Section 1. Introduction

1-1. Overview ................................................................. 1-1
1-2. WDPF Overview ....................................................... 1-2
1-3. OPC Server Overview ............................................... 1-3
1-4. Contents of This Document ............................ 1-5
1-5. Additional Reference Documents .......................... 1-6

Section 2. Getting Started

2-1. Section Overview ..................................................... 2-1
2-2. OPC Requirements .................................................. 2-1
  2-2.1. Hardware Requirements ................................. 2-1
  2-2.2. Software Requirements ............................... 2-1
  2-2.3. License Server Requirements ............................ 2-2
2-3. Setup ................................................................. 2-4
  2-3.1. Setup Procedure ............................................. 2-5
  2-3.2. External Libraries ........................................ 2-6
  2-3.3. Files Installed During Setup ............................ 2-6
  2-3.4. Registry Entries ............................................ 2-8
  2-3.5. Security ...................................................... 2-10
2-4. Un-installation ....................................................... 2-12

Section 3. Using OPC

3-1. Section Overview ..................................................... 3-1
3-2. Command Line Options .......................................... 3-2
3-3. Client Connection and Operation ............................ 3-3
  3-3.1. Information Needed ....................................... 3-3
  3-3.2. Return Values for WDPF OPC Items .................... 3-7
  3-3.3. Update Rate ............................................... 3-7
  3-3.4. OPC Quality ............................................... 3-8
  3-3.5. Automation Interface .................................. 3-9
3-4. OPC Utilities ........................................................ 3-13
  3-4.1. OPC Client ................................................... 3-13
  3-4.2. Basic Operations in OPC Client ....................... 3-14

Glossary

Index
Section 1. Introduction

1-1. Overview

This manual describes the installation, configuration, and operation of the WDPF OLE (Object Linking and Embedding) for Process Control Server, hereafter referred to as the OPC Server.

OPC transfers WDPF process point values between the WDPF Data Highway and software packages known as clients (for example, Matrikon Process X Explorer or Intellution’s iFIX). This enables an engineer to read and write real-time values from a remote computer to the Data Highway.

Note

For more details about OPC, visit the OPC Foundation web site at http://www.opcfoundation.org
1-2. WDPF Overview

The WDPF Highway transmits data elements known as “points” throughout the WDPF system. A WDPF point has a value and various other properties (fields, point name, limits, status, quality, point record type, and ASCII descriptions). There are a number of WDPF point types, including Analog, Digital, and Packed Digital. Point Types are identified by a two-letter mnemonic (listed in Table 3-3).

WDPF points are identified by names that may be up to 16 characters in length. Information about all points is stored in a database file called the System Point Directory (SPD). The SPD also contains a network name related to the system. This network name is used in the forming of OPC tags (see Section 3-3).

The value and properties of a WDPF point are defined by fields which use two-letter abbreviations. Table 1-1 provides examples of WDPF Points (for additional information, refer to “Record Types User’s Guide” (U0-0131)).

<table>
<thead>
<tr>
<th>Point Name</th>
<th>Field Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>AI100</td>
<td>AV</td>
<td>Analog Value of point</td>
</tr>
<tr>
<td>AI101</td>
<td>AS</td>
<td>16-bit integer; status of Analog point – includes point quality</td>
</tr>
<tr>
<td>AI100</td>
<td>PN</td>
<td>ASCII Point Name, (for example, LA100)</td>
</tr>
<tr>
<td>DI101</td>
<td>DS</td>
<td>16-bit integer; status of Digital point - includes Digital value in bit 0</td>
</tr>
<tr>
<td>AI101</td>
<td>HL</td>
<td>Analog high limit</td>
</tr>
<tr>
<td>A101</td>
<td>TB</td>
<td>16-Bit integer; top of Output Scale</td>
</tr>
</tbody>
</table>
1-3. OPC Server Overview

An OPC Server is a program that runs on a Windows NT machine and supports the OPC Interface (illustrated in Figure 1-1). This Interface contains the following:

- **OPC Server (Object)**
  - Each time a client is connected to the server one of these objects is created.
  - The OPC Server object contains the following information about the state of the current connection:
    - List of groups
    - Server status
    - Information about the available items

- **OPC Group**
  - Can be multiple groups
  - Contains a logical organization of data
  - Identified by a user-defined name
  - Groups contain a set of OPC items
  - Items in a group have a common update rate

- **OPC Items (process data points)**
  - Organized into OPC Groups
  - Identified by the vendor (OPC Server) specific syntax (see Section 3-3)
  - Use a common mechanism to read and write data

Access to the OPC Server is through an OPC compliant client program. Multiple clients can connect to a single OPC Server application. Each client connects to a separate instance of an OPC server object within the OPC server application. This means that each client will have its own sets of OPC groups and items. A client may execute either on the same computer as the OPC Server or may connect through a network.
Figure 1-1. OPC Interface Overview
1-4. Contents of This Document

This manual is organized into the following sections:

- **Section 1. Introduction** describes the contents of this document and provides a list of additional reference material.

- **Section 2. Getting Started** describes the requirements, setup, and configuration necessary to begin using the WDPF OPC Server.

