

TEST BENCHES COMPUTER CONTROL SOFTWARE TOOLS DEVELOPMENT

Dr. Ing. Juraj Gerlici, Ing. Tomáš Lack, University of Zilina, Faculty of Mechanical Engineering, Department of Railway Vehicles, Engines and Lifting Equipment, Velky diel, 010 26 Zilina, Slovak Republic Tel.: +421 (41) 513 2693, 2652, 2664. Fax: +421 (41) 52 530 16, e-mail: juraj_gerlici@kkvmz.utc.sk, tomas_lack@kkvmz.utc.sk..

***Abstract:** The paper deals with the measure and control program systems and the possibility of design application program environments for measurement, experiments controlling, data scanning and data assessment. The authors describe the working principles of LAGER development system and Component Works software product. This software is successfully installed in the departmental heavy laboratory of vehicles in the computer for automated control, data acquisition and data assessment for ERRI certified Test stand bench of Railways of Slovak Republic.*

***Key words:** data acquisition, data assessment, LAGER, ComponentWorks.*

1. INTRODUCTION

This paper deals with the test benches computer control software tool LaGer developed for interactive design of programs (application forms) for continuous monitoring of measured places, automated measurement, recording of measured data, data assessment, controlling of experiments, data exports in the formats suitable for processing with CSV and TXT formats, graphical presentations with WMF export, measure protocol processing and printing. The program system is based on the philosophy of the open component system. With advantage the Component Works (National Instruments) [2, 3], the ActiveX components set for data acquisition, numerical analysis, computations and data presentation can be implemented. The program contains the Designer that allows on-line editing of the application form.

2. LAGER SYSTEM

Program system LaGer, [5] is grounded on the philosophy of component opened system. The Program contains the „Designer“, enabling editing of the application form.

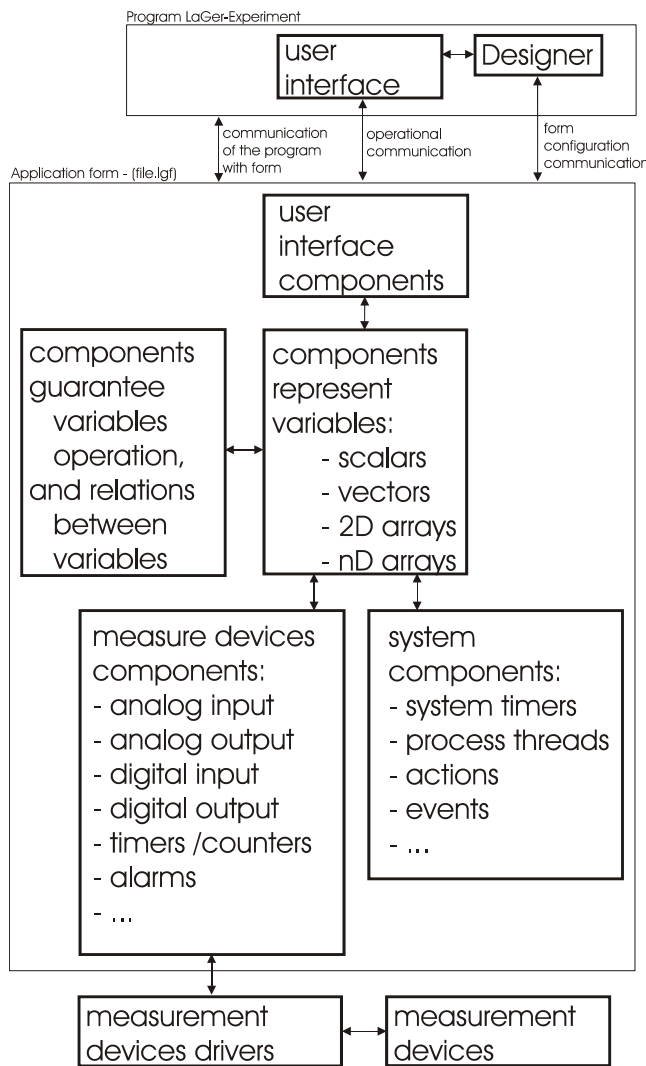


Fig. 1. Application form structure.

All activities are done during the program run. We can say that under editing of application form (Fig. 1.) we mean the creation of new instances of the registered components, their removal, settings of the published properties and assigning the methods for processing of the published events of these components. Setting of the published properties and events of the components is guaranteed by „Object inspector“ of the Designer.

