OpenSCADA 0.7.2

(Release notes)

Table of contents

Introduction

This release is a planned one of a working version of OpenSCADA, which are usually made at intervals of 3-4 months. The main purpose of this release is the stabilization and testing of key functions before releasing the next long term support (LTS) version of OpenSCADA 0.8.0. In addition to the main purposes within the limits of this release a great work for stabilization, cleaning the source code, full empowerment and testing on the embedded and mobile ARM devices is made.

This document is the processing (compilation) of <ChangeLog> document of the OpenSCADA system of 0.7.2 version, which is designed for briefly and clearly covering the new features and changes of the OpenSCADA system. 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://oscada.org/websvn/filedetails.php?repname=OpenSCADA&path=%2Ftrunk%2FOpenSCADA%2FChangeLog.

The key features of this version are:

- Implementation of planned tasks.
- Optimization, increasing of stability, sustainability and system performance.
- Improvement and stabilization of the graphics subsystem.
- Expansion of user programming API.
- System-wide expansions.
- Publication of OpenSCADA solutions.

New and updated modules:

- *Archive.DBArch (0.9.5)* — The limits for requesting data on the request time are added as well as some more extensions.
- *Archive.FSArch (1.5.0)* — The files' indexes of the values' archives read performance is increased. The adaptation of storing and reading the floating to the ARM FPA is made. The limits for requesting data on the request time are added. Some extensions. A significant stabilization.
- *DB.FireBird (0.9.7)* — Stabilization.
- *DB.MySQL (1.7.1)* — Stabilization.
- *DB.PostgreSQL (0.9.2)* — Stabilization.
- *DB.SQLite (1.6.4)* — Stabilization.
- *DAQ.BFN (0.5.1)* — Stabilization.
- *DAQ.BlockCalc (1.5.0)* — The CRON scheduling of calls is added. The start and stop blocks' call is added. Stabilization.
- *DAQ.DAQGate (0.9.5)* — The CRON scheduling of calls is added. Stabilization.
- *DAQ.DCON (0.5.1)* — Stabilization.
- *DAQ.DiamondBoards (1.2.5)* — The ability to change the parameter's type is added. Stabilization.
- *DAQ.ICP_DAS (0.8.0)* — The "ICP DAS" API library libi8k.a for the ARM architecture is added, the ability to build the module for ARM is added. Stabilization.
- *DAQ.JavaLikeCalc (1.9.5)* — The direct, dynamic call of library functions is added. The direct association of string constants is added. The conditional call of the expression's arguments (the second argument) with the logical operators ||(OR) and &&(AND) is added. The significant stabilization. Some extensions.
- *DAQ.LogicLev (1.2.0)* — Stabilization. The ability to change the parameter's type is added. The CRON scheduling of calls is added. Some extensions.
- *DAQ.ModBus (1.2.0)* — The support for "Logical" parameter's type is added for the working with template of the parameter, and also the functions of the user's API to send arbitrary, non-standard ModBus requests from the templates are added. The function is added and used the formation of alleged violations of the controller. The support of the function for group record (0x0F, 0x10) is added. The significant stabilization.
- *DAQ.OPC_UA (0.6.2)* — Stabilization. The adaptation of storing and reading the floating to the ARM FPA.
- *DAQ.SNMP (0.6.1)* — Stabilization. Some improvements.
- *DAQ.Siemens (1.3.0)* — Significant stabilization. Library LibnoDave is updated to the 0.8.4.6

version. The CRON scheduling of calls is added. The reconnection function for the "Industrial Ethernet" connections is added. Some extensions.

- *DAQ.System (1.7.5)* — Stabilization. The CRON scheduling of calls is added.
- *Transport.SSL (1.0.0)* — Stabilization. Some improvements.
- *Transport.Serial (0.7.3)* — Stabilization.
- *Transport.Sockets (1.5.0)* — Significant stabilization.
- *Protocol.HTTP (1.6.0)* — The support of user's templates for internal contents of the module is added. The generation of the user's authentication messages is added. The support of all major versions of the completion the string during HTTP-request parsing is added.
- *Protocol.ModBus (0.6.3)* — Stabilization. Some improvements.
- *Protocol.OPC_UA (0.6.2)* — Stabilization.
- *Protocol.SelfSystem (0.9.5)* — Stabilization.
- *Protocol.UserProtocol (0.6.2)* — Stabilization.
- *Special.FLibComplex1 (1.1.0)* — The direct, dynamic call of library functions is added.
- *Special.FLibMath (0.6.0)* — The direct, dynamic call of library functions is added.
- *Special.FLibSYS (1.0.0)* — The direct, dynamic call of library functions is added.
- *UI.QTCfg (2.1.0)* — Stabilization. Significant extensions and improvements. The implementation of a number of features to enhance comfort.
- *UI.QTStarter (1.6.2)* — Stabilization. The font of message in splash is fixed at 10 pixels to ensure the uniform representation.
- *UI.VCAEngine (1.2.0)* — Stabilization. Significant extensions and improvements.
- *UI.Vision (1.2.0)* — Significant stabilization. Significant extensions and improvements.
- *UI.WebCfg (1.5.6)* — Stabilization.
- *UI.WebCfgD (0.8.0)* — Stabilization. Significant extensions and improvements.
- *UI.WebUser (0.6.2)* — Stabilization.
- *UI.WebVision (1.0.0)* — Significant stabilization. Significant extensions and improvements.

