



# Analysis and Report Generation

## Worldwide Technical Support and Product Information

ni.com

### National Instruments Corporate Headquarters

11500 North Mopac Expressway      Austin, Texas 78759-3504      USA      Tel: 001 512 794 0100

### Worldwide Offices

Canada (Calgary) 403 274 9391, Canada (Montreal) 514 288 5722, Canada (Ottawa) 613 233 5949,  
Canada (Québec) 514 694 8521, Canada (Toronto) 905 785 0085, China (Shanghai) 021 6555 7838,  
China (ShenZhen) 0755 3904939, Czech Republic/Slovakia 2 24235774, Denmark 45 76 26 00,  
Finland 09 725 725 11, France 01 48 14 24 24, Germany 089 741 31 30, Greece 30 1 42 96 427,  
Hong Kong 2645 3186, India 80 535 5406, Israel 03 6393737, Italy 02 413091, Japan 03 5472 2970,  
Korea 02 596 7456, Malaysia 603 9596711, Mexico 5 280 7625, Netherlands 0348 433466,  
New Zealand 09 914 0488, Norway 32 27 73 00, Poland 0 22 528 94 06, Portugal 351 210 311 210,  
Russia 095 2387139, Singapore 2265886, Slovenia/Croatia 634 254200, South Africa 11 805 8197,  
Spain 91 640 0085, Sweden 08 587 895 00, Switzerland 056 200 51 51, Taiwan 02 2528 7227,  
United Kingdom 01635 523545

For further support information, see the *Technical Support Resources* chapter. To comment on the documentation, send e-mail to [techpubs@ni.com](mailto:techpubs@ni.com).

Copyright © 1995–2001 National Instruments (Ireland) Limited. All rights reserved.

# Important Information

## Warranty

The media on which you receive National Instruments software are warranted not to fail to execute programming instructions, due to defects in materials and workmanship, for a period of 90 days from date of shipment, as evidenced by receipts or other documentation. National Instruments will, at its option, repair or replace software media that do not execute programming instructions if National Instruments receives notice of such defects during the warranty period. National Instruments does not warrant that the operation of the software shall be uninterrupted or error free.

A Return Material Authorization (RMA) number must be obtained from the factory and clearly marked on the outside of the package before any equipment will be accepted for warranty work. National Instruments will pay the shipping costs of returning to the owner parts which are covered by warranty.

National Instruments believes that the information in this document is accurate. The document has been carefully reviewed for technical accuracy. In the event that technical or typographical errors exist, National Instruments reserves the right to make changes to subsequent editions of this document without prior notice to holders of this edition. The reader should consult National Instruments if errors are suspected. In no event shall National Instruments be liable for any damages arising out of or related to this document or the information contained in it.

EXCEPT AS SPECIFIED HEREIN, NATIONAL INSTRUMENTS MAKES NO WARRANTIES, EXPRESS OR IMPLIED, AND SPECIFICALLY DISCLAIMS ANY WARRANTY OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. CUSTOMER'S RIGHT TO RECOVER DAMAGES CAUSED BY FAULT OR NEGLIGENCE ON THE PART OF NATIONAL INSTRUMENTS SHALL BE LIMITED TO THE AMOUNT THEREFORE PAID BY THE CUSTOMER. NATIONAL INSTRUMENTS WILL NOT BE LIABLE FOR DAMAGES RESULTING FROM LOSS OF DATA, PROFITS, USE OF PRODUCTS, OR INCIDENTAL OR CONSEQUENTIAL DAMAGES, EVEN IF ADVISED OF THE POSSIBILITY THEREOF. This limitation of the liability of National Instruments will apply regardless of the form of action, whether in contract or tort, including negligence. Any action against National Instruments must be brought within one year after the cause of action accrues. National Instruments shall not be liable for any delay in performance due to causes beyond its reasonable control. The warranty provided herein does not cover damages, defects, malfunctions, or service failures caused by owner's failure to follow the National Instruments installation, operation, or maintenance instructions; owner's modification of the product; owner's abuse, misuse, or negligent acts; and power failure or surges, fire, flood, accident, actions of third parties, or other events outside reasonable control.

## Copyright

Under the copyright laws, this publication may not be reproduced or transmitted in any form, electronic or mechanical, including photocopying, recording, storing in an information retrieval system, or translating, in whole or in part, without the prior written consent of National Instruments Corporation.

## Trademarks

ComponentWorks™, DAQPad™, DataSocket™, DIAdem™, HiQ™, HiQ-Script™, IVI™, LabVIEW™, National Instruments™, NI™, ni.com™, NI-DAQ™, and SCXI™ are trademarks of National Instruments Corporation.

Product and company names mentioned herein are trademarks or trade names of their respective companies.

## WARNING REGARDING USE OF NATIONAL INSTRUMENTS PRODUCTS

(1) NATIONAL INSTRUMENTS PRODUCTS ARE NOT DESIGNED WITH COMPONENTS AND TESTING FOR A LEVEL OF RELIABILITY SUITABLE FOR USE IN OR IN CONNECTION WITH SURGICAL IMPLANTS OR AS CRITICAL COMPONENTS IN ANY LIFE SUPPORT SYSTEMS WHOSE FAILURE TO PERFORM CAN REASONABLY BE EXPECTED TO CAUSE SIGNIFICANT INJURY TO A HUMAN.

(2) IN ANY APPLICATION, INCLUDING THE ABOVE, RELIABILITY OF OPERATION OF THE SOFTWARE PRODUCTS CAN BE IMPAIRED BY ADVERSE FACTORS, INCLUDING BUT NOT LIMITED TO FLUCTUATIONS IN ELECTRICAL POWER SUPPLY, COMPUTER HARDWARE MALFUNCTIONS, COMPUTER OPERATING SYSTEM SOFTWARE FITNESS, FITNESS OF COMPILERS AND DEVELOPMENT SOFTWARE USED TO DEVELOP AN APPLICATION, INSTALLATION ERRORS, SOFTWARE AND HARDWARE COMPATIBILITY PROBLEMS, MALFUNCTIONS OR FAILURES OF ELECTRONIC MONITORING OR CONTROL DEVICES, TRANSIENT FAILURES OF ELECTRONIC SYSTEMS (HARDWARE AND/OR SOFTWARE), UNANTICIPATED USES OR MISUSES, OR ERRORS ON THE PART OF THE USER OR APPLICATIONS DESIGNER (ADVERSE FACTORS SUCH AS THESE ARE HEREAFTER COLLECTIVELY TERMED "SYSTEM FAILURES"). ANY APPLICATION WHERE A SYSTEM FAILURE WOULD CREATE A RISK OF HARM TO PROPERTY OR PERSONS (INCLUDING THE RISK OF BODILY INJURY AND DEATH) SHOULD NOT BE RELIANT SOLELY UPON ONE FORM OF ELECTRONIC SYSTEM DUE TO THE RISK OF SYSTEM FAILURE. TO AVOID DAMAGE, INJURY, OR DEATH, THE USER OR APPLICATION DESIGNER MUST TAKE REASONABLY PRUDENT STEPS TO PROTECT AGAINST SYSTEM FAILURES, INCLUDING BUT NOT LIMITED TO BACK-UP OR SHUT DOWN MECHANISMS. BECAUSE EACH END-USER SYSTEM IS CUSTOMIZED AND DIFFERS FROM NATIONAL INSTRUMENTS' TESTING PLATFORMS AND BECAUSE A USER OR APPLICATION DESIGNER MAY USE NATIONAL INSTRUMENTS PRODUCTS IN COMBINATION WITH OTHER PRODUCTS IN A MANNER NOT EVALUATED OR CONTEMPLATED BY NATIONAL INSTRUMENTS, THE USER OR APPLICATION DESIGNER IS ULTIMATELY RESPONSIBLE FOR VERIFYING AND VALIDATING THE SUITABILITY OF NATIONAL INSTRUMENTS PRODUCTS WHENEVER NATIONAL INSTRUMENTS PRODUCTS ARE INCORPORATED IN A SYSTEM OR APPLICATION, INCLUDING, WITHOUT LIMITATION, THE APPROPRIATE DESIGN, PROCESS AND SAFETY LEVEL OF SUCH SYSTEM OR APPLICATION.



# Contents

<b>About This Manual</b>	<b>ix</b>
Conventions	ix
<b>1 DIAdem Overview</b>	<b>1-1</b>
<b>1.1 The DIAdem Screen</b>	<b>1-1</b>
<b>1.2 The DIAdem Help</b>	<b>1-3</b>
<b>1.3 Settings</b>	<b>1-4</b>
Directory Structure	1-5
Saving and Loading Settings	1-6
Installing and Operating DIAdem in a PC-Network	1-6
<b>1.4 The GPI Interface</b>	<b>1-7</b>
<b>2 DIAdem-DATA: Processing and Managing Data</b>	<b>2-1</b>
The Data Area	2-2
Data Area Display	2-3
<b>2.2 Channel Overview</b>	<b>2-4</b>
<b>2.3 Channel Contents</b>	<b>2-5</b>
<b>2.4 Channel Properties</b>	<b>2-6</b>
<b>2.5 The Data Navigator</b>	<b>2-7</b>
<b>3 DIAdem-VIEW: Viewing, Measuring, and Editing Data</b>	<b>3-1</b>
Layout: Choosing a Split Screen	3-2
<b>3.2 Axis Systems: Displaying Data in Curves</b>	<b>3-2</b>
Individual Scaling of Axes	3-3
Curve legend	3-4

<b>3.3</b>	<b>Cursor: Viewing and Editing Graphs</b> _____	<b>3-5</b>
	Measuring a Graph	3-6
	Zooming and scrolling	3-7
	Cut and Print	3-8
<b>3.4</b>	<b>Documenting Evaluations</b> _____	<b>3-9</b>

## **4 DIAdem-CALC: Analyzing and Evaluating Data** \_ **4-1**

<b>4.1</b>	<b>Performing Standard Calculations</b> _____	<b>4-2</b>
	An Example: Differentiation	4-2
	Viewing Results in DIAdem-VIEW	4-3
	Example 2: Curve Smoothing	4-3
<b>4.2</b>	<b>Calculator: User-Defined Formulas</b> _____	<b>4-4</b>
	Viewing Results in DIAdem-DATA	4-6

## **5 DIAdem-GRAPH: Graphic Representation of Data** \_\_\_\_\_ **5-1**

	Image Template: The pre-defined Layout	5-2
<b>5.2</b>	<b>Arrange Objects: Move, Resize, Align</b> _____	<b>5-3</b>
	Alignment functions: Arranging Objects	5-4
	Quick Display: Sketch mode and axes without curves	5-4
<b>5.3</b>	<b>Axis Systems: Displaying Data in Graphs</b> _____	<b>5-5</b>
	Subaxes: Different Value Ranges	5-7
	Graph Legends: Labeling Curves	5-8
<b>5.4</b>	<b>Tables: Displaying Numeric Data</b> _____	<b>5-9</b>
<b>5.5</b>	<b>Texts: Labels and Comments</b> _____	<b>5-10</b>
	The Text Editor	5-11
	Text objects: Continuous Text Individually Formatted	5-12
<b>5.6</b>	<b>Illustrate: Embedding Graphics and Logos</b> _____	<b>5-14</b>
<b>5.7</b>	<b>Printing Diagrams and Saving Templates</b> _____	<b>5-15</b>

<b>6 DIAdem-AUTO: Automated Analysis and Report Generation</b>	<b>6-1</b>
Starting Autosequences at the Press of a-Button	6-2
<b>6.2 The DIAdem-AUTO Workspace</b>	<b>6-2</b>
File Overview	6-3
The Autosequence Editor	6-4
VBScript in DIAdem	6-4
<b>6.3 Teach-In-Mode (Learning by Doing)</b>	<b>6-5</b>
Teach-In-Mode Results	6-8
Autosequence Structure and Syntax	6-9
<b>6.4 The Script Debugger: Finding and Eliminating Errors</b>	<b>6-10</b>
Debugging Tools	6-11
The Information Area	6-13
<b>6.5 Customizing DIAdem</b>	<b>6-14</b>
DIAdem Standard Dialog: Interactive File Selection	6-14
Defining User Dialogs	6-15
Defining and activating variables	6-20
Channel Conversion	6-21
Manipulating Graphics Layouts	6-22
Listing for Example AUTO3	6-23
<b>6.6 Additional Autosequence Features</b>	<b>6-26</b>
Important Commands and Variables	6-26
Control Structures: Loops and Requests	6-27
Working with Channel Designators	6-28
Serial Evaluations	6-28
Interaction Mode	6-30
Full-Screen Mode	6-30
User Management	6-30
<b>6.7 Interface Functions</b>	<b>6-32</b>
OLE Interface	6-32
ODBC/SQL Interface for Database Connection	6-32
<b>6.8 Autosequence Converter from AUT to VBS</b>	<b>6-33</b>
Formulas	6-34
Procedures	6-34
For...do Control Structure	6-35
Remarks and Special Features	6-35

<b>7 Index</b>	<b>7-1</b>
<b>8 Technical Support Resources</b>	<b>8-1</b>
Web Support	8-1
NI Developer Zone	8-1
Customer Education	8-1
System Integration	8-1
Worldwide Support	8-2



# About This Manual

The *DIAdem Analysis and Report Generation Manual* describes the concept of DIAdem and how to use it. The chapters describe the DIAdem devices for data management, data viewing, mathematical evaluation, documentation, and automated analysis and report generation.

Use this manual to learn about DIAdem, including the DIAdem user interface with all its functions. This manual does not include specific information about each toolbar, function bar, menu, context menu, and dialog box. Refer to the *DIAdem Help* for more information about these items and for detailed instructions for using DIAdem features and for building specific applications.

The *DIAdem Analysis and Report Generation Manual* is also available in Portable Document Format (PDF) on the DIAdem-CD. You can access the PDF manual from the CD interface, without installing the Acrobat Reader.

## Conventions

The following conventions appear in this manual:

*Italic printed paragraph*

*Italic printed paragraphs are step-by-step actions, which you can execute directly with DIAdem on your PC. These examples show you how to work with the DIAdem device.*

Small font

Paragraphs printed in small font are remarks and comments added to the topic which has been described in the above paragraph. An icon indicates the meaning:



This icon denotes a tip, which alerts you to advisory information.



This icon denotes a note, which alerts you to important information.



This icon denotes a caution, which advises you of precautions to take to avoid injury, data loss, or a system crash.

- » The » symbol leads you through nested menu items and dialog box options to a final action. The sequence **Settings » Desktop parameters » Device specific** directs you to pull down the **Settings** menu, select the **Desktop parameters** item, and select **Device specific** options from the last dialog box.

*About This Manual*

**Bold** Bold text denotes items that you must select or click in the software, such as menu items and dialog box options.

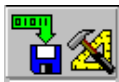
*Italic* Italic text denotes variables, emphasis, a cross reference, or an introduction to a key concept.

Monospace Text in this font denotes text or characters that you should enter from the keyboard. This font is also used for the proper names of disk drives, paths, directories, subroutines, commands, and variables.

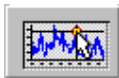
**Monospaced bold** Bold text on gray background denotes programming and syntax examples, as well as code of Autosequences.

# 1 DIAdem Overview

DIAdem is designed to perform all tasks involving technical data. Related functions are grouped into separate *DIAdem devices*. Depending on your task, you can equip your DIAdem installation with the following devices:



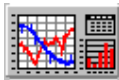
You can use DIAdem-DATA to manage and process your data. The data is arranged in channels, each displaying a data series. DIAdem imports and exports external data into files and databases.



With DIAdem-VIEW you can view, measure, and process your data as graphs in several axis systems. You can evaluate data with various kinds of cursors, scalings, and other tools.



Use DIAdem-CALC for mathematical data evaluation. Standard mathematical functions are available in extensive libraries and you can define your own formulas with the calculator.



With DIAdem-GRAPH you can generate presentation graphics to document your data series. You can access all data stored in DIAdem and present it in almost any desired way.



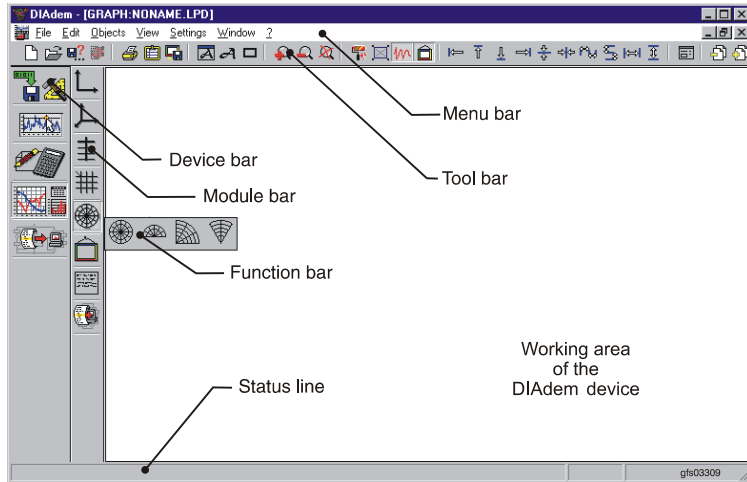
DIAdem-AUTO automates recurring work sequences as Autosequences. Interactive processing steps can be recorded in the TEACH-IN-MODE and then be reproduced.

## 1.1 The DIAdem Screen

The DIAdem screen structure is based on two principles: the features are grouped hierarchically according to task, and frequently used features are permanently visible and therefore accessible at any time.

The DIAdem devices constitute the main level, in which functions are grouped into related areas. Every DIAdem device has a module bar containing all function bars. Click on a function bar to open it and select the function you need. The most frequently used functions in the devices are available in the toolbar.

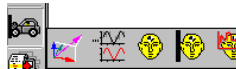
Layout of the DIAdem screen, in DIAdem-GRAPH



⇒ The device bar is on the left of the DIAdem screen and is permanent. This means that you can switch between devices at any time. Each tool is intended for a specific purpose.



⇒ Next to the device bar you will see the module bar, which contains related groups of functions. DIAdem-GRAPH has all 2D tables in one group and DIAdem-CALC has all graph functions in one group.



⇒ The function bars open out horizontally from the module bars and from there you can choose axis systems and tables in DIAdem-GRAPH, for example, or mathematical functions in DIAdem-CALC. The Autosequence function bar is the same in all the DIAdem devices, so frequently used Autosequences can be used in all devices.



⇒ The toolbar contains all the commands required to work with a particular DIAdem device.



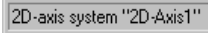
You can modify the *presettings* and several command icons using the *shortcut menu*. In DIAdem-CALC you can revert to the standard settings with reset.



⇒ The menu bar includes functions that are rarely used and are not contained in the function bars and toolbars in the device.

⇒ The DIAdem workspace automatically adapts to the selected device. DIAdem-DATA provides an overview of the data channels

and their contents. The workspace in DIAdem-GRAPH is designed as a blank page, which you can fill with graphics, tables, and text.



- ⇒ The status line includes extended Quickinfo, messages, and details about how the selected objects and actions work.
- ⇒ Shortcut menus facilitate fast access to functions. You can open them by right clicking on an icon or the workspace.
- ⇒ There are dialogs for setting axis systems, tables, and function block parameters. Basic options that are often changed are at the top of the dialog box hierarchy. Special settings, which are not modified as often, are one level down. For example, in the graph and axis definition, a quick glance reveals which data channels are displayed. You can scale and label axes in the dialog the next level down.



If icons, menu items, or devices are displayed in gray, these features are either not installed or not included in your license. You either have to register the appropriate function library (Settings >> GPI-DLL Registration) or extend your DIAdem license to include the relevant options.

## 1.2 The DIAdem Help



The program help in DIAdem provides you with four levels: the Quickinfo for the user interface, the dialog help with explanations of all the parameters in the dialogs, the help menu with subject-related descriptions, and the active help with film sequences for program operation, application examples, and user templates.

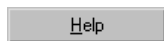
### ⇒ Quickinfo



DIAdem-CALC: Mathematics

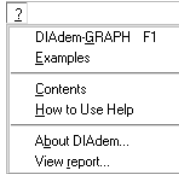
When you hold the mouse pointer over an icon, the yellow field with the symbol name appears. Click on Settings » General for the *long Quickinfo*, for a more extensive description of the icons.

