A

Networking With DDE

This appendix explains how to network Lookout*Direct* using DDE. This information is retained in Lookout*Direct* 4 for compatibility with earlier versions of Lookout*Direct*.

While in Lookout*Direct* 4 you can do most process networking with the faster and more efficient TCP/IP, you might want to use DDE to communicate with other applications, such as Microsoft Excel (as described in Chapter 5, *Dynamic Data Exchange*).

You can use Lookout*Direct* DDE networking capabilities to do the following:

- View the same or different screens simultaneously on separate nodes
- Make setpoint adjustments from any node
- Acknowledge alarms from any node
- Configure specific nodes for monitoring only
- Configure a peer-to-peer architecture
- Configure a client-server architecture
- Configure network nodes to communicate through standard telephone lines
- Configure network nodes to communicate through radios

Networking multiple Lookout*Direct* nodes is a powerful and advanced capability. This appendix refers to many concepts and terms explained elsewhere in the manual. Before you attempt to work with networking, make sure you have a strong understanding of the Lookout*Direct* architecture and are comfortable working within Lookout*Direct* on a single node.

To network using Lookout*Direct*, you must meet the following requirements:

- Purchase and install on each computer a separate, licensed copy of LookoutDirect.
- Equip each computer on the network with compatible network hardware.
- Install Microsoft Windows NT/98/95 on each computer.

Because Lookout*Direct* is a native Windows application, it supports all the networks that Microsoft Windows supports, including Microsoft Windows

Network, Microsoft LAN Server, Novell NetWare, Banyan VINES, 3Com 3+Open, DEC Pathworks, IBM LANServer, and so on.

NetDDE Networking Considerations

Networking with Lookout*Direct* involves passing data back and forth between two or more Lookout*Direct* nodes. In Lookout*Direct*, data is passed between and contained within objects. There are two basic methods for passing data between multiple Lookout*Direct* nodes: *Multilink Networking*, which requires a NetDDE (Network Dynamic Data Exchange) link for every value being passed between the nodes, and *Table Networking*, which implements a *data concentrator* at each node where only tables (which concentrate the data) are linked between nodes. Although both methods involve connecting objects over a network using NetDDE, the implementation and total effect can be quite different.

If you plan to use one of the NetDDE methods, you have to configure automatic NetDDE activation and possibly trusted DDE shares. See the *Running NETDDE.EXE Automatically* and *Adding a Trusted DDE Share* sections in this appendix for setup instructions.

There is one more method for implementing network-like capabilities between multiple Lookout*Direct* nodes. Unlike the first two methods, however, this technique does not rely on data being passed between Lookout*Direct* nodes. Instead, you use PLCs, RTUs, or I/O to share data between Lookout*Direct* nodes. This method is called *Hardware Networking*.

Multilink NetDDE Networking

Multilink networking consists of passing data from an object on one computer to another object on a different computer. When linked, any change to the value of one object is instantly propagated to and reflected by the second object on the other computer. The way in which data is passed between two objects across a network depends on whether they are controllable objects.

Linking Controllable Objects

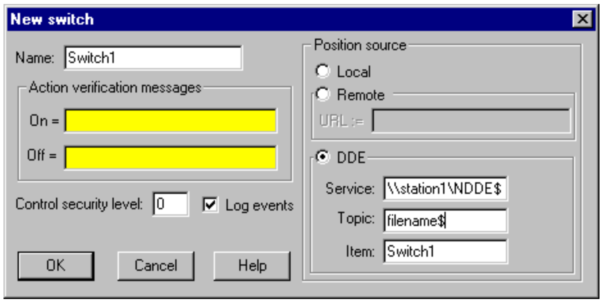
User-controlled objects have a DDE option. These objects include Pot objects, Switch objects, and Pushbutton objects.

Controllable objects each have three **Position source** options: **Local**, **Remote**, and **DDE**. When you choose DDE, you instruct the object to take its value from a remote source. This could be a cell in a spreadsheet, another DDE-aware application, or a second copy of Lookout*Direct* running on the network.

Linking Controllable Objects Together Across a Network

You need at least two computers connected by a functioning network to use the following example. This example refers to the two computers as Station1 and Station2.

1. At Station1, create a Switch object called `Switch1`. Leave **Position source** at the default setting of **Local**.
2. At Station2, create a corresponding Switch object called `Switch1`. However, change the **Position source** to **DDE**. (Although you are not required to give objects matching names over a network, it makes your documentation easier to follow and your applications easier to maintain.)



