Diadem-D User Guide

(Release 2.0 - 7/6/2002)

Distributed Applications Debug Monitor for Delphi 4 .. 7

Purpose

The purpose of Diadem-D is to display debugging information of applications executed on different PC's in a local debugging window. A server collects the debug textes of all other PC's and displays them time-sorted in it's window. Optionally this information can also be displayed on a remote viewer which e.g. is connected to the server via a dial-in connection.This makes it possible to monitor the messages on a customers plant while factory acceptance test in your office. To output debug information can be done with a minimum of effort.

Installation

- Install the Diadem-D client on all PC's where your applications want to output debug information. Set the name of the server if you want the information appear also in the Diadem-D serverr window.
- Install the Diadem-D server on one PC where you want to collect the debug information of the clients installed. This PC should be connected via LAN with the clients.
- Optionally install the Diadem-D remote viewer on the PC on which you want to watch the debug information displayed in the Diadem-D server window. This PC can be connected via LAN or by dial-in connection with the server. Set the name of the server on which you installed the Diadem-D server program.
- Install the Diadem-D-API on all those PC's where you develop applications with Delphi.

How To

Producing output in a local Diadem-D window is done in five steps:

- Insert a Uses-statement for the unit 'DiademIF' (the interface) into the Unit that has to output debug text.
- Insert WriteLn - statements anywhere you need debugging output.
- Start DiademC (the Diadem-D client).
- Set the name of the server on which you intend to install the Diadem-D server (Menue - Preferences - Server). Instead of the server name you can use it's IP-address.
- Check the option 'Send to server'.

Now your application can output debugging text in the window of DiademC. If necessary, you can start DiademC on multiple computers. Every DiademC window displays the debug text of the local application. DiademC can be used parallel by many applications.

Optionally collecting the distributed debug information displayed by different Diadem-D clients:

- Start DiademS (the Diadem-D server) on one PC only.

Now the the server (DiademS) should collect the displayed debug messages of all running clients and display them time-sorted.

Optionally show the debug information displayed by the Diadem-D server on a remote PC:

- Start DiademRV (the Diadem-D remote viewer) on one PC only.

- Set the name of the PC on which the Diadem-D server is installed.
  (Menue - Preferences - Server). Instead of the server name you can use it's IP-address.

- Connect to the Diadem-D server (e.g. by Dial-In-Connection).

Now the the remote viewer (DiademRV) should get all messages displayed by Diadem-D server and display them time-sorted.

That's all.

Contents of this description

A. Setup process

Depending on the selected options the setup program copies different files to different folders.

Setup for a PC to be a Diadem-D client:

The setup program copies the DLL with the functions needed for the testee to output text in Diadem-D client window into the Windows system directory. The program DiademC and the setup program (SetupDIA) are copied to the installation directory. DiademC is entered into the windows start menu.

Setup for a PC to be a Diadem-D client + a Diadem-D server:

The setup program copies the DLL with the functions needed for the testee to output text in Diadem-D client window into the Windows system directory. The programs DiademC, DiademS and the setup program (SetupDIA) are copied to the installation directory. DiademC and DiademS are entered into the windows start menu.

Setup for a PC to be a Diadem-D server:

The program DiademS and the setup program (SetupDIA) are copied to the installation directory. DiademS is entered into the windows start menu. For applications with a high processor load, client and server should not be installed on the same PC!

Setting the optional Delphi version, additionally the API unit (DiademIF) is copied to the Delphi library directory and the example program is copied into the Diadem-D directory. The selected programs are installed in the Delphi tools menue.

For uninstalling start the setup program (SetupDIA.exe) in the Diadem-D directory, it will automatically delete all files it has copied before.

Diadem-D can be used with or without the Delphi IDE. Output of Debug information is very fast, every information transferred from the testee to Diadem-D is buffered in a shared memory. If the information is not processed yet, a message is posted to Diadem-D to start output. If the buffersize (64 entries) would be exceeded, Diadem-D is forced to process the buffer, the application waits until done so. Normally Diadem-D works when the testee is waiting for a message, so that the testee is not slowed down because of displaying error messages.

If automatic start is checked when installing, the application that uses the Diadem-D client (DiademC) automatically starts DiademC at initialization. This feature can also be activated later on in DiademC itself.

B. Simple Usage - using the main channel

When a Uses-statement for the Unit DiademIF is inserted into the DPR-file of the project, output with the WriteLn-statement can be produced if the Diadem-D client (DiademC) is started (e.g. by the Delphi tools menu).