### ⇒ Dialog help





Each dialog contains a Help button. The dialog help explains what each parameter in the open dialog means. References are made to more detailed explanations and related topics.

⇒ *Help menu*



The help menu opens the *device-related description*, which you can also access directly with <F1> *Contents* provides the general DIAdem descriptions. The *Report* is a record of all previous DIAdem sessions, the extent of which you can set yourself in Settings.

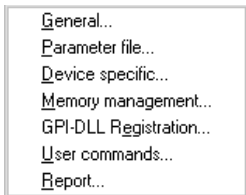
⇒ *Examples: Film sequences and demos*

Apart from film sequences and program operation, the *Examples* also include user templates and application examples. The user templates are solutions for simple tasks, which you start with  and stop with <Esc>. The information button  displays a brief task description and names the files used, so you can use the user templates for your own solutions. Most of the application examples are more extensive and demonstrate the wide range of applications available in DIAdem.



The example files are stored in the library directory `.\diadem\libr\`. If you have not installed the examples, use a *user-defined installation*

## 1.3 Settings



Desktop settings

You can use the Settings menu to adapt the program settings to your requirements.

⇒ *In the General settings you can alter the path selection and the standard files to be loaded. DIAdem always stores the path of the last file loaded as the user path, but you can also specify the library path here.*



You can use the twelfth symbol in the Autosequence device function bar to set the same user path for all devices.

⇒ *The parameter files contain desktop properties like presettings for the function bars.*

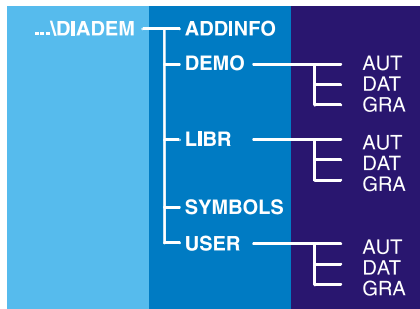
⇒ *You can enter further specifications for the DIAdem devices in the device Settings menus.*

- ⇒ Memory organization can be used to enlarge or reduce the data area in DIAdem-DATA.
- ⇒ The GPI-DLL Registration enables you to extend the range of DIAdem functions by registering function libraries, called DLLs.
- ⇒ You register user commands as VBScripts, in which you have programmed your own functions. An example would be axis labeling in DIAdem-GRAPH.
- ⇒ The Report is a record of the DIAdem program start and end, error messages, and confirmation messages.

## Directory Structure

When DIAdem is installed, it sets up a directory structure at three levels: the program level, the structural level, and the file level.

Standard directory structure



- I. The program directory, for example at c:\diadem, contains the program files (EXE, DLL), control files and Readme files.
- II. The structural level divides the files in the following way:
 

ADDINFO	system files
DEMO	practice examples from the User Manual
LIBR	examples, demos, and templates
SYMBOLS	System files
USER	For storing user files

III. The file level contains the files from each of the DIAdem devices, in various directory structures:

AUT	Autosequences (VBS, AUT), compiled Autosequences (AUC), list files (LST), variable definitions (VAS), user dialogs (SUD,COD)
DAT	Data header files (DAT), binary files (R48, I16 etc.)
GRA	Picture definitions (LPD), graphics files (WMF,BMP, TIF, JPG, etc.), DIAdem-VIEW layouts (LVD).

## Saving and Loading Settings



When working with DIAdem you adjust the standard settings to your requirements. DIAdem stores the program settings in the desktop file (DDD) when you close it. You can save your own settings in a file, which DIAdem automatically uses as a standard file. After having closed all DIAdem devices, you can generate other project-related settings. DIAdem starts with the last desktop file that has been saved.



If you work on another project, you can load the related desktop file when you have closed all the DIAdem devices. DIAdem is then restarted so the new settings can take effect.



You can start DIAdem with a special desktop file. Extend the program call with the parameter /D and the desktop file to be used, without the extension but with the following path:

```
C:\DIAdem\DIAdem.exe /Dc:\diadem\testing
```

## Installing and Operating DIAdem in a PC Network

To install DIAdem in a PC network for several workstations, perform the standard installation on the network server and install the DIAdem client version on all DIAdem workstations.

### ⇒ *DIAdem on the network server*

*For the server installation you have to select which features you need write rights for. Refer to the DIAdem Standard installation setup.*

### ⇒ *Workstation installation*

*Select network workstation (DIAdem-client) in the setup for each workstation and enter the directory for the server installation.*

When the installation is complete, start the DIAdem client so the necessary user interface files and the runtime-critical system files are copied to the local hard drive.



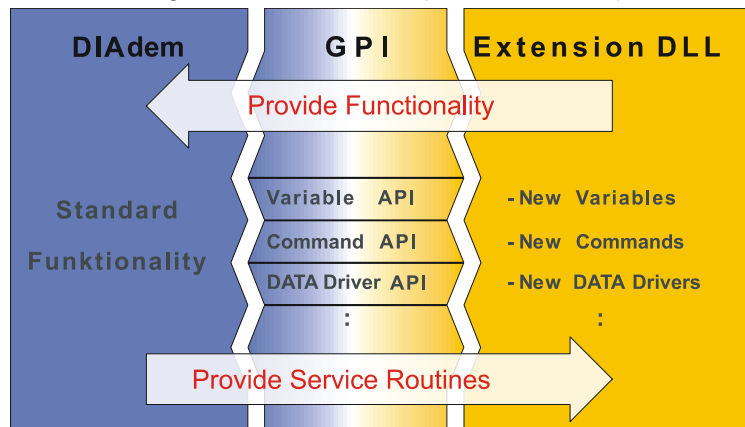


You can specify the system path as the program parameter (/S) when the program starts. If the system files are located in the System subdirectory, the call is: `diadem /SC:\diadem\system`.

## 1.4 The GPI Interface

The General Programming Interface enables you to extend various standard function areas in DIAdem. You program specific DLLs, providing additional variables, commands, file filters, or additional drivers for measurement hardware. The GPI Wizard generates the framework for a GPI-DLL, which you then only have to extend to include the functions you require. To be able to use your extensions, register the DLL in *Settings* » *GPI-DLL Registration*.

The GPI extends the DIAdem range of functions to include user-programmed functions.



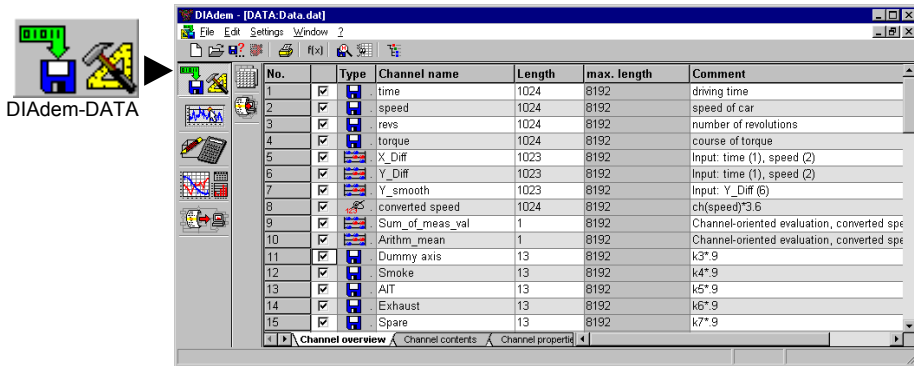
The GPI-Wizard was created for Microsoft Visual C++ (Version 6.00) and generates a basic definition with examples for the function extensions:

- ⇒ *Additional variables for Autosequences*
- ⇒ *Additional commands for Autosequences*
- ⇒ *File filters for non-standard external formats.*
- ⇒ *Service routines for accessing the data area*
- ⇒ *Additional drivers for process link hardware for the time-controlled acquisition, processing, and output of signals*

To be able to use your GPI extensions, you need DIAdem with its devices and options. You can extend these to include the functions you have programmed yourself. For example, if you program a file filter to import external data, you need DIAdem-DATA to be able to integrate the filter.





## 2 DIAdem-DATA: Processing and Managing Data





You can use DIAdem-DATA to manage and process your data. The data is arranged in channels, each displaying a data series. All the DIAdem devices read the data area and some store their results in new channels.



There are three data viewing modes in the working area: *Channel overview* shows all the channels in the data area and allows you to delete and add channels. *Channel contents* shows the data in the data channels and has features for deleting, editing and generating data. *Channel properties* shows the characteristic values of the data channels.




The toolbar contains the following functions for processing your data:

-  Delete data in DIAdem-DATA
-  Load data
-  Save data
-  Print

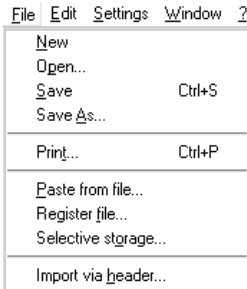
-  Data set properties
-  User-defined channel selection
-  Data Navigator
-  Formula calculation

## The Data Area

In the viewing modes, DIAdem-DATA distinguishes between data channels that are read, entered, calculated, and measured.


-  Entered: data edited or generated
-  Read: loaded from a data file
-  Calculated: results calculated by DIAdem-CALC

The data area continually adapts to the stock of data, so data channels with different lengths can follow each other in sequence. If the defined number of length of channel is exceeded, enlarge the data area. To do this, close all the windows with **Window » Close all** and click on **Settings » Memory management » Data matrix**.



The **File** menu contains features for pasting in data channels from a DIAdem-DATA file and importing external data. You can load ASCII data with *File open*. Instead of loading extensive data sets, you can register them for reading.

- ⇒ **Paste from file:**  
*You can select data channels from a DIAdem-DATA file and add them to the existing data.*
- ⇒ **Import via header:**  
*You can import external data by describing the data set properties. The description is stored in the header file.*
- ⇒ **Open ASCII files:**  
*When you load ASCII files click on File open for assistance in characterizing the data to be imported.*

 registered  
Registered data channels cannot be altered.

- ⇒ **Register file:**  
*If you are working with very large data sets, you can register them as files. The data channels then only have read-only access and are not completely loaded. These data channels can thus be displayed and used for calculations, but not overwritten.*

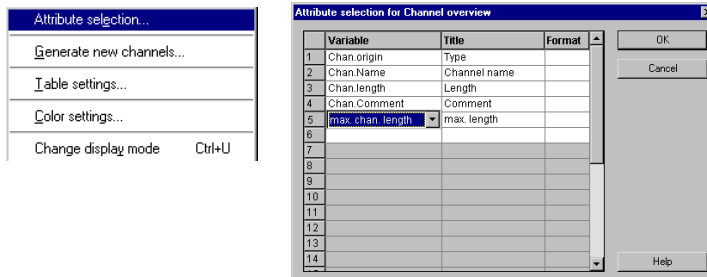
*Registering files enables you to view very large amounts of data that would take up a lot of time and space if they were loaded in the data area.*

The data area only exists when DIAdem is running. Modifications to the data area do not become permanent until the data set is saved. DIAdem saves data sets in several files. The *header files* (for example TEST.DAT) contain the data set properties and references to the binary files. The *binary files* contain the numeric data in space-saving formats. One data set may have several binary files (for example TEST.I16 and TEST.R48).

Click on **File » Selective storage** to store a selection of the data channels, you have activated in the channel overview. You can store them in an existing data file or in a new one.

## Data Area Display

To display the data area on the screen, you can modify the attribute selection, the table settings, and the order and widths of the columns. Open the *Attribute selection* by clicking on the shortcut menu in the heading bar. You can select any channel property as a **variable**. The attribute selection does not affect all three view modes; you can modify it individually for *channel overview*, *contents*, and *properties*.



You can alter the *column width* in the heading bar. The *column position* for marked columns can be altered, for example, to compare data channels in the channel contents value by value.

## 2.2 Channel Overview

### Channel overview

The channel overview provides you with a line by line list all of the channels in the data area. A standard line includes the channel number, the checkbox for activating the channel, the channel name, the channel length, and the comment field. There are two types of columns:

- ⇒ Columns with a gray background, like channel number and type. These can only be displayed and not edited.
- ⇒ Columns with a white or light gray background can be edited. For example, if you reduce the channel length by entering a lower number, the surplus data will be rejected when you save the data set.

The channel overview shows you the data at a glance

No.	Ty...	Channel name	Length	max. length	Comment
1	<input checked="" type="checkbox"/>	Time	1024	8192	driving time
2	<input checked="" type="checkbox"/>	Speed	1024	8192	speed of car
3	<input checked="" type="checkbox"/>	Revs	1024	8192	number of revolutions
4	<input checked="" type="checkbox"/>	Torque	1024	8192	course of torque
5	<input checked="" type="checkbox"/>		0	8192	
6	<input checked="" type="checkbox"/>		0	8192	
7	<input checked="" type="checkbox"/>		0	8192	
8	<input checked="" type="checkbox"/>		0	8192	
9	<input checked="" type="checkbox"/>		0	8192	
10	<input checked="" type="checkbox"/>		0	8192	
11	<input checked="" type="checkbox"/>		0	8192	



Activate data channels

The checkbox column plays a vital role: channels are only available in the channel contents and properties if they have been activated in the channel overview. If you click on a checkbox you change the activation status of the channel. You can use the shortcut menu to *activate or deactivate the entire column, or invert the activation status*.



Cutting and pasting channels

Switch to the *user-defined view* via the toolbar. You can activate other channels here than in the standard viewing mode. Your channel selection is saved with the desktop settings. You can modify the channel order, column width and the font in the user-defined view mode.

To cut, copy, paste, and delete data channels, *mark* the channel line by line, by clicking on the channel number. Marked channels have a dark background. Delete the marked channels with <Del>, copy them with <Ctrl-C>, and cut them with <Ctrl-X>. Use <Ctrl-V> to paste the channels from the marked channel onwards.

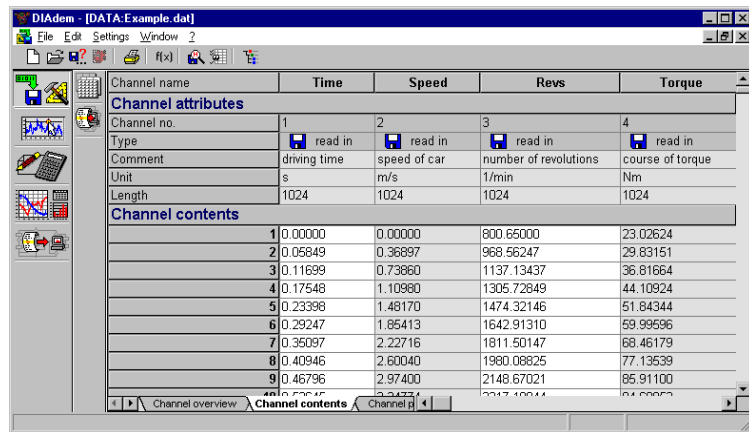
The **Generate new channels** feature is in the shortcut menu and you can use it to reserve channels by entering the channel name and length.

## 2.3 Channel Contents

### Channel contents

The channel contents displays all the data channels activated in the *channel overview*, in columns. The channel contents are underneath the channel attributes. Depending on the size of the window and the amount of data, only an excerpt of the data may be displayed.

Edit the data contents in the channel contents mode.



In the channel contents you can process the channel data individually or in blocks or channels.

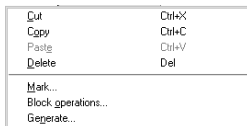
#### ⇒ Single datum

You can edit any data field that is not dark gray; this applies for the channel attributes as well as the channel contents. If you click on a data field you can delete or overwrite the contents. If you want to alter the entry, double-click on the data field.

#### ⇒ Block operations

You can mark data fields in one or more data channels and group them into a data block.

The shortcut menu contains the clipboard functions for cut, copy, and paste. Data blocks are *pasted* from the cursor position, by shifting existing values. When a data block is *deleted*, the subsequent values are moved up.



Context menu (excerpt)

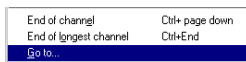
The *block operations* for copy, paste and add are performed without the clipboard. Use the *Insert data* block operation to assign a fixed value or a DIAdem variable to a data block.

#### ⇒ *Data channels*

Click on the shortcut menu to delete, copy, cut, and paste marked channels. When you move channels, you can *sort* marked channels in ascending order of the channel number.

*New data channels can be generated* in various ways: you can import data channels from other files, calculate result channels with DIAdem-CALC, and enter data manually.

You can use the *Generate* feature in the shortcut menu to generate equidistant data rows. The generated data is inserted from the data field the cursor is positioned in, or you can mark and overwrite a data block in the data channel.



In the shortcut menu you will find several *jump functions*:

⇒ *Channel end goes to the end of the channel the cursor is in.*

⇒ *Go to: enter the data field with the line and channel.*

⇒ *End of longest channel in the stock of data.*

## 2.4 Channel Properties

### Channel properties

The channel properties display the properties and characteristics stored in the header file.

⇒ *The channel attributes that can be modified include the channel name, comment, unit, length, channel type, and the format the channel contents are displayed in. In the various additional items, you can store special information about each data channel.*

⇒ *The maximum length, NoValues, monotony, and the minimum and maximum values are all channel properties that cannot be modified.*



Apart from the additional information, the channel properties also show calculated values such as extrema and monotony.

Channel name	Time	Speed	Revs	Torque
Comment	driving time	speed of car	number of revolutions	course of torque
Unit	s	m/s	1/min	Nm
Length	1024	1024	1024	1024
Channel type	Time	Numeric	Numeric	Numeric
Format	MM.dd.yyyy hh:nn:ss	dddd.dd	d.dde	e
Additional text 1				
Additional text 2				
Additional integer 4	0	0	0	0
Additional integer 5	0	0	0	0
Channel no.	1	2	3	4
Type	entered	read in	read in	read in
max. length	8192	8192	8192	8192
NoValues	No	No	No	No
Monotony	increasing	not m...	not monoton...	not monotonous
Minimum	01.01.0000 00:00:00	0	800.077998	23.00314929
Maximum	01.01.0000 00:01:00	52.76359526	6059.729223	207.1355276

Recalculate characteristic values of the active channel  
Recalculate all characteristic values

You can recalculate the characteristic values with the shortcut menu, because not every alteration to the data area automatically triggers an update. Either *the characteristic values in the active channel can be recalculated* or *all the characteristic values can be recalculated*.

## 2.5 The Data Navigator















You can use the Data Navigator to navigate through data files, file directories, or databases. You can read files, view the measurement and test data they contain, and load selected data channels into the current data area. You can also save data that is located in the data area into an ATF file (ASAM Transport Format) or to an AOP server (ASAM-ODS Protocol).

Channel values	unit	datatype
<input checked="" type="checkbox"/> PRED	kW	REAL64
<input checked="" type="checkbox"/> PB	bar	REAL64
<input checked="" type="checkbox"/> NMOT	1/min	REAL64
<input checked="" type="checkbox"/> MITOEL	grd C	REAL64
<input checked="" type="checkbox"/> MIOELFS	mm	REAL64
<input checked="" type="checkbox"/> TWA	grd C	REAL64



If the Data Navigator call fails, you can register the *Gfsods.DLL* by clicking on Settings » GPI-DLL Registration.

The Data Navigator toolbar provides these features:

-  Open model file (ATF, EXP)
-  Open file (ATF, XML)
-  Navigate on servers (AOP, SQL, Citadel) and file directories
-  Save data file or data model
-  Select measurement channels
-  Select measurements
-  User-defined configuration (STP file)
-  Process configuration
-  Update display
-  Copy measurement channels to the DIAdem-DATA area
-  Store marked DIAdem-DATA channels in the data source
-  Close current data source
-  Free icons for assignment with Autosequences
-  Close the application window using the window bar

In the *tree view* to the left of the Data Navigator, you navigate through data sets, file directories, and databases. You can also select data channels. In the *attribute list* on the right, you see the attributes associated with the selected data channels.



In the tree view you can select the view mode for measured channels or measurements. If you select *measurement channels*, you can copy marked channels into the DIAdem-DATA data area. In the *measurement* view mode, you can transfer the entire measurement to the DIAdem-DATA area. This is facilitated by being able to work parallel in DIAdem-DATA and in the Data Navigator.

