AUTOMATION OF NEUTRON SPECTROMETRY EXPERIMENTS USING NETWORK TECHNOLOGIES

Ignatovich V.K.[a], Salamatin I.M.[a], Salamatin K.M.[b], Senner A.E.[b]

[a]*JINR, FLNP, Joliot-Kurie 6, Dubna, Moscow reg., 141980 Russia*
[b]*Intern. University "Dubna", University 17, Dubna, Moscow reg., 141980 Russia*

Abstract
Standardized means of experiments description and of operations sequence control, which do not require editing system components, when experiment is changed, are proposed.

1. Introduction

A new structure of distributed software for experiments automation system (EAS) for series of research works in neutron spectrometry was developed. Every particular experiment needs a list of data recording conditions, and a method of data recording and processing. Upon completion of the experimental data analysis a new research plan with change of experimental procedures usually occurs. As a rule the new experiment requires hardware and software changes.

Modification of each EAS, as was mentioned in [1], requires a work of a team of several programmers for about 6 months. The known ideologist of the Code Gear of the Borland Software firm David Intersymon notes that such a long development time is associated with the lack of an adequate strategy for code reuse of existing programs [2]. Therefore, to shorten the new experiments preparation time one must be able to use existing software components. A goal of this paper is to show some methods, which help to make unified components and to use existing components without their modification when experiment conditions are changed.

The main question when creating a unified program is how to link individual components without their modification at such changes.

2. Linking components in software of EAS

At usual static linking the needed information (such as network addresses of interacting components and information about interfaces) is somehow fixed before the start of the EAS. The static linking is robust and allows faster execution of interaction and communication between system components. However a failure of one of the components during the EAS operation in this case leads as a rule to loss of efficiency of the entire system, and even small changes in the experimental procedure require modification of several components. The static linking significantly complicates the use of components in EAS without their modification. This defect is absent in dynamical linking, because necessary information to ensure the interoperability of components is produced during the EAS operation itself. Therefore the dynamical linking for EAS development has the advantage in the case of systems with frequently variable functions.

The dynamical linking uses network technologies, and the most popular of them were analyzed in [3,4]. As a result of this analysis a special version of a mediator program DiCME (*Distributed Components Messaging Environment*) for dynamical linking of components was developed. The mediator DiCME provides basic technological support for EAS software and

has the following key features:
1. automatic search and linking of components ;
2. asynchronous mechanism for remote execution of procedures;
3. transmission of information to multiple processes simultaneously;
4. processing of all interactions of components of EAS with the same means;
5. independence of the access to the components procedures upon the network address of the computer.

3. Possibilities of the components unification

If the experimental procedure requires new equipment not provided in advance, it becomes necessary to change the composition of the components used and to edit the sequence of operations in the experiment. There are two traditional methods of an experiment control:
1. usage of a fixed group of procedures written in a programming language, where each procedure implements a specific device of the experiment, and required parameters are set via a dialog box, or are obtained from a data file;
2. usage of an interpreter in EAS composition and preparing an experiment control program (script) in the form of a list of calls of external routines, and values of the parameters.

In both methods, in fact, the program of the operations sequence control in the experiment includes a list of procedures calls, which results in a rigid components coupling. Practice shows that error rate in written scripts can reach 70%, and the adjustment of EAS software, which uses scripts and configuration files can take up to two days [5]. Substantial simplification was proposed in [6,7], but these studies do not solve the general problem of scripts - the problem of components linking.

To solve the problem of unification and to develop a dynamical method of components linking, when the experimental technique changes, an analysis of the functional composition and of characteristics of the EAS software components was made. Based on this analysis the following features were found [8]:
• dynamical linking of components for remote procedures is necessary only for the main functions: management of conditions for registration, conversion and archiving of experimental data;
• the result of a remote procedure must contain 1) a mandatory shutdown signal, addressed to the caller (the experiment managing program or the user interface) and 2) detailed information (list of registered data, description of the managed object state, a diagnostic message , etc.) addressable to auxiliary functions;
• implementation of auxiliary functions (data visualization, feedback, etc.) requires a special method for connecting components, taking into account the spontaneous nature of requests for such operations and the independence of the basic functions (in normal conditions) upon the results of operations of the auxiliary functions.

