|  |  |
| --- | --- |
|  | Data Logging Application |

Data logging is the act of recording data over time in order to be analyzed in later stages. Data logging is the crucial part of design and analysis, especially when you are developing or testing control system software using a virtual environment. This application, called data logging application, is a software solution for recording data available on National Instrument Shared Variable Engine. Utilizing MHWirth LabVIEW client, this application can be used for recording data on xFactor DDS server.

Table of Contents

[1. Introduction 3](#_Toc432432438)

[a) Objective 3](#_Toc432432439)

[b) Main Features 3](#_Toc432432440)

[c) System Requirements 3](#_Toc432432441)

[d) Supported Data Types 3](#_Toc432432442)

[2. Quick Start-up Quide 4](#_Toc432432443)

[3. Main Functionality – Main Feature 5](#_Toc432432444)

[a) Capture information in a file 6](#_Toc432432445)

[b) TDMS File Structure 6](#_Toc432432446)

[a) Exchangeability of TDMS file format 7](#_Toc432432447)

[4. How to Use Data Logging Application 8](#_Toc432432448)

[a) User Interface Introduction 8](#_Toc432432449)

[b) Application Version Number 9](#_Toc432432450)

[c) Status Pane 9](#_Toc432432451)

[d) Setting Pane 10](#_Toc432432452)

[e) Controls Pane 12](#_Toc432432453)

[f) Variables Pane 12](#_Toc432432454)

[5. Menu Bar 14](#_Toc432432455)

[a) File Menu 14](#_Toc432432456)

[b) Settings Menu 14](#_Toc432432457)

[c) Tools Menu 15](#_Toc432432458)

[View 15](#_Toc432432459)

[Defrag 15](#_Toc432432460)

[Database 15](#_Toc432432461)

[Variables 15](#_Toc432432462)

[d) Help Menu 16](#_Toc432432463)

[6. Short Keys 16](#_Toc432432464)

[7. TDMS File Viewer 16](#_Toc432432465)

[8. Limitations 16](#_Toc432432466)

[9. Technical Information 17](#_Toc432432467)

[a) General Technical Info 17](#_Toc432432468)

[b) Database 17](#_Toc432432469)

[c) Appending Data to an Existing Log File 18](#_Toc432432470)

[d) File Fragmentation 18](#_Toc432432471)

[10. Troubleshooting 19](#_Toc432432472)

[11. Copyright & Support 19](#_Toc432432473)

# Introduction

## Objective

The main objective of this application, called *Data Logging Application*, is to store data (online while they are generated by various parts of the test setup) in a TDMS file so that one can view data at the later stages. This application is capable of storing data available on the MHWirth xFactor DDS server and/or on the National Instrument Shared Variable Engine.

Along with this application, “Data Viewer”, which is an independent application, is delivered. Data Viewer allows you to view and plot logged data while or after recording. Data Viewer can be called directly from this application.

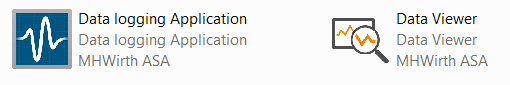


Figure Data logging application and data viewer.

## Main Features

* Stand-alone application that can be executed on an independent PC
* Storing data in a binary-size file (of TDMS format) that is readable in other applications such as Excel, MATLAB, and LabVIEW
* Storing data that are available on DDS server regardless of the source and type
* Storing data of 36 data types
* Storing data with timestamp and adjustable sampling frequency (rate)
* Storing data in a rich documented file with possibility of adding attribute and tracking appendices
* Viewing and plotting stored data on-line (while logging) and off-line
* Automatically saving and restoring data logging setting
* No need for additional hardware – The sole piece of hardware required is a PC

## System Requirements

* Windows 7 or higher
* Minimum 2GB RAM
* NI Run-time Engine 2014 (if installer is used it will be installed automatically)
* MHWirth xFactor LabVIEW Client if data are distributed on MHWirth DDS server (it requires NI Run-time Engine to be installed first)

Note: The DDS server (MHWirth product) is used for data distribution. Besides xFactor LabVIEW client, the OPC Client (MHWirth product) can also be used; however, it requires more effort for variable mapping.

## Supported Data Types

The data logging applications supports 36 data types including scalars, 1-dimentional, and 2-dimentional arrays. You can find supported data types for data logging in Table 1. These data types cover the current supported DDS data types.

