## ISaGRAF extensions. Innovative functionality, productivity and openness

There is an overview of fundamental and flexible extensions of ISaGRAF target implemented by experts of FIORD company which have essentially expanded ISaGRAF functionality and increased performance and determinism is given at this article. These ISaGRAF extensions open path for further development of system according to real inquiries of users at the expense of usage of open and with detailed documentary interfaces. Basis for ISaGRAF extensions is ISaGRAF FIORD Target, the innovative version of the target task.

### ISaGRAF FIORD Target is a basis for extensions

**ISaGRAF FIORD Target is** standard target systems from ISaGRAF of Rockwell Automation (Canada, **www.isagraf.com**) essentially processed by experts of FIORD company (**www.fiord.com**) (**www.isagraf.ru**). Processing the source code of target system has consisted first of all in usage of library ACE (Adaptive Communication Environment) for implementation of system level. It has caused carrying over of a source code of ISaGRAF target in the environment of the compiler of language C++ and abstraction from features of operation with various operating systems. What was in a result? It was possible to eliminate various "parasitic" delays and to provide execution time of the task within the limits of one cycle in a mode "Real time" from 5 micro seconds, to raise stability of a cycle with the set execution time.

ACE library has appeared during research activity and scientific developments of the Dr. of Douglas S.Schmidt at the Californian university in Invine, the templates routed on designing, representation and the analysis of object-oriented technologies promoting development of high-efficiency, distributed computing systems of real time. ACE library is the powerful toolkit intended for creation of complex multiplatform communication applications became result of operation. ACE it is freely used for any applications, including commercial, without any license agreements. ACE it is widely applied all over the world, and also is one of the most transferable C ++ libraries, it supports tens platforms and continues to develop. As all system-dependent functions have been substituted by appropriate calls of functions of library ACE, target system source texts are uniform for different operating systems. At present time it is fulfilled porting and testing of new ISaGRAF target platform for following OS: Linux, QNX 4.25, QNX Neutrino, Windows 7/8/Vista/XP, Windows Embedded, FreeBSD, OpenSolaris. ACE gives possibilities which can be used for creation of effective user's device drivers of input/output, intrinsic functions and function blocks. Also there is a simple platform to platform carrying over of drivers for protocols using standard communication devices (RS-232, Ethernet, USB) as it does not demand change of source texts of drivers. It is enough simple recompilation.

ISaGRAF FIORD Target is invariant to a platform. All finite runtime systems have identical possibilities and the uniform description in the environment of ISaGRAF Workbench, i.e. carrying over of the existing project on a configuration from other OS will not demand its recompilation.

### Fast data access system (FDA)

One of the most interesting constituents of runtime system ISaGRAF FIORD Target is system FDA (Fast Data Access), intended for data access of real time of executive system ISaGRAF FIORD Target by inquiries from OPC-server FDA-OPC. Real time data are understood here as current ISaGRAF data at the moment of reception of inquiry about their output. FDA system consists of two parts:

1. Runtime (executive) subsystem ISaGRAF Fast Data Access **(ISaFDA)** is constructed on the basis of ISaGRAF technology and fulfils functions of tracing and saving of values of the changing variables for their reading and recording by inquiries of an OPC-server. The subsystem is implemented in the form of ISaGRAF virtual device and includes also demand processing tools on output of data and installation of new values of ISaGRAF variables.
2. Server subsystem **FDA-OPC** is an OPC-server of system of ISaGRAF fast data access. FDA-OPC is intended to extract ISaGRAF data from the controller, access allocation to them by OPC-inquiries, and also for setting of new values of variables ISaGRAF. FDA-OPC is executed on MS Windows platform and corresponds to specifications OPC DA 2.x, OPC DA 3.x. OPC-server customisations allow to work simultaneously with several controllers. On operation with the selected controller the OPC-server is customised automatically at path instructions to the compiled project ISaGRAF loaded on the target controller. On fig. 1 the common diagram of system operation FDA is presented.

