PVI DDE Server

Contents (6.13.2005)

1 PVI DDE Server

1.1 General

The PVI DDE Server is a PVI application which allows other Windows applications (clients) such as Excel or self–made applications (Visual BASIC, Visual C) to access PVI variable object data using DDE.
The PVI DDE Server can only carry out write and read services for PVI variable objects. Other services or services of other PVI objects cannot be carried out.

1.2 Operating Modes of the PVI DDE Server

Two different operating modes can be selected by entering the respective program argument when starting the PVI DDE Server:

- PVI operating mode
- NET2000 operating mode

1.2.1 PVI Operating Mode

If no program argument or an invalid program argument is entered, the standard PVI mode is used. All PVI lines can be used in this mode.

1.2.2 NET2000 Operating Mode

If the program argument "net2dde" is entered, the DDE Server is operated in NET2000 mode. The text "(net2dde)" is shown in the header for the DDE Server. In this mode, only the PVI NET2000 line can be accessed. The PVI DDE Server is then compatible to the NET2000 DDE Server for Windows 3.11 (16 Bit Version).

Differences to NET2000 DDE Server for Windows 3.11:

1. The function "Read Net2000 Object Information" is not available in the PVI DDE Server.
2. The variable data accessed in the BRBINARY Clipboard Format is always transferred in the correct byte order. "Swapping" the bytes is not required.

Note:
In Windows, program arguments can be defined in the properties.

The NET2000 operating mode is only intended for applications which were written for the NET2000 DDE Server for Windows 3.11. New applications should work in PVI operating mode.

1.3 Device, Station and Variable Lists

DDE communication with a DDE Server is handled using 2 levels – topic and item. PVI recognizes more than 2 levels (object hierarchy), therefore the PVI process objects on the DDE Server are handled using 3 lists – device, station and variable list.

1.3.1 Device List

Each entry in the device list defines a PVI line object and a PVI device object. If a second device object is required for the same line, another entry must be made in the device list. When doing this, the same name (case sensitive) is entered for the line object. This sets up a PVI line object with 2 device objects on the PVI Manager. The device list is set on the DDE Server using a dialog window and stored in the initialization file "PviDDE.ini" (Menu: Options – device Parameters).

1.3.2 Station List

Each entry in the station list defines a PVI station object. Each station entry is also assigned an entry in the device list. In this way, a PVI line object and a PVI device object is also defined.

The station list is defined in the configuration file. A symbolic name can be assigned which allows DDE to access the station entry as a topic.

1.3.3 Variable List

Each entry in the variable list defines a PVI variable object. An optional CPU and task object can be defined. Additionally, each variable entry is assigned an entry in the station list.
The variable list is defined in the configuration file. A symbolic name can be assigned which allows DDE to access the variable entry as an item.