Table Supported Data Types

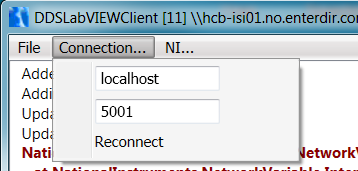
|  |  |  |  |
| --- | --- | --- | --- |
| Variables | Scalars | 1D Array of | 2D Array of |
| Boolean | 🗸 | 🗸 | 🗸 |
| Signed integer - Int (8, 16, 32, 64 bits) | 🗸 | 🗸 | 🗸 |
| Unsigned integer - UInt (8, 16, 32, 64 bits) | 🗸 | 🗸 | 🗸 |
| Single-precision floating point (32 bits) | 🗸 | 🗸 | 🗸 |
| Double-precision floating point (64 bits) | 🗸 | 🗸 | 🗸 |
| Extended-precision floating point (128 bits) | 🗴 | 🗴 | 🗴 |
| Fixed-point | 🗴 | 🗴 | 🗴 |
| String | 🗸 | 🗸 | 🗸 |

[up](#_top)

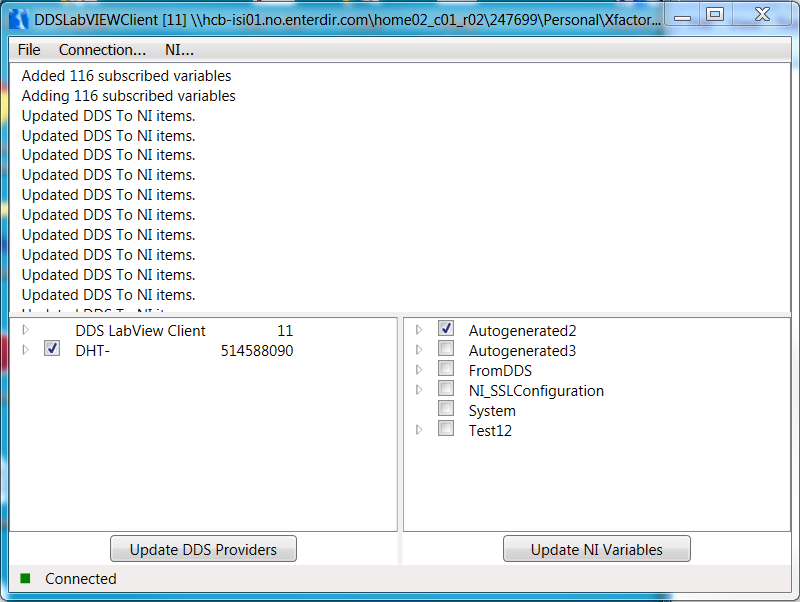
# Quick Start-up Quide

To quickly start working with the data logging application, follow the steps below.

1. Install NI Run-time Engine 2014 ([32-bit](http://www.ni.com/download/labview-run-time-engine-2014/4887/en/) – [64-bit](http://www.ni.com/download/labview-run-time-engine-2014/4889/en/)). Read about LabVIEW Run-Time Engine compatibility [here](http://digital.ni.com/public.nsf/allkb/800E68EBF895BD96862570770051FF36).
   * Alternatively, you can install “Data Logging Application Installer” that installs the run-time engine.
2. Install DDS server if data distribution shall be done using DDS server. You need to have an active dongle to use this application.
3. Install LabVIEW Client.
4. Open DDS server. Run other applications that provide variables to DDS server.
5. Open LabVIEW Client. In “Connection”, set the server address with port 5001.



1. In LabVIEW Client, select the processes (on the left side) you would like to have in NI shared variable engine, and click on “Update NI Variables”. Make sure “Added xxx subscribed variables” is shown.



1. Open Data Logging Application, and follow below steps for a quick startup.

|  |  |
| --- | --- |
|  | 1. Set the locations for the TDMS log file and the database file (A). 2. Select the file name. 3. Initialize the application. Deployed Libraries are then shown. 4. Select one deployed library. 5. Check the database status. If the database is available, skip step 4. 6. Create database. This is disabled if the database is available. 7. Select a deployed library and variables that you want to log (B). 8. Set the sample rate and the server name (C). 9. (Optional) Add more information to the log file. 10. Start recording. 11. Stop recording when you want. The logged file is found in the location you specified in step 0. |

1. If the database for a selected library is not available, the application can make a base database for data logging
2. Deployed Library from LabVIEW Client is named “FromDDS”
3. In this step, you can also set the buffer size and turn on/off Auto Defrag.

