use this to enhance mimic diagrams. You could, for example, by clicking over a particular part of the process represented on the mimic diagram send a Restore command to another Graphics application which features that item in more detail. Or you may wish to include a button on the mimic which allows the operator to start logging data to disk. Section 4 gives further details.

For more complicated sequences you can combine Graphics with another DDE Client. Graphics could be used to form the front end of a test system using DDE to send data to an Excel spreadsheet. The data may be used as a flag which results in an Excel macro being run. In this way it is possible to build a very powerful but easy-to-use system.

3.1 Execute Commands Accepted by Windmill

The target Windmill application should already be loaded and have a unique DDE Service name. In this section commands are shown in upper case and their parameters in angled brackets. Parameters should be separated by a single space.

DESTROY

Closes the Windmill application. Equivalent of the Exit command on the File menu.

INTERVAL <new interval>

This command is supported by Logger, the DDE Panel and Graphics. It determines how often the channels are read. The interval parameter specifies the interval in seconds. For example a 500 millisecond update rate would be specified as:

Interval 0.5

LOAD <setup file name>

Loads a new setup file. If the setup file is not in the current directory you must specify the complete file path. You could use this command to change the channels being monitored. For example Windmill Chart may be switched between charting temperature channels to pressure channels, or a Graphics window may be changed to show a different part of the process.

When you use DDE to load a new setup file, all the setup details are loaded apart from the DDE service name. To avoid confusion the service name cannot be changed whilst involved in a conversation with another application.

MAXIMIZE

Makes the Windmill window full screen size. This is not available on Graphics which is confined to the size of the background bitmap.

MINIMIZE

Iconises the Windmill window.

MOVE <x pos> <y pos> <x size> <y size>

Moves and resizes the Windmill window. The co-ordinates are in pixel units and refer to the position of the top left hand corner of the window. They are absolute, not relative to the current position. You can omit the size parameters if you don't want to change the dimensions of the window. To move the window to the top left of the screen, for example:

Move 0 0

PRINT

This is available with Windmill Chart and Graphics. With Chart it's equivalent of the Print command on the File menu. You could use Print, for example, in an Excel macro to produce a hard copy of a test after its completion.

With Graphics you can define your own Print button on the screen using the standard DDE output button; or you could perform a screen dump by sending a Print command from a remote application. For example Windmill Test-Seq may print the Graphics screen whenever a particular channel goes into alarm. Printing is performed to the default Windows printer, set in Control Panel. Areas not visible on the screen will not be included on the print-out.

RESTORE

Restores the Windmill window to the size it was before it was minimised or maximised.

START

Starts logging or charting.

STOP

Stops logging or charting.

3.2 Command Summary

Execute Command	DDE Panel	Logger	Chart	Graphics
Destroy	✓	✓	✓	✓
Interval	✓	✓	x	✓
Load	✓	✓	✓	✓
Maximize	✓	✓	✓	x
Minimize	✓	✓	✓	✓
Move	✓	✓	✓	✓
Print	x	x	✓	✓
Restore	✓	✓	✓	✓
Start	x	✓	✓	✓
Stop	x	✓	✓	✓

4 Using DDE Links with Graphics

DDE applications are often referred to as either Clients or Servers. A Server always responds to instructions from a Client. The standard Windows programs only act as Servers but Graphics can act as a Client or a Server. As Graphics is more powerful and therefore slightly more complicated than the other Windmill programs, a summary of using DDE with Graphics is given here. (For more on Graphics see http://www.windmill.co.uk/graphics.html)

4.1 Enabling DDE Support

Graphics cannot make DDE conversations unless it has a unique DDE Service Name. The first instance of Graphics loaded will automatically be given the service name “Graphics”. To change this, or allocate a service name to another instance of the program, use the DDE option from the Graphics File menu.

4.2 Sending DDE Commands to Graphics

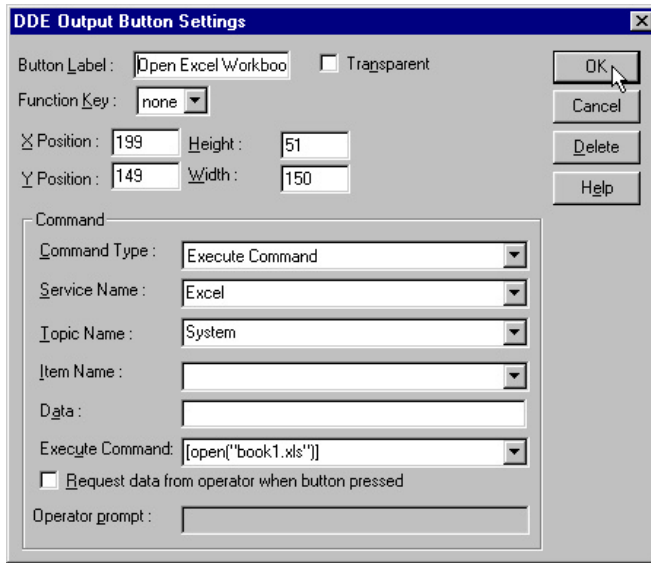
Graphics recognises nine DDE Execute commands: Destroy, Interval, Load, Minimize, Move, Print, Restore, Start and Stop. These may be sent by other Windows applications such as Excel, other instances of Graphics, or even by Graphics to itself. Graphics is here acting as the Server and receiving commands from the Client. For more details of the Execute commands see Section .

4.3 Sending DDE Commands to Other Applications

When Graphics sends commands to other applications it is acting as a Client. To send commands you must insert a DDE output button as follows.

1. Select Execute Command as the command type.
2. Select the Service name of the application to which you want to send commands.
3. If you’re sending commands to non-Windmill applications enter a Topic name. This is often the name of a file—see the Manual of your chosen target (server) application for details. If you are not sure use “System” (without quotes).

4. Enter the command you want to send. See the Manual of your chosen target (server) application for details of the commands it supports.



Inserting a button to send commands to other applications. Here Graphics will instruct Excel to open a worksheet called Book1.xls.

4.4 Sending Data to Other Applications

When Graphics instructs other applications to receive data it is acting as a Client. To send data you must insert a DDE output button (Section) as follows.

1. Select Poke Data as the command type.
2. Select the Service name of the application to which you want to send data.
3. Enter a Topic name. This is often the name of the file you want to send data to.

4. Use the Item name to specify the desired location of the data. In a spreadsheet, for instance, the Item name would be the address of a cell, or range of cells.
5. Enter the data values to be sent, or choose to defer entry until the DDE button is pressed.

Graphics does not understand requests for data from other applications: you should address the DDE Panel instead.

4.5 The SHELL Service Name

The SHELL Service name may be listed in the Graphics Service name options. Ignore it. This identifies facilities of Program Manager which are used by system developers. Graphics cannot use these, but recognises that they are there.