Having in mind these features, it is proposed in the given paper to exclude the procedure calls from the experiment description and to separate the experiment managing tool from the description of the experimental procedure. The proposed method of the experiment management is as follows:
• experiment managing program completes two abstract processes (conditions formation and data registration) and by default assumes that the data registration goes

after the formation of the conditions. The program dictates the conditions for the data registration in an experiment description file;

• specification of these abstract processes (for example, the composition of the components used) is performed dynamically using the file with conditions of data registration created by a subsystem of experiment description.

4. The subsystem of the experiment description

To describe the methodology and compose experiment description file information about the devices and components working with them (devices passports) is required. Detection system and the data input controller for the spectrometer are fixed. A set of components performing data input, conversion and archiving, are usually combined into a data acquisition (DAQ) subsystem, and call of this subsystem can be performed by default after the formation of the data registration conditions. Subsystem describing experimental procedure includes a database (DB) and two interactive programs: 1) the one used by programmers for producing passports of devices and components, managing the data registration conditions; 2) and the other one, PSJ (Preparation of Single Job), used by experimenters.

4.1. The program for devices passports compilation creates and stores in the DB the documentation: devices description in the JSON format. Documentation includes: the name of the controller and a list of devices connected to it; the type of the component working with the controller, its unique identifier (GUID), used for component addressing, and descriptions of parameters. Parameters composition (application protocol) is defined by the component developer.

4.2. The PSJ program uses the list of available devices (and components) from the database. From this database, the user selects in a dialogue the desired components in his experiment and makes a list of devices that in the process of the experiment will determine the data registration conditions. For each device in the list one can specify a list of consecutive values of the used control parameter. The result of this work is PSJ-file with a table describing the desired states of hardware system in the experiment. In each state, a data exposition is performed. The structure of the PSJ file for an experiment is shown in Fig.1.
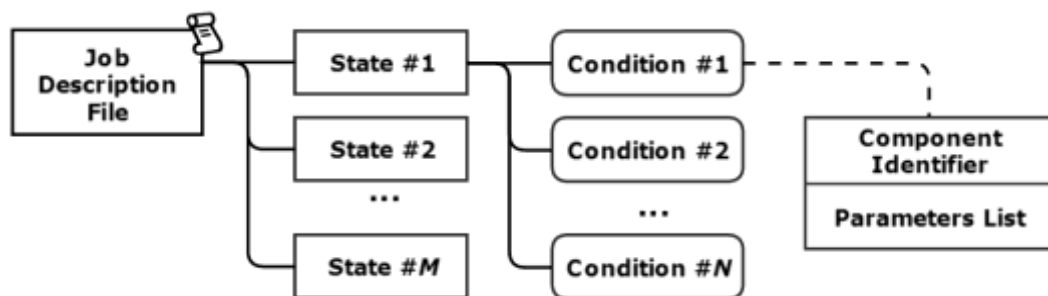

Fig.1. The structure of the PSJ file for an experiment.

A distinctive feature of the method proposed in this paper from those used in other systems is that the PSJ program produces a list of data registration conditions instead of the actions described in the form of a procedure calls list in a script file or hard-coded procedures.

Description of the experimental technique uses only experimenter's terminology: names of the spectrometer components, angular positions, etc. The program automatically

composes PSJ file with components identifiers (GUID) in the JSON format. A user interface of the job preparing program is shown in Fig.2.