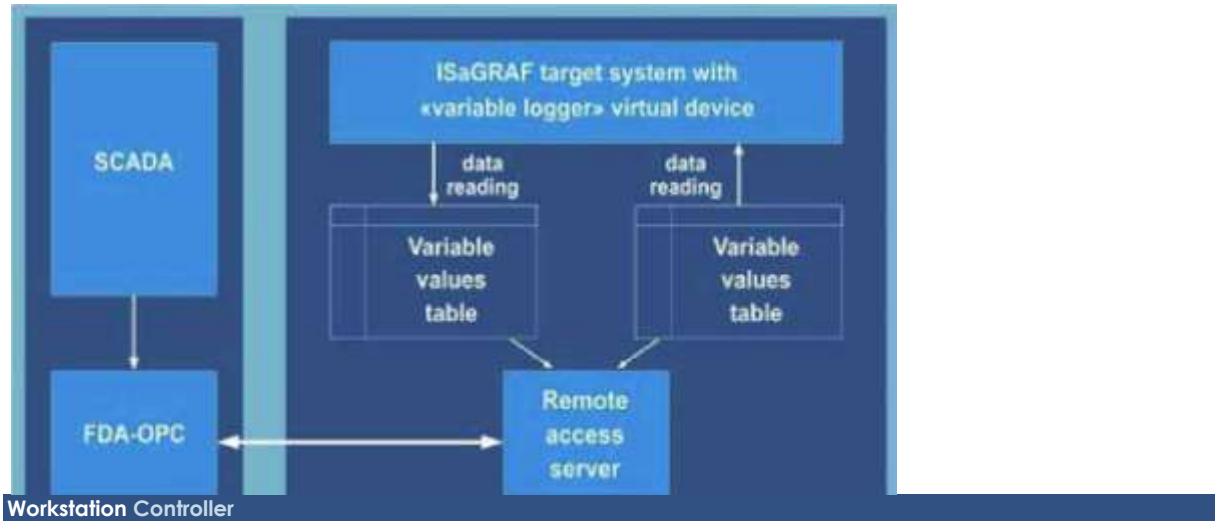


Fig. 1 System operations of FDA.

On a reading demand from the workstation fda_serv module executes data reading from the table of ISaGRAF variable values, then forms an answer and sends it back to the workstation. In the similar method new values of variables ISaGRAF are written to the controller. For detailed customization of ISaFDA subsystem the target system contains built in set of special functions defining the list of accessible for reading variables during project construction . FDA allows to work with variables of all primary ISaGRAF types (except STRING type), including derivative types (arrays, structures).
Following possibilities of configuration are offered:
■ Registration of the separate variable ISaGRAF, all user' variables of the specified type;
■ Registration by variable name , entire array or its parts (on elements),
■ Registration of structures (on fields), all fields of complex data structures (on elements);
■ The setting of a threshold of sensitivity, the permission (prohibition) of registration in each function.
The described functions also should be used only once before the operation begin. Development of the possibilities given by system FDA for registration, and also mapping of ISaGRAF variables in OPC-server FDA-OPC, is the new tool of customization, FDA Configurator. The given tool allows to refuse usage of special functions for registration of variables at project creation, and all operations on target system and OPC-server customisation to fulfill by means of the uniform tool. Thus in the project it is enough to connect only the device «Variable Logger». After setting of necessary customizations Configurator FDA creates adjusting files and loads them to corresponded controllers then restarts ISaGRAF resources on controllers to apply all changes. It allows to change the list of registered variables without modification of ISaGRAF project. FDA Configurator also creates the list of variables for FDA-OPC corresponding to customizations of resources, that allows to display in an OPC-server only the variables registered in controllers. Distinctive feature of system FDA is that only varied values of variables are registered, so on a network the superfluous information is not sent that allows to accelerate data exchange essentially. Value of each variable in the table is accompanied by the information on its type and so-called virtual ISaGRAF address. On the basis of this information from the symbol table which are in the directory with project Workbench, a variable its name can be again compared, that, purely, and
fulfils FDA-OPC.
Diagram presented on fig. 1 describes operations with one ISaGRAF resource, in presence of several resources for each of them tables of variables are built, and the utility fda_serv accesses the necessary table according to number of a resource specified in inquiry from an OPC-server or other client application. Data transfer is carried out by inquiries FDA-OPC (or other application) and represents data exchange under the method "inquiry-answer" and as the transfer protocol of data UDP protocol is selected. The choice of UDP protocol was made on a reason that powerful mechanisms supporting the reliability of TCP protocol are not mandatory for solution of described task of, besides simplicity of implementation of UDP protocol allowed to reduce development time. As UDP protocol does not provide reliability of data transfer, the specified disadvantage is compensated by usage of additional