- **Section 3. Using OPC** describes how to access OPC compliant programs and user applications.
1-5. Additional Reference Documents

While using this document, it may be helpful to refer to the documents listed in Table 1-2.

Table 1-2. Reference Documents

<table>
<thead>
<tr>
<th>Document Number</th>
<th>Title</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>U0-0131</td>
<td>Record Types User’s Guide</td>
<td>Describes WDPF point record types.</td>
</tr>
<tr>
<td>U0-2482</td>
<td>WESAPI for Microsoft Windows NT User’s Guide</td>
<td>Describes the WESAPI libraries and how to use them.</td>
</tr>
</tbody>
</table>
Section 2. Getting Started

2-1. Section Overview

This section describes the requirements, setup, and configuration needed to get started using the WDPF OPC Server.

2-2. OPC Requirements

Make sure the following requirements are met before attempting to set up the OPC Server:

2-2.1. Hardware Requirements

- Minimum of 200 MHz Pentium II or compatible PC with available ISA slot.
- Server must have a connection to the WDPF Highway (refer to “WESAPI for Microsoft Windows NT User’s Guide” (U0-2482)).
- (Optional) Ethernet network connection to other Windows 95/98/NT machines that will act as OPC Clients to the WDPF OPC Server.

2-2.2. Software Requirements

- Server must be a computer running Windows NT Version 4.0.
  - The minimum system requirements for a Windows NT computer are described in “WESAPI for Microsoft Windows NT User’s Guide” (U0-2482).
- OPC Client Application (for example, Intellution’s iFIX).
- WDPF NT software – which includes IGI device driver and SHC/SPD libraries (wdpfghc.dll).
- Westinghouse License Manager Server must be installed (see Section 2-2.3.).
2-2. OPC Requirements

2-2.3. License Server Requirements

The OPC Server requires a license to operate. If the Westinghouse License Server is not running, the WDPF OPC Server will not allow any data operations.

The OPC Server interacts with a License Server to determine if proper administrative requirements have been met before it can operate. Multiple License Servers can run on a heterogeneous network, but each License Server should contain all of the license file information.

The License Server must be running on the Ethernet network to which the WDPF OPC Server is attached. In order to ensure adequate operation of the WDPF OPC Server, the License Server should be run on the same machine. Thus, if the computer is running both the License Server and the WDPF OPC Server, this will help ensure that a license can be acquired for an operating OPC Server.

The Westinghouse License Server has its own log file to indicate license operations. It also has a control applet to set parameters. Refer to the License Server help files and documentation for more information on the operation of the License Server(s).

Licenses are based on client connections. Each client requires a single connection to an OPC server to operate. There are no licensing restrictions on the number of groups in a client connection or the number of OPC items in a group.

The default installation of the WDPF OPC Server installs the Westinghouse License Server on the same machine as the OPC Server.
License Limits

The OPC Server is licensed based on client connections. The license limits are determined when the software is purchased. The License Server consults a license feature file that has encrypted data about license limits and applicability.

The License Server allows for two types of license limits:

- **Soft limit.** If more OPC connections are licensed than the value of the license limit, a drop fault is generated. However, the License Server will still grant licenses above the soft license limit.

- **Hard limit.** An attempt to exceed the hard limit will also generate a drop fault. The License Server will NOT grant licenses in excess of the hard limit.

Procedure

Use the following procedure to get licenses for an OPC Server:

1. On the License Server, run the program called `wlmadmin.exe`.

2. This program will display the following:
   - Location of the key directory (where the license files go).
   - Computer name.
   - Server Code (an ASCII string).


   Westinghouse will use the “Server Code” to produce the “key” (another ASCII string). Westinghouse will give the Key Directory code to you. Enter `<wx_slm installation directory>/keys` as the “Key Directory.” Normally, `<wx_slm installation directory>` is `C:/Program Files/westinghouse/wx_slm`.

4. Enter the key into the appropriate field on the `wlmadmin.exe` program.

5. Press the Install Key button.

   You will get a confirmation message that the license file has been added to the key directory.
2-3. Setup

Before installing the WDPF OPC Server software, the WESAPI for Windows software must be installed (refer to “WESAPI for Microsoft Windows NT User’s Guide” (U0-2482)). The installation for WESAPI is provided on the same media as the WDPF OPC Server.

The WDPF OPC Server is installed using a standard Windows installation procedure (see Section 2-3.1 for procedure).

**Note**

Installation must be performed while logged in with Administrator privileges.

This installation program performs the following:

- Installs the WDPF OPC software onto the computer that will be the server.
- Installs the License Server.
- Installs various dynamically linked libraries (see Section 2-3.2).
- Sets the location of files in the user-selected directory (see Section 2-3.3).
- Makes entries in the registry (see Section 2-3.4).
- Registers the WDPF OPC Server and related files.

Two major software components are installed by the setup program (see Section 2-3.3):

- WDPF OPC Server Software
- License Server
2-3.1. Setup Procedure

Before installing the WDPF OPC Server software, the WESAPI for Windows software must be installed (refer to “WESAPI for Microsoft Windows NT User’s Guide” (U0-2482)).