Example:

```
WriteLn(‘This is a debug output on the standard output device (main channel) with Diadem-D’);
```

Output without naming a file-variable uses the standard output device. This device is redefined by the unit DiademIF and is the **Main Channel**.

Output in DebugDelphi’s window would be:

```
-M- [date ][time ]This is a debug output on the standard output device (main channel) with Diadem-D
```

-M- stands for main channel.

If date and time are displayed depends on the options set in DiademC.

If DiademC is not started, there is no output, also no error message. By starting and ending DiademC you decide, if you want to see messages or not.

C. Advanced usage - using the minor channels

By using ‘DiademIF’ in your application, automatically the standard output device for your application is redefined. Also 9 extra channels, **Minor Channels**, are automatically defined. The extra channels can be used by naming one of the defined file variables DBChannel in the WriteLn - statement.

Example:

```
WriteLn(DBChannel[1], ‘This is an output on channel 1 (minor channel) with Diadem-D’);
```

Output in the Diadem-D client window would be:

```
-1- [date ][time ]This is an output on channel 1 (minor channel) with Diadem-D
```

The unit DiademIF automatically creates 9 minor channels to DiademC. These channels can be used for different purposes. Every channel can be switched on and off seperately on-line. If you want to use minor channels, you need to places the ‘USES DiademIF’ statement into every unit in which you want to display something in the Diadem-D client window.

A good idea would be to build error classes in your application. Depending on the settings in the Diadem-D programs you can display these classes or not.

D. Using Diadem-D in muti threaded applications

All output done with WriteLn is done by means of the Delphi text-file device driver. This driver uses file variables for the output. As every other data too, these variables have to be protected against simultaneous access by different threads. To do this, you have different possibilities, e.g. use critical sections or mutexes. Critical sections afford an EnterCriticalSection call before the WriteLn and a LeaveCriticalSection call after the WriteLn and is a lot of coding effort.

Easier ways are:

- To use one of the standard minor channels per thread.
- To define auxiliary file variables as described in the next chapter.

E. Defining auxiliary output files

The Diadem-D interface (DiademIF.PAS) offers calls to open and close additional files for your output with WriteLn. Of course the output is not stored in a file but transfered to DebugDelphi.

```
PROCEDURE OpenAuxDebugOutput( VAR AuxChannel : Text; ChannelNo : Integer);

PROCEDURE CloseAuxDebugOutput(VAR AuxChannel : Text );

// ChannelNo must have a value between 0 and 9. (0 = Main channel / 1..9 = Minor channels)
```

Example:

```
VAR
  MyOutput : Text;
BEGIN
  OpenAuxDebugOutput(MyOutput, 5);  // Open file and use channel number 5
  WriteLn(MyChannel, ‘Write something on MyChannel using channel 5’);
  CloseAuxDebugOutput(MyOutput);
```

By this way you can open as much file variables as you like (or as Windows accepts).

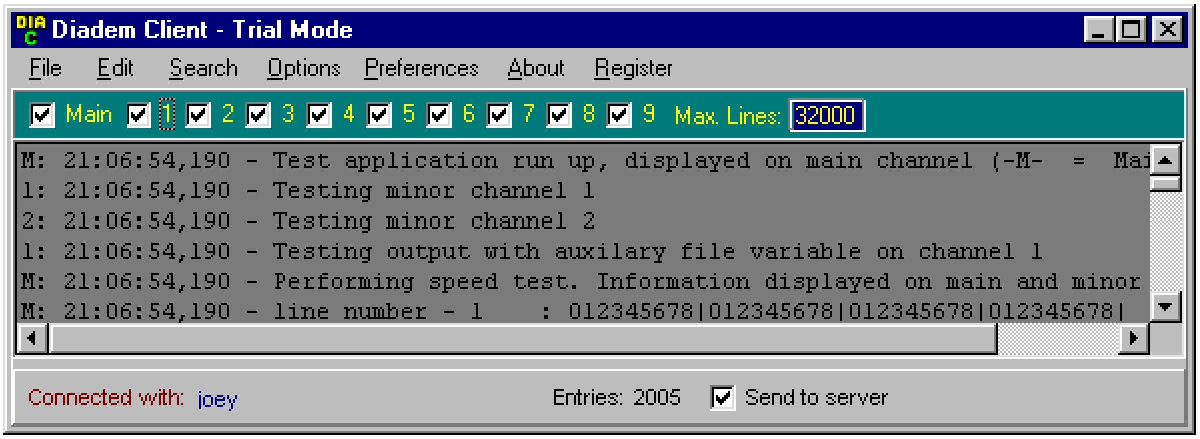
A good idea would be to let every thread use an own file variable and use the standard channels in the Main Thread.