Name: Switch1

On =

Off =

Control security level: 0 ☑ Log events

Position source

○ Local

○ Remote

URL :=

● DDE

Service: \\station1\NDDE$

Topic: filename$

Item: Switch1

3. In **Service**, enter `\\Station1\NDDE$`, where `Station1` is the network name of the other computer. The backslashes, `NDDE`, and dollar sign are required by Microsoft Windows.

4. In **Topic**, enter `Filename$`, where `Filename` is the name of the LookoutDirect process file on the other computer.

   Again, the dollar sign is required but the `.l4p` file extension is not.

5. In **Item**, enter the name of the object at the other computer that you want to link to (in this case, the other object is also called `Switch1`).
6. Click on **OK**.

Test your network by flipping the switch. When you toggle the switch at either station, it instantly flips at the other workstation. You can add to, modify, delete, or revise your logic on either computer, completely on-line, and still maintain the network link between the two objects.

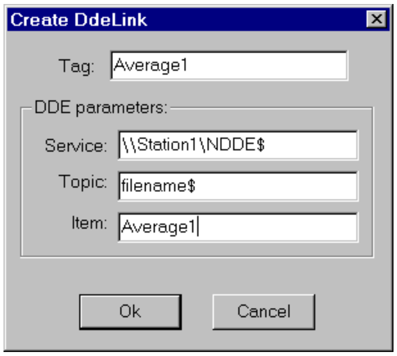
Linking Non-Controllable Objects

If a Lookout*Direct* object is not a Pot, Switch, or Pushbutton, it probably does not have built-in DDE capabilities. However, all you are really trying to do is pass data between two computers. After the data is in Lookout*Direct*, any object can access it, so you can create a DdeLink object to receive data from another application using DDE or NetDDE. For more information, refer to the discussion of the *DdeLink* object in the online PDF *Object Reference Manual* and online help. In this example, the other application is Lookout*Direct* running on a second computer.

To Access Real-Time Data at Another Computer

You need at least two computers connected by a functioning network to follow this example. This example refers to the two computers as Station1 and Station2.

1. At Station1, create an Average object. It can average whatever values you want, but give it the name `Average1`.
2. At Station2, create a DdeLink object and name it `Average1`. Although you are not required to give objects matching names over a network, it makes your applications easier to maintain.


Create DdeLink

Tag: Average1

DDE parameters:

Service: \\Station1\NDDE$

Topic: filename$

Item: Average1

3. In **Service**, enter `\\Station1\NDDE$`, where `Station1` is the network name of the other computer. The backslashes, `NDDE`, and dollar sign are required by Microsoft Windows.
4. In **Topic**, enter `Filename$`, where `Filename` is the name of the LookoutDirect process file on the other computer.

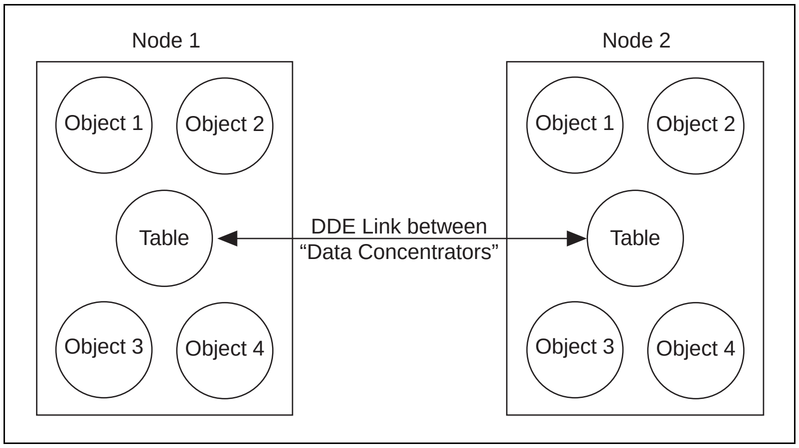
   Again, the dollar sign is required but the `.l4p` file extension is not.
5. Enter the name of the object at the other computer that you want to link to in the **Item** field (in this case, it is also called `Average1`).
6. Click on **OK**.
7. Finish the definition of the DdeLink object by inserting the DDE expression into your panel.

After you establish the link, you can connect other objects at Station2 to the `Average1` DdeLink object. Remember, just because you named the DdeLink object `Average1` does not make it a Lookout*Direct* Average object. It just receives data from the Average object at Station1.