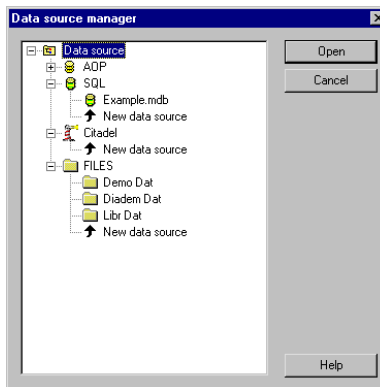
For every *data source*, a tree view is opened with an attribute list, in the Data Navigator. You can open an ATF or an XML file, a file directory or a database, as the data source.



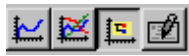
⇒ Click on **File open** to load ATF and XML files. You can import and save data channels to ATF files.



⇒ Click on **Open data source** to access File systems and databases. AOP, SQL, and Citadel databases are supported. Data can be read from, and saved to AOP servers. You can use the data source manager to define the access parameters for new data sources and to modify existing settings, and open and close data sources.



⇒ The **model file** supplies the structure, which conforms to ASAM-ODS, as an ATF or an express file.



User-defined configuration

The display of measurement channels or measurements in the Data Navigator can be configured. You can generate your own configurations with the Data Navigator Editor, save them as STEP files, and load them to the toolbar switches.



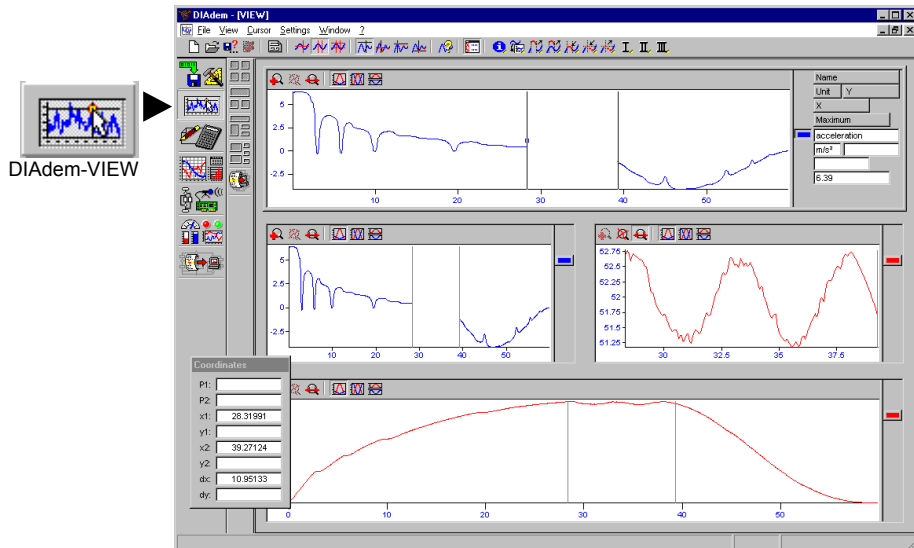
Autosequences

You can assign your own Autosequences to the Autosequence switches, to automate read and write access. Autosequences can be generated with DIAdem-AUTO, which contains special Autosequence commands and variables for the ASAM-ODS Interface.



# 3 DIAdem-VIEW: Viewing, Measuring, and Editing Data

With DIAdem-VIEW you can view, measure, and edit your data as graphs. Several axis systems for the display of multiple graphs are available. The evaluation is conducted with various kinds of cursors, scalings, and other tools.



DIAdem-VIEW



Icons in the function bar let you determine how to split the screen. Several layouts for split screens are available. The graph definition for each axis system is carried out individually and you can copy graphs with a graph icon into another axis system.

To the right of every axis system is a legend that contains the name of each channel, the current cursor coordinates, and other information.



Each axis system in DIAdem-VIEW has its own tool bar. These individual tool bars are used to modify the particular area of a graph that is displayed (zoom and scroll) and to change the scaling of the y-axis when displaying several graphs with different value ranges.



In the main tool bar are the various cursors with the different measurement modes. Use these cursors to evaluate curves and scan characteristic points.



Several icons are pre-assigned with Autosequences, which can copy, cut and paste, curve points and transferred graphs in DIAdem-GRAPH.

## Layout: Choosing a Split Screen



The default layouts in the function bar differ in number and alignment of the axis systems. You can customize the settings of a layout and save it to a file with the extension \*.LVD. The settings you can modify are the alignment of the axis systems, the legends, the scaling mode, the scroll bars, and the coordinate window. However, you cannot change the data channel assignments for each graph.



If the axis systems contain curves they are stored together with the layout.



The default DIAdem-VIEW layouts are stored as REG\*.lvd and UNR\*.lvd in the library directory. Do not overwrite these files!

In **Settings » Desktop parameters » Device specific...** you specify the image template that is to be displayed in the workspace. By default this is EXAMPLE.LVD.

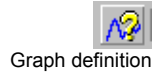


When changing the layout during a session to a layout that has more axis systems than the previous one, all graphs remain. However, if the new layout has less axis systems, the graph definitions in the surplus axis systems are lost. **Choosing yet another layout with more axis systems does not restore lost graphs.**

## 3.2 Axis Systems: Displaying Data in Curves

A highlighted frame identifies the active axis system. When moving the cursor in the active axis system, the cursors in other passive axis systems cover the same value range.

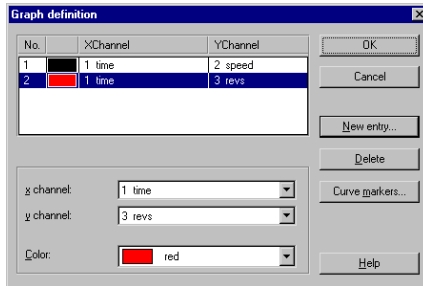
*Start the exercise with the graph definition in the upper axis system. The data set for this exercise is EXAMPLE.DAT. If you have already modified the default data in EXAMPLE.DAT, load the data file DRIVE.DAT in DIAdem-DATA for alternative use.*



Open the axis system and specify the data to be graphed by clicking on **New Entry**. Choose the following settings:

⇒ X-Channel: **1 Time**

⇒ Y-Channel: **2 Speed and 3 Revs.**



⇒

You can add curves or change the data allocation of the existing curves. In **Curve markers** you can highlight the points of a curve with colored symbols.



The upper axis system now has two graphs. Each curve is represented at the right by a graph icon of the same color. You can copy graphs into other axis systems by dragging an icon into the desired axis system.

## Individual Scaling of Axes



Each axis system has its own tool bar for zooming, scrolling, and scaling. If you want to compare curves with different value ranges with each other, as in the exercise, you can enhance the display by choosing a different scaling mode.

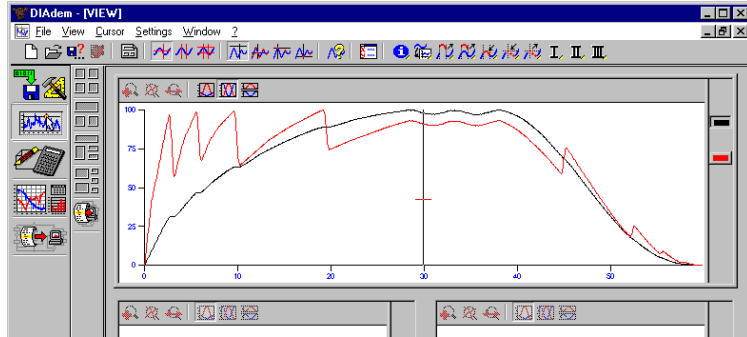
⇒ **1 System [physical]:** An axis system with a common absolute y-axis.

⇒ **1 System [%]:** An axis system with a common y-axis as a percentage.

⇒ **n Systems [phys.]:** n sub axes with a characteristic physical value range.


Since the absolute values of the rotational speeds are 100 times higher than the absolute speeds in the exercise, the speed curve is merely displayed as a line on the abscissa. Choose scaling mode **1 System [%]**.

Two curves with different value ranges with percentage scaling



### Curve legend

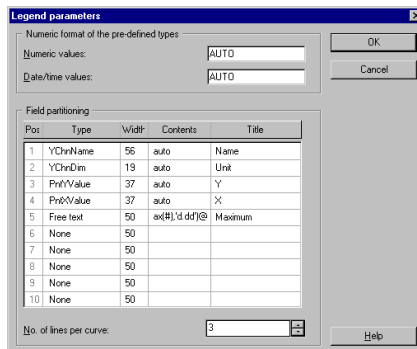
Each axis system has a legend next to it on the right. The legend shows the name and physical unit of the y-data channel and the x- and y-coordinates of the measurement cursor. You can add information to the legend and modify the order in which the information is displayed.


  
 Open legend

*Pull the right border of the axis system to the left and open the legend parameters by double-clicking on the legend. Modify position 5 as follows:*

Type	<b>free text</b>
Contents	<b>@str(CMax(#), 'd.dd')@</b>
Title	<b>Maximum</b>

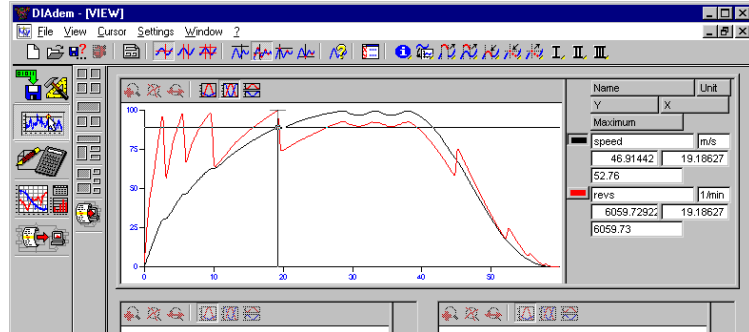
*Extend the number of lines per curve to 3.*



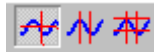


The content of the free text is a formula that contains the DIAdem variable *CMax*. The maximum value of the curve (#) is to be output as a string with two digits of precision. You must enclose such a formula with @ symbols to ensure that the formula interpreter does not interpret the expression as text.

The legend can be customized



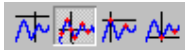
### 3.3 Cursor: Viewing and Editing Graphs



In DIAdem-VIEW cursors and various measurement modes enable you to view and measure graphs. The following cursors are available:

- ⇒ *Crosshair cursor for measurement*
- ⇒ *The crosshair cursor moves along the coordinates of a graph while you press the left mouse button.*
- ⇒ *Band cursor for measurement, zooming, scrolling, cutting, and interpolation.*
- ⇒ *The width of the sector is determined by the horizontal position of the lines to the left and right.*
- ⇒ *Frame cursor for measurement, zooming, scrolling, cutting, and interpolation.*
- ⇒ *The width of the frame is also determined by the horizontal position of the two lines. The graph shape automatically determines the height of the frame.*

## Measuring a Graph

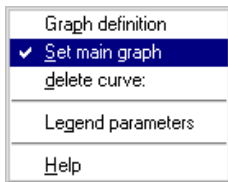


In addition to the *free cursor*, which you can move freely in an axis system, there are the following measurement modes:

- ⇨ The *Graph cursor* lets you scan any point of a curve.
- ⇨ The *Maxima cursor* lets you scan the peaks of a curve.
- ⇨ The *Minima cursor* lets you scan the valleys of a curve.



Choose the **Graph cursor**. Move along the graphs and notice that there is a small square on the one graph and a small cross on the other graph.



The square identifies the main graph that you can set in the shortcut menu of the legend symbol. By setting the main graph you determine which graph leads the cursor.

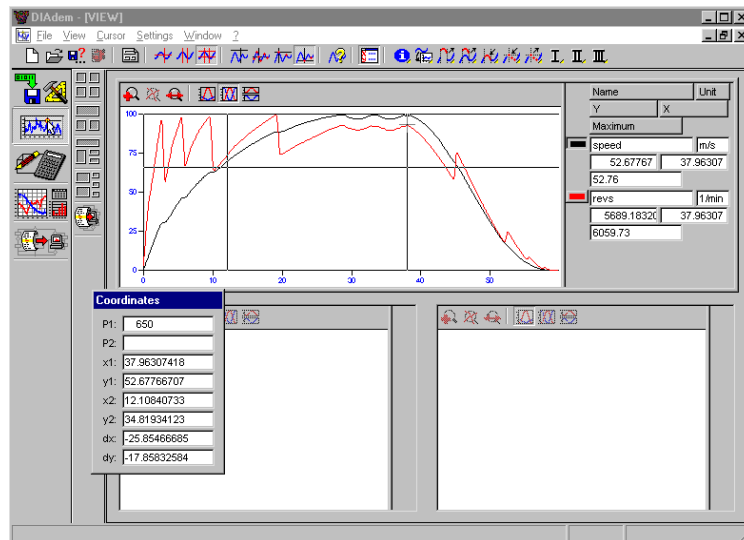
Set the **red graph (Time/Revs)** to be the main graph. Now, if you move the **Crosshair cursor**, it remains fixed on the red graph.



Switch coordinate display on/off

The *Coordinate window* is another tool to display values of the current cursor position. The coordinate window displays the measurement point number, the x and y-coordinates, and, if applicable, the distance between the section lines.

The coordinate window can be positioned anywhere on your screen



## Zooming and scrolling



The icons for zooming and scrolling are located in the individual tool bars, since these functions refer to each individual axis system. Zooming and scrolling functions are only available for the band and frame cursors.



Choose the **Band cursor** and set it to the desired width in the upper axis system. Copy the black graph to the lower left axis system and the red graph to the lower right axis system.



Zooming

The graph segment you have determined by clicking on the zoom icon now fills up the entire upper axis system. The smaller the segment the more you zoom into that segment. All curves in the axis system are enlarged equally when they are within a section.

You can zoom in further by reducing the band cursor section and clicking on the zoom icon again.



Scrolling

While the zoom captures sections on curves, scrolling lets you move the curve through the enlarged view. In each axis system you can enable horizontal and vertical *scroll bars* in the **shortcut menu**.

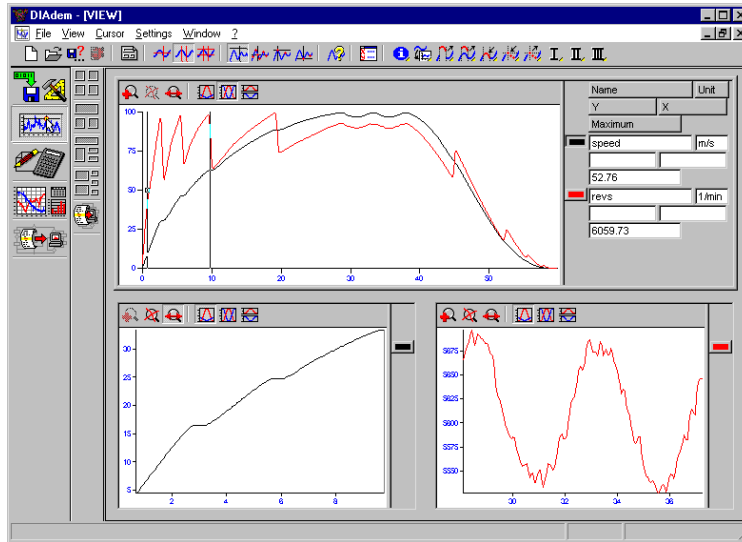


**Zoom off** switches off zooming and scrolling.

Zooming and scrolling refers to the band or frame cursor of the current axis system. However, you can also zoom into the other axis systems while moving the cursor in the current one. To do this all axis systems must use the same x-channel.

*For the lower left axis system turn on the zoom and for the lower right axis system enable scrolling. Move the band cursor along the main curve in the upper axis system.*

Viewing curves in multiple windows: In the upper window you move the band cursor, in the lower left window you zoom into a fixed section, and in the lower right window you scroll.



## Cut and Print



To edit curves and for curve documentation, the tool bar contains Autosequences. The icons with the roman numerals are not assigned to default Autosequences. You can use them for your own Autosequences to expand DIAdem-VIEW to your needs.

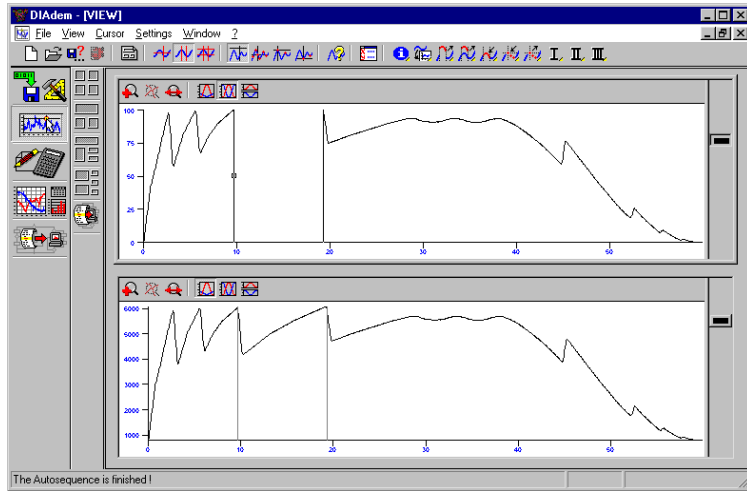
The following default autosequences are available:

- ⇒ *Information and settings for the default Autosequences in DIAdem-VIEW*
- ⇒ *Transferring the current evaluation to DIAdem-GRAPH*
- ⇒ *Deleting a segment of a graph*
- ⇒ *Copying data of a curve section to DIAdem-DATA*
- ⇒ *Interpolating a deleted curve section*
- ⇒ *Manipulating single values in curves*
- ⇒ *Writing single values of one or more curves in data channels*



Choose a graph segment with the band cursor in the upper axis system. Cut the segment using the Autosequence **AreaDel**.

Cutting a segment using a default Autosequence.



Restore the section using the Autosequence **AreaFit**. The missing section of the curve is restored by interpolating the slope of the end points.



The Autosequences only affect the upper axis system. For more information on Autosequences refer to the chapter *DIAdem-AUTO*. For the comparison of one graph in two axis system, the corresponding y-data-channel was copied, not the graph itself. **Cutting the segment as well as the interpolating it effects the data directly; the original data is overwritten.**

### 3.4 Documenting Evaluations

Two ways exist to document the results of your graphs evaluation: printing the workspace or creating a diagram in DIAdem-GRAPH.



*Print workspace* prints the current workspace.



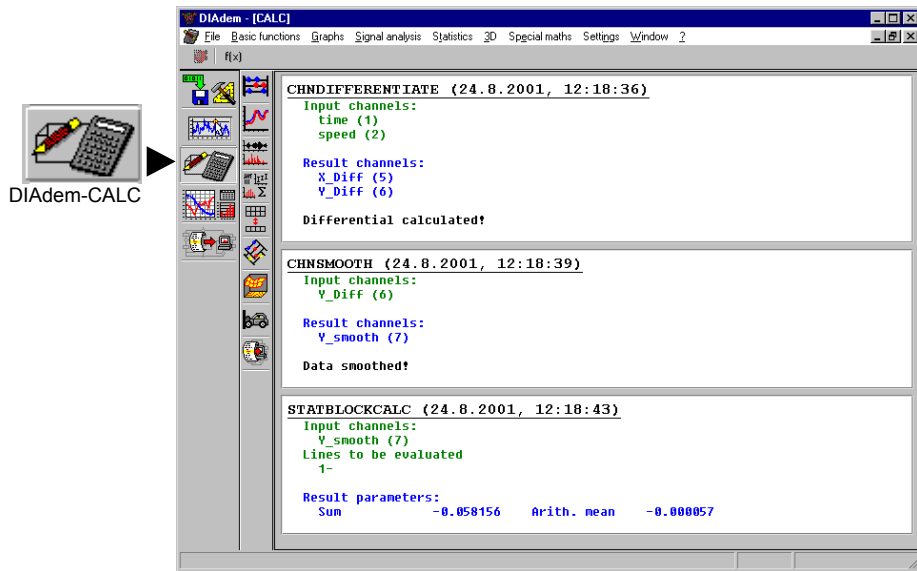
Transferring the content of your current workspace to a diagram in DIAdem-GRAPH creates a diagram that contains the exact display from DIAdem-VIEW, including zoomed or scrolled axis systems.

For more information on presentations and diagrams refer to the chapter DIAdem-GRAPH.



## 4 DIAdem-CALC: Analyzing and Evaluating Data

DIAdem-CALC includes all the mathematical functions required for calculations. The various functions are arranged by subject area in the module bar: basic mathematics, graph calculations, signal analysis, statistics, 3D functions and crash analysis.



The workspace in DIAdem-CALC displays a record of all the calculations performed in one session.

