

# OpenSCADA 0.6.4

## Contents table

<a href="#">OpenSCADA 0.6.4</a> .....	1
<a href="#">Introduction</a> .....	1
<a href="#">1 Implementation of planned tasks</a> .....	3
<a href="#">2 Optimization, stability, sustainability and system performance increasing</a> .....	4
<a href="#">3 Improvement and stabilization of the VCA</a> .....	5
<a href="#">4 Translation of documents in English and the program's interface in German. Multi-language database</a> .....	6
<a href="#">5 System-wide expansions</a> .....	6
<a href="#">6 Plans for the further development</a> .....	8

## Introduction

Release of the open SCADA (Supervisory control and data acquisition) system of version 0.6.4 is the planned release of the 0.6.0 branch. Withing the limits of this version the considerable work on the stabilization and adaptation to the practical problems is made, and also the significant improvements are implemented. This version of the OpenSCADA system is positioned as an official beta version of the stable 0.7.0 version of OpenSCADA for the platform "Linux x86/x86\_64".

This document is the processing (compilation) of <ChangeLog> document of the OpenSCADA system of 0.6.4 version, which is designed to briefly and clearly to cover the new features and changes of the OpenSCADA system. The document includes information on changes made in the stabilization releases of the 0.6.3 version. To get acquainted with all the changes in the OpenSCADA system it is possible in the file ChangeLog from the distribution kit of the system, or here: <http://wiki.oscada.org/Works/ChangeLog>.

Key features of this version are:

- Implementation of planned tasks.
- Optimization, stability, sustainability and performance increasing of the system.
- Improvement and stabilization of the VCA.
- Translation of the documentation into English and the program interface into German. Multi-language database.
- System-wide expansion.

While working on the release the following changes were made to the OpenSCADA modules:

- *Archive.FSArch* -- 1.0.1 -> 1.3.1, translation of documents in English, stabilization, modifying of the API request of the values and the subsystems at whole to version 2.
- *Archive.DBArch* -- 0.7.1 -> 0.9.0, translation of documents in English, stabilization, modifying of the API request of the values and the subsystems at whole to version 2.
- *DB.DBF* -- 1.9.1 -> 2.0.1, update of the data acquisition mechanism, the expansion of opportunities when working with keys, translation of documents in English.
- *DB.MySQL* -- 1.4.1 -> 1.6.1, stabilization, storage of the multilingual fields, the new macro for the MySQL library detection, updated the data acquisition mechanism, the expansion of opportunities when working with keys, translation of documents in English.
- *DB.SQLite* -- 1.4.1 -> 1.6.1, stabilization, storage of the multilingual fields, the new macro for the SQLite library detection, update of the data acquisition mechanism, the expansion of opportunities when working with keys, translation of documents in English.
- *DB.FireBird* -- 0.8.1 -> 0.9.1, the new macro for the FireBird library detection, update of

the data acquisition mechanism, the expansion of opportunities when working with keys, translation of documents in English.