Table Networking

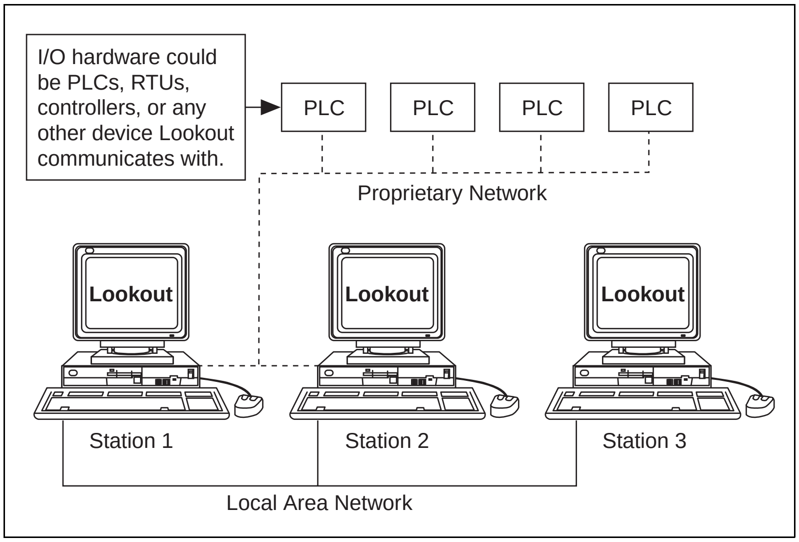
Table networking consists of consolidating all the data that needs to be shared over the network in a single table or array of values. You might think of these tables as “data concentrators”—although they can do much more than just serve as a repository for data. For more information, refer to the discussion of the *DataTable* object in the online PDF *Object Reference Manual* or the online help. You can link a table through NetDDE to a corresponding table on another computer. When linked, the tables update each other on all changes within their databases.

Note Table networking is one of the most involved and advanced concepts in Lookout*Direct*. You should have a complete understanding of objects, database connections, events, and general Lookout*Direct* architecture before you attempt table networking.

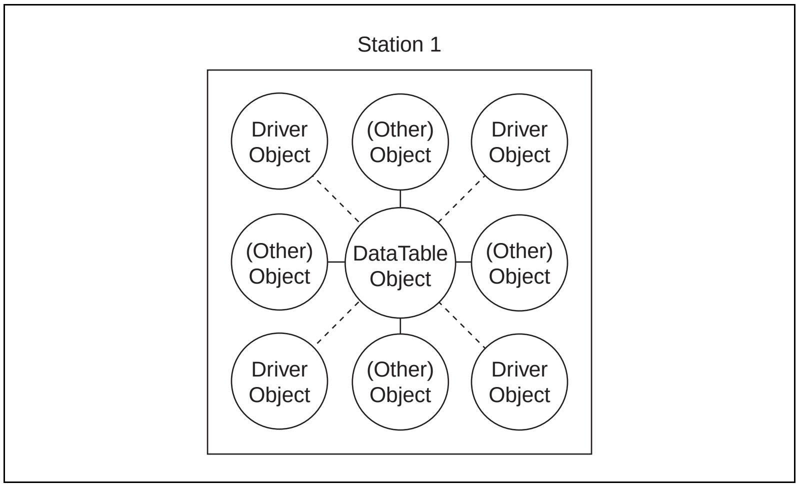


You might have better results linking multiple DataTables for networking purposes if you plan before jumping headfirst into your application. The following example shows how table networking is done.

This example consists of three computers, each running a copy of Lookout*Direct*. Set up three nodes on the network with the names Station1, Station2, and Station3. Station1 is the only node directly connected to a PLC, but you want to monitor and control the I/O from any of the three nodes. The following diagram depicts the network structure of this example.



By doing a little planning, you can configure Lookout*Direct* at Station1 so that you can reuse its process file at Station2 and Station3. The following diagram depicts the general design of the Lookout*Direct* process file at Station1. Notice that Driver objects are connected to the DataTable object using dotted lines while all other objects are connected using solid lines. This is for illustrative purposes only.



The focus of this picture is the Driver objects and their connections to the DataTable. Eventually, you will modify this process file to run at Station2 and

Station3 by deleting all connections to Driver objects, then deleting the Driver objects themselves at Stations 2 and 3.