5. An experiment management with a mediator DiCME and PSJ files

Fig.3 shows the interaction of the experiment management program (EMP) and mediator DiCME in basic operations of the EAS software. The PSJ file comes to the EMP entry. EMP selects the consecutive description of the system state (list of the data registration conditions) and sends a description of each condition to the mediator DiCME. Description of the condition carries information about the component (GUID) sufficient for its search and linking to EMP, and also a list of parameters. DiCME finds the needed component and passes to it a message describing the condition containing the list of parameters. EMP functions as a dispatcher of the components used. The choice and call of needed procedures that implements the required data registration conditions is performed in the components that drive the hardware, based on the interpretation of the referred to them conditions descriptions. For example, to manage the position of the polarizer condition description in JSON format looks as follows:

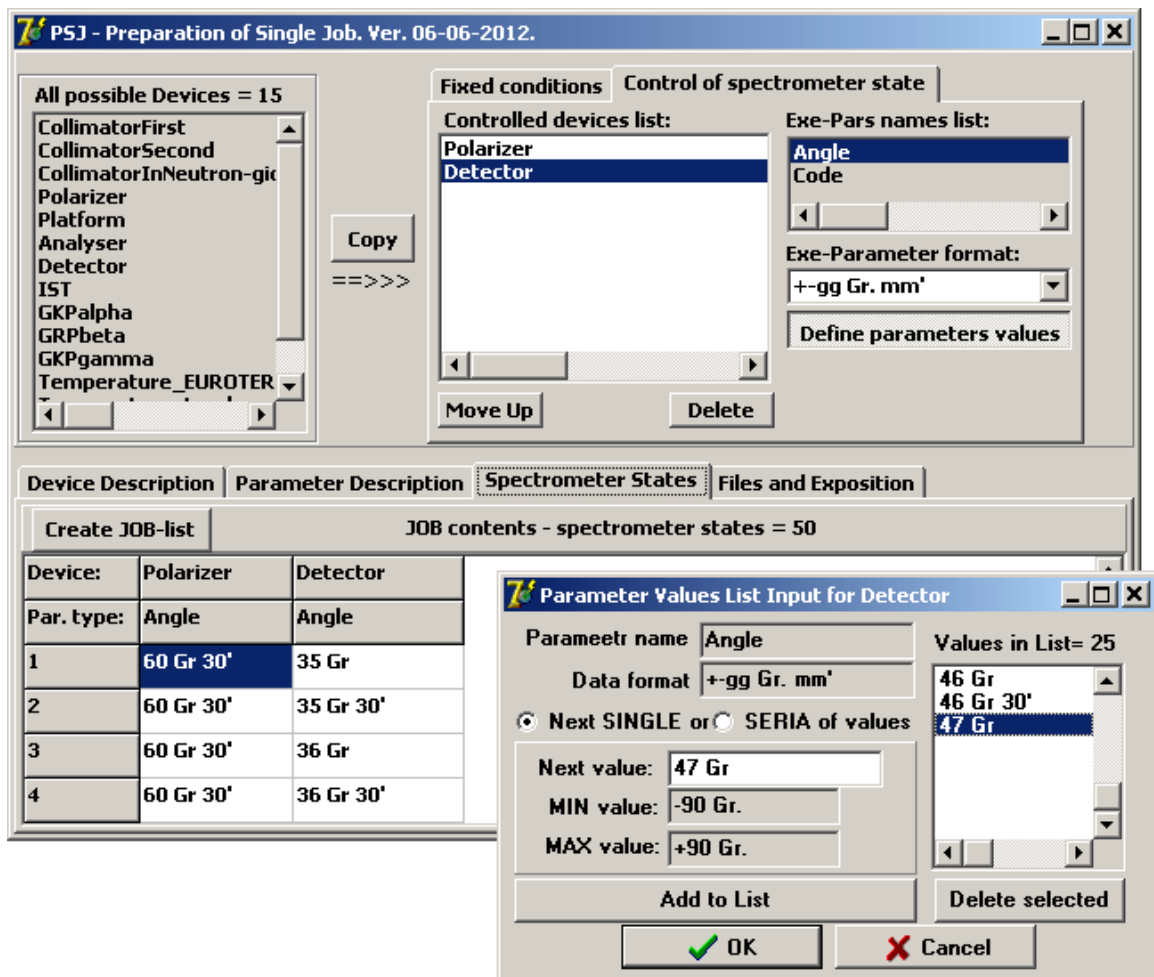{device:"polarizer", parameter:"angle", value: "30 grad"}

Fig.2. The user interface for PSJ file preparation.