[up](#_top)

# Main Functionality – Main Feature

The main functionality of this application is to record any type of data (scalar, 1D and 2D arrays) in a file. The file format is TDMS (Technical Data Management Streaming) that is developed by National Instrument. TDMS file format allows fast streaming of data while keeping the size in binary level.

This section sheds some lights on TDMS file format, its structure, and capability of reading the file using other applications.

## Capture information in a file

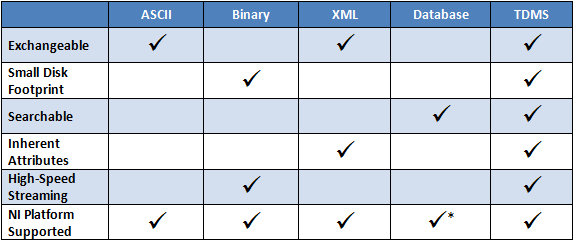
This application saves data in a TDMS (Technical Data Management Streaming) file format, which is part of the NI technical data management solution. The NI technical data management solution includes three components:

* The NI TDMS file format for saving well-documented measurement data
* The NI DataFinder for quickly locating previously stored data sets
* NI DIAdem or the LabVIEW DataFinder Toolkit for processing data and creating reports

Table 1 shows the pros and cons of some of the most commonly chosen storage options for measurement data. More detailed comparison for data types is given [here](http://www.ni.com/white-paper/9630/en/).

Table The TDMS file format combines the benefits of several data storage options in one file format

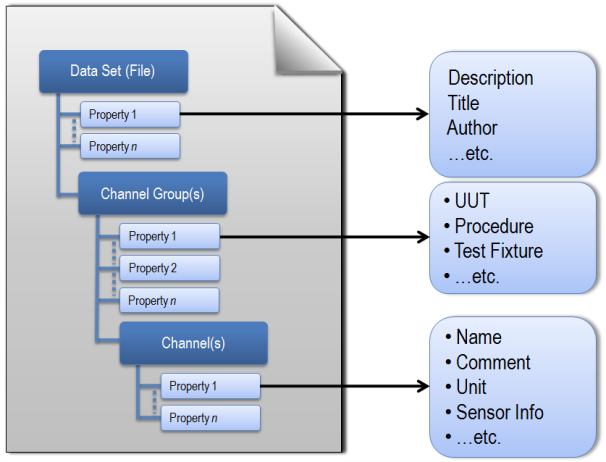
(Source ni.com – [click here](http://www.ni.com/white-paper/3727/en/))



## TDMS File Structure

The binary TDMS file format is an easily exchangeable, inherently structured, high-speed-streaming-capable file format that, when combined with the other technologies in the NI TDM solution, becomes quickly searchable without the need for complicated and expensive database design, architecture, or maintenance.

The single, most important feature to understand about the internal format of the TDMS file structure is its inherent hierarchical organization. The TDMS file format is structured using three levels of hierarchy, as shown in Figure 1 << file, group, and channel >>.



The file level can contain an unlimited number of groups, and each group can contain an unlimited number of channels. You can think of excel workbooks. A data set (TDMS file) is like an Excel workbook. Each channel group is like one sheet. Channels are like columns in sheets while each column has several inherent properties that can be set.

TDMS files also automatically generate a complimentary \*.tdms\_index file. This file provides consolidated information on all the attributes and pointers in the bulk data file that drastically speeds up read access to the data on larger data sets. This index file is not required for storage or distribution and automatically regenerates.

This is important to note that TDMS file format is a well-documented, rich file format. One can store an unlimited number of custom-defined scalar *attributes* to achieve well-documented and search-ready data files. The descriptive information located in the file is a key benefit of this file format, and provides an easy way to document the data without having to design your own header structure. As your documentation requirements increase, you do not have to redesign your application; you simply extend the model to meet your specific needs. The more custom properties you use to document your measurement data, the more easily it can be located at a later date by using an NI DataFinder client that abstracts complex database communication from the user.

## Exchangeability of TDMS file format

#### .NET languages using measurement studio

The NI Measurement Studio TDM Streaming .NET Class Library provides a fast and flexible way to interface directly with TDMS files using .NET languages. Access to the TDM Streaming .NET Class Library is restricted to the Professional and Enterprise editions of Measurement Studio.

#### TDM C DLL

NI offers the TDM C DLL as a free download. It contains the necessary functions for reading and writing TDMS files from any application development environment that is flexible enough to enable DLL communication.

#### OpenOffice.org Calc

With the free add-in for OpenOffice.org Calc, you can load and process TDMS files including descriptive information in OpenOffice.org Calc.