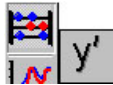
You can perform your own formula calculations with the calculator. You can evaluate channels or single values. You can also assign values to variables and retrieve them.

## 4.1 Performing Standard Calculations

The standard mathematical functions are preformatted functions that you can call, parameterize and execute.

### An Example: Differentiation

To determine the acceleration, the speed is to be differentiated dependent of time. Ensure that the data file EXAMPLE.DAT is loaded in DIAdem-DATA.



Open **Basic mathematics » Differentiation**. Enter **Time** as the *x-channel* and **Speed** as the *y-channel* and perform the differentiation.

Records are displayed for all the standard calculations in the workspace. They appear during the entire session and show you whether the calculation was completed successfully, which channels were used for input and where the results were stored.

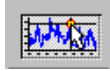
The differentiation generated two result channels in DIAdem-DATA: the time axis of the differentiated curve is in the first channel. These values were calculated as the mean of two adjacent x-values in the target input channel. The second new channel contains corresponding y-values, which are the calculated difference quotients. In terms of physics, the values in this channel correspond to acceleration.

For some mathematical operations, you can activate **Store results in the original channels**. This kind of *in place* operation overwrites the source channels with the result data!

You can also assign the results of some mathematical calculations to DIAdem variables. Open the calculation of **Statistic characteristic values » Results**. **Transfer as value** means that the result is stored in the clipboard. **Transfer as variable reference** means that the result is transferred to the relevant DIAdem variable. For example, if you have calculated the arithmetic mean, you can integrate the @StatArithMean@ variable as text for your presentation graphics in DIAdem-GRAPH. Every time the graphic is refreshed, the current variable contents are used.



## Viewing Results in DIAdem-VIEW

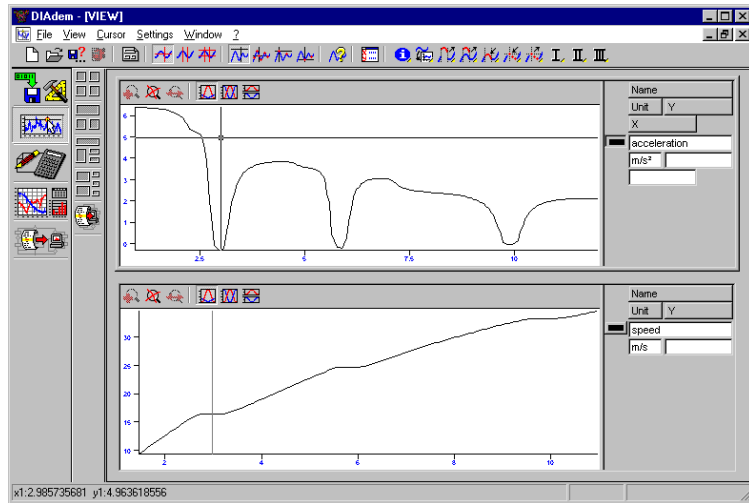


View the resulting graph in DIAdem-VIEW.

Open **Graph definition » New entry**. Select **X\_Diff** as the x-axis and **Y\_Diff** as the y-axis. Enter the original signal **Speed/Time** in the lower axis system.

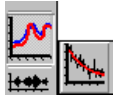
If the time is 3, 6, 10, etc. seconds, the acceleration goes down to almost zero. This represents gear changes during acceleration which are displayed as velocity plateaus.

Graphical comparison of the result and the original signal

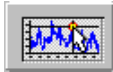


## Example 2: Curve Smoothing

In the differentiation example above, the only parameters required were the input channels to be analyzed. Of course there are also functions that require additional input parameters. One of these functions is smoothing a curve by calculating the floating mean.



Open **Curve calculation » Smooth**. Select the calculated acceleration **Y\_Diff** as the channel to be smoothed and enter **2** as the **smoothing width**. This indicates the number of neighboring values to the right and left of the coordinate currently being calculated that are to be included in the calculation of the mean.



Compare this result in DIAdem-VIEW to the previous calculation. Enter **Y\_smoothed/X\_Diff** as the new curve. If you perform several calculations with an increasing smoothing width, the result in DIAdem-VIEW could look like this.

Original signal compared to the various smoothed curves



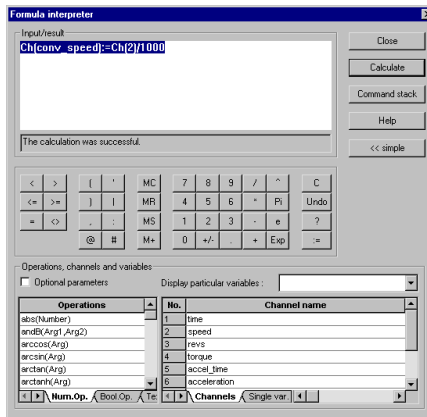
## 4.2 Calculator: User-Defined Formulas

f(x)

You can use the calculator to link channels or single points using functions. The functions range from basic mathematics and trigonometric functions to string and bit linkages. You can assign values to variables and request their current contents.

Open the calculator in the tool bar.

Advanced calculator mode



<< simple

When you use the calculator in the **standard mode**, the input field is in the top left hand corner and the keys are underneath it. Enter your formula and the result is displayed in the row beneath it.

The keys include comparison operators, special characters, memory functions, the question mark for queries, and the assignment character. There are two notable special characters: the @ symbol is used for variable contents and the # symbol introduces a time format.

To the right of the input field, click **Command stack** for a list of the formulas that have been processed: the last command is at the top. Copy the formula to the input field with a double click.

extended >>

In the advanced mode, there are two more tables with **operations** (numeric, Boolean, text operations) and **variables** (data channels, vectors, single and special variables). If you activate **Optional parameters**, the corresponding number of decimal places appear in the bracketed expression when the operations have been selected. You can specify which variables are to be displayed in the **Display certain variables** listbox.

*Convert Channel 2 speed from m/s to km/h in 2 steps.*

```
Ch("conv_speed") := Ch(2) / 1000
```

```
Ch("conv_speed") := Ch("conv_speed") * 3600
```

In formula assignments you can use channel names and channel numbers. If the channel specified on the left does not exist, it is either generated. You can also use just the channel name.

```
Ch("conv_speed") := conv_speed * 3600
```

If you want to write the result to the next free data channel, use the # as the variable.

```
Ch(#) := Ch("conv_speed") * 3600
```

Follow the following *Syntax* in the formula definition:

⇒ *Strings may not be longer than 16 characters.*

⇒ *Write "strings" in inverted commas.*

⇒ *Formulas are assigned with :=*

⇒ *Use a decimal point.*

Apart from the channel conversions listed above, you can also use the formula interpreter as a standard calculator.

```
3.3 + 5.8
```

```
2Pi6.5 (Pi stands for π)
```

```
3Exp5
```



If the result of a calculation is not defined, such as division by zero, DIAdem assigns NoValue as the result.

In the advanced calculator mode, you can access various operations and DIAdem variables.

**Operations** For example, a channel number is to be queried. Open the **Operations » Text.Op.** tab and enter **CNo(Channel name)** with a double-click. Set the channel name **Channels » Time**. Close the expression with a question mark : **CNo('Time')**? The number of the Time channel appears in the result row.

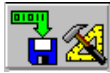
**Variables** In the right table there are various variables that are frequently used in formula expressions: single variables, vector and temporary variables, and several variables related to data sets. You can use all DIAdem variables.

You want, for example, to use a register variable to convert kilometers to miles. Select **Single var. » R1** and assign **1.61** for a mile, to the register variable. Enter the distance in km as follows: **Distance/R1**. The result is the distance in miles.



Variable assignments apply to all devices in DIAdem and are only modified when a new assignment is made or deleted when DIAdem is closed.

## Viewing Results in DIAdem-DATA



Open DIAdem-DATA to see where and how your new speed data is stored in the data area. The new result channels appear as the last entries in the data overview.

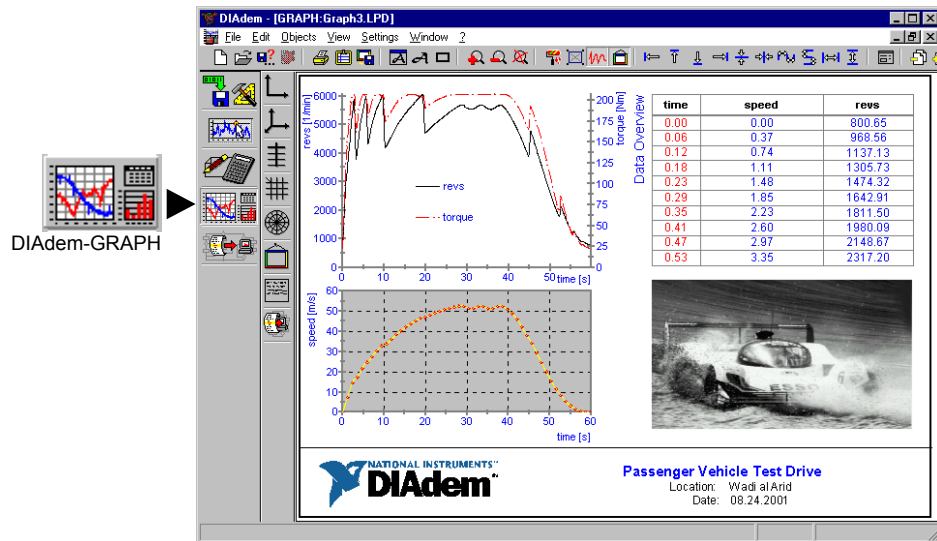
The length of the data channels is also shown in the data overview. You will note that the differentiation result channels differ from the original channels by 1. This is because during differentiation the difference-quotient of two neighboring x input values is calculated.

Modify the following entries:

	Channel overview » Channel name	Channel contents » Units
<b>Chan. 5</b>	accel_time	sec
<b>Chan. 6</b>	acceleration	m/s <sup>2</sup>
<b>Chan. 7</b>	accel_smoothed	m/s <sup>2</sup>
<b>Chan. 8</b>	conv_speed	km/h

# 5 DIAdem-GRAPH: Graphic Representation of Data

In DIAdem-GRAPH you can create presentation graphics to document your data. You can access all data stored in DIAdem and present it in almost any desired way.

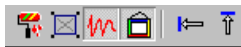


DIAdem-GRAPH



The various graphical display objects are located in the module bar: axis systems, tables, polar displays, graphics, and text objects. The function bars contain various default graphic objects.

A diagram is created directly in the workspace. You can combine all objects, position them, and scale them to your specific needs. After opening an object, you can assign and parameterize the data.

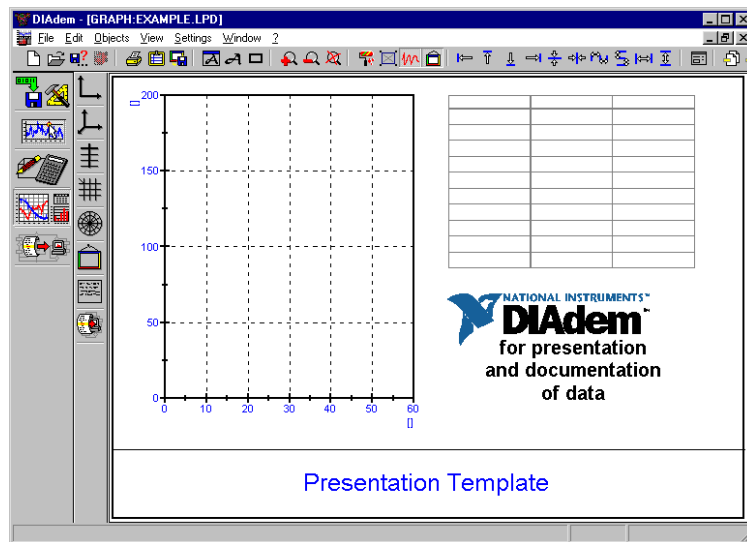


In the tool bar you can access functions like aligning objects, displaying diagrams, and creating text.

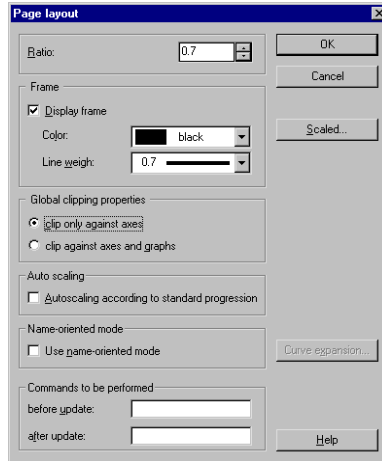
## Image Template: The pre-defined Layout

The graphic description is independent of your data and is saved individually in a LPD file. You can reuse this file for any data set. In the object dialogs, you assign the data as well as modify the layout of curves, tables, and text.

In **Settings » Desktop parameters » Device specific** you specify the image template that appears in the workspace by default. The file EXAMPLE.LPD contains a pre-defined axis system, table, logo, and text. The axis system and the table do not contain data since no data has been assigned yet. You can, however, save a layout with assigned data. If you then load suitable data, the layout is displayed containing this data.

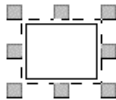


The layout shown above has a frame that you can put in the page layout. Graphic objects may be moved, scaled, or rotated only inside this frame. Set the page layout before creating the layout, so the alignment of graphic objects does not change. The page layout is stored with the picture definition because all relative position specifications for the graphic object definition are related to it. To modify the page layout choose **Settings » Page layout**.



Set the paper size and the alignment in the **Height/Width Ratio**. Use the **Clipping properties** to determine the clipping of text against axes and curves. When you **Reference by name**, you assign the data channel via the channel name, not the channel number. Use **Commands** from **Settings » Script Registration** to execute registered script-based command extensions. In order to **display** the diagram **to scale**, set the width and height of the diagram.

## 5.2 Arrange Objects: Move, Resize, Align



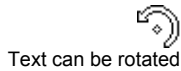
Objects of the various function bars can be grouped together and positioned anywhere on the workspace.

You can select objects by clicking on them or by using the <Tab> key which eases selecting objects that overlap. To select several objects, click anywhere in the workspace and drag the cursor diagonally across the workspace while holding the left mouse button. Alternatively, hold the <Ctrl> key and select the desired objects. To clear your selection, click outside the selection box or press <Esc> or <Ctrl-N>.



You can move or resize selected objects. If you resize an object by dragging the sizing boxes, the height-to-width ratio does not change. Press the <Shift> key while resizing an object to disregard

the size ratio. Objects that have sizing boxes only in the corners, always keep their height-to-width ratio. To resize an object *concentrically*, drag the size handles while holding the <Ctrl> key. The ratio of the object is maintained.

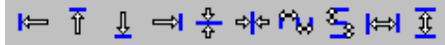


A single-lined *text box* can be rotated by selecting it and clicking on a sizing box

You can select and move axis systems and tables as well as their subobjects. **Subobjects** of an axis system are the axes, the axes labels, the axis scale, and the curves. Every subobject can be moved and parameterized independently of the others. While pressing the <Shift> key you can select objects more easily, especially if objects overlap or other components of an axis system have already been selected.

### Alignment functions: Arranging Objects

To arrange objects, use the icons in the toolbar for alignment and size adjustment.

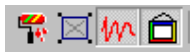


You can align several objects to the left, right, top, etc.; for example, where the object is furthest to the left functions as the benchmark. You can also align objects horizontally, equalize the distance between objects, or make all selected objects the same size.



Groups of selected objects can be edited as individual objects. You can move, enlarge, reduce (in size), and parameterize groups of objects. Note however that modifications must be valid for all objects of the group. For example, if one of the objects cannot be enlarged, this applies to the entire group.

### Quick Display: Sketch mode and axes without curves



To display complex diagrams quickly, you can turn off certain display layers.



With **Refresh** you update a diagram, for example when the diagram is redrawn, contents of variables are updated, and the objects are displayed in the actual order. Note that an object that you move or parameterize is the only one that is redrawn and appears on top.





With **Sketch Mode** you can hide selected objects and display them as gray frames. These objects are displayed again once you have *refreshed* the screen.



With the **Graphic Layer** you can hide all background graphics. They are replaced by gray placeholders and the file names. To display the background graphics again, you must turn the graphic layer back on.



With the **Data Layer** you can hide all curves and table contents. Axis systems and tables are empty. To display the data again, you must turn the data layer back on.

This function accelerates the display when large background graphics or particularly long data channels are used. The graphic and data layers are global modes that rank above the sketch mode. Thus, when you hide the graphic layer for example and then display it again, all graphics are displayed even if you have previously hidden individual graphics in sketch mode.

## 5.3 Axis Systems: Displaying Data in Graphs

Start the exercise by assigning data and parameterizing the axis system in the image template EXAMPLE.LPD. If you have already modified the default data in EXAMPLE.DAT, you can load the data file DRIVE.DAT in DIAdem-DATA for alternative use.

*Open the axis system by double-clicking on it. In the dialog for graph and axis definition, you can define the axis system as well as the graphs.*



Assign the data using **New Entry**. Choose the following settings:

⇒ X-Channel: **1 Time**

⇒ Y-Channel: **2 Speed**.

⇒ Line color: **yellow**

Graph param...

Open **Graph parameters** and make the following settings for this graph:

<b>Display</b>	Line Thickness	0.7
<b>Marker</b>	Marking Style	Rhomb
	Size	2
	Marker Color	<ul style="list-style-type: none"> <li>• unlike graph color,</li> <li>• red</li> </ul>
	Repeat Marker	<ul style="list-style-type: none"> <li>• Every n-th point</li> </ul>
	n	20

Axis definition

Open **Axis definition** in the dialog for graph and axis definition. Parameterize the **Axis system**:

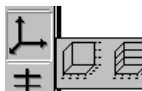
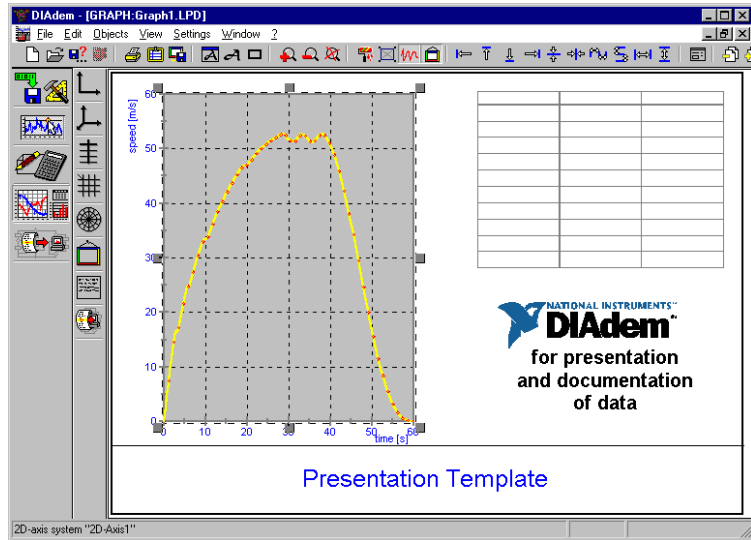
⇒ Background color: *gray*

⇒ Color: *dark gray*



If you go down to the bottom of the list of available colors, you will find the entry **other colors** where you can define customized colors. The resulting diagram should look like this:

Activated 2D axis system with defined graph



If you want to display your data in a 3D axis system, creating a curve in space, you need triple values. Thus, you must create three equally long data channels. For *surface display* an x- and y data channels as well as a z- data matrix are required.

## Subaxes: Different Value Ranges

Axis systems can contain more than one graph, even if the y-channels differ greatly in their value ranges. In that case, use axis systems with two or more different y-axes or subaxes.



Reduce the size of the axis system on your workspace. Select the 2D axis system that has two y-axes from the function bar. Place it above the first axis system.