- *DAQ.DiamondBoards* -- 1.1.0 -> 1.2.0, implementation of redundancy, interface translation in German and documentation in English.
- *DAQ.System* -- 1.6.2 -> 1.7.0, implementation of redundancy, interface translation in German and documentation in English.
- *DAQ.BlockCalc* -- 1.1.1 -> 1.2.1, stabilization, implementation of redundancy, interface translation in German and documentation in English.
- *DAQ.JavaLikeCalc* -- 1.3.0 -> 1.5.0, stabilization, increasing the limit on the number of registers in the function to 65535, the implementation of objects, implementation of redundancy, interface translation in German and documentation in English.
- *DAQ.LogicLev* -- 0.9.2 -> 1.0.1, some features enhancements, implementation of redundancy, interface translation in German and documentation in English.
- *DAQ.SNMP* -- 0.3.3 -> 0.4.0, implementation of redundancy, interface translation in German and documentation in English.
- *DAQ.Siemens* -- 1.1.1 -> 1.2.1, stabilization, implementation of redundancy, interface translation in German and documentation in English.
- *DAQ.ModBus* -- 0.9.1 -> 1.0.1, increasing the functionality, a significant stabilization, selection of the protocol part into the separate module *Protocol.ModBus*, replacing its own mechanism of serial transports with the module *Transport.Serial*, the use of an individual mechanism of blocking the outbound transports, implementation of redundancy, interface translation in German and documentation in English.
- *DAQ.DCON* -- 0.1.0 -> 0.3.0, stabilization, expanding the list of supported hardware, implementation of redundancy, replacing its own mechanism of serial transports with the module *Transport.Serial*, interface translation in German and documentation in English.
- *DAQ.DAQGate* -- 0.3.3 -> 0.8.0, renamed from *DAQ.Transporter*, stabilization, increasing of the functionality, the implementation of vertical and horizontal redundancy, support for forwarding the archives of remote stations, interface translation in German and documentation in English.
- *DAQ.SoundCard* -- 0.6.0, new module of the implementation of access to input data of the sound card, implementation of redundancy, interface translation in German and documentation in English.
- *DAQ.ICP\_DAS* -- 0.5.0, new module of the implementation the support of the ICP DAS equipment.
- *Transport.Sockets* - 1.3.6 -> 1.4.0, stabilization, some features enhancements, translation of the documentation in English.
- *Transport.SSL* -- 0.8.0 -> 0.8.2, stabilization, some features enhancements, translation of the documentation in English.
- *Transport.Serial* -- 0.5.0, new module.
- *Protocol.HTTP* -- 1.3.2 -> 1.4.1, stabilization, the support of the Keep-Alive mode, enhancement of the module mechanism of the HTTP protocol, interface translation in German and documentation in English.
- *Protocol.SelfSystem* -- 0.8.0 -> 0.9.1, stabilization, the new API of the outgoing requests of the "Protocols" subsystem, interface translation in German and documentation in English.
- *Protocol.ModBus* -- 0.5.0, new module of the ModBus protocol selected from the module *DAQ.ModBus*, it supports the server or slave mode of the ModBus network.
- *Special.FLibComplex1* -- 1.0.3 -> 1.0.4, some improvements in the function of PID-regulator, interface translation in German and documentation in English.
- *Special.FLibMath* -- 0.5.1, interface translation in German and documentation in English.
- *Special.FLibSYS* -- 0.7.0 -> 0.9.0, addition of the user objects and functions to work with them, a significant expansion of library at whole, the stabilization, interface translation in German and documentation in English.
- *Special.SystemTests* -- 1.3.6 -> 1.3.7, changed by changing API, interface translation in German.
- *UI.QTStarter* -- 1.5.1 -> 1.5.3, addition of the insert the function of the own section of the menu in the opened QT window, QT translation of internal QT messages, interface translation in German

and documentation in English.

- *UI.QTCfg* -- 1.7.0 -> 1.8.1, stabilization, support of the dynamic table structures, unification of the OpenSCADA management interface, search engine for the element in the branch, the new version of API of the OpenSCADA protocols, selection of several elements of the tree for their group removal, interface translation in German and documentation in English.
- *UI.WebCfg* -- 1.5.2 -> 1.5.3, stabilization, support of the dynamic table structures, using of the general authentication mechanism, interface translation in German and documentation in English.
- *UI.WebCfgD* -- 0.5.0 -> 0.6.2, stabilization, support of the dynamic table structures, unification of the OpenSCADA management interface, the new version of API of the OpenSCADA protocols, using of the general authentication mechanism, interface translation in German and documentation in English.
- *UI.VCAEngine* -- 0.8.0 -> 0.9.0, stabilization, support of styles, expanding the visual primitive "Protocol", version of the database structure increases to 2, the support of the multi-language data, optimization of data storage, presentation of "Frequency spectrum" in the primitive "Diagram", support of the context menu in the execution mode, interface translation in German and documentation in English.
- *UI.Vision* -- 0.8.0 -> 0.9.0, stabilization, support of styles, expanding the visual primitive "Protocol", presentation of "Frequency spectrum" in the primitive "Diagram", support of the context menu in the execution mode, interface translation in German and documentation in English.
- *UI.WebVision* -- 0.6.0 -> 0.7.0, stabilization, expanding the visual primitive "Protocol", presentation of "Frequency spectrum" in the primitive "Diagram", New version of API of OpenSCADA protocols, using of the general authentication mechanism, optimizing of the speed and memory usage, support of the context menu, interface translation in German and documentation in English.