Use the following procedure to run the setup.exe program.

1. Insert the WDPF OPC CDROM containing the setup.exe program into the computer that will be the OPC Server.

2. Execute the setup.exe program that is found in the wdpfopc.2.4 directory on the CD.
2-3. Setup

2-3.2. External Libraries

The WDPF OPC Server uses the SHC library routines in `wdpfghc.dll` to access and write WDPF point fields. This is installed during the WESAPI installation.

The following libraries, installed during the setup process, must be accessible to the OPC Server program.

<table>
<thead>
<tr>
<th>Library</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>OPCDAuto.dll</td>
<td>%WINDIR%\system32</td>
</tr>
<tr>
<td>OPCComm_ps.dll</td>
<td>%WINDIR%\system32</td>
</tr>
<tr>
<td>OPCProxy.dll</td>
<td>%WINDIR%\system32</td>
</tr>
</tbody>
</table>

2-3.3. Files Installed During Setup

Two software components are installed by the setup program.

- WDPF OPC Server Software
- License Server

The setup program installs software in the user-selected directory, makes entries into the registry, and registers the WDPF OPC Server and related files. The registration process also makes registry entries; this is what allows OPC clients to actually remotely connect to and retrieve values from the WDPF OPC server.

You must have administrative security privileges on the Windows NT machine in order to run the setup program.

The setup program includes several system dynamically linked libraries (DLL) including a file called `mfc42u.dll`. This DLL may already be present on the hard drive. The setup program will only attempt to overwrite the existing file if the setup program has a newer version. A pop-up dialog box may be presented to give the installer the option to overwrite the existing file. If in doubt, do not overwrite the existing file.

The setup program will install most files under the `C:\wdpf\opc` directory. If desired, this may be altered by the user during installation.

Some configuration and data files are installed under the directory tree specified by the WDPF_HOME environment variable.
These files are installed during setup:

<table>
<thead>
<tr>
<th>Filename</th>
<th>Location</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>WDPFOPC.exe</td>
<td>c:\WDPF\opc\</td>
<td>OPC Server</td>
</tr>
<tr>
<td>OPCEnum.exe</td>
<td>%WINDIR%\system32</td>
<td>OPC Server Enumerator</td>
</tr>
<tr>
<td>OPCClient.exe</td>
<td>c:\WDPF\opc\</td>
<td>Client test program</td>
</tr>
<tr>
<td>mfc42u.dll</td>
<td>%WINDIR%\system32</td>
<td>System DLL</td>
</tr>
<tr>
<td>msvcrtd.dll</td>
<td>%WINDIR%\system32</td>
<td>System DLL</td>
</tr>
<tr>
<td>wx_slmd.dll</td>
<td>%WINDIR%\system32</td>
<td>License client DLL</td>
</tr>
</tbody>
</table>
2-3. Setup

2-3.4. Registry Entries

The WDPF OPC Server uses Microsoft services to communicate across the network. DCOM (Distributed Common Object Model) and its associated security model is the underlying mechanism for an OPC Server and OPC Client to communicate.

The WDPF OPC Server uses DCOM and therefore relies heavily on a number of registry entries for correct configuration and operation. These values are set during program installation and should not be changed.

You can re-register the WDPF OPC Server with DCOM and reset the registry values by running the program from a command line:

C:\> c:\WDPF\opc\programs\WDPFOPC.exe /service

Alternately, WDPF OPC can be set up as a DCOM Server (not an NT service) by using the following command:

C:\> c:\WDPF\opc\programs\WDPFOPC.exe /RegServer

Similarly, you can remove the WDPF OPC Server from DCOM by running:

C:\> c:\WDPF\opc\programs\WDPFOPC.exe /UnregServer

The dynamic link library, OPCProxy.dll, is used to marshall interfaces for OPC clients and seamlessly transmit data over the network. This DLL is registered with DCOM during setup. You can re-register by running this program from a command line:

C:\winnt\system32\> regsvr32 OPCProxy.dll

Note

The current directory must be the same as the actual location of the DLL. If this operation is successful, you should get a confirming message box.

The dynamic link library, OPCDaAuto.dll, implements the automation interface. This allows the OPC Server to be used with Visual Basic, Visual Basic for Applications and Microsoft J++ (Java). This DLL is registered with DCOM during setup. You can re-register by running the program from a command line with this syntax:

C:\winnt\system32\> regsvr32 OPCDaAuto.dll
2-3. Setup

Note

The current directory must be the same as the actual location of the DLL. If this operation is successful, you should get a confirming message box.

The dynamic link library, OPCComn_ps.dll marshals interfaces for OPC Servers. This DLL is registered with DCOM during setup. You can re-register by running the program from a command line with this syntax:

```
c:\winnt\system32\> rgsvr32 OPCComn_ps.dll
```

Note

The current directory must be the same as the actual location of the DLL. If this operation is successful, you should get a confirming message box.

The program OPCEnum.exe is used to provide OPC Server information to remote machines. It is registered with DCOM during installation. You can re-register by running the program from a command line with this syntax:

```
c:\winnit\system 32> OPCEnum_service
```
2-3. Setup

2-3.5. Security

The WDPF OPC Server security is configured with dcomcnfg.exe, a program supplied with the Windows NT operating system.