Open the axis system and make the following settings:

<b>New entry</b>	X-Channel	<b>Time</b>
	Y-Channels	<ul style="list-style-type: none"> <li>• Speed</li> <li>• Revs</li> </ul>
<b>Curve parameters</b>	Line Weight	0 . 5
<b>Axis definition</b>	Axis system: min. line weight	0 . 7
	Display	<b>Axis</b>



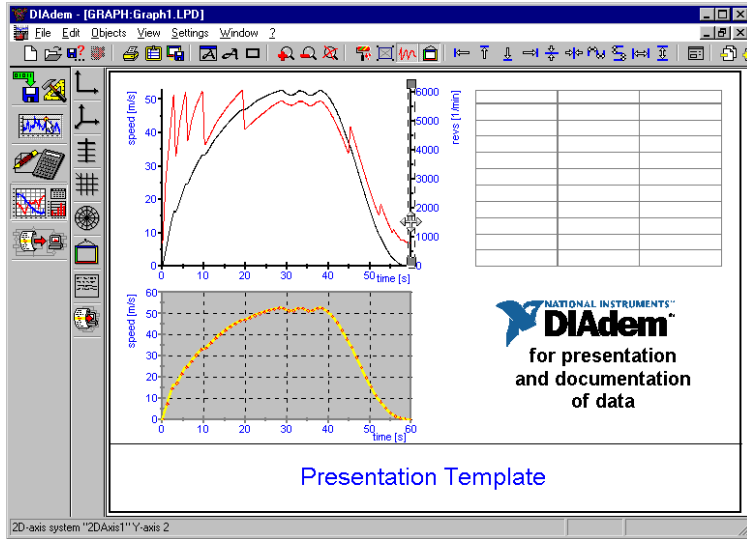
The assignment of the graphs to the y-axes is done in the Graph and axis definition dialog. In the dialog **Subaxes** you can modify existing subaxes or define additional y-axes.

Select the graph **2 Time/Revs** and allocate the **subaxis-system 2 X-1/Y-2 (0.0,0.0)** to it.

Two differently scaled graphs appear in the workspace. Subaxes can be shifted and reduced in size. When they are reduced, the display of the assigned curves changes in accordance with the reference axis.

Select the left y-subaxis and change the **Tick style to left**. Set the right y-subaxis to **right**. The diagram should now look like this.

Y-subaxes can be moved and reduced in size.



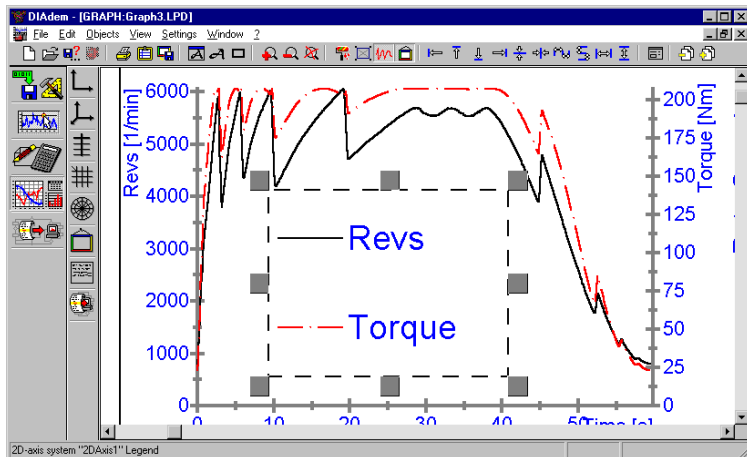
### Graph Legends: Labeling Curves

When multiple curves are displayed in an axis system, legends make their allocation to the displayed data easier.



Open the upper axis system and check **Legend**. Uncheck the **Frame** around the entire legend box. Center the legend in the upper axis system.

A curve legend can be positioned anywhere in an axis system. (GRAPH1.LPD)



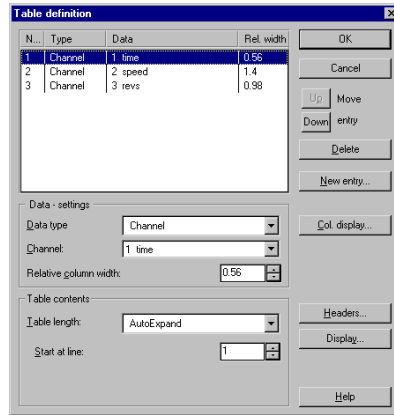
Color palette...

Note: Do not confuse the **Legend** function with the *color legend* of a 3D axis system. The color legend assigns values from the data area to the palette colors that have been selected.

## 5.4 Tables: Displaying Numeric Data

The display of numeric data in a table is similar to an axis system.

*Open the Table and in New entry allocate the first three channels Time, Speed, and Revs to the table.*



With *Up/Down* you can move the data channels listed in the table.



*In Table contents choose **AutoExpand** for Table length. This creates a table consisting of several pages in which the data is displayed consecutively on several pages. You can browse the pages using the icons in the tool bar.*

The appearance of the table can be modified in the submenus **Headings**, **Col. Display**, and **Display**.

HEADINGS		
	1. Heading	@CN ( # ) @
<b>Parameters</b>	Font attributes	<b>bold</b>

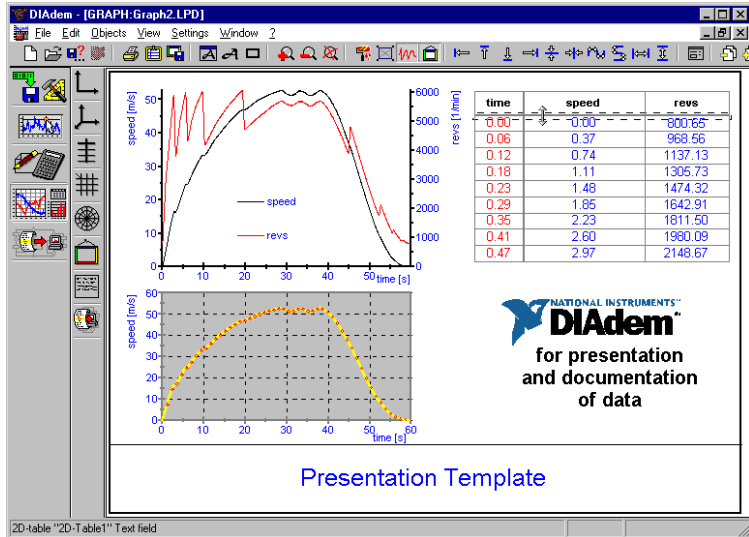
COL. DISPLAY		
Column 1	Format	<b>d . dd</b>
	font color	<b>red</b>
Columns 2 and 3	Format	<b>d . dd</b>
	font color	<b>blue</b>



You determine the channel name with the DIAdem variable *CN* and the dimension with *CD*. The @ tag causes the entry to be recognized as a variable and not as text.

A three-columned table filled with data appears next to the axis system in the diagram on your screen. Browse the various pages of the table. You can resize the column width and the row height by shifting the corresponding separating lines.

2D tables can be individually designed. (GRAPH2.LPD)



## 5.5 Texts: Labels and Comments

Technical designs usually contain a text box with the name of the displayed object, the name of the author, and the creation date in it. Also, every presentation graphic has at least one heading. In

addition, you may want to include comments and explanations for graphics. Text can be added to other objects. Such texts can be clipped within axis systems and tables.

There are two ways to insert texts in DIAdem-GRAPH: using the text editor from the tool bar or the text object from the module bar. Use the text editor for short texts with only one single line and text object for long texts with several lines that can contain more than one text format.

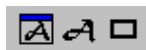
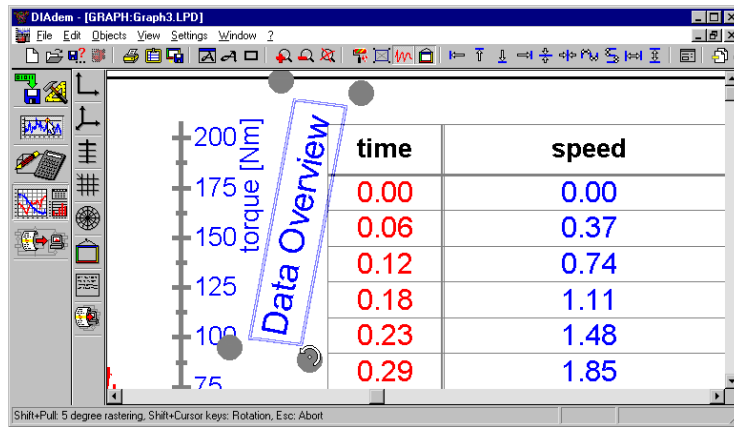
## The Text Editor



Choose the text editor from the tool bar. Click on the position in the diagram where you want to create text. To start a new line, press the <Enter> key. Each line you create like this is independent text. The text can contain formulas and DIAdem variables that are updated together with the diagram. Leave the text editor by clicking on any blank position in the workspace or press the <Esc> key. You can rotate text that you have selected by clicking on one of the sizing boxes.

*For the exercise choose the **Text editor**. Enter the text **Data Overview**. Open the dialog **Properties** and set the font to **Arial** and the color to **blue**. Rotate the text and place it vertically next to the table.*

Inserting and rotating free text (zoomed view)



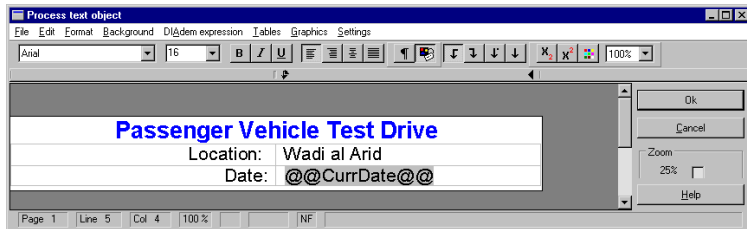
In addition to the text editor, the toolbar contains the *Text parameter* for setting global text parameters and the *Frame* function for creating frames. To draw a simple line, create a frame and set the horizontal or vertical positions to zero.

## Text objects: Continuous Text Individually Formatted

Text objects can have more than one line and have hard or soft line breaks as well as individual formats. They can be grouped into subject areas using tab settings and tables, and texts can be stored with graphics. As in the text editor, you can use DIAdem variables and formulas in text objects.



The example is to contain a heading with the name of the object, the authors name, and the creation date. Delete the title **Sample Template of a Presentation Graphic** and replace it with a **Text object**. Open the text object.



The white text field appears as large as it was created in the diagram.



For the *character format* choose from the following settings: font, font style, size, subscript, superscript, and color.



For *paragraph formatting* the following options are available: alignment, bullets, numbering, and left and right flushed tabs, etc.



Control characters and DIAdem expressions can be *displayed* and zoomed in on.

Format the *background* and insert *DIAdem expressions*, *tables*, and *graphics* by choosing these options from the menu bar.

Write a heading **Passenger Vehicle Test Drive** into the first line and insert a table with two columns and two rows below from **Tables » Insert new table**. Parameterize the table as follows:

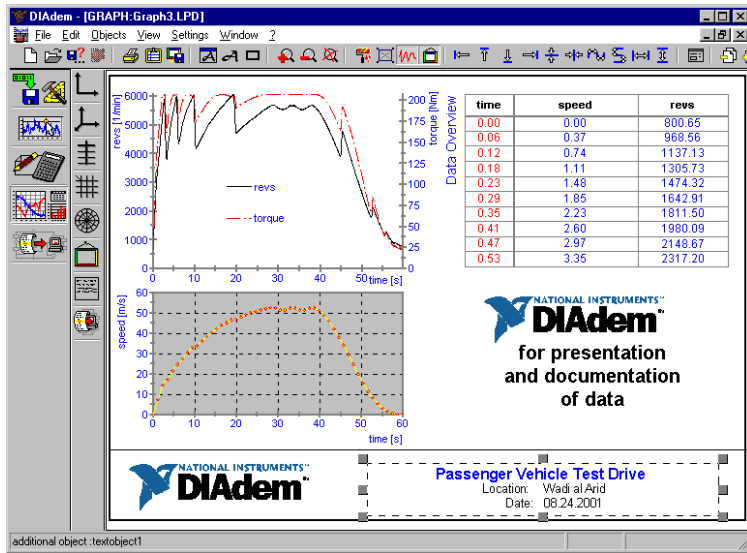


HEADING		
<b>Passenger Vehicle Test Drive</b>		
Paragraph		Centered
font size		20
font color		blue
TABLE CELLS		
Location:		Wadi al Arid
date:		DIAdem variable»New: CurrDate
font size	16	16
Paragraph	align right	align left



By entering the DIAdem variable *CurrDate* you have inserted the current date. If **Display DIAdem expressions** is enabled, the necessary control character @ appears in front and behind the entry.

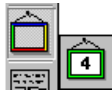
**Disable Background » Frame and enable Background » Transparent.** The diagram could now look like this.



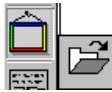
Text objects differ from the other text and axis labels with respect to their text size definitions. The text size of the text object is specified absolutely as a point size and not relative to the diagram width.

## 5.6 Illustrate: Embedding Graphics and Logos

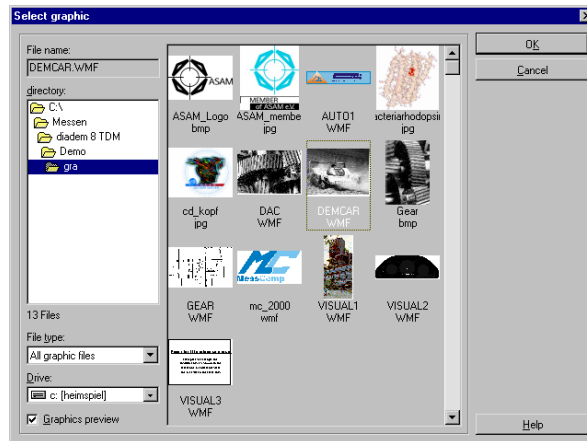
Graphics, images, or sketches add more information to diagrams. There are two ways to embed graphics in DIAdem-GRAPH: choose pre-defined icons from the function bar or load a new graphic.



Graphics that you use often can be allocated to the four icons in the *function bar*. Place the cursor over an icon, open the shortcut menu and choose Default settings to allocate a file to the icon. In DIAdem-GRAPH the National Instruments and the DIAdem logo are assigned to the icons by default.



Use **Load background graphic** to insert graphics from files into the diagram. DIAdem supports WMF, BMP, JPG, TIF (uncompressed), and GIF files.



Choose **Demcar** and place it below the table. After you delete the texts, place the DIAdem logo into the text field on the lower left.

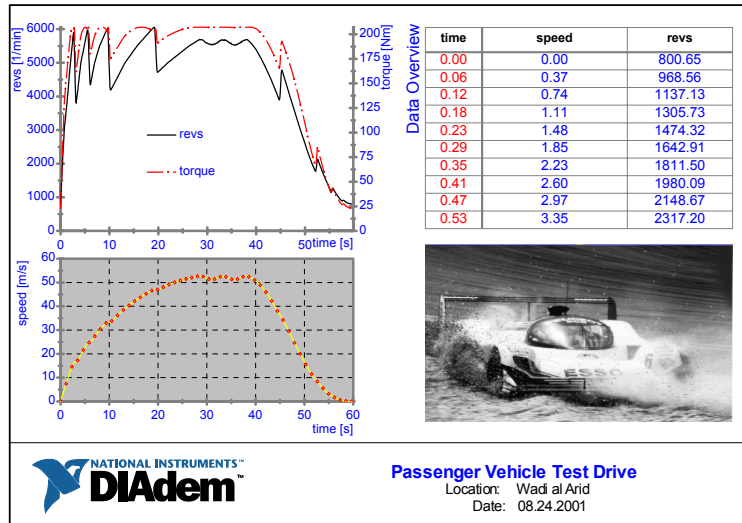
You can place graphics beneath tables and axis systems because they are transparent. To do this, you must insert the graphic before inserting an axis system for example. This is the order in which the view is refreshed.

## 5.7 Printing Diagrams and Saving Templates



You can print a diagram on any installed printer or copy it to the *clipboard* and paste it into a text document for example.

Diagram with curves, tables, images, and a text field (GRAPH3.LPD)



⇒ Use *Save as* to export a diagram as a graphic in various formats.



Also, you can save a diagram as a layout. Saving a diagram as a layout usually saves all data assignments in graphic objects. This means that you have to load a suitable set of data with the layout, for example using an Autosequence. If you want to use the layout independently of the data structure for other presentations, delete the data allocations in the objects before saving.

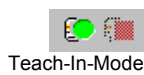
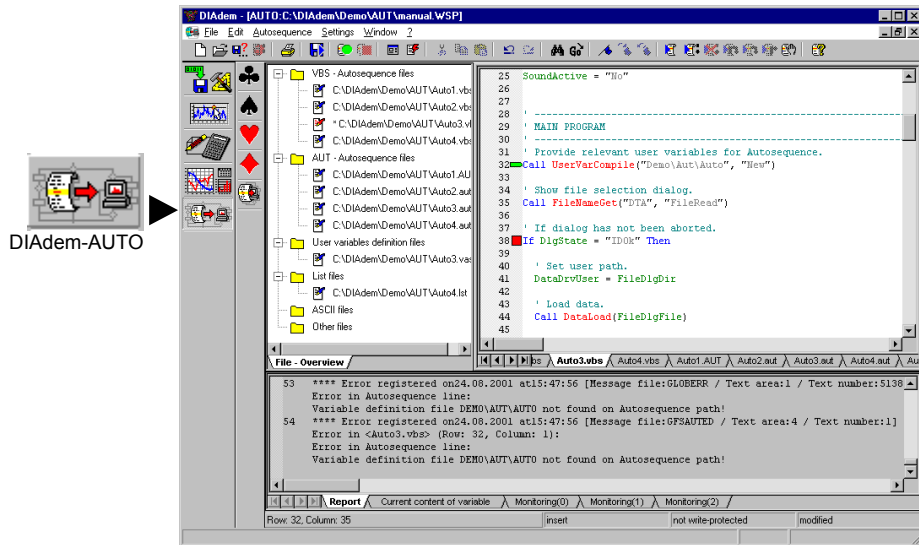


Every file can have a **Picture information**. The name of the Author and the project as well as comments make future identification easier.



# 6 DIAdem-AUTO: Automated Analysis and Report Generation

DIAdem-AUTO automates recurring work sequences as Autosequences. Autosequences consist of commands that are processed by DIAdem and VBScript.



In the TEACH-IN-MODE, the program actions that are performed interactively are registered and logged as Autosequence commands. The sequence of commands appears automatically in the editor and can be saved for repeated use.

The editor is of central importance in DIAdem-AUTO. Each Autosequence file is displayed and edited in the editor. The syntax color coding allows you to distinguish clearly between comments, commands, and variables. The Quickinfo shows the correct syntax and the context-related help provides more details on each command.

The debugger integrated into the editor helps you to find errors. You can see the current error report and monitor the most important variables in the information area.

The file overview, is a list of all the open files in a tree structure. You can save the current tree structure with all file references.

Language elements such as loops, condition-related branching, procedures, and user-defined dialogs allow you to process sequential Autosequences more flexibility.

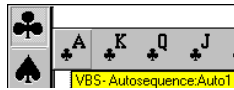


Also a dialog editor exists for you to generate complex dialogs with texts, input fields, selection fields, list fields, marking fields, radio buttons, spin boxes, graphics, and ActiveX controls.

## Starting Autosequences at the Press of a Button



The DIAdem-AUTO *module bar* contains groups of Autosequence symbols that you can assign your Autosequences to. Assign Autosequences to the function bar by clicking on the pre-settings: **Clubs function bar » Shortcut menu Ace of clubs » Presettings » AUTO1.**



Open the top function bar and position the mouse cursor on the **ace of clubs**. The Quickinfo with the name of the Autosequence now appears. Start the Autosequence.

As you can see in the example, Autosequences use functions from all the DIAdem devices.



Function bar that uses features from all DIAdem devices

The bottom function bar is available for the user in all DIAdem devices. This enables you to start the Autosequences from each of the devices. You can assign the Autosequences from all devices.

You can also use the keyboard to start Autosequences in this function bar. Use <Shift-F5> to start the Autosequence assigned to the fifth symbol.

## 6.2 The DIAdem-AUTO Workspace

The workspace in DIAdem-AUTO consists of the file overview, the editor, and the information area.

There is a tree-shaped list of the directories and files in the *File overview* on the left. You can use it to define and delete directories, define directory properties, and open, activate, and close files.

The contents of files are displayed and processed in the *Editor window* on the right. The editor provides features like syntax coloring, line numbering, bookmarks, and find and replace. The status line indicates where the cursor is currently located, whether the overwrite mode and write-protection are activated, and whether the file has been modified.

The *information area* at the bottom displays the error report and variable contents in the *debug mode*. The debugger facilitates finding and eliminating errors in Autosequences.