1.3.3.1 Diagram
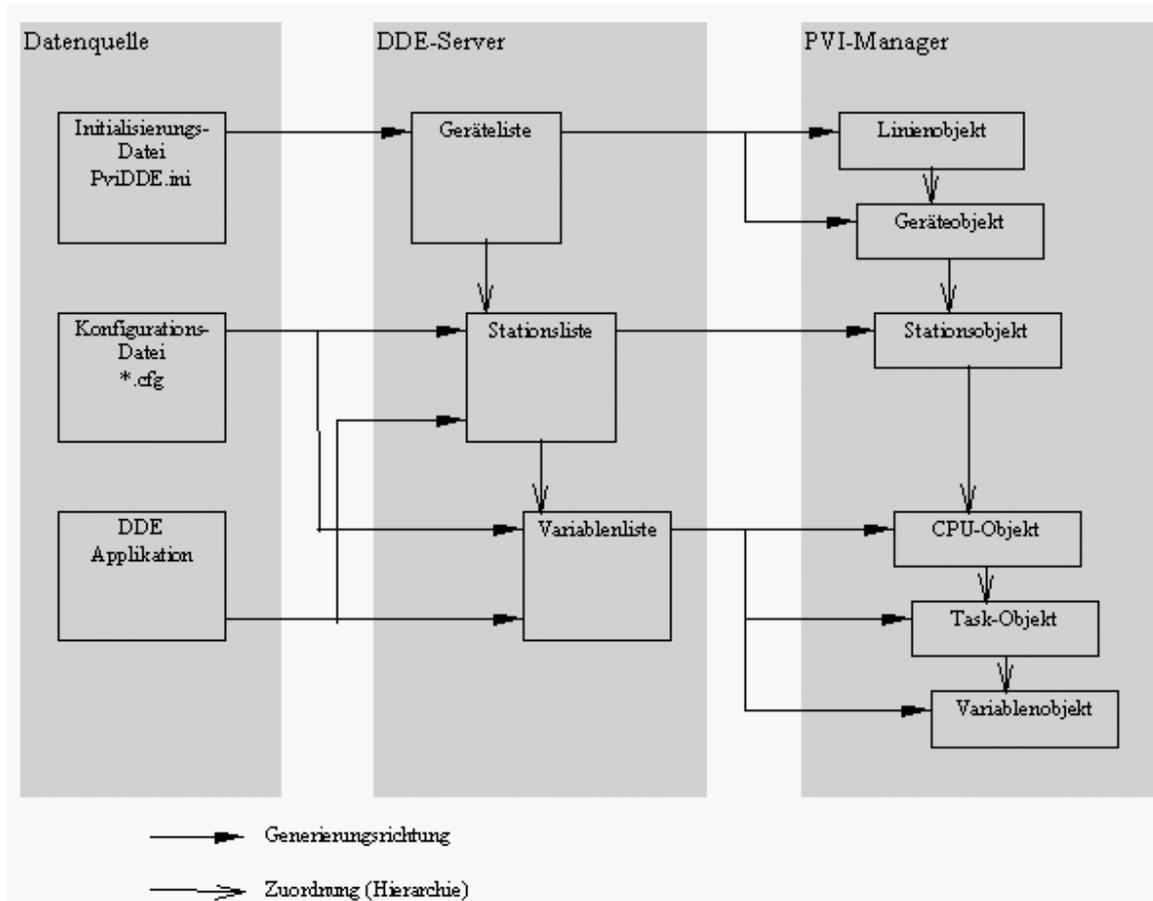
1.4 Operating the PVI DDE Server

1.4.1 Opening the Configuration

The menu function "File / Open" can be used to select a configuration file and set the configuration data contained in it. Additionally, the DDE service name (pvidde) is registered (XTYP_REGISTER).
After restarting the DDE Server, the configuration last selected is automatically opened.

1.4.2 Closing the Configuration

The menu function "File / Close" can be used to close an open configuration. The registration for the DDE service name is cancelled (XTYP_UNREGISTER).

1.4.3 Calling the configuration file editor

The menu function "File / Editor" starts the editor assigned to the file (file extension – Windows function). If the file is not assigned to a program, the Windows text editor "notepad" is used.
The argument given to the editor is the path name of the last selected (opened) configuration file.
Edited configuration data is not set automatically after exiting the editor. The configuration has to be opened again.

1.4.4 Display Global Status

The global status display is activated with the menu function "View / Global".

This display shows the following information:		
Stations, variables:	Total:	Shows the number of stations and variables used in the DDE Server.
	Actv	Shows the number of stations with active variables or the number of variables with active dynamic update "Advise Loop".
	Error:	Shows the number of stations or variables that have an error.
Configuration file:	Path of the selected configuration file.	
Service cycle:	The set or current refresh cycle. If there is no display, either no configuration is open or the connection to the PVI Manager is broken	
Last error:	Shows the last Net2000 error: $\#<ErrorNumber> <StationName> \| <VariableName>$ If the error refers to all variables for a station, the character "*" is shown instead of the variable name. If the error refers to all stations * | * is shown instead of the text. The error display can be cleared with the menu function "View / Clear Error Display".	

1.4.5 Display Station List

The station list display is activated with the menu function "View / Station List". The station list contains all stations defined on the DDE Server .

This display shows the following information:		
Status:	Shows the current status of the station:	
	L ...	Status = Link, connection to the PVI Manager is established.
	E ...	Status = Error, station error.
Error:	Error number (if error has occurred).	
PVs:	Number of process variables assigned.	
Device:	Symbolic name of the device used on the DDE Server.	
Station:	Symbolic name of the station on the DDE Server. The name of the station object on the PVI Manager is shown in brackets "(...)" if it is different than the symbolic name on the DDE Server.	

1.4.6 Display Process Variable List

The process variable list display is activated with the menu function "View / Variable List". The list contains all variables defined in the DDE Server. The display is sorted according to station name and process variable name.

This display shows the following information:	
Type:	PVI data type of the process variable.
Length:	Length of the process variable in bytes

Status:		Shows the current status of the process variable:
	L ...	Status = Link, connection to the PVI Manager is established.
	R ...	Status = Ready, process variable is identified, PV data is valid.
	E ...	Status = Error, process variable error.
	A ...	Status = Active, process variable is active (dynamic data update).
Attr.:		Attribute for the process variable:
	r ...	Read attribute.
	w ...	Write attribute.
Refresh:		Refresh time for the PV data.
Station | P		Symbolic name of the station and the process variable on the DDE Server. The name of the variable object on the PVI Manager is shown in brackets "(...)" if it is different than the symbolic name on the DDE Server.