F. Using Write and WriteLn

Instead of producing a line in the Diadem-D client window by a single WriteLn statement you also can use several Write - statement plus a WriteLn - statement. The output in the Diadem-D client window appears after the WriteLn - statement.

The total text length may not exceed 255 characters.

Example for a DiademC window:


Diadem Client - Trial Mode
File
Edit
Search
Options
Preferences
About
Register
☑ Main ☑ 1 ☑ 2 ☑ 3 ☑ 4 ☑ 5 ☑ 6 ☑ 7 ☑ 8 ☑ 9
Max. Lines: 32000

```
M: 21:06:54,190 - Test application run up, displayed on main channel (-M-   =  Mai
1: 21:06:54,190 - Testing minor channel 1
2: 21:06:54,190 - Testing minor channel 2
1: 21:06:54,190 - Testing output with auxilary file variable on channel 1
M: 21:06:54,190 - Performing speed test. Information displayed on main and minor
M: 21:06:54,190 - line number - 1     : 012345678|012345678|012345678|012345678|
```

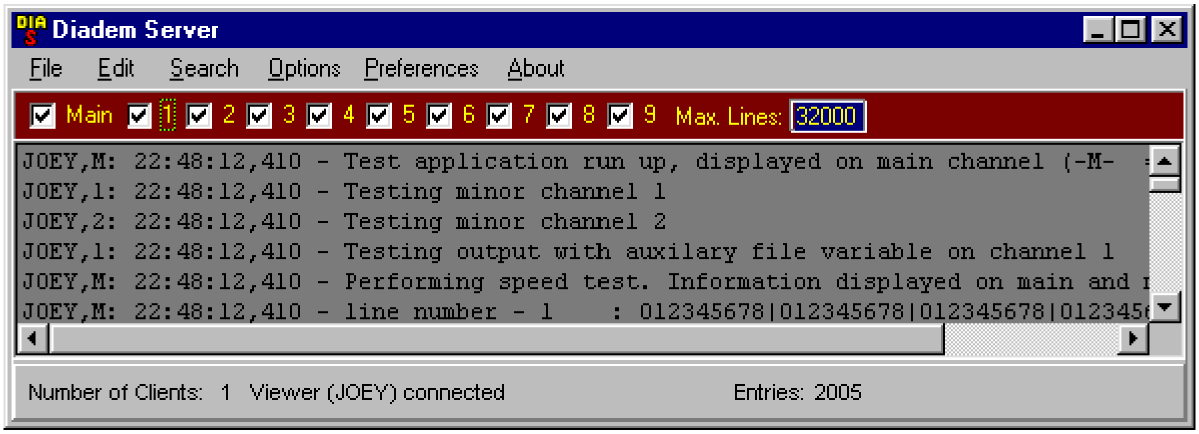
Connected with: joey   Entries: 2005   ☑ Send to server

G. Using the server

Just start the server program of Diadem-D (DiademS) on one of the PC's connected with those PC's on wich the client programs of Diadem-D (DiademC) are executed. If DiademC ist configured with the correct server name and the checkbox 'Send to server' is set, DiademC tries to open a connection to the server. If this fails, every 5 seconds this try is repeated. When the connection is succesful, all currently displayed messages are sent to the server. As this is done by every client, the server displays all messages of all clients time sorted. For a correct sorting of the messages, the hardware clocks of the client PC's should be synchronized. Synchronizing the times is not done by Diadem-D. It might be implemented in a later release.

If DiademS is started on the same PC as DiademC, it is unnescessary to set the server name for the local DiademC programm.

Example for a DiademS window:



```
JOEY,M: 22:48:12,410 - Test application run up, displayed on main channel (-M-
JOEY,1: 22:48:12,410 - Testing minor channel 1
JOEY,2: 22:48:12,410 - Testing minor channel 2
JOEY,1: 22:48:12,410 - Testing output with auxilary file variable on channel 1
JOEY,M: 22:48:12,410 - Performing speed test. Information displayed on main and
JOEY,M: 22:48:12,410 - line number - 1     : 012345678|012345678|012345678|012345
```