mechanisms to rise the reliability, built in applications implementing data exchange. Besides static timeouts and repeated inquiries of data in case of occurrence of errors, these mechanisms use serial numbers of packages to guarantee that the received answer corresponds to the transmitted inquiry. Integrity of transferred data is provided at the expense of the field "length" UDP-datagramm, and also at the expense of usage built in count of check total UDP. When the inquiry of data output arrives the attempt to read the next chunk of data from a table of variables is been implemented. The size of a chunk of data can be varied depending on presence of data in the table at the moment of reading attempt. The maximum size of data produced for one inquiry is defined by settings of demand processing utility in which parameters possible size of outgoing buffer UDP is defined. Thus, each answer represents one UDP package that eliminates the necessity to divide and then to gather packages by transmission of a considerable quantity of data.
The protocol allows working with data of each resource of ISaGRAF runtime system functioning on a controller if the data logging mode is included for it. As UDP protocol does not require installation and connection procedure, it has allowed to raise considerably a transfer rate in comparison with TCP and it is essential to simplify connection. The description of transfer protocol of data is delivered together with manual of FDA system in structure development tools FDA DevKit that allows to develop the own client applications which are carrying out data exchange with runtime system ISaGRAF FIORD Target.
The main benefit of FDA is reliability and data transfer high speed (up to **500 thousand ISaGRAF variables per second).**

## The distributed ISaGRAF Archive System (IAS)

**ISaGRAF Archive System** is intended to carry on archives of historical data on controllers with ISaGRAF target system, to collect stored information to a uniform archive base and to perform further analysis of archive records. IAS represents 4 levels system of program components:
1. **IAS Logger** is a lower layer of the distributed historical data system of IAS. It is intended for accumulation of archive records delivered by ISaGRAF target system and saving it to a disk of the controller, and also for providing access to saved data locally or on Ethernet network by inquiries from a top level. IAS Logger is constructed on the basis of ISaGRAF technology and fulfils functions of ISaGRAF data saving to a local archive of the controller. The subsystem is implemented as ISaGRAF virtual device and includes also demand processing tools for data output and the handle of the archives intended for transportation of given local archives on a server by inquiries. Subsystem customization is carried out by means of the configuration files containing the list of archived variables and set for each ISaGRAF resource. For each variable the threshold of sensitivity for record can be set. IAS Logger customisations allow to limit also the maximum size of disk base of the local controller, thus on filling of the taken away space new data are written instead of the old. Thus, the system can store the actual archive records stored during defined time segment.