1.4.7 PV Data Monitor

A PV data monitor window can be opened for individual variables in the process variable list (double click with the mouse or mark the line and call menu function "View / PV Data Monitor"). The current process variable data can be shown with the PV data monitor. Up to 32 PV data monitor windows can be open at the same time.

This display shows the following information:	
Advise:	Advise counter. The counter shows the number of set "Advise Loops" (dynamic update).
Value:	Value of the process variable data (if it can be shown) or error number (PV error).

1.4.8 Display Error List

The error list display is activated with the menu function "View / Error List". The error list shows the last 200 PVI and DDEML errors. The error list can be cleared with the menu function "View / Clear Error Display".

1.4.9 Format of the Error Display

Error *<Number>*: *<ErrorNumber>* *<StationName>* | *<VariableName>*

<Date> *<Time>* >> *<ErrorText>* (*<Function>*)

***

<Number>:	Incremented number for error.
<ErrorNumber>:	PVI (#...) or DDEML error number (0x....).
<StationName>:	Name of the station.
<VariableName>:	Name of the process variable (visualization name). If the error refers to all variables for a station, the character "*" is shown instead of the variable name.
<Date> *<Time>*:	Time stamp.

|	
<ErrorText>:	Description of error.
<Function>:	Shows the function in which the error occurred.

1.4.10 Setting Device Parameters

The dialog window used to define the device parameters is opened with the menu function "Options / Device Parameters". The device parameters set become active immediately after closing the window with "OK".
In the first dialog window (title: "Device List"), the symbolic device list for the PVI DDE Server (maximum 32 entries possible) can be edited (see "Device, Station, and Variable Lists").
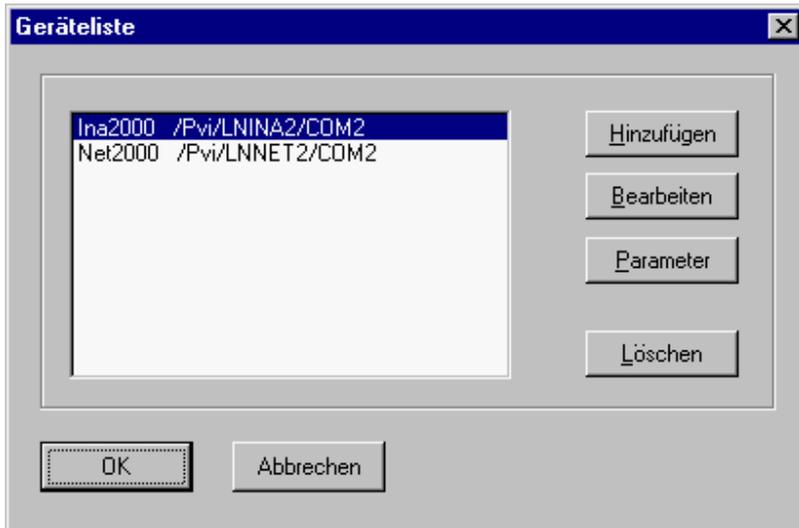
Functions in the "Device List" dialog box:	
Add:	Inserts new entry in the device list.
Edit:	Edits selected entry in the device list.
Delete:	Deletes selected entry in the device list.
Parameters:	Edits PVI device parameters for selected entry in the device list.

In the second dialog window (title: "Device"), an entry for the symbolic device list for the PVI DDE Server can be edited (see section "Device, Station, and Variable Lists"). This dialog window is opened with the functions "Add" or "Edit".
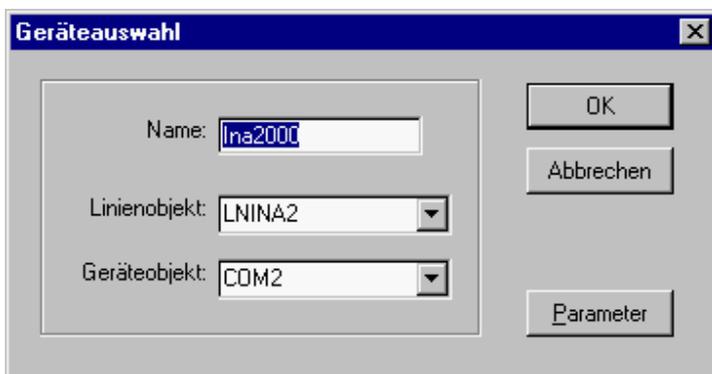
| Parameters in the "Device" dialog box:	
Name:	Symbolic name of the device on the DDE Server.
Line object:	Name of the PVI line object.