It is important to route all signals to and from Driver objects through the DataTable. If you want to display an analog input from a PLC on a control panel, do not connect the PLC signal directly to the control panel. Instead, first connect it to the DataTable and then connect the appropriate DataTable cell to the control panel. If you need to connect an operator setpoint to an output on a PLC, do not connect the setpoint directly to the PLC. First connect the setpoint to the DataTable, then connect the appropriate DataTable cell to the output on your PLC. This is only necessary for signals that originate from, or are destined for, a Driver object. Although this procedure adds another step to your Lookout*Direct* configuration for Station1, it greatly simplifies the development of the process files for Station2 and Station3.

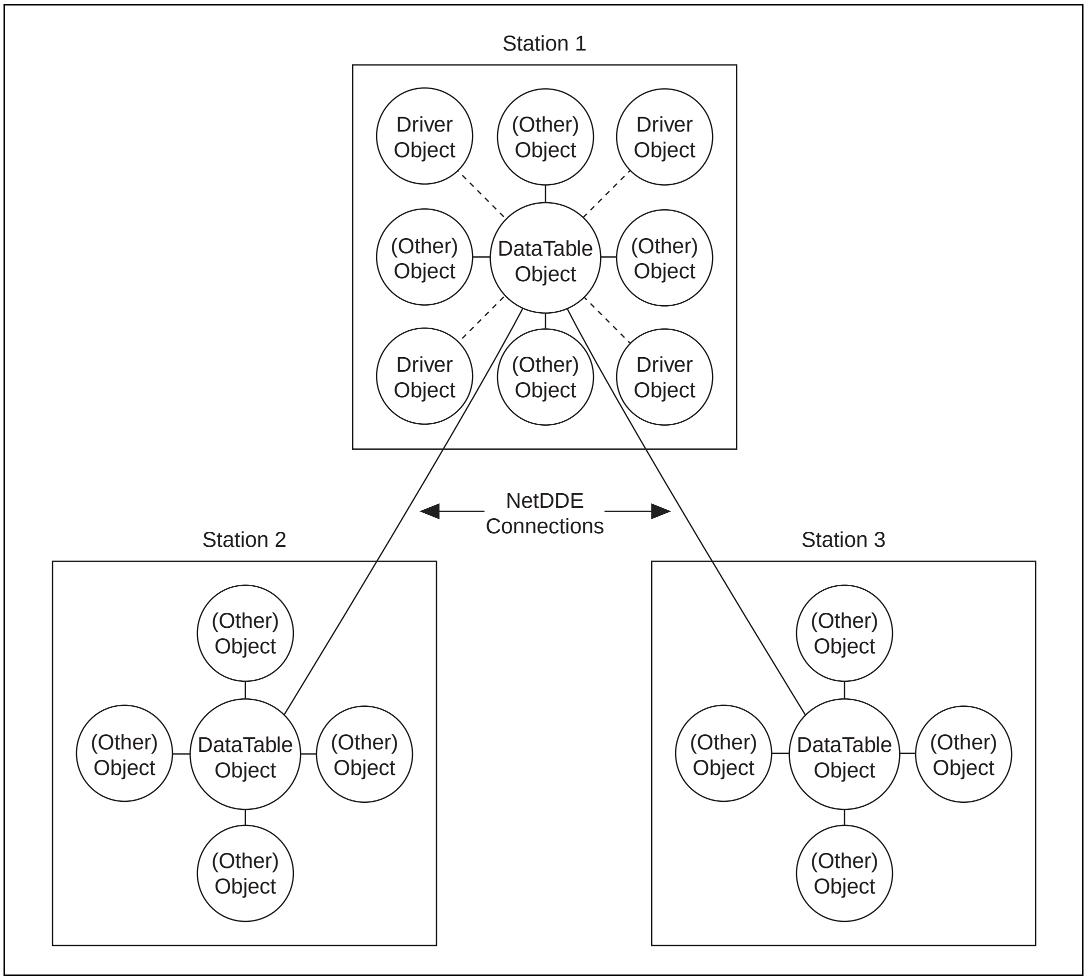
Now that you have developed an application at Station1, move over to Station2. Copy the process file to the Station2 computer and delete all the connections to and from your Driver objects. Then, delete the Driver objects themselves.

Remember, you do not want Station2 attempting to communicate with the PLCs because there is no physical connection to that proprietary network. Station2 should get its information from Station1.