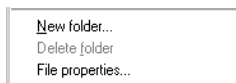


Changing the size of the editor window changes the sizes of the three areas.

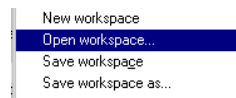
## File Overview

The file overview gives you a view of all the loaded files. The directories contain files of the same type. Click on the file you want to work on in the editor. The following standard directories are set up and cannot be deleted.

- ⇒ VBS Autosequence files (\*.VBS)
- ⇒ AUT Autosequence files (\*.AUT)
- ⇒ User variable definition files (\*.VAS)
- ⇒ List files (\*.LST)
- ⇒ ASCII files (\*.ASC):
- ⇒ Other files for storing any other file types.



If you need a directory for a frequently used file type, you can use the shortcut menu to generate one. When you do this, no files may be loaded. You also define directory properties such as syntax color coding for the editor, or the font, in this menu.



Save the workspace settings in a **workspace** (WSP) file to provide DIAdem-AUTO with the files required for each project. The **MANUAL.WSP** in the demo directory is an example. You delete the file overview with **New workspace**: the standard tree structure remains.



When you open DIAdem-AUTO a prepared workspace can be loaded immediately. To do this, store the current desktop settings by clicking on the Save icon in the toolbar, after closing all the devices with **Windows » Close all**.

## The Autosequence Editor

You can view and alter Autosequence files in the editor. Autosequences can be modified, for example, to influence the basically linear sequence of commands by control structures, to integrate request dialogs, or to enable the user to take interactive operating steps during the Autosequence.

Select the Autosequence file in the File overview. The syntax color coding allows you to distinguish clearly between comments, commands, and variables. You can specify the colors and font in the directory properties: **File overview » Shortcut menu » Directory properties » Colors/Font**.



Go to line



Find (you can find Replace in the Edit menu)



Undo and reconstruct alterations



Bookmarks are jump marks. If you insert a line, the subsequent jump marks and command lines are shifted. You can delete single jump marks with the toolbar or use the shortcut menu to delete all of them simultaneously

The *status line* shows the current cursor position and the insert or overwrite mode. It also shows whether the Autosequence has been altered or is write-protected.

## VBScript in DIAdem

In the Autosequence editor you can edit AUT and VBS Autosequences. Several of the new functions, like debugging, are based on Visual Basic Script and are only available in the VBS Autosequences. The emphasis therefore is on the VBS Autosequences here.

Up to Version 7 three script machines were developed for DIAdem: the AUT Autosequences, the control file drivers, and the internal dialog control. They differ in their language range and speed.

The Visual Basic Scripting Edition (VBScript, VBS) from Microsoft is the script engine most widely used with Windows and is used in DIAdem. VBScript has been used in the Script DAC Driver since



Version 7. The Autosequences and the new user dialogs, with logic that can be controlled with VBScript, are new VBS applications. VBS generally means:

- ⇒ Higher performance for Autosequences
- ⇒ Errors can be found more easily with the debugger
- ⇒ Function calls with parameters
- ⇒ Declaration of local variables

*Visual Basic* is a complete programming development tool. *VBA* (Visual Basic for Applications), which is familiar from Office applications, provides powerful Visual Basic tools in the environments of existing applications. VBScript is the most recent element of the Visual Basic group and includes a sub-group of the Visual Basic features.



If VBS is not installed when you install DIAdem, the script engine is automatically installed. VBS is not entered in the registry, it is only installed locally for DIAdem. Other applications can therefore not access the script engine. This means that VBS files cannot be started; the script engine is not accessible for viruses. VBS files cannot be executed by other programs until the entire VBS environment is installed. The debugger also requires this.

## 6.3 Teach-In-Mode (Learning by Doing)

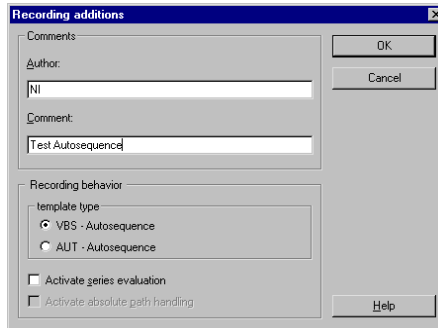
You can generate Autosequences in two ways: manually, which requires you to be familiar with Autosequence syntax and structure, or by simply activating the TEACH-IN-MODE.

When you activate the TEACH-IN-MODE the program registers the actions of the Autosequence author and logs the text commands in the editor. When the Teach-In-Mode has been deactivated, the sequence of commands can be repeated at any time by calling the Autosequence. The procedure for generating an Autosequence in the Teach-In-Mode involves three steps:

1. Activate the TEACH-IN-MODE
2. Execute the functions/operations
3. Deactivate the TEACH-IN-MODE and save the Autosequence.



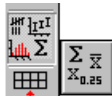
*Step 1: Activate the **TEACH-IN-MODE**. The recording details are included in the Autosequence heading.*



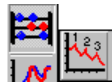
The record button in the status line indicates that all subsequent function calls will be recorded.

**Step 2a: Load *AUTO1.DAT* in *DIAdem-DATA*.**

*AUTO1.DAT* contains the measured angle deviation between the table and the ram of a gear-shaping machine and the calculated frequency spectrum of the time signal.



**Step 2b: Calculate the minimum value, maximum value and the range of the *Channel 2 deviation* in *DIAdem-CALC*. Click on **Statistics** » **Statistical characteristic values**.**



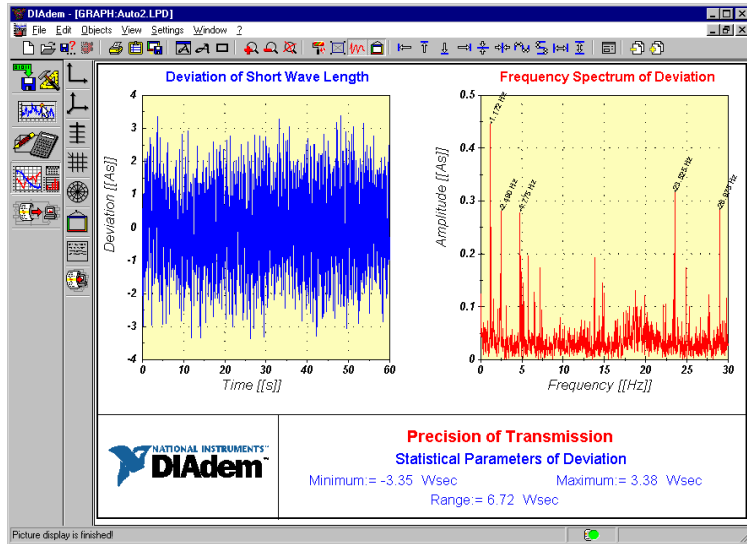
**Step 2c: Click on **Basic mathematics** » **Peak search** and make the following settings:**

X-Channel	<b>Frequency</b>
Amplitude channel	<b>Amplitude</b>
Search criterion	<b>Maximum peaks</b>
Maximum no. peaks	5

Based on the amplitudes recorded at specific frequencies and the knowledge of the gearing layout we can draw conclusions about the transmission accuracy of individual drive components.

**Step 2d: Load the graphic layout *AUTO2.LPD* in *DIAdem-GRAPH*, to display your calculation results.**

Result graphic in DIAdem-GRAPH



Deactivate Teach-In-Mode

Step 3: When the task is complete, deactivate the *TEACH-IN-MODE* in DIAdem-AUTO. You will see all recorded commands in the editor.

The recorded Autosequence is displayed in the editor (AUTO2.VBS).

```

23 STATSEL(16)      =""
24 STATSEL(17)      =""
25 STATSEL(18)      =""
26 STATSEL(19)      =""
27 STATSEL(20)      =""
28 STATSEL(21)      =""
29 STATSEL(22)      =""
30 STATCLIPCOPY     =0
31 STATCLIPVALUE    =0
32 STATFORMAT       =""
33
34 Call STATELOCKCALC("Channel","1-","2") ... STATDIPEC,ROWNOFF
35 Call CHMPEAKFIND("Frequency","Amplitude","X_Peak","Y_Peak",S,
36 Call PICLOAD("Auto2") ... PICFILE
37 Call PICUPDATE()
38
    
```

File - Overview

TeachIn(2).VBS /

```

63 **** Message registered on 24.08.2001 at 15:55:08
64 C:\MESSEN\DIADDEM 8 TDM\LIBR\GRA\EXAMPLE.LPD being loaded...
65 **** Message registered on 24.08.2001 at 15:55:08
66 Loading header file "C:\MESSEN\DIADDEM 8 TDM\LIBR\DAT\EXAMPLE.DAT" ...
67 **** Message registered on 24.08.2001 at 15:55:08
    
```

DIAdem-GRAPH: Creating presentation graphics

Click on **Execute** to ascertain whether the recorded sequence produces the result required.

If everything has run smoothly you can **save** the Autosequence as a VBS file.

Use the <Esc> key if you want to stop an Autosequence. The <Esc> key may have to be pressed several times for graphics functions or measurements.

## Teach-In-Mode Results

Now that you have generated and saved an Autosequence, we will show you how the single steps have been registered.

AUTO2.VBS

```

-----
'-- VBS - Autosequence - file
'-- Generated on 08/21/2001 16:57:20
'-- Author: NIC
'-- Comment: Test
-----
Call DATALOAD("Auto1")          '... DATAFILE
STATSEL (1)          = "No"
STATSEL (2)          = "No"
STATSEL (3)          = "No"
STATSEL (4)          = "Yes"
STATSEL (5)          = "Yes"
STATSEL (6)          = "No"
STATSEL (7)          = "No"
STATSEL (8)          = "No"
STATSEL (9)          = "No"
STATSEL (10)         = "No"
STATSEL (11)         = "No"
STATSEL (12)         = "No"
STATSEL (13)         = "Yes"
STATSEL (14)         = "No"
STATSEL (15)         = "No"
STATSEL (16)         = "No"
STATSEL (17)         = "No"
STATSEL (18)         = "No"
STATSEL (19)         = "No"
STATSEL (20)         = "No"
STATSEL (21)         = "No"
STATSEL (22)         = "No"
STATCLIPCOPY        = 0
STATCLIPVALUE       = 0
STATFORMAT          = ""

Call STATBLOCKCALC ("Channel", "1-", "2") ' STATDIREC, ROWNOSTR, CHNNOSTR
Call
CHNPEAKFIND ("Frequency", "Amplitude", "X_Peak", "Y_Peak", 5, "Max. Peaks",
"Amplitude")          '... X, Y, E, E, PEAKNO, PEAKTYPE, PEAKSORT
Call PICLOAD ("Auto2")          '... PICFILE
Call PICUPDATE ()

```

Comments

The heading includes the Autosequence extras, with the date and time it was recorded This is generated automatically in the TEACH-IN-MODE. All comments lines start with apostrophes or REM. Comment lines help to make Autosequences more comprehensible and can be inserted at any point.

Command for loading data	The command for loading the data set, <code>DataLoad("AUTO1")</code> , is in the first instruction line. DIAdem commands are called with <i>Call</i> .
Characteristic statistical Values	The following block prepares the calculation of the characteristic statistical values. As specified in the dialog, the characteristic values minimum value <i>StatSel(4)</i> , maximum value <i>StatSel(5)</i> , and range have been set on YES. The variables <i>StatClipCopy</i> , <i>StatClipValue</i> , and <i>StatFormat</i> define the result transfer to the clipboard. The <i>StatBlockCalc(...)</i> command actually calculates the characteristic values.
Peak search	The Peak search <i>ChnPeakFind(...)</i> command transfers the entries required for performing the functions, as a list of parameters. The comment lines clarify which parameter is indicated in the parentheses.
Load picture template - Display diagram	The <i>PicLoad(...)</i> command loads the picture template and <i>PicUpdate</i> displays the picture.

## Autosequence Structure and Syntax

Operation commands	<p>Operation commands perform internal program functions. The majority of these commands require parameters, which are transferred in a parameter list. This is the structure of an operation command:</p> <pre>Call command(parameter list)</pre> <p>In the parameter list, channel and variable parameters can be transferred. The parameter list must come directly after the command - it may not be separated by blank spaces!</p>
Variable commands/assignments	<p>A variable command assigns a numeric value or string to a program or user variable. The assignments are made with equal signs:</p> <pre>Variable=Formula expression</pre> <p>A formula expression is defined here as a combination of operands as defined by the operations. User variables have to be declared beforehand and have defined status.</p>
String	<p>If a command parameter or an assignment is a string, it has to be in quotation marks:</p> <pre>T1="Text "</pre>
Formula calculations	<p>Calling the DIAdem formula interpreter with <code>FormulaCalc</code> enables you to calculate entire data channels in Autosequences, not just single values.</p>

- Empty lines    Blank lines in Autosequences are ignored during processing, but improve readability.
- Procedures    To maintain clarity in the Autosequences, you can remove recurring groups of instructions in procedures. Call the procedure name to execute it.

## 6.4 The Script Debugger: Finding and Eliminating Errors

You can use the debugger to check the Autosequence scripts for errors. In the information area an error report is output and the current contents of the selected variables are displayed.



For licensing reasons, the debugger is not installed with DIAdem. The debugger icons thus first appear in the toolbar in gray. Install the script debugger from the Microsoft homepage. For more information please refer to the Online Help.



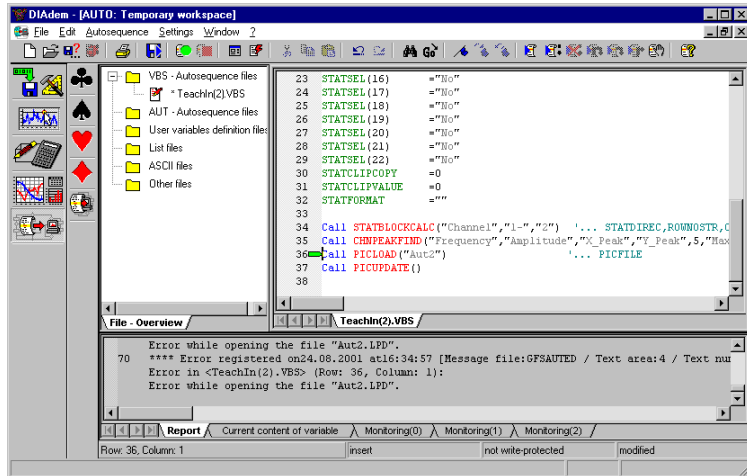
Restart DIAdem after you have installed the debugger. The debugger icon now appears next to the execute icon in the toolbar.

	<Ctrl> <F5>	Start Autosequence currently in the editor
	<F5>	Start debugger
	<F10>	Jump from line to line
	<F8>	Jump to a call and process it
		Jump to a call and stop at the end of the call
		Abort debugging
	<F9>	Set/delete breakpoints


As soon as an error occurs the debugger stops at the Autosequence line with the error and inserts a green marker. The related *error report* is located in the information area. You can use the shortcut menu to print it. Each error is logged with date, time, line number and type of error.

*Include an error: for example, alter the name of the picture template in AUT2. Then debug the Autosequence.*

Errors that occur are logged in the report window.



The report window only shows the error messages from the current session. DIAdem inserts the report window contents into the general report file, which continues on from session to session. Apart from error messages, confirmation messages and warnings are also reported. In the menu bar, the report can be viewed (? » **View report**) and configured (**Settings** » **Desktop parameters** » **Report**). These presettings are saved in the desktop file.

 Set/delete breakpoint

To enclose errors, you can set *breakpoints*, which are saved in the Autosequence file. Breakpoints are indicated by a red square next to the line number the debugger is to stop at.



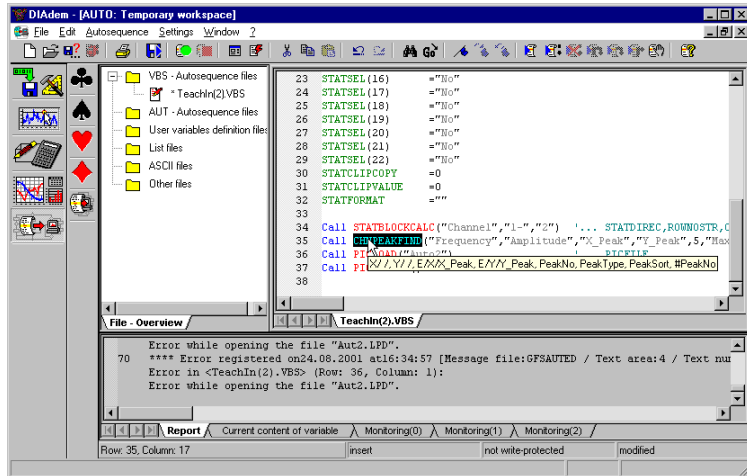
In the status line, the appropriate icon shows whether the Autosequence is in the debug mode.

## Debugging Tools

The editor provides you with various tools for the error search: Quickinfo, context-related help calls and the overview of commands and variables.

You can use the Quickinfo to call the parameters for the particular command or variable. For example, double click on the *ChnPeakFind* command and hold the cursor over it for a moment. The associated list of parameters then appears as a Quickinfo. In the debug mode, the current contents of variables are displayed.

Parameter list as Quickinfo

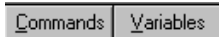
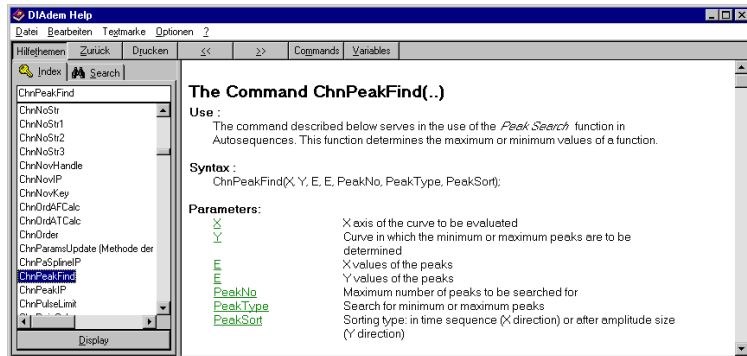


Online help

If the Quickinfo is insufficient, you can call a detailed description of the command in the *online help*.

Press <F1> and the online help appears for the marked command **ChnPeakFind**. The command and all its parameters are explained here.

Context sensitive jump to the online help



Command and variable index

The online help toolbar contains two buttons for the *command index* and the *variable index*. All the DIAdem commands and variables are listed alphabetically. If you only have a rough knowledge of the command, you can use the first letter to find the correct spelling and the parameter list.



## The Information Area

The DIAdem-AUTO information area is used to monitor the contents of variables. Use the index cards to choose among the *current variable contents* in one of the three *monitor windows*. This enables you to display various variable contents from different areas of the Autosequence separately.

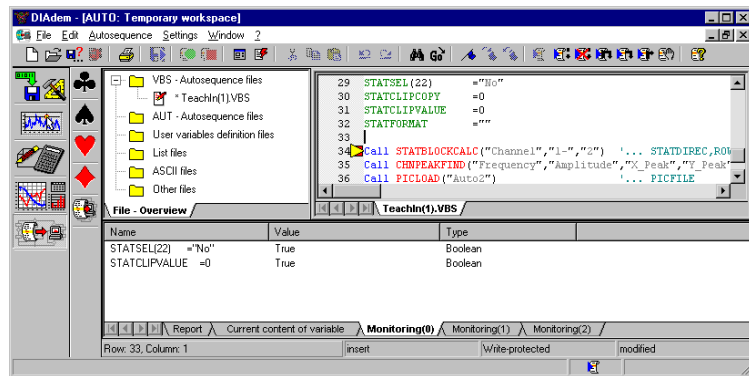


Select the first monitor window. Mark the **StatSel (22)** variable from statistical calculations, and use *Drag&Drop* to copy it to the monitor window. Then add **StatClipValue**.



Now set a breakpoint in the script at the end of the calculation and debug the Autosequence.

Variable contents are displayed in the info area



The current value and type of the variable appear after the variable name in the monitor window. If the variable does not have a value at this point, this is indicated by a question mark. You can use the shortcut menu to delete and add entries, and to modify the name and value.