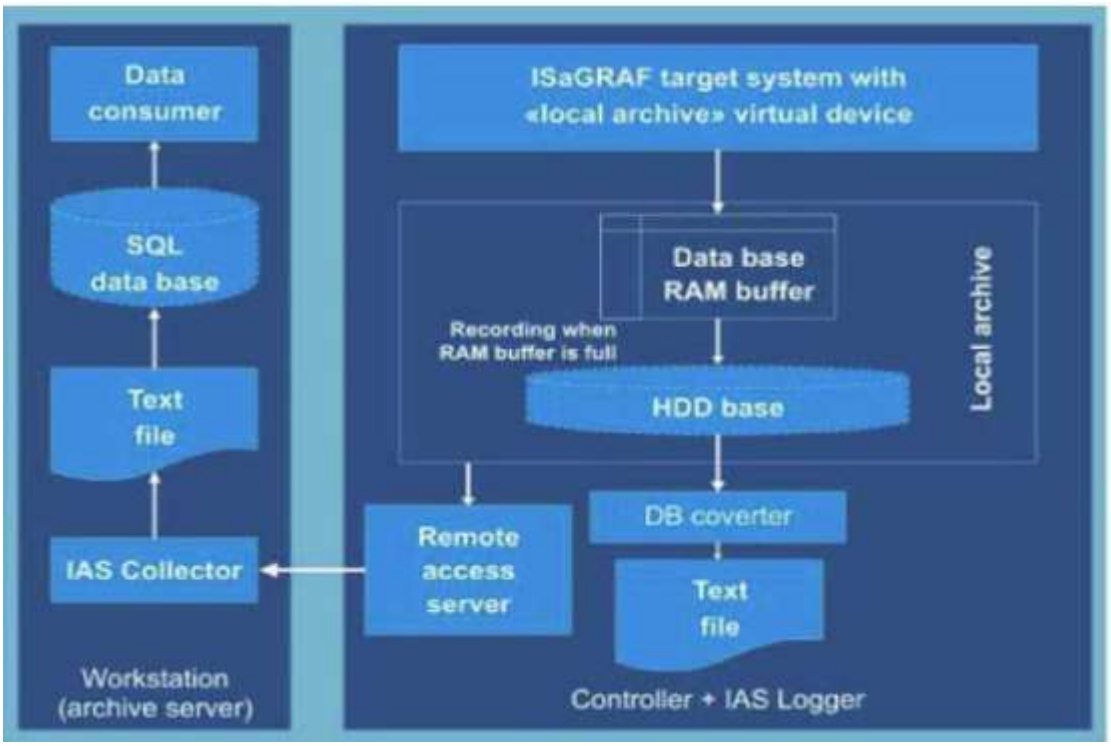


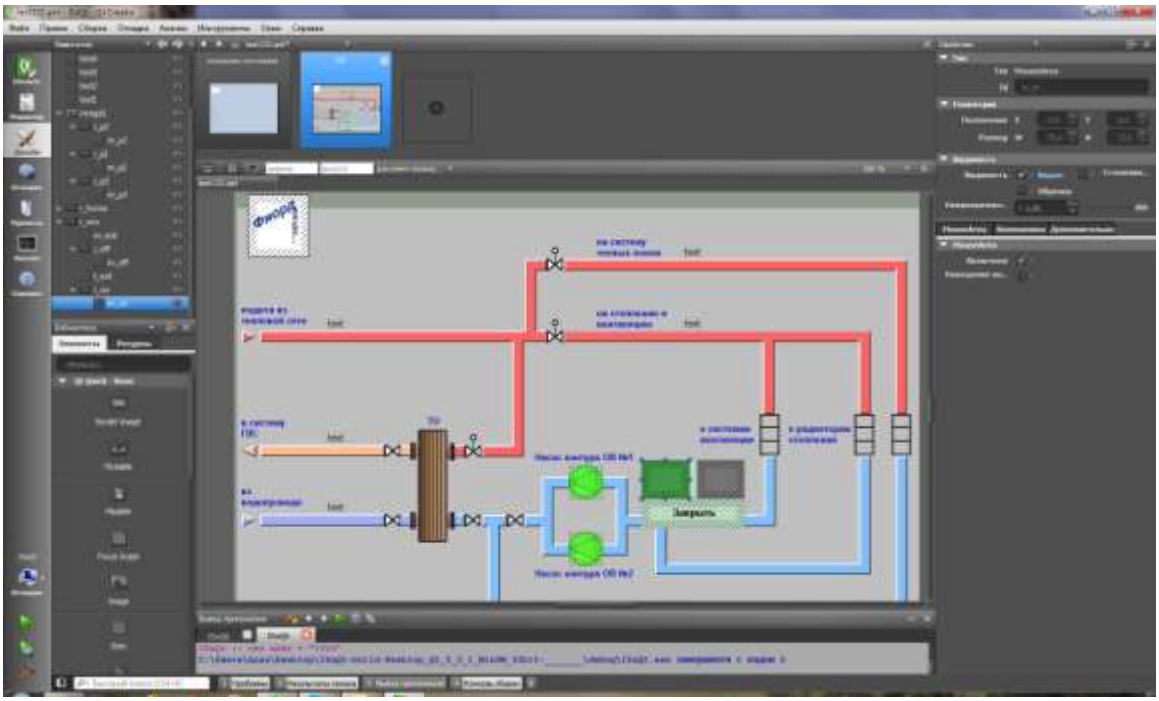
Fig. 2 The distributed archive system operation diagram.