The only thing left to do is modify the DataTable object at Station2. Change its **Table location** parameter to **DDE**. In **Service**, enter `\\Station1\NDDE$`. In **Topic**, enter `Filename$`, where `Filename` is the name of the process file at Station1. In **Item**, enter the name of the DataTable object at Station1. Your DataTables are now bidirectionally connected through a NetDDE link.

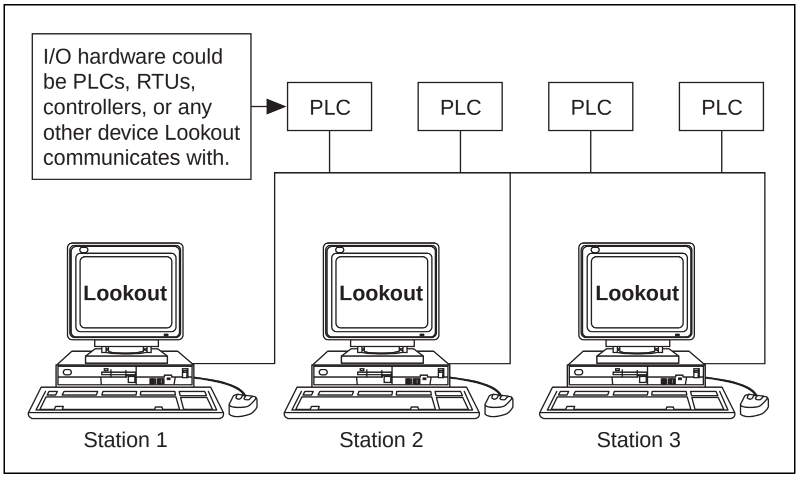
At this stage, you can add more operator stations to the network. Copy the process file from Station2 to Station3, or to any number of other Lookout*Direct* nodes on the network. No changes to the files themselves are necessary.

Your Lookout*Direct* network topography should now look something like this.


Station 1
Driver Object
(Other) Object
Driver Object
(Other) Object
DataTable Object
(Other) Object
Driver Object
(Other) Object
Driver Object
NetDDE Connections
Station 2
(Other) Object
(Other) Object
DataTable Object
(Other) Object
(Other) Object
Station 3
(Other) Object
(Other) Object
DataTable Object
(Other) Object
(Other) Object

Hardware Networking

In general, you should try to avoid hardware networking as described here with Lookout*Direct* 4. To implement the hardware networking method, all *LookoutDirect* nodes you want networked must have direct communication access to all your hardware as displayed in the following diagram. Because of this, the hardware networking method is only applicable in certain circumstances.


I/O hardware could be PLCs, RTUs, controllers, or any other device Lookout communicates with.
PLC
PLC
PLC
PLC
Lookout
Lookout
Lookout
Station 1
Station 2
Station 3

This is the most straightforward of the three networking methods to implement. Configure a process file at one node and copy it to all other nodes that have direct communication access to your hardware. The only thing to remember when building your process file is to configure all Pots, Switches, and Pushbuttons with the **Remote** parameter setting. This ensures that they stay in sync with the point they are writing to. In other words, if a switch at one Lookout*Direct* node changes the status of an output, all the respective switches at the other LookoutDirect nodes will reflect the change and automatically flip to stay in sync. For more information on the **Remote** parameter, refer to the *Pot*, *Switch*, and *Pushbutton* descriptions in the online PDF LookoutDirect *Object Reference Manual* in the online help.

Your hardware, network, and communications topology are the determining factors on whether you can use hardware networking. The hardware networking method is seldom available because it is rare that all your Lookout*Direct* nodes can communicate directly with all your hardware.

Multilink and Table Networking Comparison

There are advantages and disadvantages to both multilink and table networking. Multilink is quick and easy for small jobs. However, every NetDDE link requires processor time, so the more links you have, the more CPU overhead is used. Remember, each value passed between computers requires a separate DdeLink object. Your computer can quickly become bogged down if hundreds of links are constantly sending values back and forth.

On the other hand, while the table method might require more initial planning and development time, there will be only one NetDDE link between computers connecting the DataTables. This method dramatically reduces the amount of DDE traffic on the network. That is why table networking is the method of choice for large systems where computers are sharing hundreds or thousands of values across the network.

So how many points are too many? There is no single answer, but Lookout*Direct* has been used with the table networking method on systems having over 6000 values, and the network update time is well under one second with only three percent CPU usage.

Setting up Networking with DDE

Lookout*Direct* uses Network DDE (Dynamic Data Exchange) for client/server networking.