1 Implementation of planned tasks

In accordance with the plane of the release there were made the following tasks:

- *The implementation of the undo/redo editing changes mechanism in the Vision.* — As part of the widget's visual editing window the multi-level changes' undo/redo mechanism is implemented for all basic operations: a visual changes in the geometry, changing the value of the widget's attribute, add/remove the widget, copy the widget and editing the widgets, based on the primitive "ElFigure".
- *Adaptation of OpenSCADA to work on the ARM hardware platform.* — In addition to the primary adaptation, in the last release for the N800 tablet (http://wiki.oscada.org/Works/Tests/ARM) the build and adaption of OpenSCADA for the complex (very old) program environment of the LP-5451 controller was made, and also the builds for the SMH2Gi controller and for the Nokia smartphones: N900, N950, N9 were made. As part of the build and adaptation to different mobile devices the following tasks were implemented:
  - *DAQ.ICP_DAS*: The "ICP DAS" API library libi8k.a for the ARM architecture is added, the ability to build the module for ARM is added.
  - *SYS, DAQ.OPC_UA*: The system functions floatLE(), floatLErev(), doubleLE(), doubleLErev() are added to convert the storage format of a real number on different architectures.
  - *SYS*: The adaptation is made to build with the GLibC version less than 2.5 (2.3.2).
  - *Archive.FSArch*:
    - The LE format real number storage in the archive on the ARM architecture is fixed.
    - The algorithm of the quickly count the number of bits uses the function of not aligned reading TSYS::getUnalign32().
    - The encoding field size in the plane archive of messages expanded from 9 to 99 characters.

2 Optimization, stability, sustainability and system performance increasing

During the work on this version, as well as during its practical adaptation, it was discovered and corrected in a total of about 170 error. Also the work on cleaning code from the compiler's warnings with the "-Wall" flag was continued.

Lets list the most significant errors correction of which significantly affected the increasing of stability:

- *SYS*:
    - The unscheduled calls on the schedule, related to desync of the functions values time() and clock_gettime(CLOCK_REALTIME,&sp_tm), are fixed.
    - The checking and waiting for threads to access due to the overlapping with the previous, eponymous one's stop.
    - The waiting for initialization of the disconnectable tasks is disabled in order to prevent freezing in anticipation of rapidly closing tasks.
    - The work of not aligned reading functions getUnalign*() is fixed on the ARM. The Problem is detected on PXA270.
- *Archive.FSArch*:
    - The reading data from the archive's buffer in accordance with the buffer's size in the algorithm of fast calculation of the bits' number is fixed.
    - The placement of the offset in the cache from the algorithm of fast calculation of the bits' number is fixed.
    - The incorrect handling of the end of not aligned blocks of the fast algorithm for calculation of the bits' number is fixed.
    - The exclusive resource to the calcVlOff() function's call is added in order to prevent the incorrect operation with cache.
- *Transport, DAQ.AMRDevs, DAQ.DCON, DAQ.ICP_DAS, DAQ.ModBus, DAQ.OPC_UA, Protocol.HTTP*: The usage of the TTransportOut::messIO() function at the request of the tail and getting the zero response is fixed.
- *DAQ.ModBus, Transport.Serial, UI.WebCfg, UI.WebCfgD, UI.WebUser, UI.WebVision*: The incorrect using of the string terminator '\n' for a variety of software platforms UNIX, MAC, DOS/Windows is fixed by replacing it with "\x0A".
- *Transport.**: The using of unsigned type in the return of the read() and write() functions is fixed. The resources for the counters of the i/o requests are added.
- *Transport.Sockets*: The missing of the initialization of the size of the return value of the getsockopt() function is fixed.
- *DAQ.JavaLikeCalc*: The crash of the replace() function of the "RegExp" object is fixed.
- *DAQ.Siemens*: The receiving and recording the values of real types is fixed. The reconnect in the case of error is fixed. The using of the size property of the integer, real and string is fixed. The cleaning of the parameter's links at its disabling is added.
- *UI.Vision, UI.WebVision*: The access abroad the vector's size during the rendering of the primitives "FormEls" graphics is fixed.
- *UI.VCAEngine*: The possibility of changes' loss in the visualization server is precluded when displayed in the visualizers by preliminary saving the value of clock counter in the "openList" service request.
- *UI.WebVision*:
    - The opening and replacing the root page is fixed.
    - The handling of nested pages' scale is fixed, taking into account the scale of the root one.

