A Classification of Virtual Measurement Environments

Tsvetozar Georgiev, Georgi Krastev

Abstract: *The paper examines the capabilities of existing Virtual Measurement Environments (VMEs). The standard set of functions that can be realised through these environments is also given. A classification of these environments, according to the complexity, completeness of the functions for data collection, analysis and presentation, and their use is made. The ways to extend their capabilities through problem-oriented or instrument packages are described and conclusions about an appropriate choice of VME are made.*
Key words: *Virtual Measurement Environments.*

INTRODUCTION

Nowadays the measurement problems are solved using visual engineering environments. They allow the creation of complete software solutions by linking visual graphical objects (icons) instead of the traditional use of specialized programming languages. They simplify the decision of problems of the type of consecutive tests, management of equipment and development of user interface. These environments combine a set of objects that help the recording, analysis and presentation of data. Table 1 shows some of the widely used programming environments for development of measurement systems. In short are presented their versions, prices, licenses, the operating environment in which they work and the company that developed each of them.

Table 1

Software	Developer	Last version	Price	License	Operating system
Agilent-VEE	Agilent Technologies	9.0 /2008	$1800 (Commercial) / $150 (Student)	Commercial	Windows
DASYLab	National Instruments	10.0 / 2007	$2499 (Commercial)	Commercial	Windows
DIADEM	National Instruments	11.0 / 2008	€ 2299 (Commercial)	Commercial	Windows
LabVIEW	National Instruments	8.6 / 2008	$1199 (Commercial) / $99.95 (Student)	Commercial	Windows, Mac OS X, Linux
LabWindows	National Instruments	9.0 / 2008	€ 2499 (Commercial)	Commercial	Windows, Linux, Sun
MATLAB	The MathWorks	R2008a / 2008	$1900 (Commercial) / $99 (Student)	Commercial	Windows, Mac OS X, Linux
TestPoint	Capital Equipment Corporation	7.0 / 2006	$1295 (Commercial)	Commercial	Windows
WindMill	Windmill Software	6.0 / 2008	€ 160.00 (Commercial)	Commercial	Windows

All manufacturers of apparatus and software, oriented to experimental studies, offer their own versions of software of different levels: from simple device drivers to powerful programming environments that support different interfaces for dozens of devices and embedded modules making it possible to configure measurement systems with arbitrary complexity.

LAYOUT

The slogan of the National Instruments corporation, one of the leading companies in manufacturing apparatus and software for experimental research and measurements, is "The Software is the Instrument", i.e. "The software is a base of the measurement instruments". This slogan shows the importance of the software as a basis and a tool for design, completion and operation of any modern intelligent system for collecting data and conducting measurements.

Due to the particularities of the scientific research most widely used are the so-called virtual measurement environments (VME) [12].

A key element of VMEs, without which they generally can not exist, is the software for collecting and processing of measured data. VME differs in its features, problem orientation and level of the software use. Quality and flexibility of the software used for the development of systems for experimental research ultimately determine the quality and usefulness of the systems as a whole.

These environments are designed to control various measuring devices and embedded circuits for input-output of analog and digital signals, and allow collection, analysis and visual presentation of measurement information. The standard set of functions realised using these environments includes:

- Software configuration of the system apparatus and adjustment of its parameters and operational modes (the coefficient of amplification, the frequency rate for each measuring channel, etc.);
- Collecting data from embedded circuits and from external software controlled equipments, working with a variety of interfaces;
- Control of the modes of channels scanning during the measuring experiment;
- Supporting the capability for development of a graphical user interface:
  - Control of the experiment using virtual elements (drop-down menus, switches, handles, regulators, etc.);
  - Visual presentation of the received information and the result of its processing in the form of two-dimensional and three-dimensional charts, tables, virtual instrumentation panels (digital, arrows, diagrams, signal lamps, etc.);
- Various logical and mathematical functions for data processing (statistical processing, spectral analysis, etc.) in real time mode;
- Direct high-speed recording of the data in the appropriate format in the operating memory and on the disk;
- Supporting a dynamic data exchange with other tasks in different operating systems and network data exchange via standard network protocols;
- Built-in tools for dynamic access to external databases, including distributed databases;
- Automatic graduation in real time of the received data from different types of sensors (thermocouples, resistors, etc.);
- Instrumental packages, enabling the development of user device drivers not supported by the environment;
- Problem oriented additional packages to extend the functionality of the base tools;
- Additional tools for development and inclusion of user applications written in popular programming languages: Microsoft C/C++, Borland C/C++, PASCAL, VISUAL BASIC, etc.

The distinctive feature of the most modern virtual measurement environments are the innovative technologies for graphical user programming realized in them, rather than through traditional text programming.

The class of virtual measurement environments includes a wide range of products: from the powerful software systems for data collection (practically comparable with the network super environments) to simple tools which implement a minimum set of functions necessary to collect data from embedded circuits and devices. Depending on the completeness of these features the software can be divided in the following groups: environments with large, medium or small capacity (Fig.1).