	If the respective line object is already set on the PVI Manager (e.g: by another PVI application), the name can of the object can be entered. If the line object is set by the DDE Server, the name of the line DLL (entry without ".dll") must be given as object name. Examples: "LnNet2" for Net2000 line, "LnIna2" for INA2000 line, etc.
Device object:	Name of the PVI device object. Examples: "Com1", "Ls251", etc.
Functions in the "Device" dialog box:	
Parameters:	Edit PVI device parameters.

In the third dialog window (Title: "Device Parameter"), the connection description can be entered for the PVI device object in string format. The entries for the device parameters depend on the line used and are described in the respective line documentation (section "Process objects", subsection "Device object").
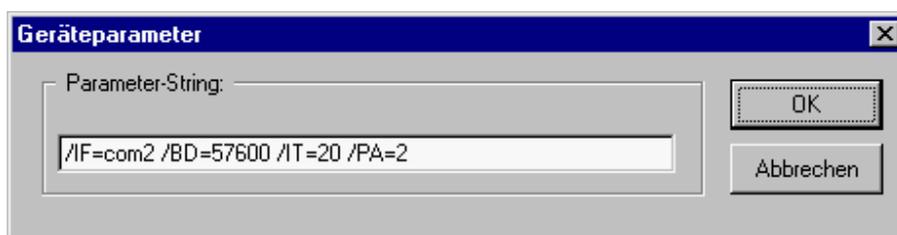
1.4.11
Setting Server Parameters

The dialog window used to set server parameters is opened with the menu function "Options / Server Parameter". The defined server parameters become active immediately after closing the window with "OK".
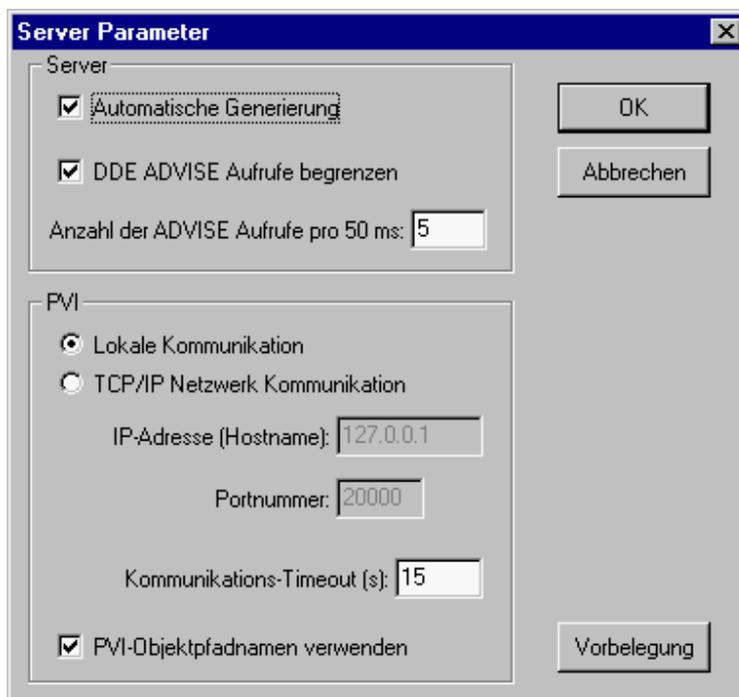
1.4.12
Server Parameters

Automatic generation:

☐ off: The DDE Server only handles the stations and process variables entered in the configuration file.

☒ on: The DDE Server automatically sets up all station and process variables which are not available when accessing the DDE interface. Default values are used for parameters and optional names.

Limit DDE ADVISE messages:

☐ off: The DDE Server immediately sends the respective DDE ADVISE message to the client (application) when data changes for an active process variable.

☒ on: If DDE ADVISE messages are triggered at the same time, they are sent to the client (application) over a longer time frame.
The maximum number of DDE ADVISE messages per timer tick (approx. 50 msec) can be defined (1 to 255).

1.4.13 PVI Parameters

Communication:

⊙ Local communication:	The DDE server and PVI manager are located on the same PC. Communication with the PVI manager is carried out directly via shared memory.
⊙ Network communication:	The DDE server and PVI manager are not located on the same PC. Communication occurs via TCP/IP network. The TCP/IP parameter IP address or host name and port number must be set for this type of communication. The parameters correspond to the PVI parameters IP and PN. (see function PviInitialize for further information).