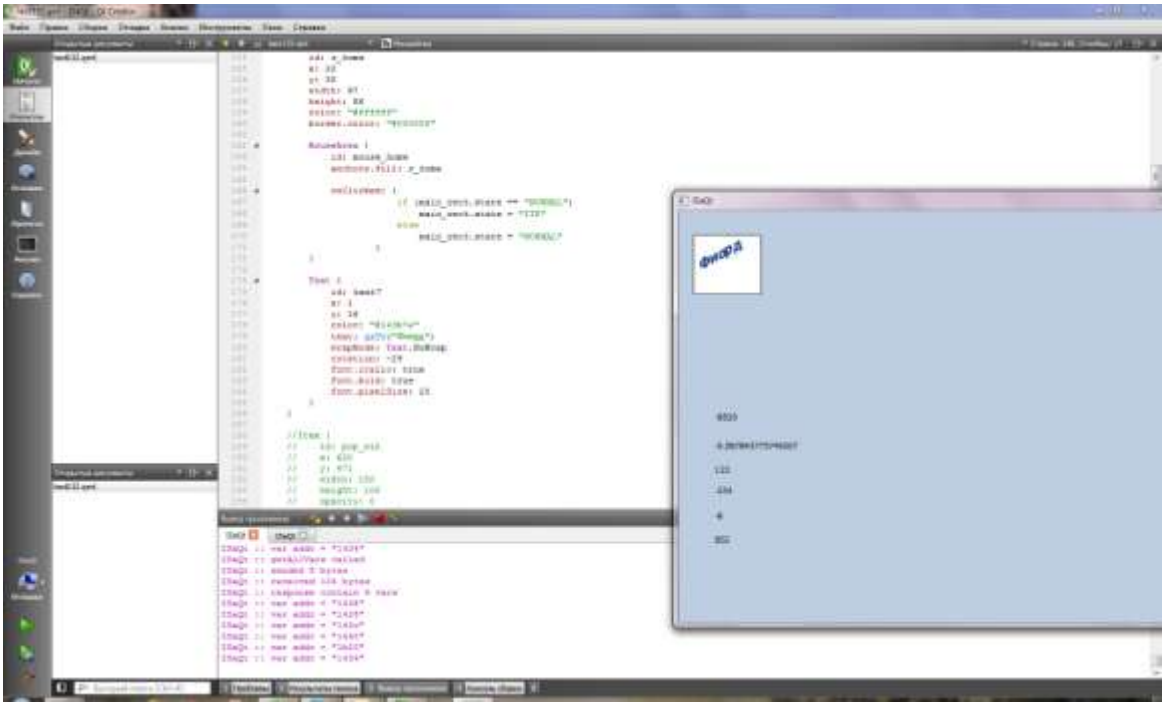
2. **IAS Configurator** is the archive system configurator, is intended for import of projects ISaGRAF to SQL database (MS SQL or PostgreSQL), customizations of the variables which are archive subject, creations of configuration files and their loading in appropriate controllers. The configurator also creates the scripts allowing to automate process uploading archive records from controllers and their import to SQL database. The program has the interface similar to FDA Configurator nterface. Possible application is remote configuration
of local archiving system, convenient tools for configuration of the big systems, replacement of "hand-held" configuration.
3. **IAS Collector** is a system for collection of archive records from controllers, data record in text files or in MS SQL Server database, PostgreSQL. Possible application is archives remote uploading to a database, the centralised collection of archives from various sources in a uniform database.
The carrying on archives system can function in two modes:
1. Continuous data storing on a local controller and periodic transportation of data to an archive server. In this mode data stored on a hard disk of a controller from the moment of the last transfer are transferred.
2. Continuous data storing at continuous transportation of data to an archive server. In this mode data are been taken directly from the RAM buffer, disk writing is performed on network faults only. The mode of functioning carrying on archives system is defined by quantity of the variables which are subject of storing, frequency of change of their values, and also of network transmission settings and the RAM buffer size.
Archived data are been put to the RAM buffer and, when it is full, data are been written to a controller disk. The stored data can be extracted from controller archive locally (by means of the special converter) or on network Ethernet (using the remote access mechanism). Each value in archive is accompanied by a time mark to within a millisecond. The main properties of carrying on archives system:
• Storing only the changed variables and periodic delivery of stored data to an archive server;
• Storing variable of all primary ISaGRAF types (except type STRING), including the types derivative of them (arrays, structures);
• Reliability and data transfer high speed (up to **200 thousand ISaGRAF variables per second);**
• Remote control by local archives and statistics reception about a state of local archives in real time;
• Continuous data gathering and saving on a disk of the controller in case of network faults;

• Possibility of integration with SCADA-systems;
• Backup of archive records on several archive servers.