Setting of the component properties influences the behavior of the component and in the visual components the appearance and the location too. By assigning of the method to the given event of the component, we can guarantee the processing of the event by assigned method. As an example of the event (of the visual component) we can introduce

the press of the mouse klick over the component area, or the example of the event of the non-visual component is the elapse of the time interval.

The central groups of the structure of the application form are the components representing the variables. These components serve as the data interface among the other components and they create the application data base too. These components can be, from the point of view of the order that they represent, separated in scalar, vector, 2D array and nD array groups.

The visual base of the application form is made from the visual components of the user interface. By help of these components the application form communicates with the user. The aim of these components is to visualize the contents of the application data that represent the variables. The examples of those components are the buttons, roller menu, popup menu, components for text editing, components for graphs viewing and so on. Their common property is the reality, that they are visible.

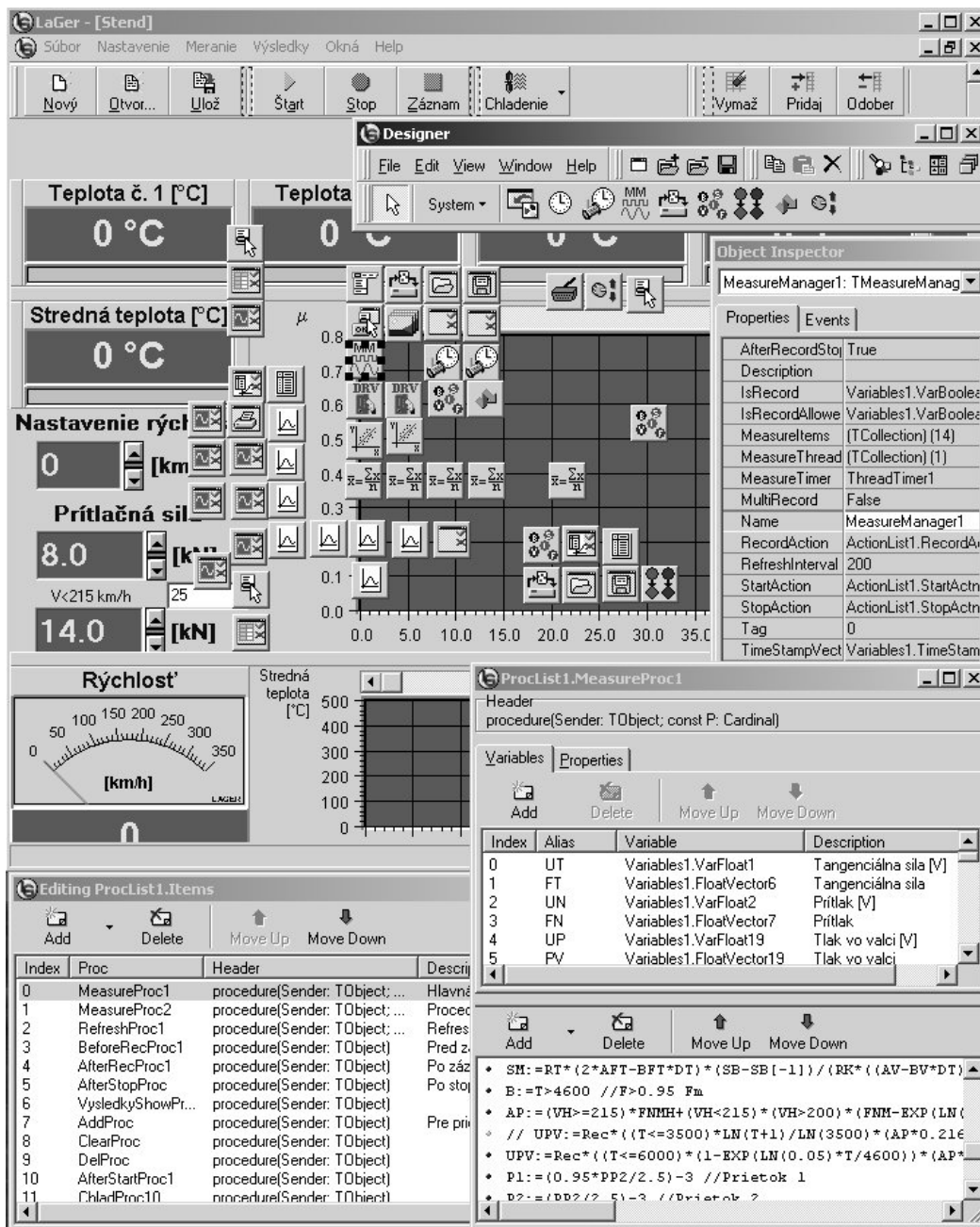


Fig. 2. The screen of application form in the program LaGer in the regime of active Designer.