## 1 Implementation of planned tasks

In accordance with the plan of the release there were performed the following tasks:

- *Extension of the API of user programming with the functions:*
  - For the session management it has been added the library of user functions to the module *UI.VCAEngine*. The library includes the following functions: *vcaWdgList()*, *vcaNodePresent()*, *vcaAttrList()*, *vcaAttrGet()*, *vcaAttrSet()* и *vcaSesUser()*.
  - Management of the controllers of the subsystem "Data acquisition" and of the other OpenSCADA objects was implemented by adding the type of data Object and creation of the user request *FLibSYS.xmlCntrReq()* to the management interface.
- The module *DAQ.ModBus* is supplemented with the protocol module *ModBus*, that allowed to work in server (slave) mode of the ModBus network.
- The redundancy mechanisms are implemented. In the subsystem "Data acquisition" it is added the mechanism of horizontal redundancy and all the modules of this subsystem are adapted to support the horizontal redundancy. The functions of the vertical redundancy are added in the module *DAQ.DAQGate*, including the restoration of archives and archives locking of the remote stations. The mechanism of the hot compensation of the holes in the archives from the archives of the back-up sources is implemented.
- Into the *DAQ.JavaLikeCalc* it is added the full support of the objects and arrays. The mechanisms of access to the elements ".prop" and "[prop]" are implemented. The functions of objects and of their execution are implemented. The language is unified at whole to work with different data types with the help of the single command of the virtual machine.
- The styles support for the VCA interfaces is added.
- *Implementation of the primitives "Link" and "Function" of the VCA.* -- This work was not done and is moved on to the next release.

## 2 Optimization, stability, sustainability and system performance increasing.

While working on this version, as well as its practical adaptation, it was discovered and corrected in a total of more than 200 errors.

Except the correction of the errors the work under the optimization of the various components of OpenSCADA. Particularly the noticeable optimization was made in the visualization control area (VCA) and in its visualizers. The main vector in the optimization was the optimization of memory usage.

Here are the most significant error, correction of which has the significant effect on improving of the stability:

- Wide-system:
  - The configuration files for the remote connection by default are fixed.
  - Compiling of the main starting program is separated from the library compiling into the directory `src_call/` in order to ensure the error-free compiling of RPM-packages in the multithreaded mode.
  - Starting script of the user mode "openscada\_start" is corrected in order to copy the configuration file `oscada_start.xml` to the `./openscada/oscada.xml` and to local using of it.
  - The adaption for the QT 4.5 using is done. The variable of the environment `LC_NUMERIC = C` is globally added in order to finally fix the problems of the separator in the real numbers in the different locales.
  - Mode of cache using by the system database is changed to use only in the case of absence of the main database.
  - The resource for access to the XML-tree of configuration file is added. To access the XML-tree not the mode of the generation of the errors exceptions is used, this corrected the leakage of resources (Stack) in some contexts.
  - The resource conflict arises when trying to delete one and the same child element of the OpenSCADA component from the different threads is fixed.
  - The resource for access to the container structure of the elements *TElem* is added. Conflicts were detected using the general *TElem* from different *TConfig* of the different threads in *UI.WebVision*.
- The support for updating of the status of connection of the remote stations in the module *UI.QTCfg* is added.
- Into the module *Protocol.HTTP*:
  - The support for the "~ KeepAlive" mode is added, which is needed for normal operation of the Opera WEB-browser.
  - The omission and not full processing of the the character '\r' at the end of the HTTP protocol lines is fixed. Fixed the work in Konqueror 4.
- VCA:
  - A lot of fixes and improvements in conjunction of the visualization and archiving are made.
  - The work of the trend primitive "Diagram" of the VCA is fixed when working on long time intervals. The work with individually selected archiver is fixed.
  - Algorithm of the identification of the fact of modifying the attributes and widgets of the module *UI.VCAEngine* while the execution of projects is submitted to the function *modifChk()* and retreated to take into account the possibility of overflow and reset of the cycles of execution counter of the visualization interface.
  - A lot of bug fixes are made in the module *DAQ.ModBus* and the reconstruction for the support of various interfaces, including ModBus/ASCII is implemented. The record to the controller is made only in the case of a real change in values and record of the not EVAL values.
  - For the outgoing transport of the module *Transport.Sockets* the reconnection and restoration of the session is fixed, it is also added the controlled timeout of the connection.
  - The export of large areas of the archive is fixed, as well as the function of management of the

requests restrictions for buffers with the fixed size.