#### Microsoft Excel Office

Using the free TDM Excel Add-in for Microsoft Excel, you can load and work with TDM and TDMS files in Excel. TDMS file can be opened and processed using Excel. It also can be saved as Excel workbooks. See details [here](http://www.ni.com/tutorial/12561/en/).

#### The MathWorks, Inc. MATLAB® Software

National Instruments has developed an example demonstrating how to adapt the TDM C DLL with a customized header file to interface with TDMS files from within The MathWorks, Inc. MATLAB® software. To learn more and download the example, see [MATLAB TDM Example User Guide](http://www.ni.com/white-paper/7446/en/).

[up](#_top)

# How to Use Data Logging Application

This section explains the functionality of the application in details.

Figure 2shows the user interface for data logging application.

Users can interact with this application in three ways:

1. Controls on the user interface
2. Menu bar
3. Short keys

In the following, every function is clarified.

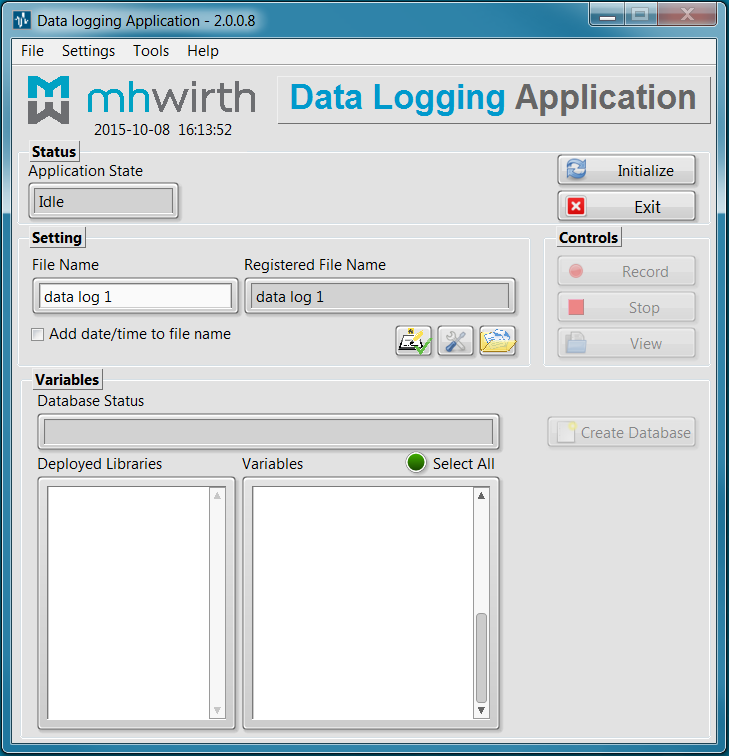


Figure User Interface

[up](#_top)

## User Interface Introduction

Main parts of the user interface are introduced in this subsection.

|  |  |
| --- | --- |
|  | 1. Application version number 2. Current date and time 3. Status pane   to show the status of the application, initialize and exit   1. Setting pane   to adjust the data logging setting   1. Controls pane   to control the main function (start and stop recording, and view the file)   1. Variables pane   to select variables for logging   1. Menu bar |

## Application Version Number

The version number is shown on top of the application. The syntax for the version number is as below

{Major version}.{Minor version}.{Fix}.{Build number}

Changes in the application are registered internally in TFS. Main changes are given in the release notes that can be found in menu as Help -> Release Note.

[up](#_top)

## Status Pane



The status pane shows the internal status of the application. It can take one of the following in each instant of time.

* Initialize 1, …, 7
* Idle
* Preparation 1, …, 4
* Record
* Stop recording
* Defrag
* View
* Create Database
* Replace Database
* Defrag Prompt File
* View Prompt File

The status pane also has two controls.

|  |  |  |
| --- | --- | --- |
| Location | Button Name | Function |
| Status Pane | Initialize | Initializes the application  Applies changes in “Folder settings”, “Preference”, and “Log filename”.  Does the following:   * Set the file name * Get deployed process list and updates the Deployed Libraries. |
| Status Pane | Exit | Releases resources and exit the application |

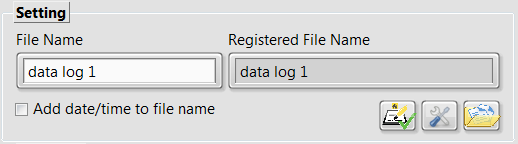
**Note:** changes in Preference, Folder settings, and Log filename are not implemented until “Initialize” is pressed.