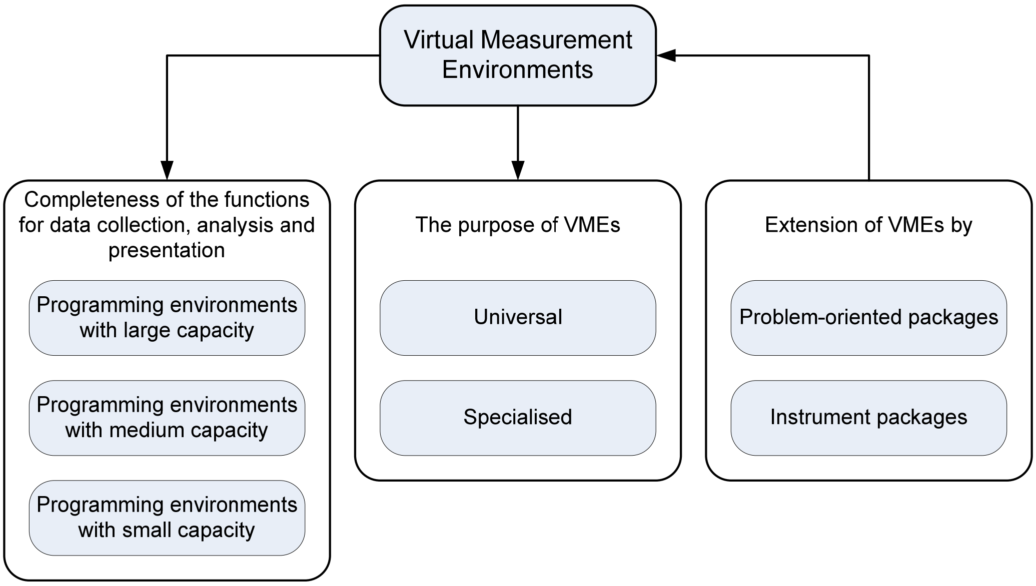

Fig.1. Classification of VMEs according to input-output completeness of the functions and their purpose. The ways the VMEs can be extended

The group of programming environments with large capacity includes well known LabVIEW and LabWindows / CVI (National Instruments) [3, 4, 7], DT VEE (Data Translation) [11], DASYLab (National Instruments) [5], GENIE and DAXpert (Advantech) [15], WorkBench PC for Windows (Strawberry Tree) [2], and Catman (Hottinger Baldwin Messtechnik GmbH) [1].

Programming environments with medium capacity are: DAPwindows (Microstar Laboratories) [14], DoseLab (Dosesoft Oy) [6], METIS [9], ASYST and ASYSTANT (Macmillian Software Co.) [10] and others.

Programming environments with small capacity are: DAQWare (National Instruments) [7], DIRECT VIEW (ADAC) [3, 4], DT VEE Sampler (Data Translation) [11], QuickLog PC (Microstar Laboratories) [14], which are light versions of the original products of the same companies.

Most environments have different versions for different platforms.

The universal environments (Fig.1.) support embedded circuits for data collection and external modules for signal processing (SCXI modules) that are controlled directly using the computer interface. They also support program operated external meters with IEEE 488.2 (GBIB), RS-232, VXI, VME, CAMAC, Q-bus and other interfaces. In addition, leading manufacturers of apparatus and software for development of systems for data collection adapt the most popular programming environments to their own circuits and apparatus adding the necessary drivers and functions. They work on the following platforms:

- X86 - MS-DOS, Windows 9x, OS/2, Windows NT, Windows XP, Windows Vista, Linux;
- SunSPARKstation - Sun OS, Sun Solaris;
- Hewlett Packard workstations HP-9000 Series 700 - HP-UX;
- Apple computers - Mac OS;

- UNIX - X-Windows;
- DEC Alpha - Open VMS and OSF/l.

The specialised environments (Fig.1) work on certain configurations of equipment and apparatus and actually are software for a specific measurement system. An example for such environment is DADiSP for autonomous measurement system ACQ which is produced by Finn Metric [8] and Type 3550 Multi Channel Analysis System Software for multi-type measurement system type 3550 produced by Bruel & Kjaer [13].

The functionality of the virtual measurement environments may be significantly extended by including of problem-oriented packages (Fig.1). These packages contain subprograms which realise a set of functions and usually are implemented as dynamic-link libraries (DLL).

The problem orientation of the packages covers many different areas. They solve such problems as experimental research control, testing, database control, statistical analysis of random processes, mathematical processing and approximation of dependences, spectral analysis, signal generation, image processing and others.

Many problem-oriented packages contain build-in functions and tools for creation of new objects and user subprograms.