Optimization and productivity improvement:

- Working mode with the DB tables is changed to the one which do not close the tables immediately after the completion of work with them. Closing is made in a minute after the last use of the table. This approach significantly increased the speed of access to the database.
- The cache of the resources is implemented in the module *UI.WebVision*. It increases the productivity in the construction of images of *ElFigure* primitives based on the images.
- Database Structure of the VCA is changed to improve the productivity and to avoid the lost of records in the database. DB API version is set to 2.
- The individual mechanism to block the outgoing transports is added. It is used for the exclusive access to transport without blocking of the entire protocol in the *DAQ.ModBus* and *Protocol.SelfSystem*.
- Optimizing of the memory usage:
  - The data type *TVariant* for the clear storage of date of different types. The object is used to obtain the direct access to the attributes of widgets "id" and "path" and to avoid the excessive storage of these attributes in the module *VCA UI.VCAEngine*.
  - The global resource object and its use are standardized. For the resource to the read-write the POSIX functions `pthread_rwlock_*` are used.
  - The container of the children groups in the *TCntrNode* is made the dynamic one.
  - The mechanism for storing the attributes in the VCA widget is simplified.
  - The description object of the VCA attributes *TFld* is used only once for the inherited attributes. The memory using is reduced by the 30%.

### 3 Improvement and stabilization of the VCA.

Significant changes have been implemented within the visual control area (VCA), namely in the VCA engine modules *UI.VCAEngine*, in the visualizers *UI.Vision* and *~UI.WebVision*. The changes were aimed to the stabilizing, optimizing of memory usage and improvement of the user functions of the VCA.

Improvements of the VCA:

- The transfer of the translation of the main QT dialogues to the standard I18N.
- The functions of the user API for enhanced access to the elements of the session of the project from the scripts of elements'(widgets') of visualization are added.
- The support of alpha channel, ie translucency, in the indication of colors of all elements of the VCA primitive *ElFigure* is added.
- Attribute "root" of the VCA is moved to the main widgets' attributes' container . Attribute "parent" is added to provide information about the parent widget. The attributes "tipTool" and "tipStatus" are added to provide an opportunity for providing the VCA widgets with the contextual help in the module of the visualization engine *UI.VCAEngine*.
- In the *UI.WebVision* module the conception of forming the image of the primitive *ElFigure* is changed to the formation of images with filling in the individual images and their subsequent integration into the resulting image.
- The mechanism of the VCA widgets' changes clear is implemented. It is often need to reset the local changes and to return to the inherited properties.
- The support for the context menu for widgets in the execution mode of the VCA projects is implemented.
- The hiding of the the menu bar in the mode of execution of the project by the *UI.Vision* module for unprivileged users is implemented. Menu Bar in the execution mode is now positioned as the tool of the OpenSCADA's administrator, which contain system-wide functions. Print and export functions are duplicated on the status bar.
- The clock on the status panel is added in the execution mode of the project by the *UI.Vision* module for full-screen performance.
- The basic support for the multi-display systems is implemented in the *UI.Vision* module.
- Changes in the structure of the VCA database are added to improve performance and to avoid

the lost of records in the database. The version of the DB structure of the VCA is increased to 2.

- Into the VCA primitive "Diagram" it is added the representation of the frequency spectrum of the archive data values.
- The user function `vcaSesUser()` into the user API of the VCA is added.
- VCA: Into the primitive "Protocol" the property of the sorting order and description of elements' design property are added. Check for disable of the trends, including the condition of the not correct color are added to the primitive Diagram.
- The support of styles is added.