**Note:** changes that happened in shared variable engine (e.g. added or removed processes and changes in variables) are not seen by the application until “initialize” is pressed.

[up](#_top)

## Setting Pane

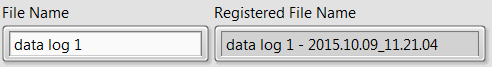
This pane is to adjust the setting for data logging.



The followings are related to the log filename.

|  |  |  |
| --- | --- | --- |
| Location | Name | Function |
| Setting Pane | File Name | Set the file name or the base for file name |
| Setting Pane | Add date/time to file name | If checked, date and time are added to the filename base. |
| Setting Pane | Registered File Name | It shows the filename that is going to be used for data logging. |

Example of how date and time are added to file name base.

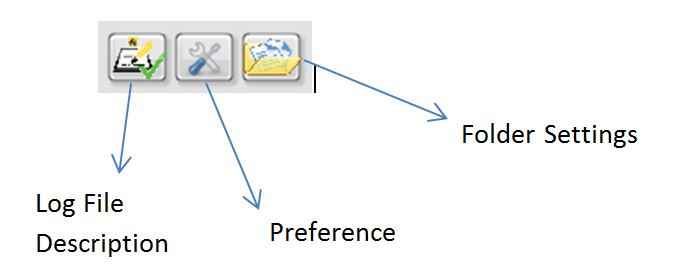


**Note:** When “Initialize” is pressed, “registered file name” is updated.

**Note:** If a log file is re-used for logging, new data are appended to the log file. That means data are not lost if file name is not changed. An attribute in the log file indicates to every appendix; see the section “[Appending Data to an Existing Log File](#_Appending_Data_to)”.

If any of “File Name” or “Add date/time to file name” changes, the “Initialize” button starts blinking to notify user to implement changes by clicking on “Initialize”.

In the settings pane, one can set other properties of data logging. There are three buttons in this pane as below.

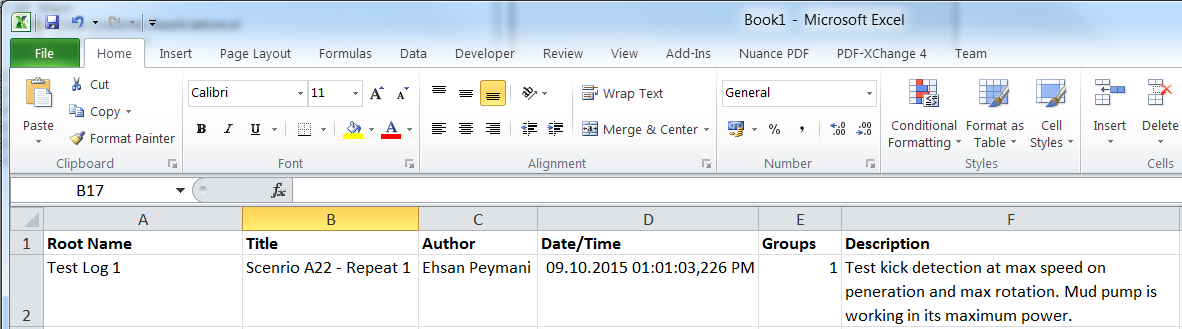


**Note:** Hovering mouse on each button activates tip strip for the function.

|  |  |  |
| --- | --- | --- |
| Location | Button | Function |
| Setting Pane | Folder Setting  Ctrl + 1 | * Locate the folder to save the log file * Locate the folder to get the database |
| Setting Pane | Preference  Ctrl + 2 | adjusts data logging preference   * Server name for shared variable engine * Data logging sampling period * Automatic defragmentation * Buffer size for data logging |
| Setting Pane | Log File Description  Ctrl + 3 | Add meta/attribute to log data   * Title for data logging * Description of data logging * Creator of data logging |

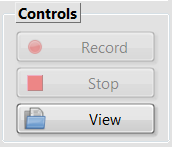
See [File Fragmentation](#_File_Fragmentation) for details on Auto Defrag and Buffer Size.

Log File Description: You can add more information to the log file. An example is shown below when the log file is opened in Excel.



[up](#_top)

## Controls Pane



The controls pane is to start and stop recording. It also has a control to view the file in [TDMS file viewer](#_TDMS_File_Viewer).