Note: The "network communication" can be operated when the DDE server and PVI manager are located on the same PC if the respective TCP/IP parameters are defined. This type of communication takes longer than local communication via shared memory.

1.4.13.1 Communication Timeout

Defines the life sign monitor time–span (in seconds) for communication between the PVI DDE server and the PVI Manager. Standard values for this parameter are 5sec to 20sec, depending on the load on the computer and also the type of communication used. For more information, see PVI User Documentation.
Note: The timeout value should not be set too large because connection interruptions or termination of the PVI manager are only detected by the PVI DDE Server after the time out has expired.

1.4.13.2 Use PVI Object Path Names

☐ off: The entire object path is used as PVI object name.

☒ on: PVI object path names are used (by default).

1.5 Configuration File

The project specific visualization data (global parameters, stations, process variables) is defined in the configuration file. This file is evaluated by the DDE Server when the configuration is opened (after starting the program or using the menu function "File / Open"). If a configuration file is not available or if individual parameters are not defined, default values are used by the DDE Server. The configuration file is an ASCII file and can be edited with any Windows text editor (e.g. NOTEPAD).

The configuration is separated into 3 sections:

Global parameters	Cycle time

Stations	Definition of stations (symbolic names).
Process variables	Definition of variables (symbolic names).

The individual parameters are differentiated in the configuration file using defined keywords. Entries for stations and process variables are made using the colon character ':'. An entry for a global parameter, a station or a process variable is terminated with the character ';' or at the end of the line.
All characters between a '*' and the end of the line are interpreted as a comment.
If syntax characters (e.g. * : #, etc.) or spaces are to be used in a station name, a process variable name, or a parameter assignment, the entry must be made using with ("...") quotation marks (e.g. ERR="E#";).

Syntax of the configuration file:

(All entries shown with *[...]* are optional.)

Global parameters:	*<Parameter> = <Value> [;]*
<Parameters>	Keyword for global parameters.
TI	Time base for process variable refresh cycle in msec: 50 .. 60000, Default: 250 msec.
Stations:	*[<Name>]* : "#*[<DeviceName>/]<StationName>[{<PVI>}]"* *[, <Parameter> = <Value>] [;]*
<Name>	Symbolic name for the station (maximum 32 characters). The station (topic) is defined in DDE applications using the name given here.
<DeviceName>	Symbolic device name. Assigns the station an entry in the device list. If the symbolic device name is not given, the first entry in the device list is assigned.
<StationName>	Name of the PVI station object.
<PVI>	Connection description for the PVI station object. If "{*<PVI>*}" is not given, the name of the PVI station object is used as connection description. The entry is sent to the PVI Manager as CD=*<ConnectionDescription>*.
<Parameter>	Keyword for station parameters.
ERR	Text message for damaged process variables (only valid in clipboard format "TEXT"), Default: no message.
RF	Refresh factor: 0 .. 255, Default: 1. The value given here is used as default for the process variable parameter RF.
AT	Attribute for the process variable: r / w, Default: rw. The value given here is used as default for the process variable parameter AT.
Process variables:	*[<Name>]* : "*[<CPUName>[{<PVI>}]/[<TaskName>[{<PVI>}]/]]* *<VariableName>[{<PVI>}]"[, <Parameter> = <Value>] [;]*
<Name>	Symbolic name for the variables (maximum 128 characters). The data point (item) in DDE applications is defined using the name given here.
<CPUName>	Name of the PVI CPU object. If the name is not given, the PVI variable object is set up directly under the PVI station object. The line used determines if a CPU object is required
<TaskName>	Name of the PVI task object. If the name is not given, the PVI variable object is set up directly under the PVI station object or under the PVI CPU object (if given). The line used determines if a task object is required
<VariableName>	Name of the PVI variable object.

<PVI>	Connection description for the PVI object. If "{*<PVI>*}" is not given, the name of the respective PVI object is used as connection description. If the character '!' precedes the connection description, the PVI Event attribute is set and variable is used as PLC event variable (as long as the line used supports PLC event variables). The entry is sent to the PVI Manager as CD=*<ConnectionDescription>*.
<Parameter>	Keyword for process variable parameters.
RF	Refresh factor: 0 .. 255, Default: Station parameter RF. The refresh cycle time for the process variable is calculated by multiplying the refresh factor by the time base (global parameter TI). The process variable data on the PLC is polled using the refresh cycle time. If 0 is given as refresh factor, process variable data is not refreshed cyclically (e.g. for constants).
AT	Attribute for the process variable: r / w, Default: Station parameter AT. r ... "read access" allowed for process variable data. w ... "write access" allowed for process variable data.
DA	Name of the station assigned. Default: The last station given.