Two components of Microsoft Windows are required when using Network DDE: the Network DDE agent (`NETDDE.EXE`) and the NetDDE Share Database Manager (DSDM). Exactly how these components are used depends on the operating system on which Lookout*Direct* runs: Windows 98/95 or Windows NT.

The program `NETDDE.EXE` is an agent that runs in the background and transfers DDE messages between computers. This program should be running before an instance of Lookout*Direct* is started if you intend for that session to act as a server. Lookout*Direct* does not automatically run `NETDDE.EXE` itself.

Running NETDDE.EXE Automatically

The following sections show how to make sure that `NETDDE.EXE` runs every time you start Windows.

Windows 98/95

1. In Windows 98/95, Click **Start»Settings»Taskbar**.
2. In the **Taskbar Properties** dialog box, choose the **Start Menu Programs** tab.
3. Click on the **Add** button.
4. In the **Command line** field of the **Create Shortcut** dialog box, enter `NETDDE.EXE`, then click **Next**.
5. Choose the **Start Menu»Programs»StartUp** folder, then click **Next**.
6. Select a name (such as `NetDDE`) for the shortcut; select **Finish** and **OK**.

Windows NT

1. In the Program Manager, double-click on the **Control Panel** icon, or run `CONTROL.EXE` using the Program Manager **File»Run** dialog box. If you are using NT version 4.0, click on **Start**, point to **Settings**, and choose the **Control Panel**.
2. In the **Control Panel** window, double-click on the **Services** icon.
3. From the list of services, select `Network DDE`.
4. Click on the **Startup** button.
5. In the **Startup Type** box, select `Automatic`.

6. Select **OK** and **Close** to exit the **Control Panel**.

Adding a Trusted DDE Share

To allow someone else to connect to a DDE share (for example, `process$`) on your computer when you are logged in, you must *trust* the DDE share. When other users connect to the share remotely, the application they connect to is running in your security context because you are the logged-on user. You must give permission for the other users to access the share. Even another person who is an administrator cannot trust a share for your account.

In Windows 98/95, trusted shares are automatically created by Lookout*Direct*.

In Windows NT, you must use the program `DDESHARE.EXE` to add a trusted share. Exactly how you set this up depends on the level of security you require. The following sequence of steps describes how to set up a share that gives everyone on the network full DDE access to LookoutDirect.

1. From the Program Manager, select **File»Run**. If you are using Windows NT version 4.0, click **Start** and select **Run**.
2. Enter `DDESHARE.EXE`.
3. Select **Shares»DDE Shares**.
4. In the **DDE Shares** dialog box, select **Add a Share**.
5. In the **DDE Share Properties** dialog box, do the following:
    a. For the **Share Name**, enter the name of the LookoutDirect process file that you want to share, omitting the `.LKP` extension, and adding a '`$`' at the end. For example, if your process file was `PROCESS.LKP`, you would enter

       `PROCESS$`

       on this line.
    b. For the **Static Application Name**, enter LookoutDirect.
    c. For the **Static Topic Name**, enter the process file without extension, such as `PROCESS`.
    d. Leave the **Allow start application** and **is service** boxes unchecked.
    e. In the **Item Security** box, select **Grant access to all items**.
    f. Click on the **Permissions** button.
    g. In the **DDE Share Name Permissions** dialog box, do the following:
        - In the **Names** list, select `EVERYONE`. If there is no entry for `EVERYONE`, use the **Add** button and associated dialog box to create it.

- In the **Type of Access** box, select `Full Control`.
- Select **OK**.

h. Select **OK**.

6. You should now be back in the **DDE Shares** dialog box. Select the share you just created and click on the **Trust Share** button.
7. In the **Trusted Share Properties** dialog box, select the **Initiate to Application Enable** button, then click on **OK**.
8. In the **DDE Shares** dialog box, click on **OK**.
9. Exit the DDE Shares program.

Note For the DDE share to be available, it must be trusted by the user currently logged in to the computer (using steps 6 through 9). Alternatively, the share can be provided as a service by selecting the **Is service** box in the **DDE Share Properties** dialog box. In this case, the share is always available to anyone listed in the **DDE Share Name Permissions** dialog box.

Note When a Lookout*Direct* client tries to connect to a DDE server on a computer running NT, it might query the user for a user name and password that are valid on the server computer. In this case, the user password should not be NULL, or the attempt to connect to the server will fail.