The components that guarantee the variables operation and inter-variables relationship are necessary for realization of the application intention (Fig.2.). The main representative of these components is the component that transforms the equation in the text form into the result in the desiderative form, most frequent in the numerical form.

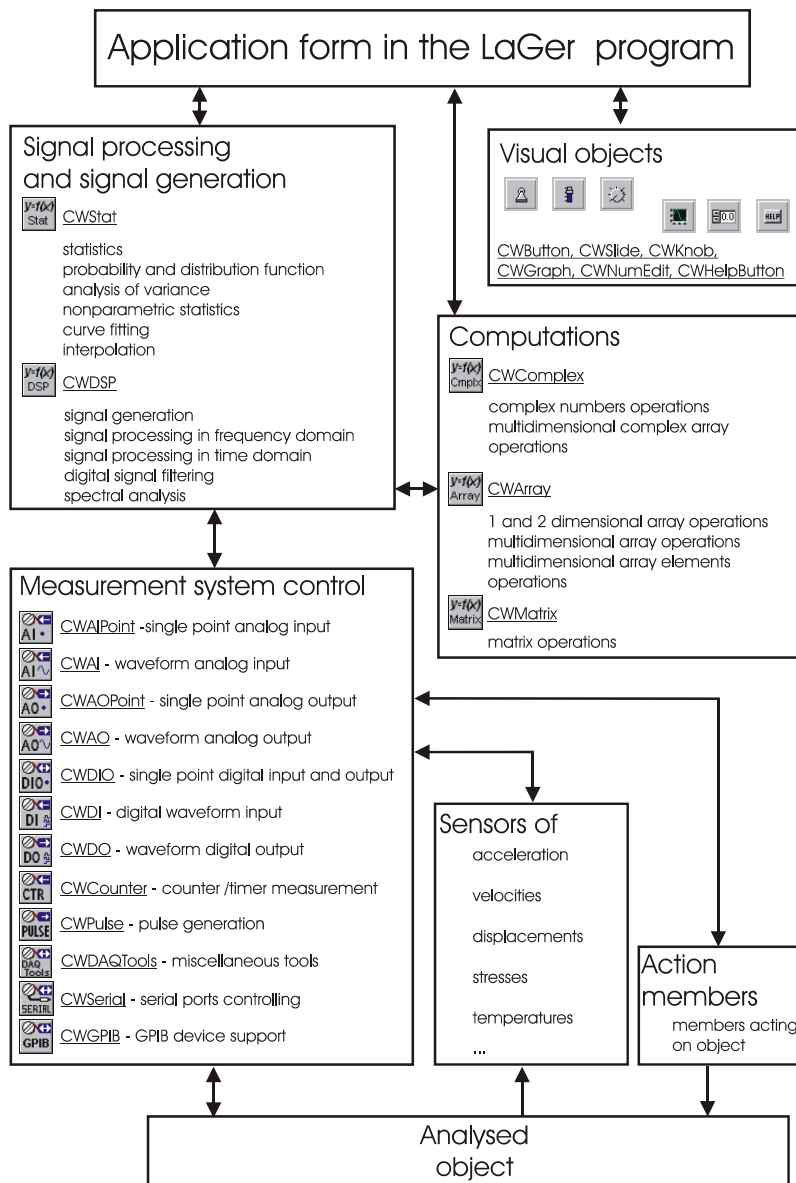


Fig. 3. Schematic layout of system and its usage.

System component group guarantees the creation, control and timing of the program process thread for application form. Such characteristic thread of the measurement application is the measurement thread, in that the activity of the measurement devices component acts. Another function of these components is the operation centralization of the action, invoked by the user and the transmission of these actions to the relevant components. The example such components are the components representing the threads, system timer/counters, actions, methods and so on. The components of the measurement devices

guarantee the co-operation of the application form with the drivers of the measurement devices. The variability of the connection is guaranteed by the reality, that the components are designed for each object of the measurement device has to correspond to the one component. It means, if the measurement device has the n-channels for analog input then in the application form are located n component for intercession analog input too and they are owned by the component representing the device.

Another component group are the components that allow the measurement protocol – form printing on the printer or the form can be exported in the Windows Meta File (WMF / EMF) format. The form can be compound from text arrays, tables, graphs and pictures.