## 6.5 Customizing DIAdem

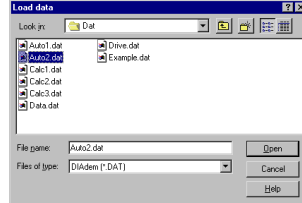
Customizing is what we call the tailoring of the standard DIAdem software for special applications. This is a solution for users that are neither familiar with DIAdem nor with the task at hand. To show what we mean, we will expand the example Autosequence to include the following features:

- ⇒ *Creating DIAdem standard dialog for interactive data set selection*
- ⇒ *Creating a user dialog for entering operational parameters.*
- ⇒ *Using special variables*
- ⇒ *Converting channels*
- ⇒ *Manipulating graphics layouts*

### DIAdem Standard Dialog: Interactive File Selection

If you want to select a file in an Autosequence, call the standard DIAdem dialog *FileNameGet*.

Tips for loading files:



This is the dialog call for file selection:

Loading a data file  
(Part of AUTO3)

```
' Open dialog for loading data.
Call FileNameGet("DATA", "FileRead")
' If data loading dialog was not aborted.
IfDlgState = "IDOk" Then
    ' Set user path.
    DataDrvUser := FileDlgDir;
    ' Load data set.
    Call DataLoad(FileDlgFile)
.
Else
    ' If the data set is o.k.
    Call MsgBoxDisp("No file was loaded!
                    Autosequence ended.", "MB_Ok")
End If
```

The *FileNameGet* command has these parameters:

<b>FileNameGet</b> (FileTargetDevice, FileOperation)	
FileTargetDevice	Device for which the file is chosen ('DATA', 'GRAPH')
FileOperation	Target for file selection ('FileRead' for file open, 'FileWrite' for file save)



The *FileNameGet* command has other optional parameters for loading or saving any file types. Start the online help with !.

Select the file name, confirm with **Open** and the entries will be assigned to three variables. *DlgState* contains the status in which the dialog was left. *FileDlgDir* contains the data path and *FileDlgFile* has the name of the selected data file. In the example, these commands and variables are integrated in several *If.. then.. else* structures in order to adapt the course of the Autosequence to the user entries.

If request The first *If request* checks the contents of the **DlgState** variable. If data loading was aborted, **IDCancel** is output as the status and you jump to the appropriate else instruction. Here you will see the **MsgBoxDisp** command, which displays a message box.

If the status is **IDOk**, since the dialog was ended with Open, the data path becomes the new user path:

```
DataDrvUser=FileDlgDir .
```

The data file, the name of which is read from the **FileDlgFile** variable, is now read:

```
Call DataLoad(FileDlgFile) .
```

## Defining User Dialogs

The second addition to the example shows how user entries can be read using user-defined dialogs. The entries can be used further on in the Autosequence for calculations and diagram labeling, for example.

The scriptable user dialogs not only have all the standard control and input elements, they can also be extended with VBS to include logical links. These scriptable dialogs are saved as SUD files.



You can also define other less complex user dialogs in DIAdem with other editors. You can specify which dialog editor you want to use by clicking on **Settings » Desktop parameters » Device-specific » Dialog editor**.

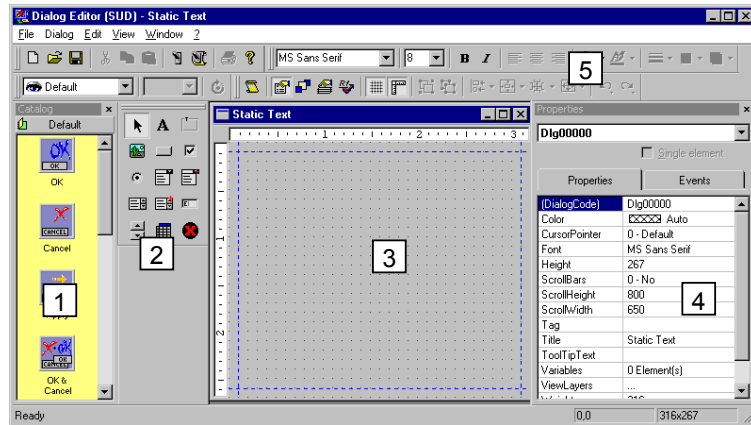
The *prepared user dialog* shows the easiest kind of dialog definition. In a table, you can enter the heading, assign labels to variables and define the function of the OK button. The Cancel button is defined. This definition is saved as a dialog Autosequence in the AUT format.

Define extensive dialogs with the *Alphanumeric editor* or the *Graphic editor*. In the alphanumeric editor you generate each dialog object using input dialogs in which you enter the parameters. In the graphic editor you define all the objects with the mouse and set the parameters in the properties window. Both of the editors enable you to define static texts, input fields, selection fields, list fields, selection fields, input and display tables, and group frames and graphics. The dialogs are saved in COD files that may contain several dialogs.

The best way to describe your dialog depends on the functionality the dialog requires. The scriptable user dialogs (SUD), which are described below, provide the most extensive features.

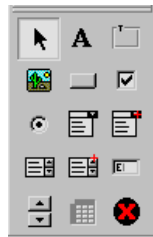








Open the dialog editor using the toolbar.










The dialog editor consists of the work area (3), the properties window (4), the control element bar (2) and the catalog window (1). The toolbars (5) for defining layouts and parameters for the control elements, are located above these.

The control panel contains all the objects for the dialogs.



















-  Mouse pointer for selecting and shifting the control elements
-  Text
-  Frames
-  Bitmap graphic (BMP, ICO, DIP)
-  Button (for example, OK, Cancel, Execute, Help)
-  Checkbox multiple selection

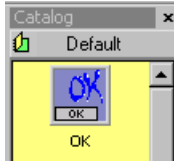
-  Radio button for choosing one of several items
-  Selection (combobox and enumcombobox)
-  Selection list that can be switched through (listbox and enumlistbox)
-  Input field (editbox)
-  Spinbox
-  Table
-  ActiveX

Layout bar



The *Layout bar* has the following editor functions.

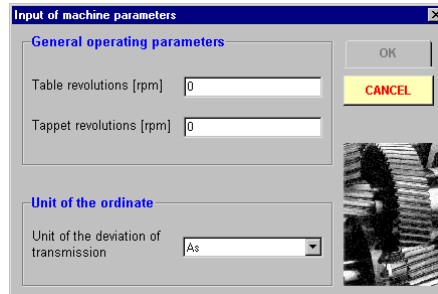
	<Ctrl-XI>	<b>Preview on/off:</b> after saving the dialog you can check its design.
	<F5>	<b>Preview and test on/off:</b> you can check the functions as well as the design.
	<CTRL-E>	<b>Script editor on/off:</b> for generating scripts for control elements
	<Ctrl-L>	<b>Property window on/off:</b> set parameters for control elements
	<Ctrl-D>	<b>Tabulator sequence:</b> list of all elements
		Add or delete <b>levels</b>
		<b>Grid on/off</b>
		<b>Ruler on/off</b>
		<b>Group control elements</b>
		<b>Ungroup elements</b>
		Move to <b>foreground or background</b>
The first marked object provides reference point and reference size		
		<b>Align to the edges</b>
		<b>Center</b> vertically and horizontally
		<b>Arrange</b> equally vertically and horizontally
		<b>Equalize</b> size
		<b>Undo and reconstruct</b>



Scriptable user dialog  
AUTO3.SUD

You can save parts of a dialog, including all the parameters and scripts, as a *catalog*. Activate the required objects and move them into the catalog window, while keeping the <Alt> key pressed. Catalogs are saved as CAT files.

For the example, rpms have to be entered and the y-axis dimension specified in the following dialog.



1. Modify the following template properties:

DialogCode	<b>Example</b>
Title	<b>Enter operating parameters</b>



2. Create a frame in the upper left area and change the following settings:

Text	<b>General operating parameters</b>
Font	<b>Arial bold</b>
ForeColor	<b>Blue</b>



- 2a. Insert two **texts** in the frame:  
**Bench rpm [revolutions/min]**  
**Ram rpm [revolutions/min]**



- 2b. Position an **input field** next to it.



3. Set a second frame underneath the first one.

Text	<b>Unit of the y-axis</b>
Font	<b>Arial bold</b>
Forecolor	<b>Blue</b>

- 3a. Insert a **text**:  
**Display a transfer deviation**



- 3b. Place an **EnumComboBox** to the right of the text.

4. Now insert two more **buttons** on the upper right. Insert the following properties for the **OK** button and the **Cancel** button:

Parameters	OK button	Cancel button
ObjectCode/Name	OK	Cancel
BackColor	-	Yellow
ButtonShape	1 - inactive	-
ButtonType	4 - OK	0 - Cancel
Enable	0 - No	1 - Yes
Font	Arial bold	
ForeColor	-	Red
Text	OK	CANCEL



5. Position a graphic beneath the buttons. When the dialog opens, load the graphic **GEAR.BMP** from the graphic directory in the demo path.

To define the dialog functionality, first enter the values for the variables containing the rpm and the y-axis unit values.



6. Without selecting a control element, click on the dialog and open the variable list in the **Variables** properties window. Enter the following list of variables.

Name	Type	Storage mode
R1	Scalar	Real
R2	Scalar	Real
Unit_	Scalar	Enumeration



This dialog is only for entering the variables. They are defined and activated in the following section.



A script is now written, which does not release the OK button until the parameters have been entered in the fields. The OK button is therefore defined to be initially inactive, in step 4.

7. Switch to the script editor by double clicking on the top **EditBox1** input field. Select the **EventChange** function and enter the following three lines in bold type. The script activates the OK button and modifies the button and font colors. Repeat the entry for the **EditBox2** input field.

```
Sub EditBox1_EventChange()
Dim This : Set This = EditBox1
    OK.Enable=1
    OK.ButtonShape=0
    OK.ForeColor=rgb(0,128,0)
End Sub
```

Load and display dialog     Since an SUD file may contain several dialogs, you usually need the file name and the dialog name to call the dialog:

```
SUDDlgShow(Dialog name, file name)
```

### Defining and activating variables

DIAdem has many variables, with which you can set parameters for functions or commands, or even update graphic layouts. There are three types of variables: system, temporary, and user variables.

- ⇒ **System variables** are all the command-specific variables in DIAdem that are used for results or to set parameters for functions. These variables can always be accessed. One such variable is the key for calculating the statistical characteristic value, maximum value (*Statse1(4) = 'Yes'*).
- ⇒ **Temporary variables** differ from system variables in that they are not assigned to any function and can be used for any purpose.

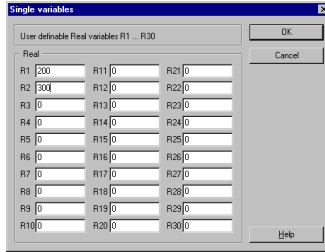
Type	Name	Value range
Boolean	B1... B10	0 or 1
Long Integer	L1... L30 LV1[1.. 15], LV2[1.. 15], LV3[1.. 15]	-2147483648... 2147483647
Real	R1... R30 RV1[1.. 15], LV2[1.. 15], LV3[1.. 15]	-1E35... -1E35...
Texts	T1... T10 TV1[1.. 10]	max. 40 characters max. 16 characters
Size lists	G1... G10 GV1[1.. 10]	max. 16 characters max. 16 characters

Choose the real auxiliary variable dialog **Edit » Temporary variables » Real**. Assign the two real variables assigned in the dialog definition:

R1	200
R2	300



Temporary variables do not have to be activated.



⇒ **User variables** are tailored for specific tasks in variable definition files (VAS files). They have to be activated for them to become accessible. To avoid delays, do not use user-defined variables as counter variables in Autosequence loops or in online calculations. In the dialog definition, you have entered `Unit_` as the enumeration variable in the combobox. You can select text or numbers in the dialog.



Load the variable definition file **AUTO3.VAS**, which has been prepared. Search for the following definition line in the editor:

<b>Unit_</b>	<b>: A (Asec', 'Degree') &lt;'Asec'&gt;</b>
Unit_	The underline _ designates the variable name.
A	Enumeration variable type
('Asec', 'Degree')	Selection list
<'Asec'>:	Presetting



Activating user variables

You can activate user variables either manually from the toolbar or by using the Autosequence command.

<b>UserVarCompile</b> (DefFileName, UserVarMode)	
'DefFileName'	Name of the VAS file without extension Text variables cannot be used
'UserVarMode'	Mode: New – Renew variable stock Append – Add variable stock

## Channel Conversion

In the example, the numeric values in data channel 2 have to be divided by 3,600 to convert the transfer deviation from angular seconds to degrees. Alter the unit in the data channel too:

```
FormulaCalc CH(2)=CH(2)/3600
ChnDim(2)=" [Degree] "
```

You use an If-control structure to request the unit selected in the user dialog.

Excerpt from AUTO3

```
' If unit is specified in degrees.
If unit_ = "degrees" Then
' Convert second channel.
Call FormulaCalc("Ch(2) := Ch(2) / 3600")
' Set unit.
ChnDim(2) = "[Degrees]"
End If
```

## Manipulating Graphics Layouts

By manipulating graphic layouts in Autosequences, you can adapt diagrams to your current data or task. You can modify graphic elements such as graphs, texts, or axis parameters while the application is running. In the example, several text parameters are set in association with the *Unit\_* variable. Set the graphic parameters like this:

```
Open graphics object
    Set graphics parameters
    Open sub-object
        Set parameters
    Close sub-object
Close graphics object
```

In the example, the manipulation of the text parameters is separated off into a procedure. This maintains clarity and allows you to use this procedure at several points in the Autosequence.

Procedure for manipulating graphic parameters (excerpt from AUTO3)

```
' -----
' Name:      Set_Text_Par
' Purpose:   set text parameters.
' Parameters: None
' -----
Sub Set_Text_Par()
' Open graphic object for parameterization, set text and close
graphic object.
Call GraphObjOpen(TxtObj(2))
    TxtTxt = "Bench rpm=" & Str(R1) & " rpm"
Call GraphObjClose(TxtObj(2))
' Open graphic object for parameterization, set text and close
graphic object.
Call GraphObjOpen(TxtObj(2))
    TxtTxt = "Ram rpm=" & Str(R1) & " rpm"
Call GraphObjClose(TxtObj(2))
' Open graphic object for parameterization, set text and close
graphic object.
Call GraphObjOpen(TxtObj(7))
' Set text in association with transfer deviation.
' If transfer deviation in Asec.
If unit_ = "Asec" Then
```

```

    TxtTxt = "Range=" & Str(StatRange, "d.ddd") & " Asec"
' If transfer deviation in degrees.
Else
    TxtTxt = "Range=" & Str(StatRange, "d.ddde") & " Degrees"
End If
' Close graphics object.
Call GraphObjClose(TxtObj(7))
End Sub

```

Procedures are called by name. In the example, the graphic layout to be manipulated, Auto3.LPD, is loaded before Set\_Text\_Par is called. Before the procedure is called, the graphics display is updated so that the new parameters are included.

Calling the procedure  
for graphics manipulation

```

' Load graphics file.
Call PicLoad("Demo\Gra\Auto3")
' Set text parameters.
Call Set_Text_Par
' Display graphic.
Call PicUpdate

```



Objects are inserted and parameters set for objects in the same way in DIAdem-VIEW layouts.

## Listing for Example AUTO3

```

-----
' Name:          Auto3.vbs
' Copyright(c):  National Instruments
' Date generated: 24.04.2001
' Time generated: 08:03:22
' Author:        RR, AT
' Purpose:       example Autosequence (user manual)
' Description:   complete manual example:
'               Monitor transfer precision for drive components
' -----
Option Explicit
' Initialize Autosequence.
' Suppress display of the current command.
AutoEcho = "No"    ' No acoustic signals.
SoundActive = "No"
' -----
' MAIN SECTION
' -----
' Initialize user variable.
Call UserVarCompile("Demo\Aut\Auto3", "New")
' Open dialog for loading data.
Call FileNameGet("DATA", "FileRead")
' If data loading dialog was not aborted.
If DlgState = "IDOk" Then
' Set user path.
DataDrvUser := FileDlgDir;
' Load data set.
Call DataLoad(FileDlgFile)

```

```

' If the data set is o.k.
If DataSetTitle = "Autdemo" Then
  ' Display dialog.
  Call SudDlgShow("Example")
  ' If dialog was not aborted.
  If DlgState = "IDOk" Then
    ' If unit is specified in degrees.
    If unit_ = "degrees" Then
      ' Convert second channel.
      Call FormulaCalc("Ch(2) := Ch(2) / 3600")
      ' Set unit.
      ChnDim(2) = "[Degrees]"
    End If
    ' Calculate statistical characteristic values.
    Call Stat_characteristicvalues
    ' Peak search.
    Call ChnPeakFind(3, 4, 5, 6, 5, _
      "Max.Peaks", "Amplitude")
    ' Load graphics file.
    Call PicLoad("Demo\Gra\Auto3")
    ' Set text parameters.
    Call Set_Text_Par
    ' Display graphic.
    Call PicUpdate
    ' If dialog was not aborted.
    Else
      ' Output message.
      Call MsgBoxDisp("Do not enter parameters! Autosequence
        ended.", "MB_Ok")
    End If
  ' If the data set is not o.k.
  Else
    ' Output message.
    Call MsgBoxDisp("No data set for evaluation.", "MB_Ok")
  End If
' If data loading dialog was aborted.
Else
  ' If the data set is o.k.
  Call MsgBoxDisp("No file was loaded! Autosequence ended.", "MB_Ok")
End If
' -----
' PROCEDURES
' -----
' Name:      Stat_characteristicvalues
' Purpose:   Calculate statistical characteritic values.
' Parameters: None
' -----
Sub Stat_characteristicvalues()
  ' Set selection of statistic characteristic values.
  StatSel(1) = "No"
  StatSel(2) = "No"
  StatSel(3) = "No"
  StatSel(4) = "Yes"           ' Calculate minimum value
  StatSel(5) = "Yes"         ' Calculate maximum values
  StatSel(6) = "No"
  StatSel(7) = "No"
  StatSel(8) = "No"

```

```

StatSel(9) = "No"
StatSel(10) = "No"
StatSel(11) = "No"
StatSel(12) = "No"
StatSel(13) = "Yes"           ' Calculate range
StatSel(14) = "No"
StatSel(15) = "No"
StatSel(16) = "No"
StatSel(17) = "No"
StatSel(18) = "No"
StatSel(19) = "No"
StatSel(20) = "No"
StatSel(21) = "No"
StatSel(22) = "No"

' Set other parameters for statistical evaluation.
StatClipCopy = 0
StatClipValue = 0
StatFormat = ""

' Calculate statistical characteristic values.
Call StatBlockCalc("Channel", "1-", "2")

End Sub

' -----
' Name:      Set_Text_Par
' Purpose:   set text parameters.
' Parameters: None
' -----

Sub Set_Text_Par()

' Open graphic object for parameterization, set text and close
graphic object.
Call GraphObjOpen(TxtObj(2))
  TxtTxt = "Bench rpm=" & Str(R1) & " rpm"
Call GraphObjClose(TxtObj(2))

' Open graphic object for parameterization, set text and close
graphic object.
Call GraphObjOpen(TxtObj(2))
  TxtTxt = "Ram rpm=" & Str(R1) & " rpm"
Call GraphObjClose(TxtObj(2))

' Open graphic object for parameterization, set text and close
graphic object.
Call GraphObjOpen(TxtObj(7))

' Set text in association with transfer deviation.
' If transfer deviation in Asec.
If unit_ = "Asec" Then
  TxtTxt = "Range=" & Str(StatRange, "d.ddd") & " Asec"

' If transfer deviation in degrees.
Else
  TxtTxt = "Range=" & Str(StatRange, "d.ddde") & " Degrees"
End If

' Close graphics object.
Call GraphObjClose(TxtObj(7))

End Sub

```

## 6.6 Additional Autosequence Features

### Important Commands and Variables

The following is a selection of important Autosequence commands and variables. If you have double clicked on a command or variable in the editor to mark it, stop the mouse pointer for a moment and the quickinfo will be displayed. You can now use <F1> to branch into the help. The command and variable index provides a complete overview in the online help.