## 4 Translation of documents in English and the program's interface in German. Multi-language database.

Much has been done to translate the program interface into German by Irina Popkova. The OpenSCADA kernel interfaces have been translated and the module ones: `UI.WebVision`, `UI.QTCfg`, `DAQ.DCON`, `DAQ.LogicLev`, `DAQ.DAQGate`, `DAQ.DiamondBoards`, `DAQ.ModBus`, `DAQ.SoundCard`, `DAQ.Siemens`, `DAQ.SNMP`, `DAQ.System`, `DAQ.JavaLikeCalc`, `Protocol.HTTP`, `Protocol.SelfSystem`, `Special.FLibComplex1`, `Special.FLibMath`, `Special.FLibSYS`, `Transport.Serial`, `Special.SystemTests`, `UI.VCAEngine`, `UI.QTStarter`, `UI.Vision`, `UI.WebCfg`. The patch of the Klaus Fetscher for the correction of the interface translation into German is included.

The English translation of documentation for system-wide documentation and modules is done by Maxim Lysenko: `DB.SQLite`, `DB.MySQL`, `DB.FireBird`, `DB.DBF`, `Archive.DBArch`, `Archive.FSArch`, `DAQ.System`, `DAQ.LogicLev`, `DAQ.DAQGate`, `DAQ.JavaLikeCalc`, `DAQ.ModBus`, `DAQ.SoundCard`, `DAQ.Siemens`, `DAQ.SNMP`, `DAQ.DiamondBoards`, `DAQ.BlockCalc`, `Transport.Sockets`, `UI.WebCfgD`, `UI.QTStarter`, `UI.QTCfg`, `Transport.Serial`, `Transport.SSL`, `Protocol.SelfSystem`, `Protocol.HTTP`.

To the OpenSCADA system kernel the support for multilingual text variables is added. Owing to this feature the demonstration databases of individual languages merged into one multilingual database, which includes English (basic), Russian and Ukrainian languages. Multi-language is supported by the modules and databases `DB.SQLite` `DB.MySQL`. Demo configuration file is also translated into English (basic), Russian and Ukrainian languages.

## 5 System-wide expansions.

For this release it was made a lot of system-wide changes aimed to enhancing of the functionality.

Extension of the user programming API:

- Into the user's library of the system API `Special.FLibSYS` are added:
  - Functions: `sysCall()`, `int2str()`, `str2real()`, `str2int()`. `sysCall()` function opens wide possibilities to the OpenSCADA user by calling any system software, utilities and scripts, as well as by way of access to the huge volume of system data.
  - The functions `dbReqSQL()` and `messGet()` are added, which are used as return objects.
  - The `XMLNodeObj` object is added, the function `xmlNode ()` for the creation of the object and function `xmlCntrReq ()` for sending the request to the management interface of OpenSCADA are added.
  - The `VArchObj` object and `vArh()`, `vArhBuf()` functions are added for it's creation. It is used for full access to the archives. The user function for the planning in th standard `Cron tmCron ()` is added.
  - The FFT function is added into the archive object `VArchObj` for processing of the archive with the Fast Fourier Transform.
  - The encode/decode of the binary data to the text functions are added: `strEnc2Bin()` and `strDec4Bin()`.
- Into the object of the instance (`TValFunc`) of the system functions of OpenSCADA execution (`TFunction`) it is added the attribute of the user allowing the sharing of rights to execute the user's

## API.

- The data type TVariant for the clear different types data storage is implemented. The object is used to obtain the direct access to the attributes of widgets "id" and "path" and to avoid the excessive storage of these attributes in the VCA UI.VCAEngine module.
- The limit of the number of variables (registers) used by the user in the function, is increased from 255 to 65535 in DAQ.JavaLikeCalc.
- The user objects are implemented. The base object TVarObj and the array object TAreaObj are supported. The support for output parameters in the functions of objects is added.
- The full support for objects and arrays in DAQ.JavaLikeCalc is added. The mechanisms to access the elements ".prop" and "[prop]" are implemented. The functions of objects and their executions are implemented. Language at whole is unified to work with different types of data using the single command of the virtual machine. The support for installing the run-time of the controller in the standard Cron is added.
- The direct functions to access the archives are removed. Instead them it should be used the built-in archive object functions.
- The primary support of the object model of OpenSCADA (DOM) to access from user programming languages.
- In the outgoing transport it is added the DOM-function messIO() to provide the user with the possibility to send the direct requests through the transports.