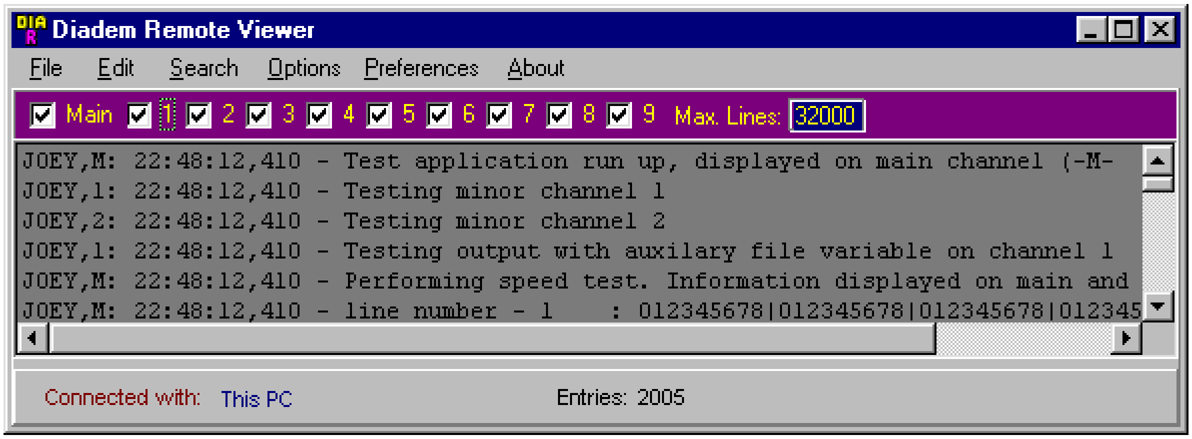
Number of Clients: 1 Viewer (JOEY) connected   Entries: 2005

In this example DiademC and DiademS are executed on the same PC.

H. Using the remote viewer

Just start the viewer program of Diadem-D (DiademRV) on a remote PC. Set the server name to that PC which runs DiademS, alternatively the IP-address can be used. Diadem RV cyclically tries to open a connection to the Diade-D server. When the connection is successful, all currently displayed messages are sent to DiademRV by the server. The messagages are displayed time-sorted. DiademRV can e.g. be connected via Dial-In to the Diadem-D server.

With the remote viewer e.g. the appearing messages on a customers plant can be monitored. The Diadem-D server shows on the bottom of it's form that the connection to the viewer is active.



☑ Main ☑ 1 ☑ 2 ☑ 3 ☑ 4 ☑ 5 ☑ 6 ☑ 7 ☑ 8 ☑ 9 Max. Lines: 32000

```
JOEY,M: 22:48:12,410 - Test application run up, displayed on main channel (-M-
JOEY,1: 22:48:12,410 - Testing minor channel 1
JOEY,2: 22:48:12,410 - Testing minor channel 2
JOEY,1: 22:48:12,410 - Testing output with auxilary file variable on channel 1
JOEY,M: 22:48:12,410 - Performing speed test. Information displayed on main and
JOEY,M: 22:48:12,410 - line number - 1    : 012345678|012345678|012345678|012345
```

Connected with: This PC

Entries: 2005

In this example DiademRV is executed on the save PC like DiademS.

I. Limitation of use

- Length of the strings to be displayed: max. 255 characters (Diadem-D adds channel / Date / Time, the server and the viewer program add computer name of the client who sent the data).

- Maximum of stored and displayed lines in a Diadem-D window: 32000.

- A Diadem-D server can collect the messages of 16 clients.

J. Registration

A registration key can be ordered via ShareIt registration service (http://www.shareit.com). After entering the registration you can use Diadem-D for an unlimited period.

K. Built-in functions

Diadem-D has o lot of features:

- Saving all lines to a file,
- Copying, deleting or printig selected lines (copying to clipboard),
- Deleting all lines,
- Search of text,
- Switch date on and off,
- Automatic start at application start (Diadem-D client only),
- Starting with a minimized window,
- Keep the window on top,
- Using a pseudonym instead of the computer name or IP-address when transferring data to the server (for companies that use cryptic names for their PC's),
- Set automatic scrolling on and off,
- Select a display font and
- Set the window colour.

L. History

Version 1.0 : 4/02	First release
Version 1.1 : 4/02	Setup program bug fixed (Entry into start menu did not work for Client on Windows NT/2000/XP)
	Autostart option in the Client was not saved
	Registration neaded to be done twice
Version 1.2 : 5/02	Bugfix: Displaying strings with more than 127 characters caused rande check error.
Version 2.0 : 7/02	Adaption to Delphi 7.