Each component, which receives a message from EMP should return to EMP signal (DONE/ ERROR) of the operation completion. After receiving of such signals from all the devices listed in the description of the desired EAS state, the EMP starts DAQ subsystem. Completion signal of the data acquisition allows the EMP to switch to processing of the next state in the job description file.

Using dynamic linking method [4], contrary to the CORBA etc. technologies, one does not need preparatory dialogue between the components to configure remote execution procedure. The described linking method does not limit the development of the experimental procedure, and its changes do not affect the EMP and inter-component communication mediator DiCME, because the EMP and DiCME are transparent to the parameters lists.
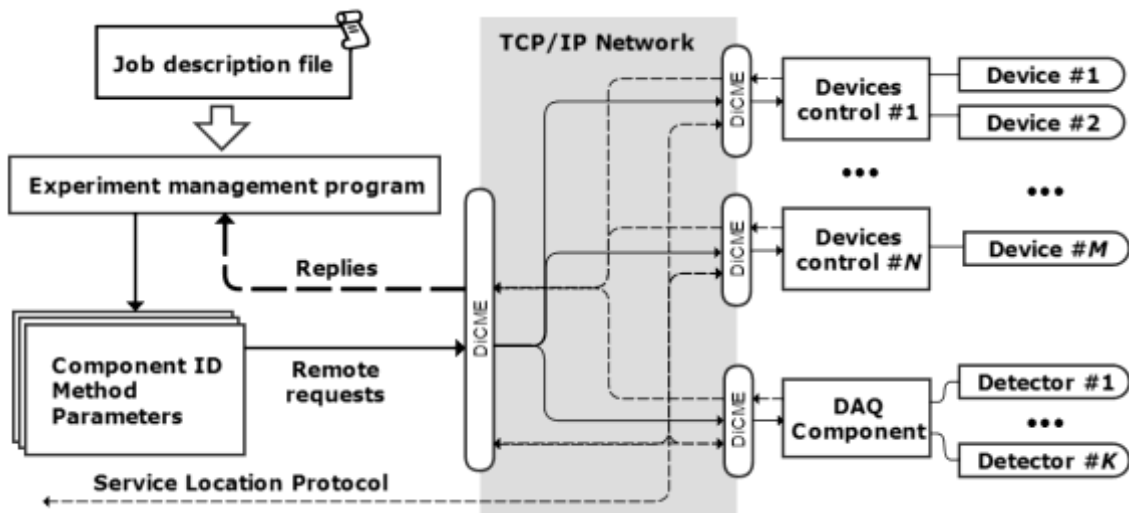

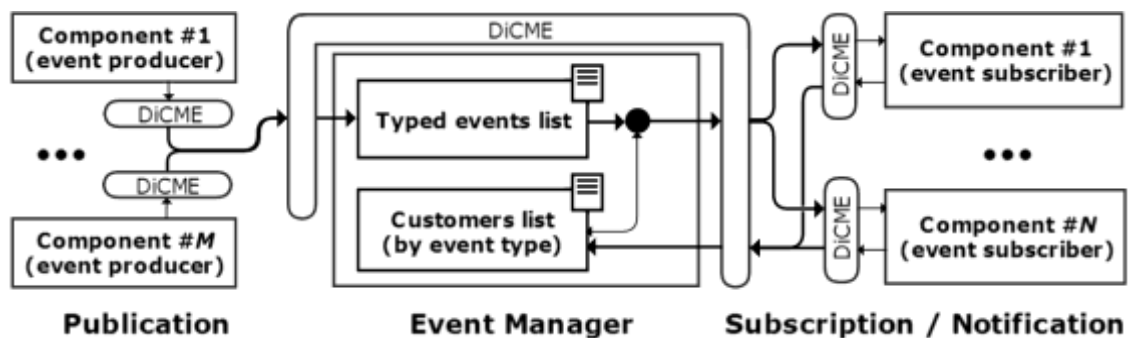Fig.3. Scheme of interaction of EMP and mediator DiCME in implementing the basic tasks of the EAS.