## Changes in the demo DB:

- Full-fledged alarm objects.
- The template of the frame of summary charts containing up to 80 charts simultaneously.
- The context sensitive help to the elements of the mnemonic schemes is added.
- The navigation bar is updated and buttons of the quittance of the alarm messages are added.
- The three-dimensional elements of the pipeline are added.
- The context menu for ball valves is added.
- The font size of the values of elements of analog signals is increased.
- The control panels of diagrams and documents are updated.
- Demo database is updated to implement the database structure of VCA of version 2.
- The translation of the VCA's database in English and the translation into Ukrainian is added.
- The algorithm for creating the archival documents is updated.
- The document functions for the summation and averaging of the archive section are corrected to use the latest value.
- The username of the session for the placing of it in the actions protocol is added.
- Database and configuration file are reconfigured to set the English as the default language.
- The functionality of the main page of alarm objects is expanded.
- The fonts' size of the contour frame are increased, as well as the buttons of the step changing of the value are added.

## Miscellaneous:

- The ability to self-boot from the selected database or configuration file is added.
- Full support of the lists of permitted and prohibited modules, replacing the list of automatic modules are implemented. The control table of the loaded shared libraries (modules) is implemented. The manual connection/disconnection of the modules is implemented.
- Mode of the working with the tables of database is changed to the mode which don't close the table immediately after the finishing of work with them. Closure is made in a minute after the last using of the table. This approach significantly increased the speed of access to the database.
- The microsecond part of the time is added into the system messages of OpenSCADA.
- The special flag of the selection of key for the using of it in the request functions to the database dataSeek() and dataDel(), which allowed to choose the keys for the request not only the text type. Flag is used in all the interfaces.
- The mechanism of the status for the controller objects of the subsystem "Data acquisition" is added.
- The document of the OpenSCADA guide is formatted for printing and added to the repository.

- The version of the API of subsystem modules: "Archives" and "Data Acquisition" is increased to 3 and the "Protocol" to 2.
- The mechanism of the status for the major subsystems and their modules is implemented.
- The x86-dependent mechanism for measuring of the short time intervals is everywhere replaced by the standard. Function to calculate the CPU clock is turned off for non-x86 and x86\_64 architectures.
- The tab of the formation of the SQL-request to the database configuration page is added.
- The support for fully dynamic structures of the table element in the configurator UI.QTCfg, UI.WebCfg, UI.WebCfgD is added.
- The mechanism of signals based on the API messages is added.
- DAQ.DAQGate:
  - The DAQ.Transporter module is renamed to the DAQ.DAQGate and expanded to support both: the horizontal and the vertical redundancy, as well as for working with archives.
  - Constant requests to the remote controller are grouped into the single one.
  - The updating of the list of parameters and the connection of the archives with them are added.
  - The conception of synchronization for the DAQ.DAQGate and redundancy of data acquisition is changed.
  - The order of the remote stations is made the meaningful one.
  - Some clean of the code is done.
  - The access to the attribute of the error "err" is fixed.
  - The mechanism of recovery from the archives is revised a little to correct the loss of values in DAQ.DAQGate and in the redundancy system.
- DAQ.DCON: The patch of the Almaz Karimov for the set of fixes and improvements is included.
- The function TSYS::cron() for the planning of the tasks execution is added.
- DB: The record of the value in the default language in case of the creation of new record and using the not default language is added. Ability to use the translations from the multilingual database in the monolingual configuration.
- Changes in the building system and in the specification file for the formation of RPM-package:
  - The template module "=Tmpl=" for the subsystems: "Transport" and "Protocols" is created.
  - The spec-file of the formation of RPM-packages is updated. For the requirements of the Fedora distribution it is adapted the separate spec-file by the Alexey Popkov and with the participation of Peter Lemenkov. The autoconf macros for the more efficient resource discovery of the databases MySQL, FireBird and SQLite is added. The building of the modules is fixed and adapted to libtool 2.2.
  - The files from the repository, which can be created by the command "autoreconf-iv" are deleted.
  - Library version of the OpenSCADA is changed to 0:11:0.

## 6 Plans for the further development

To the release of the industrial version 0.7.0 it is planned the solution of the following tasks:

- Implementation of the VCA primitives "Connection" and "Function" in order to provide the graphical programming tool.
- Implementation of the database ODBC, PostgreSQL and transport module of the database.
- Implementation of the new mechanism/module of the OpenSCADA tests.
- Large-scale testing and stabilization.