You will not be able to change DCOM configuration under Windows NT unless you are logged on with Administrator Privileges.

Caution

Changing default values can have far reaching consequences; therefore, changing these values should be carefully considered. Before making changes, record the existing values so that you can return to the original values if your experimentation fails.

The settings shown in Table 2-3 are working values. Your own application may require higher security requirements:

Modifying Security

1. Start the DCOM configuration program. You will see a tabbed dialog box. If the WDPF OPC Server is installed correctly and registered with DCOM, you should see a "WDPF OPC Server" entry in the Applications list on the Application tab.

2. Select the Default Properties Tab. The Default values are described in Table 2-3.

Table 2-3. Default Properties Values

<table>
<thead>
<tr>
<th>Entry</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Enable Distributed COM on this computer</td>
<td>Checked</td>
</tr>
<tr>
<td>Enable COM Internet Services on this computer</td>
<td>N/A</td>
</tr>
<tr>
<td>Default Authentication Level</td>
<td>Connect</td>
</tr>
<tr>
<td>Default Impersonation Level</td>
<td>Identify</td>
</tr>
<tr>
<td>Provide Additional Security for Reference Tracking</td>
<td>N/A</td>
</tr>
</tbody>
</table>
3. Select the Application Tab and click on **WDPF OPC Server**. Click on the Properties button. You should see another tabbed dialog box. These settings are recommended, but site security policies may dictate different settings. These settings are generally unrestricted and will allow any user that can be authenticated on the local machine to access the WDPF OPC Server:

### Table 2-4. Application Values

<table>
<thead>
<tr>
<th>Tab</th>
<th>Entry</th>
<th>Value</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>General</td>
<td>Authentication Level</td>
<td>Default</td>
<td>—</td>
</tr>
<tr>
<td>Location</td>
<td>Run Application on Computer where data is located</td>
<td>Unchecked</td>
<td>—</td>
</tr>
<tr>
<td>Location</td>
<td>Run Application on this Computer</td>
<td>Checked</td>
<td>—</td>
</tr>
<tr>
<td>Location</td>
<td>Run Application on the following computer</td>
<td>Unchecked</td>
<td>—</td>
</tr>
<tr>
<td>Security</td>
<td>Use Default Access Permissions</td>
<td>Select Radio button</td>
<td>Customize this if default permissions are too restrictive.</td>
</tr>
<tr>
<td>Security</td>
<td>Use Default Launch Permissions</td>
<td>Select Radio button</td>
<td>Customize this if default permissions are too restrictive.</td>
</tr>
<tr>
<td>Security</td>
<td>Use Default Configuration Permissions</td>
<td>Select Radio Button</td>
<td>Customize this if default permissions are too restrictive.</td>
</tr>
<tr>
<td>Identity</td>
<td>The Launching User</td>
<td>Select Radio Button</td>
<td>—</td>
</tr>
<tr>
<td>Endpoints</td>
<td>—</td>
<td>Default System Protocols</td>
<td>—</td>
</tr>
</tbody>
</table>

4. You can modify the default launch and access permissions with the controls on the Default Securities tab.

5. Windows NT 3.51/4.0 security is certified to the C2 level. There are no security restrictions for local OPC access to the WDPF OPC server.
2-4. Un-installation

Note

You must have administrative security privileges on the Windows NT drop in order to remove the WDPF OPC server.

1. Disconnect all clients that are using the WDPF OPC Server. This is done from the client machines.

2. Launch the Control Panel from the Start menu bar or My Computer (Figure 2-1).

3. Start the Add/Remove Programs control applet from the Control Panel window (Figure 2-2). Select and remove the entry labeled WDPF OPC Server.
Section 3. Using OPC

3-1. Section Overview

This section describes how to access data in OPC compliant programs and in user applications.

Note

Normal program operation of the WDPF OPC Server is automatic. This program is normally not visible when running. The WDPF OPC Server is an NT service which starts when the system is started.

You can tell if the program is running by using the taskmgr.exe program and looking for an entry, WDPFOPC.exe. The WDPFOPC.exe program depends on the correct operation of WDPF software. Under normal conditions the WDPF Services start automatically when the WDPF NT drop starts up.
3-2. Command Line Options

You can perform some configuration operations with the WDPF OPC Server by running the following program from a command prompt, and by using the command line switches described in Table 3-1.

Table 3-1. WDPFOPC.exe Command Line Switches

<table>
<thead>
<tr>
<th>Command Line Switch</th>
<th>Argument</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>/RegServer</td>
<td>—</td>
<td>Registers the WDPF OPC Server with DCOM.</td>
</tr>
<tr>
<td>/UnRegserver</td>
<td>—</td>
<td>Unregisters the WDPF OPC Server with DCOM.</td>
</tr>
<tr>
<td>/Service</td>
<td>—</td>
<td>Installs WDPFOPC.exe as an NT Service.</td>
</tr>
</tbody>
</table>
3-3. Client Connection and Operation

Note

This section assumes that the user is familiar with the operation and specification of data on a WDPF Highway. For more detail, refer to the “Record Types User’s Guide” (U0-0131).