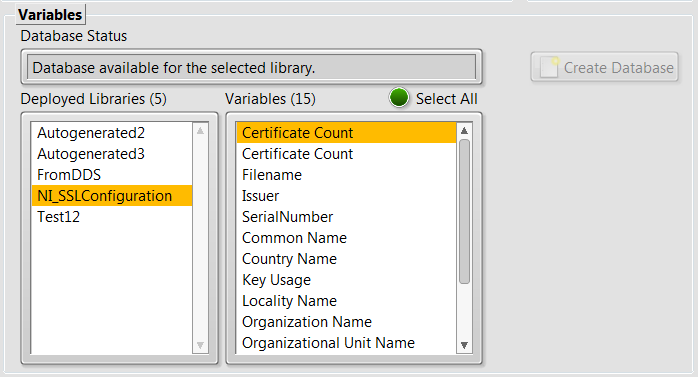
|  |  |  |
| --- | --- | --- |
| Location | Button | Function |
| Controls Pane | Record | Starts recording data as selected in “Variables” in the file whose name is as “Registered File Name”.  The rest of settings are according to the settings pane. |
| Controls Pane | Stop | Stops recording. |
| Controls Pane | View | Opens the set TDMS file in [TDMS file viewer](#_TDMS_File_Viewer). |

**Note:** You can open prompt to view a TDMS file using menu bar: Tools -> View TDMS file -> Prompt TDMS file

[up](#_top)

## Variables Pane

Using this pane, the user can select variables for logging.



|  |  |  |
| --- | --- | --- |
| Location | Control | Function |
| Variables Pane | Deployed Library | Shows all the deployed libraries/processes on the shared variable engine.  “FromDDS” is the library containing all signals imported by LabVIEW Client from DDS server.  Selection mode: 1 item |
| Variables Pane | Variables | Shows all variables in the selected library  Selection mode: 1 or more items |
| Variables Pane | Select All | Selects all variables in the selected library |
| Variables Pane | Database Status | Shows the status of database. The application looks for the database with the exact same name as the selected deployed library. If not found, a message in red is displayed. |
| Variables Pane | Create Database | Creates a database for the selected deployed library, and saves it in the designated folder with the exact same name of the deployed library.  If the database is not found by the application, this button is enabled. Otherwise, it is disabled and greyed out. |

If the application cannot find the database for the selected “deployed library” in the designated folder, it is reported in the “database status”. Consequently, the button “Create Database” is enabled. See Figure 3.

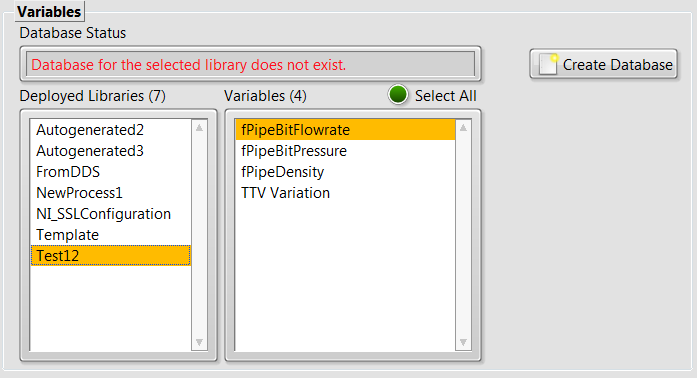


Figure How the application responds to lack of database

[up](#_top)

# Menu Bar

Menu bar is designed in order to have a better user interface. This application has 4 main items in the menu bar as mentioned below:

|  |  |  |
| --- | --- | --- |
| Menu | no.  sub-items | Function |
| File | 2 | Saving and restoring data logging configuration |
| Settings | 4 | Adjusting data logging configuration |
| Tools | 4 | Having several tools to   * view and defrag the log file, * creating and replacing database, * creating and managing processes |
| Help | 4 | Having user guide, database templates, and release note |

## File Menu

File menu has 2 sub-items.

Load configuration: allows the user to reload a saved configuration into the program.

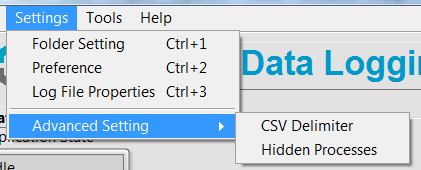
Save configuration: allows the user to save configuration for later use.