Fig.4. Scheme of the event manager and mediator DiCME serving auxiliary operations.

The most significant features of auxiliary operations are as follows:
• spontaneous nature of information about events and the ability to postpone its treatment;
• a necessity to transmit information to multiple processes, the composition of which, generally speaking, is not known to the source of information and can be changed by EAS user during the EAS software working.

To link the auxiliary components a version of the "subscription" algorithm is used. In this algorithm the component-consumer once declares interest in a particular type of information. Then a special component (Event Manager, a part of DiCME) caters for all "signed" consumers with the interested information, when it appears. A schematic diagram of a mediator DiCME serving auxiliary operations in EAS is shown in Fig.4.

As a result, the implementation of auxiliary operations through DiCME allows a user to dynamically arrange the system with the necessary auxiliary functions simply by running on any computer of local network of the desired component.

6. Experiment management program

The method of constructing a unified EMP and the assignment to the experiment is described above in detail. The practice of precise measurements requires some additions. Fig.5 shows a diagram of the management program in the high precision measurements. The diagram shows that many hours of exposure in each state ("States control cycle" in fig. 5) of the experimental setup is divided ("Statistics accumulation cycle" in fig. 5) into several (usually identical) shorter runs, the results of which can be summed. Due to this, comparison of the data in separate files, obtained under same conditions, permits to detect possible drift of background, of detector efficiency and other effects, and filter out incorrect data.
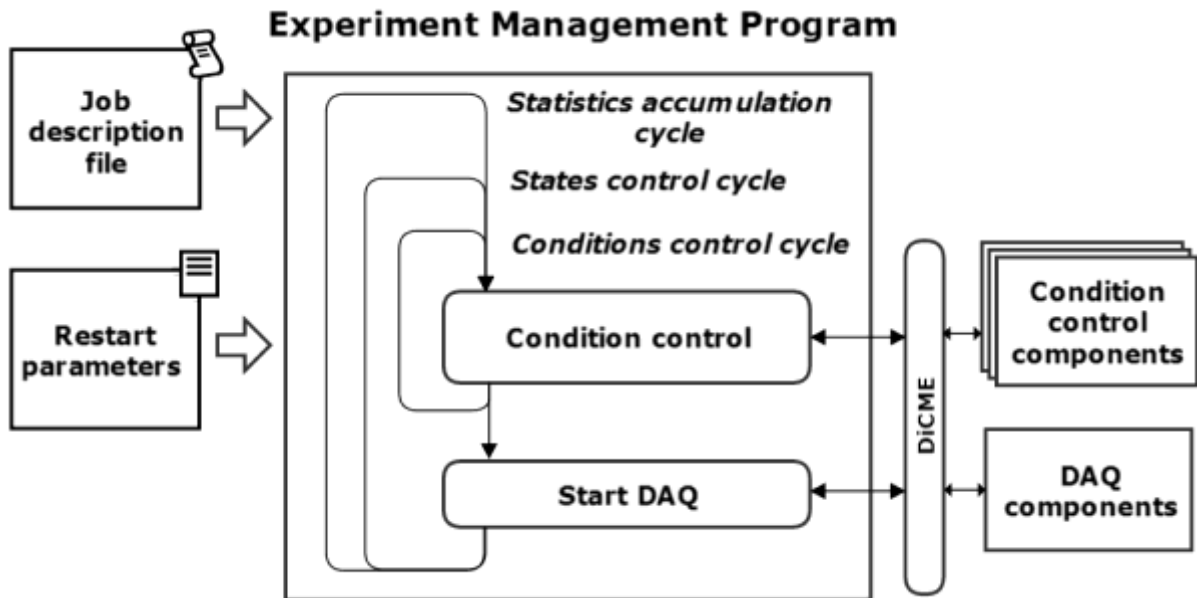

Fig.5. Scheme of the experiment management program.