3-3.1. Information Needed

A client must specify three pieces of information in order to be able to read from or write to an OPC item:

- **Server Name** — Typically, a connection dialog box will enumerate available OPC Servers on a machine. The name of the WDPF OPC Server is “WDPF.OPC” or “WDPF.OPC.n” where “n” is a revision number. To connect to a remote server you must indicate both the Machine Name and the OPC Server Name.

- **Group Name** — After a successful connection, the client must create and add one or more OPC Groups to the OPC Server. This step is sometimes hidden as in the case of the OPC Client.exe utility program. This simple OPC Client program automatically adds a single group to the OPC Server after connection.

- **Item Name** — The client program creates and adds OPC Items to existing OPC Groups. The OPC Items are identified by a text string. The syntax of the text string is unique for each OPC Server type. The WDPF OPC Server supports two types of syntax for OPC Items:
  1. The **First Syntax** requires the point name or fully qualified point name as the OPC Item. The syntax of a fully qualified point name is:

     PointName@NetworkName

     Systems with a single network do not need to specify the network name.

     PointName

     For the first syntax, the point type as indicated by the point name implies a default field name. For example, Analog (AI, AL, AC, AM, AX) and Extended Analog (AB) points have the Analog Value (AV) field as the default field name. Digital (DI, DL, DC, DM, DX) points have bit ø of the Digital Status (DS) as the default field.
2. The **Second Syntax** includes a two-character record type Field Name. The Field Name must be legal for the point record type implied by the Point Name.

   \[ \text{PointName@NetworkName.FieldName} \]

   \[ OR \]

   \[ \text{PointName.FieldName} \]

The Field Name will imply the data type of the returned value. For example, the AV field is a 32-bit floating point value. The RT (Record Type) field is an 8-bit Byte value.

Table 3-2 provides examples of the syntax for Item Names:

**Table 3-2. Item Name Syntax Examples**

<table>
<thead>
<tr>
<th>Point Name</th>
<th>Field Name</th>
<th>OPC Item Specification</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>AI100@Net3</td>
<td>—</td>
<td>AI100@Net3</td>
<td>This is an analog point which implies the “AV” or floating point value field as the field name. This is a fully qualified point name.</td>
</tr>
<tr>
<td>DI101@Net3</td>
<td>—</td>
<td>AI101@Net3</td>
<td>This is a digital point which implies the “DS” or First Status Field. Furthermore, the value bit (Bit 0) is extracted from the data to return a Boolean value.</td>
</tr>
<tr>
<td>AI101@Net1</td>
<td>HL</td>
<td><a href="mailto:AI101@Net1.HL">AI101@Net1.HL</a></td>
<td>This is the high limit field (a floating point value) for a fully qualified point name.</td>
</tr>
<tr>
<td>PB200</td>
<td>A2</td>
<td>PB200.A2</td>
<td>This is a Packet Digital point that returns a 16-bit integer value.</td>
</tr>
<tr>
<td>AI100</td>
<td>HL</td>
<td>AI100.HL</td>
<td>This is a High Limit field of an analog point that returns a 32-bit floating point value.</td>
</tr>
<tr>
<td>DI101</td>
<td>DS</td>
<td>DI101.DS</td>
<td>First Status Word of a digital point. Because the Field Name is explicitly named, the return value is the entire 16-bit integer value.</td>
</tr>
</tbody>
</table>
Table 3-3 lists the Default Fields for Point Record Types:

<table>
<thead>
<tr>
<th>Point Record Type</th>
<th>Default Field</th>
<th>Data Type</th>
<th>Point Record Mnemonics</th>
</tr>
</thead>
<tbody>
<tr>
<td>Analog</td>
<td>AV</td>
<td>Float (VT_R4)</td>
<td>AB, AI, AL, AC, AM, AX</td>
</tr>
<tr>
<td>Digital</td>
<td>DS</td>
<td>Boolean (VT_BOOL)</td>
<td>DI, DL, DC, DM, DX 1</td>
</tr>
<tr>
<td>Packed Digital</td>
<td>AV</td>
<td>long (VT_I4)</td>
<td>PB, PX</td>
</tr>
<tr>
<td>Algorithm (group 1)</td>
<td>CS</td>
<td>Short (VT_I2)</td>
<td>KF, KV, L2, L4, L8, LF, LV, P1, P2, P3 2</td>
</tr>
<tr>
<td>Algorithm (group 2)</td>
<td>SW</td>
<td>Short (VT_I2)</td>
<td>MP, MA, MV 2</td>
</tr>
<tr>
<td>Drop Status</td>
<td>FC</td>
<td>Short (VT_I2)</td>
<td>DU, VU</td>
</tr>
<tr>
<td>Packed Group Alarm</td>
<td>A2</td>
<td>I16 (VT_I2)</td>
<td>BG, BN, BX</td>
</tr>
<tr>
<td>Packed Group</td>
<td>A2</td>
<td>I16 (VT_I2)</td>
<td>GP, GX</td>
</tr>
<tr>
<td>Device</td>
<td>DS</td>
<td>I6 (VT_I2)</td>
<td>VC, VX</td>
</tr>
</tbody>
</table>

1 Special processing is performed for Digital Points. The entire 16 bit DS field is read, the Value bit (BIT 0) is extracted, and the Value bit is returned as a boolean.