Example:

```
TI=200

S1: "#Net2000/St1{127}", ERR="E1="

:temp, RF=5

:stand, RF=50

MaxTemp: max_temp, RF=0

MinTemp: min_temp, RF=0

S2: "#Ina2000/St0", ERR="E2="

Temperature:"CPU1{/SP=/temp/az }/task1/temp", RF=5

"last change": timer, DA=plc1, AT=r, RF=1
```

1.6 DDE Interface

Communication between applications and the DDE Server is made with the help of the DDEML (Dynamic Data Exchange Manager Library) from Windows. According to the definition, the DDE Server is always operated as a server and all applications are operated as clients.
DDE communication is normally initiated by the client which sends a message presently in memory to all DDE Servers (not only PVI DDE Servers). The ID for the data connection given with the message is determined by the DDE Server. After establishing the data connection, the data provided by the DDE Server (PVI DDE Server process variables) can be read. The ID for the data connection and the data is made up of three strings, the application, the topic and the item.

1.6.1 Application, topic and item

Application, topic and item are defined as follows for the PVI DDE Server:

|	
Application:	Server name (PVIDDE or compatible mode NET2DDE).
Topic:	Symbolic station name.
Item:	Symbolic variable name (name of the variable in the DDE Server).

The entries topic and item on the DDE Server are case sensitive. This is necessary because the process variable name (name of the variable on the PLC) and the visualization name (name of the variable on the DDE Server) cannot be differentiated between when using automatic configuration. In order to prevent errors, process

variable names, visualization names and station names should always be entered without using constraints of capitalization.

1.6.2 Dynamic Update

The data for active process variables is read cyclically by the DDE Server using the configured refresh time (polling). When the data changes, the server sends a message to all connected DDE applications. These applications can then request process variable data from the server.

Diagram:
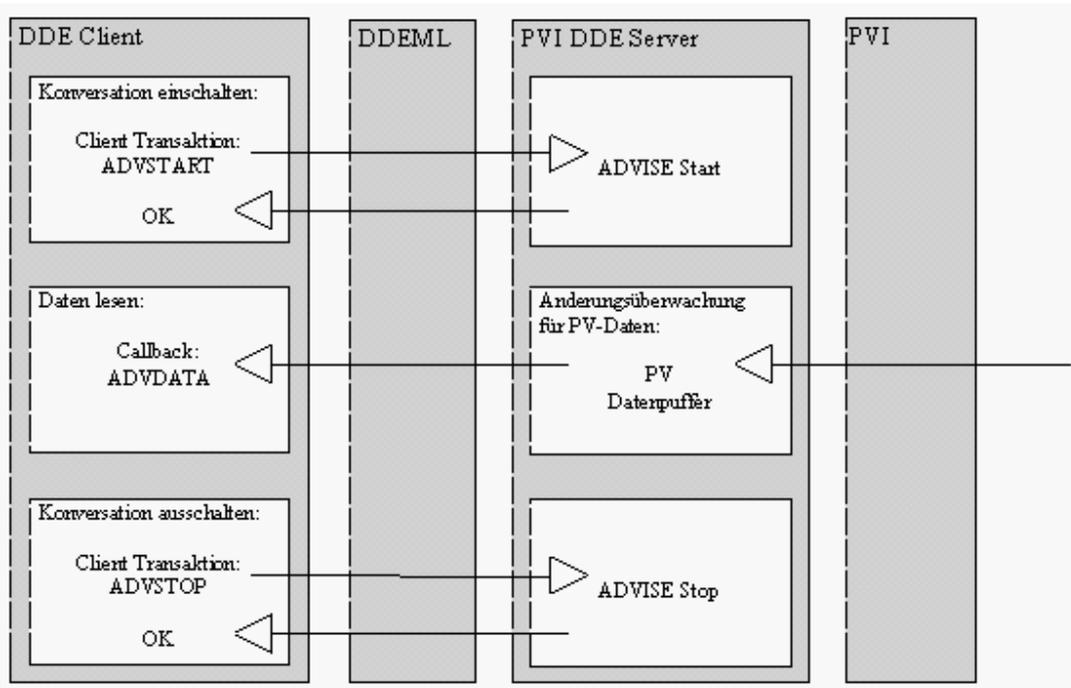
The DDE Server only sends the message "Data changed" for active process variables. In order to make a process variable "active", the application for the client transaction ADVSTART (starts an advise loop) is carried out for the desired process variable.
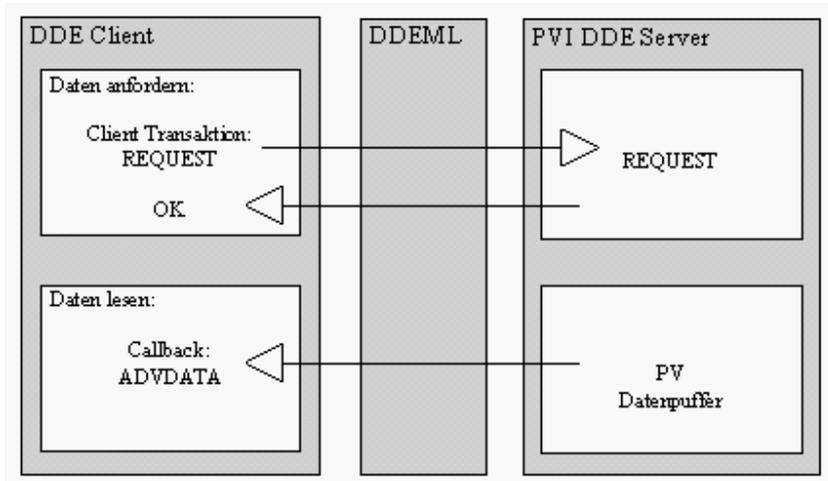
If automatic configuration is active (see Server Parameters) and the process variable is not available when the call is made, the DDE Server makes the setting automatically. For the process variable parameters (refresh factor and attribute), the default values are used. The process variable name is identical to the visualization name.