The items that form configuration include those that are in “Folder Setting”, “Preference”, and “Log File Description”. To have a better overview, these are collected below:

|  |  |  |
| --- | --- | --- |
|  |  |  |

## Settings Menu

The settings menu has 4 sub-items.



|  |  |  |
| --- | --- | --- |
| Menu | Short key | Function |
| Folder Setting | Ctrl + 1 | The same function as  in [Setting pane](#_Setting_Pane). |
| Preference | Ctrl + 2 | The same function as  in [Setting pane](#_Setting_Pane). |
| Log File Properties | Ctrl + 3 | The same function as  in [Setting pane](#_Setting_Pane). |
| Advanced Setting | - | It has two sun-items:  CSV Delimiter – set the delimiter of CSV file. It can be one of {comma, semi-colon, tab}  Hidden Processes – specify what processes you do not want to show in the deployed libraries |

**Note:**  Hiding deployed processes in the user interface is different from un-deployment.

## Tools Menu

The tools menu has four items.

|  |  |  |
| --- | --- | --- |
| Menu | Short key | Function |
| View | - | View and plot the TDMS file data |
| Defrag | - | Defrag TDMS files to remove fragmentation |
| Database | - | Create or replace database |
| Variables | - | Create new libraries and deploy them + SVE management |

### View

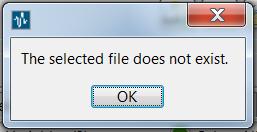
There are two options to view a log file.

Logged Data Viewer which is an application in order to view and plot the logged data. It can be used while recording and after that.

TDMS File Viewer which is a tool to see file properties and view data in tables. It has limited plotting capabilities.

You can prompt for any TDMS file using this function.

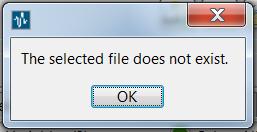
If the file does not exist, the following message is shown.



### Defrag

Manual defrag of the current TDMS file or any TDMS file is doable using this function.

If the file does not exist, the following message is shown.



### Database

This function allows you to

Create a missing database

Replace an existing database

Only one of them is enabled. “Create a missing database” is the same function as “Create Database” button on the variables pane.

See section [Database](#_Database) for more information.

### Variables

It opens a wizard to create shared variables and to manage the shared variable engine.

## Help Menu

Using the help menu, the user can find the user guide. The user guide is available in HTML format and as a Word document.

Release Notes can be found in the help menu.

Application information page is shown in the help menu.

To facilitate creating databases, templates for database are included in the application and can be found under help menu.

[up](#_top)

# Short Keys

To facilitate use of the applications, the following short keys have been implemented.

|  |  |  |  |
| --- | --- | --- | --- |
|  | Function | Short key |  |
| Buttons | Initialize | Ctrl + I |  |
| Exit | Ctrl + Q and Ctrl + E |  |
| Record | Ctrl + R |  |
| Stop | Ctrl + S |  |
| View | Ctrl + V |  |
|  | Function | Short key |  |
| Buttons and Menus | Folder Settings | Ctrl + 1 |  |
| Preference | Ctrl + 2 |  |
| Log File Properties | Ctrl + 3 |  |
|  | Function | Short key |  |
|  | Help | F1 |  |

[up](#_top)

# TDMS File Viewer

There are two ways to view the log file:

Logged Data Viewer – capable of plotting logged data both while recording and after recording.

TDMS File Viewer – capable of viewing data in tables and viewing file properties

TDMS File Viewer allows you to view the log file.

When viewing a TDMS file, all keys remain disabled.

[up](#_top)

# Limitations

Here, there is the list of current limitations

|  |  |  |  |
| --- | --- | --- | --- |
|  | **Limitation** | **Application Response** | **Application Handling Method** |
| 1 | Database is required for data logging | If app cannot find it, a message is shown to the user. | User can make a database using this application. |
| 2 | Variable selection is only possible within one selected deployed library | Deployed Library selection mode is limited to one item. |  |
| 3 | Data Type is limited to the list in [Section 1.D](#_Supported_Data_Types) | Any variable not supported, an error is generated. | Recording stops. |

[up](#_top)

# Technical Information

This section includes some of the technical information for the data logging application.

## General Technical Info

* The base cycle time of the application is 100 msec. This cycle time is different from data logging sampling time.
* Data logging sampling time can be adjusted by user. The minimum is 1 msec.
* Tip strips are shown if mouse is hovered over the buttons.

## Database

Database is an excel sheet or a text file which contains the following necessary information about the variables.