2 For some record types, there is no Value field. For these record types, the default field is arbitrarily designated.
Individual Bit Access

Some 16 bit integer fields allow access to individual bits of the field. Refer to “Record Types User’s Guide” (U0-0131) for information about data types of record fields.

To access a single bit in a record field, the following syntax would be used:

```
<point name>.<record field> : <bit #>
```

OR

```
<point name> @ <Network>.<record field>: <bit #>
```

This syntax identifies a single bit in the record field. Multiple bits cannot be accessed. Access the entire field instead. The data type of individual bits is Boolean.

<table>
<thead>
<tr>
<th>Record Type</th>
<th>Bit Number</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Digital Point</td>
<td>D100.DS:15</td>
<td>Accesses time-out bit in status word.</td>
</tr>
<tr>
<td>Packed Group</td>
<td>GP100.A2:4</td>
<td>Accesses bit 4 of the value field.</td>
</tr>
<tr>
<td>Analog</td>
<td>GP100.A2:4</td>
<td>Quality latch bit of analog second status word.</td>
</tr>
<tr>
<td>Drop Status</td>
<td>Drop1.HC:6</td>
<td>Timekeeper bit in drop status DHC status word.</td>
</tr>
</tbody>
</table>
3-3.2. Return Values for WDPF OPC Items

As described in Table 3-3, the record type Field Name determines the basic data type of the return data. OPC Items return a VARIANT data structure. The VARIANT data structure is a union of a large number of simple data types. The basic data types of Fields in a WDPF point map into the VARIANT data structure (Table 3-5) in a straightforward fashion. The exception to this is the ASCII data type. The basic string data type in the VARIANT is called a BSTR. BSTR data types encapsulate the actual string of characters and a length.

<table>
<thead>
<tr>
<th>WDPF Field Type</th>
<th>Variant Type</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>(B) Byte - 8 Bit integer</td>
<td>V_11</td>
<td>No real distinction in WDPF Field Types between signed and unsigned integer types.</td>
</tr>
<tr>
<td>(I16) Integer - 16 Bit Integer</td>
<td>V_12</td>
<td>No real distinction in WDPF Field Types between signed and unsigned integer types.</td>
</tr>
<tr>
<td>(I32) Long Integer - 32 Bit Integer</td>
<td>V_14</td>
<td>No real distinction in WDPF Field Types between signed and unsigned integer types.</td>
</tr>
<tr>
<td>(R32) Floating Point - 32 Bit</td>
<td>V_R4</td>
<td>Double (64 Bit) Floating point values are not supported in WDPF.</td>
</tr>
<tr>
<td>C6 - Six byte ASCII</td>
<td>BSTR</td>
<td>—</td>
</tr>
<tr>
<td>C8 - Eight byte ASCII</td>
<td>BSTR</td>
<td>—</td>
</tr>
<tr>
<td>C16 - Sixteen byte ASCII</td>
<td>BSTR</td>
<td>—</td>
</tr>
<tr>
<td>C30 - Thirty byte ASCII</td>
<td>BSTR</td>
<td>—</td>
</tr>
</tbody>
</table>

The normal rules for reading and writing WDPF points apply. You cannot write to the value field (AV, DS, ...) of a point unless it is originated on the drop. You can modify the other fields such as the HL or high-limit field of an analog point.

3-3.3. Update Rate

When a group is created, the user specifies an interval in milliseconds for updating the data. In OPC, only changes to data are broadcast. If you define the interval of a group as 1000 milliseconds (1 second), then the OPC Server will read the value of each item in the group every 1000 milliseconds. If a particular item has not changed value, then the OPC server will not transmit that value to the client. Only changed item values are sent.

The minimum update rate is 100 milliseconds for WDPF.
3-3. Client Connection and Operation

3-3.4. OPC Quality

OPC has a definition of quality that differs from that of WDPF quality (see Table 3-6). WDPF point quality is associated with the point value. This is the AV field for Analog Points. For fields other than the value fields of a point, the OPC point quality will always be OPC_QUALITY_GOOD, unless the point is timed out or not available. For the point value field of a point, the WDPF point quality maps to the OPC quality.

Table 3-6. WDPF Quality/OPC Quality

<table>
<thead>
<tr>
<th>WDPF Quality</th>
<th>OPC Quality</th>
</tr>
</thead>
<tbody>
<tr>
<td>Good</td>
<td>OPC_QUALITY_GOOD</td>
</tr>
<tr>
<td>Fair</td>
<td>OPC_QUALITY_UNCERTAIN</td>
</tr>
<tr>
<td>Poor</td>
<td>OPC_QUALITY_BAD</td>
</tr>
<tr>
<td>Bad</td>
<td>OPC_QUALITY_BAD</td>
</tr>
<tr>
<td>SHC library call fails</td>
<td>OCP_QUALITY_UNCERTAIN</td>
</tr>
</tbody>
</table>
3-3. Client Connection and Operation

3-3.5. Automation Interface

Note
You can use Microsoft ® Visual Basic, or Visual Basic for Applications and the automation interface to WDPF OPC Server to provide specialized programming tasks. This section assumes that the reader is familiar with Visual Basic programming.