This EMP protocols the job: program remembers the last command, the number of the last completed state of the system (executed jobs line), and some other data, and uses it to automatically generate file names. This means that at failures it is possible to restart the system. After restart it will continue to work starting with the last unfinished state. Local menu provides several debug modes.

7. Key findings

The subsystem which describes the experimental technique forms the information base for components dynamic linking algorithms, for specifying the functional composition of the processes performed by the experiment management program, and for automatic integration of components into the EAS software. This subsystem significantly reduces the software adjustment for a new experiment and facilitates the work of the experimenter, which is a key element, because it made possible to transfer the EAS software to experimenters themselves to manipulate it, when the experiment is changed.

Model of EMP performing two abstract processes (formation of data registration conditions and data registration), is appropriate for different experiments and in other systems.

Proposed EAS software is a problem-oriented distributed software package of applied programs, including managing components for the package (means of interoperability of components and experiment management program) and the standardized application components. Subsystem of experimental technique descriptions provides a language for package management. The proposed structure can be used in other areas of human activity, for example, for technology processes automation.

The approach presented here was applied to the development of EAS software for spectrometers used in experiments on neutron sources IBR-2 and IREN at JINR [9- 11].

References

1. *P. Levis, G. Cooper, F. Trouw et al.* Data Acquisition and Instrument Control at the Lujan Center: An Update // NOBUGS 8, Preliminary Agendas NOBUGS34, 2010.
2. *Derril Taft.* What we wait from the year 2027-th. // PC Week/RE №8 (614) 11-17 Mar. 2008.
3. *Salamatin K.M.* The choice of technology for constructing software systems for automation of experiments in the field of neutron spectrometry // JINR P13-2013-72, Dubna, 2013, 20 p. (in Russian).
4. *Salamatin K.M.* DiCME - distributed components messaging environment for automation of experiments for low energy physics // Software Engineering number 3, pp. 3-11, 2014. (in Russian).
5. *M. Konnecke, N. Hauser, F. Franceshcini et al.* Treepath Based Instrument Control // NOBUGS 6, №132.
6. *T. Nakatani, Y. Inamura, T. Ito, T et al.* Present status of the computing environment for the experimental instruments in J-PARC/MLF // NOBUGS 8, Preliminary Agendas NOBUGS40, 2010.
7. *M. Clarke, T. Nakatani, Y. Inamura et al.* Making life easier for scientists: developing an Experiment Scheduler for the MLF // NOBUGS 8, Posters NOBUGS20, 2010.
8. *Salamatin I.M, Salamatin K.M.* Development EAS for low energy physics through the use of network technologies // JINR P13-2013-74, Dubna, 2013, 24 p. (in Russian).
9. *Tsulaia M.I., Salamatin I.M., Sirotin A.P. et al.* The Kolkhida Setup Upgrade // ISINN-21 (Alushta, Ukraine, May 20-25, 2013), Dubna E3-2013-40. Dubna: JINR, 2013, p. 77.
10. *Kuznetsov V.L., Kuznetsova E.V., Sedyshev P.V.* Investigation of Possibilities for the Measurement of Parity Violation in Neutron Diffraction at the IBR-2M Reactor // Proc. of Internat. Seminar ISINN-20 (Dubna, May 27-29, 2012), Dubna E3-2013-22. Dubna: JINR, 2013, pp. 66-68.
11. *Shvetsov V.N., Alpatov S.V., Astakhova N.V. et al.* 8-input system for neutron-nuclear research by the method of time-of-flight // Instruments and Experimental Techniques, Vol. **55**, No. 5, pp. 561–568, 2012.