If an active process variable is no longer required in the application (e.g: after the page is changed or the program is interrupted), the application has to change the status to "not active". This is done by the client transaction ADVSTOP (ends an advise loop).

1.6.3 Reading Process Variable Data

Reading process variable data in an application normally takes place as a reaction to the message "Data changed" (see "Dynamic update"). Basically, process variable data can be read at any time (client transaction REQUEST). Take note that data from process variables which are not active cannot be current because it is not read cyclically by the PLC.

Diagram:

The client transaction REQUEST (request PV data) can be used to read data without using dynamic updates ("Advise Loop"). If automatic configuration is active (see Server parameters) and the process variable is not available when the call is made, the DDE Server makes the setting automatically. However, no valid PV data is available immediately after the setting is made. Therefore, the first REQUEST is always handled as an "illegal read access" in this case. Solution: Enter process variables in the configuration file or make REQUEST transactions for all required process variables in the initialization phase of the application and ignore the response data.

If a process variable has an error, an illegal read access occurs or read access causes an error, depending on the clipboard format used and the station parameter ERR, the error is transferred in the response data as follows:

Process variable has an error:

Clipboard format:	Parameter "ERR=E# given:	Parameter "ERR" not given:
CF_TEXT:	"E#xxx" xxx ... error code	String with length 0.
BRBINARY:	Word 1 is not 0 (error code), Word 2 is 0 (data length).	

Illegal read access:

Clipboard format:	
CF_TEXT:	String with length 0.
BRBINARY:	Word 1 equal to 0 (error code), Word 2 equal to 0 (data length).

Causes:
Read access denied (PV attribute AT).
Invalid clipboard format (e.g.: read structure data in clipboard format CF_TEXT).
PV data not valid (object identification not yet made).
Read access causes error:

Read access denied on DDE level (DDE error):

Causes:
Unknown clipboard format.
Unknown station or process variable (Server Parameter: AUTO=OFF).
Internal error such as not enough free memory, etc.

1.6.4 Writing Process Variable Data

Writing process variable data with the application is carried out using the client transaction POKE.
After the write procedure, the data is transferred from the DDE Server to the PLC (station) via PVI, a read procedure is carried out again and a response is sent to the application in the form of the message "Data

changes". An additional write procedure (write echo) allows the application to recognize possible errors caused by the write procedure. However, this mechanism is only effective for active process variables. With process variables which are not active, the DDE Server does not carry out a write echo.

The order in which the data for different process variables is sent does not depend on the order of the write procedures (POKE transactions). If a certain order is to be used, a separate handshake mechanism must be used.