## Graphic interface ISaQT

ISaQT system is intended for support of interactive graphics visualization of real time data directly in runtime system of ISaGRAF FIORD Target, using FDA protocol. So ISaQT can render the data changes on the PLC in real time both locally and remotely from another computer. ISaQT may interact with multiple controllers at a time, so you can do the visualization not only for autonomous, but also for distributed automation systems. ISaQT is already implemented for Linux and Windows. To create the mimics it is necessary to use Qt Creator, which forms the interface on the QML 2 language, dynamic component on the JavaScript. Qt Creator is free cross platform IDE for developing in C, C++ and QML on Qt Framework. Generated file is interpreted by ISaQT. As a result, the user gets a GUI on the controller.





## Additional function libraries in ISaGRAF FIORD Target

The structure of runtime system includes **library of functions for fast data processing** allowing considerably to accelerate real type variables arrays and matrixes processing. The given functions are executed in the target system with speed of the executed machine code and allow to implement in ISaGRAF applications high-efficiency algorithms of data arrays processing which necessary, for example, for operation with audio and video streams, the impulse forms registered AD converters:

1. ***The set of functions Fastarray*** is intended for fast performance of mathematical operations with arrays of real type variables and includes following functions:

■ Filling of the array with a preset value or linearly changing values;
■ Calculation of the sum, element multiplication or scalar multiplication of arrays;
■ Multiplication of all array elements to the set number (scaling);
■ Finding the sum of elements, minimum or a maximum element in the array;
■ Copying of the contained array with other array.

Operations with arrays of real type variables with usage of given functions are implemented **10 times faster** than similar operations in program on ST language.

2. ***The set of functions Fast matrix*** is intended for fast performance of mathematical operations with matrixes and gives following possibilities:

■ Zeroing of all elements of a matrix or filling of a matrix with a preset value;
■ Calculation the sum of matrixes, matrix multiplication on a vector or multiplication of all matrix elements to a set number;
■ Copying of contents of one matrix in another;
■ Solution the system of linear equations.

Operations with matrixes with usage of given functions by results of tests are implemented **60 times faster** than similar operations in program on ST language. All operations with arrays and matrixes are made using the special objects-indexes representing structures of certain sort. The given objects-indexes preliminary form in the dictionary of variables Workbench and are initialized by means of special functions of creation of arrays and matrixes accordingly. Functions of creation (initialization) for each desirable array or a matrix should be called once in the operation beginning, it can be made in the program in ST language, using a variable of type BOOL as a flag dumped after performance of demanded operations.

***3. Functions of measurement of time with a high-resolution (microsecond).***
***4. Functions of signals processing (FFT, digital filters) on base fastarray***
***5. Functions of operation with containers C ++ (vector, list, hash, bitset).***

***6. Functions for operation with COM-ports:*** opening, closings of COM-port, reading and data record from () it, installation of signals DTR and RTS COM-port and some other.
***7. PID-REGULATOR***
8. ***IODevKit:*** gives to the developer of drivers possibility of usage of language C++ and libraries ACE for creation of effective device drivers and intrinsic functions.

**Conclusion**
Extensions of target task ISaGRAF on the basis of runtime system ISaGRAF FIORD Target give a complete complex of solutions for high-speed processing, handling and delivering data to a top level of the information management system, archive storing and graphics visualization of data at controller level. Base possibilities ISaGRAF FIORD Target open path for further development of system according to real inquiries of users at the expense of usage of open and in details documentary interfaces On sites www.isagraf.ru and www.fiord.com evaluation versions of ISaGRAF FIORD Target runtime system are accessible to various operating systems, and also the documentation and demos of various extensions.