Variable Name – First Column

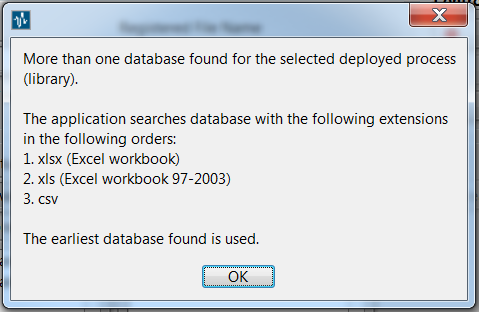
Variable Data Type – Second Column

Database might contain more information about the variables and the library (e.g. binding information). For the purpose of data logging, the only required information is variable name and variable data type.

You can use the provided templates for creating database as an excel sheet or as a text file using menu bar: Help -> Database Templates.

**Note:** The application expects to find a database with the exact same name as the selected deployed library in the designated folder as specified in [Folder Setting](#_Setting_Pane) (Ctrl+1) . If it cannot find the corresponding database, a message is shown to the user as database status, and “Create Database” button is enabled. Pressing “Create Database” will create the database and save it in the designated folder.

The application looks for database with the same name as the deployed library with extensions “.xlsx”, “.xls”, and “.csv”, in order. If more than one database is found, the first found database is used but the following message is shown to the user.



If no database files are found, “create database” will create a database of the format “.xlsx”.

You can create a new database or replace an existing database using menu bar: Tools -> Database.

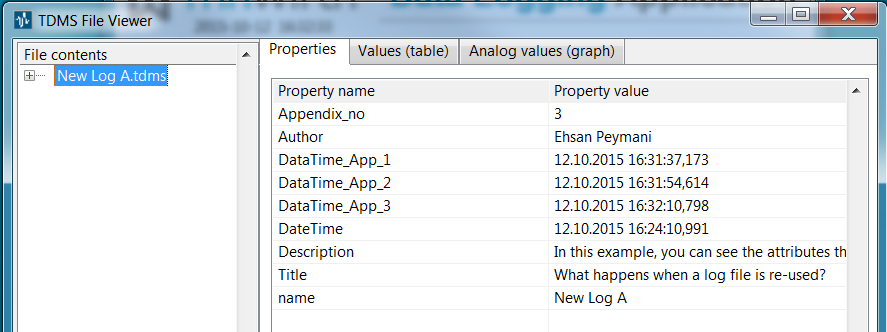
## Appending Data to an Existing Log File

If you start recording on an existing file, new data are appended to the file. In the file properties, there are 2 properties related to this.

Appendix\_no which shows the number of re-use of the file

DateTime\_App\_{i} which shows the timestamp for one specific append

See the below example for a file which has been used for logging three times.



## File Fragmentation

A file is fragmented when data are not stored in sequentially/continuously. File fragmentation may result in large file size and may degrade performance (e.g. may slow down reading process).

In the data logging application, there are two approaches to avoid/cure fragmented files.

File Defragment

This option defrags a TDMS file by removing fragmentations, and making the log file contiguous.

There is an Auto Defrag option in Preference in Setting Pane (alternatively: Ctrl+2 or menu bar: Settings -> Preference). If it is ON, the TDMS file is defragged after stopping recording.

It is also possible to defrag a selected TDMS file using menu: go to Tools -> Defrag

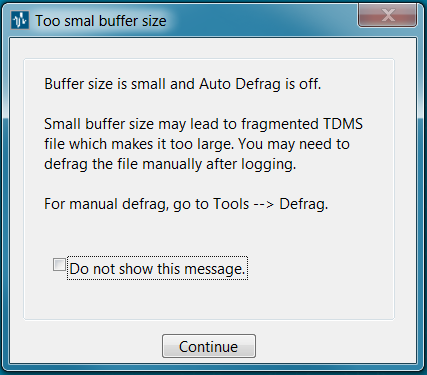
Adjustable Buffer Size

With small buffer size, it is more likely to have fragmented files. Therefore, increasing the buffer size will increase the size of data chunks that are written in memory blocks. In turn, file fragmentation is smaller.

The buffer size can be adjusted in Preference in Setting Pane (alternatively: Ctrl+2 or menu bar: Settings -> Preference).

**Note:** With large buffer size, the Logged Data Viewer in online mode is updated when the new buffer of data is written in TDMS file.

If the user switch of Auto Defrag and the Buffer Size is less than 50, the following warning is shown to user before recording.





# Troubleshooting

# Copyright & Support

MHWirth is the owner of this application. This application, Data Logging Application, is copyright protected; © 2015 MHWirth. All rights reserved.

For questions, contact MHWirth - Ehsan Peymani (ehsan.peymani@mhwirth.com)