Diagram:
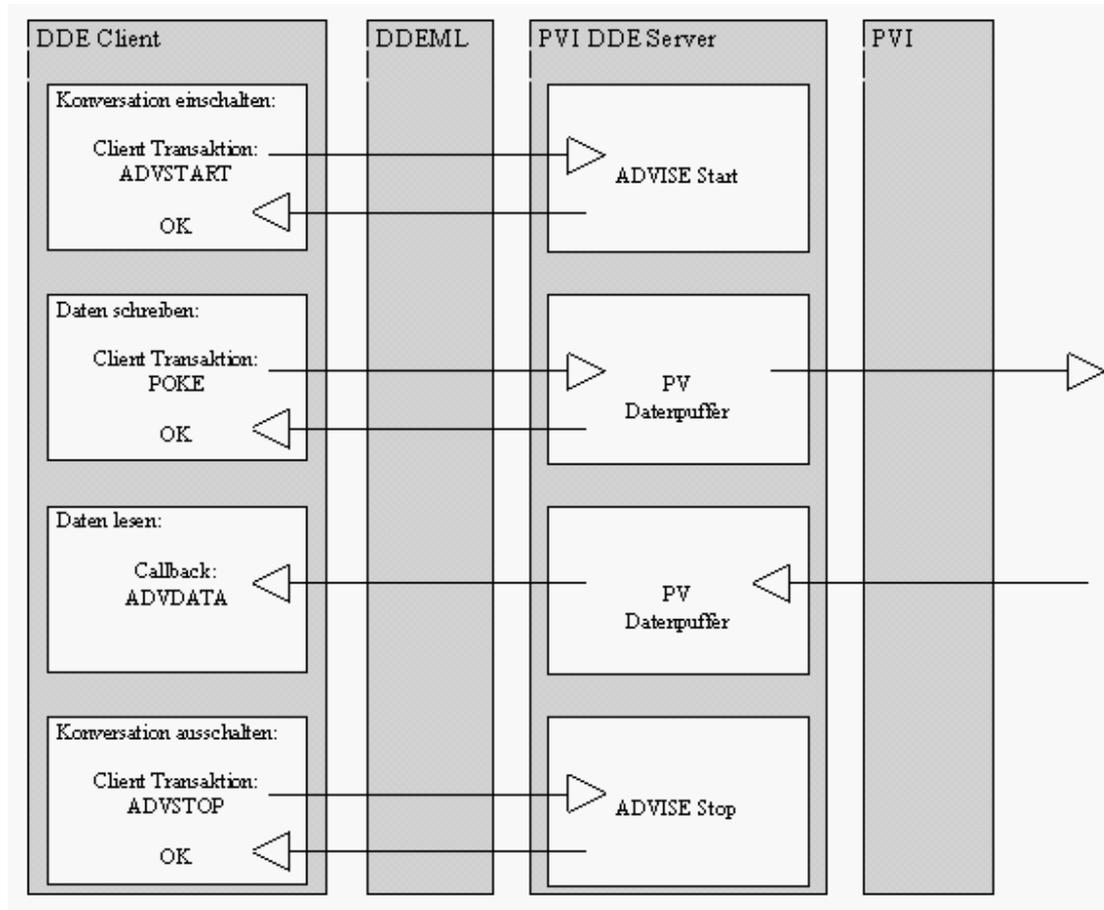
If automatic configurable is active (see Server parameters) and the process variables is not available when the POKE transaction is made, the DDE Server sets up the process variable automatically. However, the DDE Server cannot handle write procedures immediately after setup. In this case, the POKE transaction shows the server status "BUSY" (DDE error). Help: see section "Reading process variable data".

1.6.5 Clipboard Formats

The clipboard format defines the organization of the PV data for data transfer between the DDE Server and the application.

The DDE Server supports the following formats:

TEXT:	Text format.
BRBINARY:	Binary format (Original Byte Order).

1.6.6 Text Format

The clipboard format TEXT is a Windows format (format code: CF_TEXT) and can therefore be used for standard applications (e.g: Excel).

With this format, only data from single variables, arrays elements and structure elements, as well as text strings (array with data type BYTE) can be transferred. Data from single variables, arrays elements and structure elements are transferred in the form of decimal text strings.

1.6.7 BRBINARY format

Clipboard format BRBINARY is not a Windows format. The DDE Server sets this clipboard format up under the name "BRBINARY". Applications that use this format have to set the format code using the Windows SDK function "RegisterClipboardFormat".

With this format, single variables, arrays or structure elements, and also total arrays or structures can be transferred. The PV data is transferred in binary form.

Read format:	
Error code	2 bytes
Length of PV data	2 bytes
PV data	n bytes
Write format:	
PV data	n bytes

When reading process variable data, additional information concerning the process variable data is provided by the DDE Server:

Error code:
If an error occurs, the current error status is sent.

Length of the PV data:
Number of PV data bytes sent. If an error occurs, 0 is given.