Example
The brief coding example below connects to the WDPF OPC Server, adds a group, adds items, reads values, and writes values.

1. Start up the Visual Basic Program or the Visual Basic for Applications environment for an Microsoft Office Product. Add the OPC Automation Library 2.0 (opcdaauto.dll) to the list of references using the Tools\References menu item.

2. Copy the example subroutine below into a module. Execute the code from the debugger.

Option Explicit

' This is an example of the use of WDPF OPC
' This simple example will have to be modified to be
' used by an Ovation system. The correct machine name
' and correct point names must be inserted in the appropriate
' variables

Sub OPCExample()
    Dim myOPCServer As OPCServer
    Dim myOPCBrowser As OPCBrowser
    Dim myOPCGroup As OPCGroup
    Dim myOPCItem(3) As OPCItem
    Dim OPCItemTemp As OPCItem
    Dim myServerName As String
    Dim myNodeName As String
    Dim myGroupName As String
    Dim myItemName(3) As String

    Dim i As Integer
    Dim lCount As Long
    Dim lClientHandle As Long
    Dim myClientHandles(3) As Long
    Dim myServerHandles() As Long
    Dim myErrors() As Long
Dim lReturn As Long

Dim myValue As Variant
Dim myQuality As Variant
Dim myTimeStamp As Variant

' Modify these as needed for a specific OPC Server/group/item

myServerName = "WDPF.OPC.2"
myNodeName = "KingFisher"
myGroupName = "Kendo"

' This name must definitely be modified
to match a legal name on the WDPF Highway.
The syntax for a fully qualified point names is:
PointName@Network

' ex. 1a006-122@w3

' Create a new server
Set myOPCServer = New OPCServer

' Connect to the WDPF OPC Server
On Error GoTo NoConnect
Call myOPCServer.Connect(myServerName, myNodeName)
Set myOPCBrowser = myOPCServer.CreateBrowser

' Add a group
On Error GoTo NoGroup
Set myOPCGroup = myOPCServer.OPCGroups.Add(myGroupName)

' modify some group values

myOPCGroup.UpdateRate = 500
myGroupName = "Pandora"
myOPCGroup.Name = myGroupName

' add 3 OPC Items
On Error GoTo NoAddItem

' initialize data
myItemName(1) = "la006-122"
myItemName(2) = "da006-122"
myItemName(3) = "ld006-122"
myClientHandles(1) = 2500
myClientHandles(2) = 2501
myClientHandles(3) = 2502
Call myOPCGroup.OPCItems.AddItem(3, _
    myItemName, _
    myClientHandles, _
    myServerHandles, _
    myErrors)

' read the data
For Each OPCItemTemp In myOPCGroup.OPCItems
    Call OPCItemTemp.Read(0, myValue, myQuality, myTimeStamp)
    Debug.Print OPCItemTemp.ItemID & myValue & " quality " & myQuality & " time " & myTimeStamp
Next OPCItemTemp

' write the data
On Error GoTo NoWrite
Set OPCItemTemp = myOPCGroup.OPCItems.GetItem(myServerHandles(1))
OPCItemTemp.Write (OPCItemTemp.Value + 1#)
Set OPCItemTemp = myOPCGroup.OPCItems.GetItem(myServerHandles(2))
OPCItemTemp.Write (OPCItemTemp.Value + 1#)
Set OPCItemTemp = myOPCGroup.OPCItems.GetItem(myServerHandles(3))
OPCItemTemp.Write (Not OPCItemTemp.Value)

For Each OPCItemTemp In myOPCGroup.OPCItems
    OPCItemTemp.Write (OPCItemTemp.Value + 1)
    Call OPCItemTemp.Read(0, myValue, myQuality, myTimeStamp)
    Debug.Print OPCItemTemp.ItemID & myValue & " quality " & myQuality & " time " & myTimeStamp
Next OPCItemTemp

' Leave with the quiet confidence of a job well done

On Error GoTo RemoveItemsFault
lCount = myOPCGroup.OPCItems.Count
Call myOPCGroup.OPCItems.Remove(lCount, myServerHandles, myErrors)
myOPCServer.OPCGroups.Remove (myGroupName)
myOPCServer.Disconnect
Exit Sub

NoWrite:
    Debug.Print "Write Item " & " returns error; " & Err.Number & "; (" & Err.Description & "); ""
    Resume Next

NoAddItem:
    Debug.Print "Add Item " & myOPCItem(i).ItemID & " returns error " & Err.Number & " (" & Err.Description & ")"
Resume Next

RemoveItemsFault:
    Debug.Print "Remove Items returns error " & Err.Number & " (" & Err.Description & ")"
    myOPCServer.OPCGroups.Remove (myGroupName)
    myOPCServer.Disconnect
    Exit Sub

NoGroup:
    MsgBox "Can't add group " & myGroupName
    myOPCServer.Disconnect
    Exit Sub