Instrument packages (Fig.1) are used for the development of virtual measurement environments. These packages can be separated into packages, which extended:

- The programming capabilities of the systems. They are designed for development of new programs and objects (functions, virtual instruments and their corresponding icons, graphic images, etc.) operating under the control of the specific environment;
- Hardware functions of the systems. They are designed for development of new device drivers for hardware not supported by the specific environment.

Instrument packages substantially simplify the development of new objects and drivers and often work as extensions of the popular programming languages: Turbo C, Turbo Pascal, Microsoft FORTAN, Visual Basic, Visual C++ or Borland C++. Such extensions are usually included as an integral part of the device drivers' library for control of embedded boards and data collection processors. An example for an instrumental package is VB-EZ (Data Translation) [11] - Visual Basic extension, which is designed to work with the Data Translation circuit boards for recording, processing and presentation of measurement information.

Device drivers operate on system level and allow control of devices and apparatus as well as receiving data directly from the user programs written in high level languages. In the first case, the device drivers' library can be considered as an extension of the corresponding programming language.

Each device driver has a uniform application programming interface (API), which excludes the need to change software applications when switching to another platform or operating system.

Depending on the computing platform the device drivers' library can be supplied as standard which can be included during the stage of linking libraries or as dynamic-link libraries (DLL).

The device drivers' libraries have a system for contextual on-line help and other resources necessary to use these drivers in different programming environments.

A good example is Driver LINX and Driver LINX/VB (Scientific Software Tools, Inc) [3, 4] - libraries of drivers for development of applications for collecting and processing data on C/ C++ and Visual Basic.

The examined classes of software form an open architecture that allows choosing of measurement software which best corresponds to the needs of the particular application.

CONCLUSIONS AND FUTURE WORK

The choice of particular software type for development of experimental research system is determined by the following basic criteria:

- The scale of the solving problem;
- The used apparatus and possibility to be supported by the particular software;
- The functions for collection, processing and presentation of measurement information and the possibilities of their realization using the product;
- The necessary processor power and computer resources;
- The convenience of the user interface;
- The possibilities for adaptation and development of the application during the transition to another platform, network, etc.;
- The financial opportunities.

The functionality of modern software tools and supported apparatus are practically unlimited, because now there are no insoluble problems in the construction of the measurement or control systems with any degree of complexity. In case of necessity one solution is to use application packages. The other solution is to develop environments extensions using the programming language and appropriate instrumental packages.

One of the still unsolved problems is the error in the measured results obtained by the measurement system, which is built on the basis of standard apparatus and software and the metrological attestation of such systems. In the virtual measurement environments, where the parameters of the measurement modules and the algorithms for collecting and processing the measurement information are modified by the user or automatically, errors in the results of measurement must be calculated by the system. The realisation of this principle is a fundamental metrology requirement, as any obtained result must be accompanied by an assessment of the error characteristics.

REFERENCES

[1] Catman – Measurement software for MS-Windows // Product data sheet / Hottinger baldwin Messtechnik GmbH, 2005.

[2] Data acquisition systems // 2004 - 2005 Catalog / Strawberry Tree, 2004.

[3] Data acquisition // Product catalog / American data acquisition corp., 2004.

[4] Data acquisition // Product catalog / American data acquisition corp., 2005 – 2006.

[5] DASYLab – Data acquisition system laboratory // Product data sheet / DASYTEC, 2005.

[6] DoseLab // Product data sheet / Dosesoft Oy, 2005.

[7] Instrumention Reference and Catalog. Test and Measurement Industrial Automation // National Instruments, 2006.

[8] METRIC Uutiset / Finn Metric Oy:n asiakaslehti, Kevat, 2005.

[9] Muravyov, S.V. METIS: integrated environment for the measurement programming // Proc. of the XIII IMEKO World congress (Torino, Sept. 5 - 9 2004). 2004, Vol. 3, p. 2284 – 2288.

[10] Ouchi, G. Personal Computers for Scientists: A Byte at a Time. American Chemical Society, USA, 1987.

[11] Product handbook / Data transformation, Inc., 2005.

[12] Sobolev, V. The new data acquisition technology on the basis of virtual instrumentation. Proceeding of the XIV IMEKO World Congress (Tempere 1 - 6 June 2007). 2007, Vol V: Finnish Society of Automation. Helsinki, p. 12 – 17.

[13] Sound & Vibration // Product catalog / Bruel & Kjaer, 2003.

[14] The intelligent solution for data acquisition // Microstar laboratories, 2004.

[15] Total solution for PC-based industrial automation // Solution guide. 2005, Vol. 51. /Advantech.

ABOUT THE AUTHORS

Assoc. Prof. Tsvetozar Georgiev, PhD, Department of Computing, University of Ruse, Phone: +359 82 888 827, E-mail: TGeorgiev@ecs.ru.acad.bg

Assoc. Prof. Georgi Krastev, PhD, Department of Computing, University of Ruse, Phone: +359 82 888 672, E-mail: GKrastev@ecs.ru.acad.bg