The program allows inclusion of the Component Works objects into the LaGer form too. The Component Works is the ActiveX components set (from the National Instruments measurement software production) for the data acquisition, data assessment, numerical analysis and the data graphic presentation. The interface with analyzed object is guaranteed by the sensors of the various types (acceleration, velocity, movement, stress, strain, temperature...) and by action members. The data are transferred to action members, or they are recorded by the program. For data assessment the computational components are available, that implement the operations with complex algebra, multidimensional arrays, the members of the arrays and matrix operations. Until data in the correct form are, it is possible to use the components oriented for data processing and signal generation. The CW components communicate with the main program by the help of the properties, methods and events. It is possible to set the components properties just from the program, or from the pages of the properties, that are characteristic for ActiveX components. The visual components are the inseparable part of the CW. They wider the basic components of the development environment for the possibility of interactive program control and suitable visualization of the acquired and assessed data.

3. COMPONENT WORKS SYSTEM COOPERATION

Component Works™ (CW) [1, 4] is a collection of 32-bit ActiveX components for data acquisition, numerical analysis, numerical problem solution and data presentation. It is possible to implement them into other programs dedicated to requested aim by help of program means, that support the ActiveX component technology as MS Visual Basic, MS Visual C++, Borland Delphi, Borland C++ Builder. In the Fig. 3 is schematic layout such a program. Program is connected with analysed object through the sensors of various types (accelerations, velocities, position, voltages, temperatures...) and active members from which are data by means components aimed to measurement chain control (CWAIPoint, CWAI, CWAOPoint, CWAO, CWDIO, CWDI, CWDO, CWCounter, CWPulse and CWDAQTools) transported to active members, or saved by program. The program input are boundary conditions, start conditions and data that characterise the parameters of model for comparison and possible correction of computational model, computational method, computational parameters or algorithm of computation. The computational section of program can be programmed through available computational components that are implementing the complex arithmetic operation, multidimensional complex array, 1D, 2D and multidimensional array and array elements and matrix operations (CWComplex, CWArray and CWMatrix). If the

data are not in suitable form, they can be processed by components for signal processing and signal generation (CWStat and CWDSP). Components CW are communicating with the main program by means of properties, methods and events. The component properties can be set up right from program or through property pages, that are own for ActiveX components. Important part of CW are visual components (CWButton, CWSlide, CWKnob, CWGraph and CWNumEdit), that enhance the basic components of development environment with the possibility of interactive program control and suitable visualisation of measured and computed data.

4. CONCLUSION

During the experimental works in the laboratory of Department of railway vehicles, engines and lifting equipment has proven these approaches worth: to use the combination and cooperation of our own program system tool with the NI ComponentWorks tool. This combination yields the advantage of the sourced components utilization and simultaneously guarantees the full access to the program.

5. REFERENCES

- [1] Gerlici J., Lack T.: Mechanical systems analysis with Component Works™. 10. International scientific conference: Communications on the edge of the milleniums, University of Žilina, EDIS Žilina 1998.
- [2] Gerlici J., Lack T.: Vývoj programového systému pre skúšky spaľovacích motorov. Scientific Papers of the Univerzity of Pardubice, series B - The Jan Perner Transport Faculty 4, 1998, University of Pardubice 1999.
- [3] Kriššák, P.: Experimentálne meranie dynamických vlastností a identifikácia hydraulického stavu. In: Medzinárodná konferencia mladých, SPU Nitra, ISBN: 80-8069-085, 2002.
- [4] Lack T., Gerlici J.: Analýza mechanických sústav s využitím Component Works™. Scientific Papers of the University of Pardubice, series B - The Jan Perner Transport Faculty 4, 1998, University of Pardubice 1999.
- [5] Lack T., Gerlici J.: Aplikácia programového systému LAGER-EXPERIMENT na brzdový stav, 2. Scientific conference of the Jana Pernera Transportation faculty, DFJP UP, Pardubice 1999.
- [6] Lack T., Gerlici J., Řezníček R.: Aplikácia meracieho programu na brzdový stav KKVMZ, In: Zborník: XIV. International conference Current problems in rail vehicles - PRORAIL'99, ŽU Žilina, EDIS Žilina 1999.
- [7] Strážovec I., Kriššák, P.: Využitie multifunkčnej karty AT-MIO 16 v experimentálnej oblasti. In Sborník referátů z XIX. Semináře ASR 96 – Počítače v měření, diagnostice a řízení, VŠB-TU Ostrava.
- [8] Strážovec, I., Kriššák, P., Božek J.: Visual Programming in Lab View. In. Zborník XVII. Medzinárodnej vedeckej konferencie katedier mechaniky tekutín a termomechaniky“, Herľany 1998.