NoConnect:
    ' fail to connect
    MsgBox "Connect to " & myNodeName & " Server " & myServerName
    Exit Sub

End Sub
3-4. OPC Utilities

3-4.1. OPC Client

**OPC Client.exe** (Figure 3-1) is a simple OPC Client program that can be used to check connections to an OPC Server. Start the program from the command line or Windows Explorer. The file is loaded into `C:/wdpf/opc` by default during installation. You should see this window:

![Figure 3-1. OPC Client.exe Window](image)

**Figure 3-1. OPC Client.exe Window**
3-4.2. Basic Operations in OPC Client

1. Click on **OPC/Connect** menu to access the Connection dialog box.

![Select OPC Server Dialog Box](image)

2. A list of OPC servers on the local machine should appear (Figure 3-2). You can attempt to connect to Version 1.0 or Version 2.0 OPC Servers. If the WDPF OPC Server exists on the local machine, there should be a “WDPF.OPC” or “WDPF.OPC.n” entry (where “n” is a revision number).

3. To check connections to a remote machine, enter a Windows NT machine name in the Server Node text box and click the **Refresh List** dialog box. This can be time consuming. If you know that the server exists on the remote machine, enter the correct OPC Server and machine name in the text boxes and attempt to connect.

4. Double click on an OPC Server name from the list of available servers, or manually enter a name in the Server Name text box. Click **OK** to attempt to connect. Remote connections sometimes take a while to complete. If there is an error, a message box will appear with an indication of the error.
5. If there is a successful connection, the program automatically creates an OPC Group. You can view and modify some group properties, but this simple program does not support multiple groups.

6. Using the **OPC Client.exe** window (Figure 3-1) click on **OPC/Item/Add** menu item to access an OPC dialog box. Enter a legal WDPF OPC item name in the **Item Name** text box (Figure 3-3).

   OR

   To get the default value of a point, enter the POINT NAME in the **Item Name** text box. Use the “**PointName.FieldName**” format (for example, “ai100.av”) to retrieve a specific field. Leave the **Access Path** text box empty. Click **Add Item** to add the OPC Item to the OPC Group.

![Figure 3-3. Add Item Display Box](image)

If browsing is enabled, then you can use a tree view of the points to select a particular point name (Figure 3-3).

7. If the point name is legal and the WDPF drop software is operating correctly, the item should appear in the OPC Client window with the current value.

8. Use the **OPC/Disconnect** menu item in the **OPC Client.exe** window to remove all OPC Items in the group, delete the group, and disconnect from the OPC Server.
Index

A
ASCII data type  3-7

B
Boolean  3-6
BSTR  3-7

C
client connection and operation  3-3
information needed  3-3
  group name  3-3
  item name  3-3
  syntax  3-4
  server name  3-3
coding example  3-9
command line options  3-2
command line switches  3-2
  RegServer  3-2
  Service  3-2
  UnRegserver  3-2
command prompt  3-2

D
DCOM  2-8
Default values  2-10
dynamic link library
  mfc42u.dll  2-7
  msvcr7.dll  2-7
  OPCComm_ps.dll  2-6
  OPCDAAuto.dll  2-6, 2-8
  OPCProxy.dll  2-6, 2-8
  wx_slmd.dll  2-7

F
First Syntax  3-3

G
getting started  2-1

I
individual bit access  3-6
Intellution iFix  2-1

K
key  2-3
Key code  2-3
key directory  2-3

L
libraries  2-6
  SHC  2-6
License Server  2-2

M
Microsoft J++  2-8
Microsoft Visual Basic  2-8, 3-9
Microsoft Visual Basic for Applications  2-8,
  3-9
Multiple License Servers  2-2

O
OLE  1-1
OPC  2-4
  interface
    items  1-3
  using  3-1
    command line options  3-2
    command line switches  3-2
    configuration operations  3-2
OPC Automation Library  3-9
OPC Foundation  1-1
OPC quality  3-8
OPC Server
  getting started  2-1
  interface  1-3
    group  1-3
  overview  1-1, 1-3
  removing from DCOM  2-8
  requirements  2-1
    hardware  2-1
    software  2-1
    OPC Client application  2-1
      Windows NT 4.0  2-1
  setup  2-4
OPC Utilities  3-13
OPC Client  3-13
  basic operations  3-14
  OPC Client.exe  3-13

P
Point Record Types
  default fields  3-5

R
reference documents  1-6
registry
  re-register 2-8
requirements 2-1
return values 3-7

S
Second Syntax 3-4
  field name 3-4
Security 2-10, 2-11
  modifying 2-10
  working values 2-10
Server Code 2-3
setup 2-4
  files installed 2-6
  procedure 2-5
  security 2-11
setup.exe program 2-5
System Point Directory (SPD) 1-2

T
taskmgr.exe 3-1

U
Un-installation 2-12
update rate 3-7

V
VARIANT 3-7
variant types 3-7

W
WDPF 1-2
  highway 1-2
  points 1-2
    properties 1-2
    types 1-2
      analog 1-2
      digital 1-2
      packed digital 1-2
wladmin.exe program 2-3