Commands

Command	Purpose
AutolconSet(...)	Assign Autosequences to symbols
AutoQuit(...)	Stops an Autosequence
ChnAlloc(...)	Reserves and names channels
ChnGet(...)	Selects channels in Autosequences
DirLstWrite(...)	Creates a list file
ExtProgram(...)	Calls an external program
FileClose(...)	Closes an ASCII file
FileEx(...)	Checks whether an ASCII file exists
FileNameGet(...)	Calls a standard dialog box to enter a file name
FileOpen(...)	Opens an ASCII file
FileWriteLn(...)	Writes an ASCII line to a file
FR(...)	Reads text lines from an ASCII file
KeyWait	Pauses an Autosequence until next keystroke
InterActionOn	Interrupts an Autosequence
MsgBoxDisp(...)	Displays a user message in a message box
MsgLineDisp(...)	Displays a user message in a Status line
Pause(...)	Pauses an Autosequence (pause length in seconds)
SUDDlgShow(...)	Loads and displays an SUD user dialog
UserDlgOpen	Opens a user dialog
UserVarCompile(...)	Activates a user variable file
WndShow(...)	Opens/closes DIAdem devices

Variables

Variable	Purpose
ActDrv	Deals with current drive path
AutoAbort	Activates the <Esc> key for aborting the Autosequence
AutoEcho	Displays the commands during an Autosequence
AutoIgnoreError	Ignore Autosequence error messages
ChnCommentOver	Operations overwrite channel comments
ChnNameOver	Operations overwrite channel names
DlgState	Dialog status after it has been left
LastErrorNo	Number of the error that occurred last
MsgState	Final status of a message
SoundActive	Activates acoustic signal output

## Control Structures: Loops and Requests

You control the course of the Autosequence with control structures like counter loops, condition-related loops, and simple and complex branching.

Condition-dependent loops

```
while...do...           (reject)
repeat...until         (no reject)
```

Simple branching

```
if...then
if...then...else       (with alternative)
```

Compound branching

```
case...of...end
case...of...else...end (with alternative)
```

Adhere to the following rules when using control structures:

- ⇒ *One instruction per line*
- ⇒ *Begin and end are always in separate lines.*
- ⇒ *Indent instructions in branches and loops.*
- ⇒ *Write consistently, for example, keywords are always in the upper case*

## Working with Channel Designators

The channel can be referenced either by channel name or channel number when Autosequences are recorded. Select the channel reference by clicking on **Settings » Desktop parameters » Device-specific » Other settings**.

For example, if the speed (channel 2) is to be differentiated according to time (channel 1), the TEACH-IN-MODE returns the following Autosequence line, depending on which channel reference has been selected:

Channel numbers	<code>Call CHNDIFFERENTIATE (1, 2, 5, 6)</code>
Channel names	<code>Call CHNDIFFERENTIATE ("Time", "Speed", "X_Diff", "Y_Diff")</code>

When you work with channel numbers, the x- and y-channels that are to be evaluated, and the two result channels, are specified by their numbers. When you are working with channel names, all channels are specified by name.

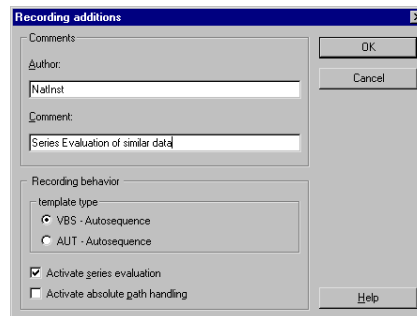
## Serial Evaluations

Serial evaluation means that several of the same type of data sets are analyzed and documented identically. It can come into effect when the Autosequence is recorded.



Activate the **TEACH-IN-MODE** and click on **Activate serial evaluation** in the recording additions.

Serial evaluation is activated in the additional information.







With serial evaluation you can click on *absolute path handling*, which logs all loaded files (picture template, data files, block diagrams) with the complete path: for example `C:\diadem\demo\dat\auto1.dat`. If absolute path handling is not activated, the path is specified relatively, using the current path definition in the desktop settings. Relative path handling is recommended when Autosequences are used on different PCs.

When a serial evaluation is created in the `TEACH-IN-MODE`, a list file is generated with the same name as the Autosequence file. The `LST` file contains a list of the names of all the files to be processed by the Autosequence. In the Autosequence, there is a reference to the list file after the loading command `DataLoad`.

**Call DataLoad("UseFileList")**



Mind the order of the file names in the list file if you use several file load or file save commands. If a load command and later a save command, is used in an Autosequence with `UseFileList`, the first file listed will be loaded and the result will be saved under the second name in the list.

You can modify existing Autosequences to use them in serial evaluation.

- (1) Secure your Autosequence file by saving it.
- (2) Create a list file with the same name, for example by copying an existing list file.
- (3) Replace the file names after loading and storing commands in the Autosequence by `UseFileList`.
- (4) Enter the files to be processed in the list file.

Alter the **AUTO2** Autosequence accordingly.

Modify loading command	<b>Call DataLoad("UseFileList")</b>
Insert pause	<b>Call Pause(3)</b>
Create list file	<b>ASCII file with the extension LST</b>
Edit list file	<b>Auto1 Auto2</b>
Save under same name	<b>Test.VBS Test.LST</b>



If you start the **TEST** Autosequence, the status line shows you the paused second countdown before the second evaluation is processed. See examples **AUTO4.VBS** and **AUTO4.LST** in the demo directory.

## Interaction Mode



You can use the interaction mode to interrupt an Autosequence and work interactively in DIAdem. In the Autosequence, for example the *InterActionOn* command is called at the appropriate point to view the loaded data manually in DIAdem-VIEW. As soon as you click on the **End interaction** switch in the toolbar, the Autosequence continues. During the interaction an Autosequence can be called again, in turn initiating interaction. An example of the interaction mode is provided in the Autosequence **VIEW\_FFT**.

## Full-Screen Mode

The full-screen mode described for DIAdem-VISUAL can also be used in Autosequences. Instead of <Ctrl-U>, the command *WndShow* is used in the Autosequence for activation and deactivation.

<b>WndShow</b> ( "WndName" , "WndMode" )	
WndName	Name of the DIAdem device (SHELL, DATA, GRAPH, VIEW, CALC, AUTO)
WndMode	Opening mode
	Open: Open device
	Show: Activate device
	Maximize: Maximize device
	Minimize: Iconize device
	Plain: Device not maximized
	Fullsize: Generate full-screen

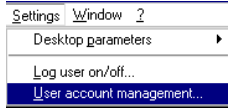
The full-screen mode has to be called before the display is started.

Activate the full-Screen mode before the start of a display

```
call WndShow("SHELL","Fullsize") ' Activate full-screen mode
call PicLoad("GRAPH3")          ' Load template
call PicUpdate                   ' Refresh
call WndShow("SHELL","Normal")  ' Deactivate full-screen
mode
```

## User Management

The user management can be used in Autosequences to protect applications generated in DIAdem-AUTO from unauthorized access. The DIAdem administrator sets up user accounts with rights and passwords, which are saved in a coded ADM file. In the Autosequence, the features to be protected are provided with the appropriate requests. The test rig user has to log in and can only perform the functions granted in the user's account.



The DIAdem administrator provides user accounts with the appropriate rights and generates new rights for projects with **Settings » User management**. First you log in: name Administrator, password diadem. See the Online Help for more details.

The following user management functions can be performed by Autosequences.

<b>ADMLogIn;</b>	Logs users in and out
<b>AskADMPermission(ADMRight);</b>	Requests the authorization
ADMRight	Name of the right
ADMCurUser	Current user the request is for
ADMPermission	Boolean variable with the request result
<b>ADMAdmin;</b>	User management with log in at first call
<b>ADMCreateRight (ADMRight,ADMDescription)</b>	Generate right
ADMRight	Name of the right
ADMDescription	Description of the right

The following Autosequence excerpt defines a new right and requests the authorization from the user account.

```
'Creating a right for the Alpha project
ADMRight= "Project Alpha"
ADMDescription= "Right to process Alpha project Autosequences"
call ADMCreateRight(ADMRight, ADMDescription)

'Request authorization of user currently logged in
call AskADMPermission("Project Alpha");
if ADMPermission then
    call MsgBoxDisp("You may")
else
    call MsgBoxDisp("You have no Alpha project rights")
end if
```

## 6.7 Interface Functions

You can use various DIAdem interfaces in Autosequences. OLE enables you to communicate with other Windows applications or a second DIAdem, and ODBC/SQL enables you to read and write in databases.

### OLE Interface



DIAdem exchanges data and instructions with other Windows applications via the OLE interface (Object Linking and Embedding). DIAdem can be the client as well as the server. Using OLE, DIAdem can be accessed by another program or another instance of DIAdem.

OLE connections are made with the Autosequence command *CreateObject*. This is a VBS command that is called without *Call*.

<b>CreateObject (interface)</b>	Initialize OLE link
interface:	
<b>DIAdem.TOCommand</b>	Command interface and access to single variables.
<b>DIAdem.TODataSheet</b>	Data interface



The DIAdem OLE interface provides two interfaces: the Command interface and the Data Sheet interface. The program directory contains the type library DIAdem.TLB, from which all the necessary OLE information comes (method names, parameter lists, etc.).

The Help contains OLE application examples for DIAdem (**Help » Examples » Application examples » Interfaces**).

### ODBC/SQL Interface for Database Connection

You can use the ODBC/SQL interface to connect DIAdem to your database and manage your data with any additional details. These functions are based on the ODBC (Open Data Base Connectivity) database interface, with SQL (Structured Query Language) as the manufacturer-independent query language. These Autosequence commands all start with the letters SQL.

Example Autosequence for database connection

```
' Connection to the ODBC data source Dbl as user1
call SQL_Connect("DB1","user1","","")
call SQL_AutoCommit(0)
call SQL_Select("Time,Pressure,Acceleration","Measurement23","Time>10
ORDER BY time")
T1 = SQL_Result(2,3)
T2 = SQL_ColName(1)
call SQL_BindVar("Pressure","TestVariable","R","APPEND")
L1 = SQL_ResultRows
call SQL_BindChannel("3","Acceleration","n")
call SQL_BindChannel("Time","n")
do while "SQL_Next"
  call SQL_FetChNext()
loop
call SQL_Update("Measurement23","Pressure","0.7","Time=7","")
call SQL_Transact("Rollback")
call SQL_Update("Measurement23","Pressure","7.0","Time=7","")
call SQL_Transact("Commit")
call SQL_Insert("Measurement23","Time,Pressure","1200,0.12")
call SQL_Transact("Commit")
call SQL_Delete("Measurement23","Time > 1000")
call SQL_Transact("Commit")
L1 = SQL_ResultRows
call SQL_ExecDirect("DROP TABLE Measurement23","","","")
call SQL_Disconnect
```

## 6.8 Autosequence Converter from AUT to VBS

In DIAdem-AUTO you can load, edit, and generate AUT Autosequences and VBS Autosequences. You can use the converter to convert an AUT Autosequence into a VBS Autosequence, so you can use features such as the debugger. Call the converter by clicking on **Autosequence » AUT/VBS Autosequence converter**.

The two types of Autosequence have the following differences:

	AUT	VBS
User variables	Uvariable\$	Uvariable_
Command call	Command name()	Call command name()
Formulas	Formula	FormulaCalc("Formula")
End of line	With semicolon	Without special characters
Assignment	:=	=

## Formulas

VBS cannot perform channel operations directly. Single point operations are defined directly by using the equal sign, and channel operations are performed in VBS syntax by the *FormulaCalc* command.

	AUT	VBS
Single point operation	L1:=Len("test")+3;	L1=Len("test")+3
Channel operation	Ch(1):=Ch(1)+2;	Call FormulaCalc ("Ch(1):=Ch(1)+2")

## Procedures

Procedures can be called when they have been converted by *CALL*. In VBS Autosequences there are two types of procedures: SUB and FUNCTION, and the latter can return a value. Procedures without parameters can be called VBS with parentheses, without parentheses, or with the *CALL* keyword with parentheses.

AUT	VBS
Test;	Test;
	Test()
	Call Test()
	Call MsgBoxDisp("MsgBoxDisp with two parameters", "MB_OK")

The converter converts AUT procedures into SUBs. The two AUT commands *AutoStart* and *AutoInclude* are changed to *ScriptStart*.

AUT	VBS
AutoStart('AUTToVBSSub');	Call ScriptStart("AUTToVBSSub")
AutoInclude('AUTToVBSSub');	Call ScriptStart("AUTToVBSSub")
AutoInclude('AUTToVBSSub', ' ', 'Proc');	Call ScriptStart("AUTToVBSSub", "Proc")
AutoInclude('AUTToVBSSub', 'L1 > 5', 'Proc');	If L1>5 Then Call ScriptStart("AUTToVBSSub", "Proc") End If



The AUT-VBS converter cannot convert nested procedures. You have to convert them manually.

The name conventions in VBScript do not allow underlines at the start of procedure names. If an Autosequence contains a procedure that starts with an underline, the converter replaces the underline in the name with an X.

## For...do Control Structure

For-loops in AUT Autosequences are converted into Do-While-loops for VBS, because the loop variable of a for-loop may not be a DIAdem variable in VBS syntax.

AUT	VBS
<pre>for L1:= 1 to 3 do begin   MsgBoxDisp('L1: ' + Str(L1)); end;</pre>	<pre>L1=1 Do While L1&lt;=3   Call MSGBOXDISP("L1:   "+Str(L1))   L1=L1+1 ' &lt;&lt; Step 1 or Step -1 ??? Loop</pre>

The conversion inserts the comment " << Step 1 or Step -1 ???", because For-loops in can run ascending or descending order in AUT Autosequences. As you cannot define the direction automatically, you are requested to check the direction ( +1 or -1).

## Remarks and Special Features

- ⇒ If you access an object in a DIAdem device in the AUT Autosequence, for example, the x-axis of an axis system in DIAdem-GRAPH, open the device before starting the conversion.
- ⇒ Accessing data channels with the variable CHD is invalid and can be converted manually into the variable ChDX. Access to channel values is greatly accelerated.
- ⇒ Several operations are no longer performed by the DIAdem formula interpreter when they have been converted. VBS performs them itself because they are VBS keywords: abs, sin, cos, tan, exp, log, len, ltrim, rtrim, trim, round, sqr.
- ⇒ Multiple instructions are converted in a way that there is only one instruction in each line.





# 7 Index

## A

- Alignment functions 5-4
- ASAM-ODS-Protocol 2-9
- ASCII file 2-2
- ATF file 2-9
- Autosequence 3-2, 3-9, 6-1
  - Channel conversion 6-21
  - Channel reference 6-28
  - Command 6-26
  - Control structure 6-27
  - Converter 6-33
  - Debugger 6-10
  - Dialog editor 6-16
  - Full-screen mode 6-30
  - Function bar 6-2
  - Graphic manipulation 6-22
  - List file 6-29
  - OLE 6-32
  - Presetting 6-2
  - Procedure 6-10, 6-23
  - Serial evaluation 6-28
  - Syntax 6-9
  - User management 6-30
  - Variable 6-26
- Autosequences
  - ODBC/SQL 6-32
- Axis definition 5-6
- Axis system 5-5

## B

- Background graphic 5-14
- Binary file 2-3

## C

- Calculation 4-1
- Calculator 4-4
- Channel Reference Mode 5-3
- Channel selection 2-4
- Clipping 5-3

- Color legend 5-9
- Command 6-26
- Command index 6-12
- Configuration 1-4
  - Data Navigator 2-9
- Control structure 6-17
- Cursors 3-2, 3-5
- Curve
  - Definition 3-2
  - Markers 3-3
- Curves measure 3-6
- Customizing 6-14

## D

- Data channel
  - Compare 2-3
  - Edit 2-4
  - Generate 2-5
  - Registered 2-2
  - Sort 2-6
- Data layer 5-5
- Data Navigator 2-7
  - Autosequence 2-9
  - Configuration 2-9
  - Databases 2-7
  - Model file 2-9
- Data set properties 2-2
- Desktop file 1-6
- DIAdem-AUTO 1-1, 6-1
  - Customizing 6-14
  - Debugger 6-10
  - Dialog editor 6-16
  - Editor 6-4
  - File overview 6-3
  - Information area 6-13
  - Interaction mode 6-30
  - Online help 6-12
  - Quickinfo 6-12
  - TEACH-IN-MODE 6-5
  - Variable 6-20
  - Workspace 6-3

- DIAdem-CALC 1-1, 4-1
- DIAdem-DATA
  - Block operations 2-5
  - Channel contents 2-5
  - Channel overview 2-4
  - Channel properties 2-6
- DIAdem-GRAPH 1-1, 5-1
  - Aligning objects 5-4
  - Axis system 5-5
  - Frame 5-2
  - Layout 5-2
  - Page layout 5-2
  - Subaxes 5-7
  - Subobject 5-4
  - Table 5-9
  - Text 5-11
- DIAdem-VIEW 1-1, 3-1, 4-3
  - Coordinate window 3-6
  - Curve editing 3-5
  - Diagram 3-9
  - Legend 3-4
  - Measure 3-6
  - Scaling 3-3
  - Scrolling 3-7
  - Zooming 3-7
- Diagram 5-1
  - DIAdem-VIEW 3-9
- Dialog editor 6-16
  - Layout bar 6-17
  - Property window 6-19
  - Script editor 6-19
- Directory structure 1-5
- Display layer 5-4
- Documentation 5-1

## E

- Editor 6-4
- Examples 1-4

## F

- Formula 3-5, 4-4
  - Syntax 4-5
- Frames 5-11

## G

- General Programming Interface (GPI) 1-7
- Graph definition 4-3
- Graph parameters 5-6
- Graphic export 5-15
- Graphic formats 5-14
- Graphic Layer 5-5

## H

- Header file 2-3
- Height-to-Width Ratio 5-3
- Help 1-3

## I

- Illustration 5-14
- Image template 5-2
- Import
  - Data channels 2-2
  - Using Headers 2-2
- In place-Operation 4-2
- Interface
  - GPI 1-7

## L

- Layout 3-2, 5-2
- Legend 3-4, 5-8

## M

- Math 4-1
- Memory management 2-2

## N

- Network installation 1-6
- NoValue 4-5

## O

- ODBC/SQL interface 6-32
- OLE interfae 6-32
- Operation 4-6

**P**

Path handling 6-29  
Presetting 1-2, 1-4  
Program start with Parameter 1-6

**R**

Report 1-5, 6-11  
Resize objects 5-3

**S**

Script Custom Functions 5-3  
Scroll bars 3-7  
Serial evaluation 6-28  
Sketch Mode 5-5  
Subaxis 5-7

**T**

Table 5-9  
  AutoExpand 5-9  
TEACH-IN-MODE 6-5

Templates 1-4  
Text editor 5-11  
Text object 5-12  
  Tables 5-12

**U**

User management 6-30

**V**

Variable 4-4, 4-6, 6-21, 6-26  
  Text object 5-12  
  Variable index 6-12  
VBScript 6-4

**W**

Workspace 6-3

**X**

XML file 2-8



# 8 Technical Support Resources

## Web Support

National Instruments Web support is your first stop for help in solving installation, configuration, and application problems and questions. Online problem-solving and diagnostic resources include frequently asked questions, knowledge bases, product-specific troubleshooting wizards, manuals, drivers, software updates, and more. Web support is available through the Technical Support section of [ni.com](http://ni.com).

## NI Developer Zone

The NI Developer Zone at [ni.com/zone](http://ni.com/zone) is the essential resource for building measurement and automation systems. At the NI Developer Zone, you can easily access the latest example programs, system configurators, tutorials, technical news, as well as a community of developers ready to share their own techniques.

## Customer Education

National Instruments provides a number of alternatives to satisfy your training needs, from self-paced tutorials, videos, and interactive CDs to instructor-led hands-on courses at locations around the world. Visit the Customer Education section of [ni.com](http://ni.com) for online course schedules, syllabi, training centers, and class registration.

## System Integration

If you have time constraints, limited in-house technical resources, or other dilemmas, you may prefer to employ consulting or system integration services. You can rely on the expertise available through our worldwide network of Alliance Program members. To find out more about our Alliance system integration solutions, visit the System Integration section of [ni.com](http://ni.com).

## **Worldwide Support**

National Instruments has offices located around the world to help address your support needs. You can access our branch office Web sites from the Worldwide Offices section of [ni.com](http://ni.com). Branch office Web sites provide up-to-date contact information, support phone numbers, e-mail addresses, and current events.

If you have searched the technical support resources on our Web site and still cannot find the answers you need, contact your local office or National Instruments corporate. Phone numbers for our worldwide offices are listed at the front of this manual.