Optimization and performance improvement:

- *SYS*: The mechanism of the "ResString" object resource is replaced from the RW-locks with mutex in order to improve performance and reduce memory consumption.
- *Archive.FSArch*: The processing of the values' archive index table is significantly accelerated through the use of fast algorithm for computing the number of bits in a 32-bit integer.
- *UI.WebCfgD*: The group request of the page's contents is used to greatly improve the performance on slow and high latency channels.
- *UI.Vision*: The rendering performance of the fills of the "ElFigure" primitive is improved.

- *UI.VCAEngine*:
    - The measures to increase the speed of project's session initiation are implemented: the initialization of the basic attributes because of their subsequent inheritance is skipped.
    - The translation of certain messages of the runtime session is removed.
    - The enabling of only the necessary pages at startup of the session is implemented. Other pages of the running session are enabled when accessed. This has greatly increase the speed of session's startup, as well as reduce the dependence of the starting speed on the complexity of the visualization project.
    - Implementation of the archive mode of the "Document" primitive is changed to direct work with the database, which let to expand the depth of the archive up to the 1000000 documents, and without prejudice to the RAM.

3 Improvement and stabilization of the graphics subsystem

The notable work was made within the graphics system, namely in the VCA engine modules UI.VCAEngine, visualizers UI.Vision and UI.WebVision, as well as in the configurators. The changes were provided for stabilization, the optimization of memory consumption and improvement of user's properties.

Improvements of the graphics subsystem:

- *UI.VCAEngine, UI.Vision, UI.WebVision*:
    - The handling abroad the array (vector) in the construction of trends of the "Diagram" primitive is fixed.
    - The width property of trend of the "Diagram" primitive is added and implemented.
    - The periodic inspection of the widgets' tree of the page is added in order to detect the fact of widgets' removing by requesting the complete list of widgets.
    - The display of trends, graphics group of the "Diagram" primitive in the scale of values is implemented in the case of differences between the scales of all graphs not more than 20%.
    - The ability to select the confirmation mode for the items of the line and text editing of the primitive "FormEl".
    - The possibility of loosing of some changes in the model, when displayed in the visualizers is fixed.
    - The property of the values' count of the one pixel in the trends of the "Diagram" primitive is added and implemented in order to control granularity of export to CSV etc.
    - The logarithmic scale of values for the trends of "Diagram" is implemented.
    - The storage of aspect ratio property of the main page during the maximization of the main window of the project's session execution is added and implemented.
- *UI.VCAEngine*:
    - The enabling of project's session pages when it is really necessary is implemented in order to speed up the startup and optimize memory usage.
    - The "Document" primitive is remade for keeping the archives fully in the database, as well as the function of user API to access the elements of the archive is added.
    - The periodic (30 minutes) check and the closing of the unused (orphaned) projects' sessions are added.
    - The previous one (controversial) rollback mechanism of nested widgets deleting is removed and it is replaced by the full-fledged rollback mechanism in the *UI.Vision* visualizer.
    - Optimization of memory usage by the session's elements up to 20% is implemented.
    - The support of the first, out of date, version of the database structure of the visual area is removed.
    - The restriction on the time of the document's generation in the "Document" primitive in 5 seconds is added.
    - The support of the attribute's type "Object" and "Text" is added.
    - The forced initialization of new user's attributes in EVAL is disabled.
    - The "Document" primitive is switched to XML parsing in order to fully preserve the XHTML markup.

    - The display of the correct status of the links in the form of "(+)" at the end of the line is added.
- *UI.Vision*:
    - The implementation of the full-fledged undo and redo the changes in visual editing widgets is added.
    - Improvements in the sensitivity and updating the changes in the project's session: the rendering performance of the "ElFigure" primitives is optimized, handling of the changes in the "geomZ" level of the widget is added.
    - The export to CSV format from the "Diagram" and "Document" primitives is added.
    - The limit on the size of the input dialogs is added and an adaptive determination of their sizes by their contents is added.
    - The rules' execution mechanism for syntax highlighting is completely rewritten.
    - The new session creation after the restoration of remote connection and absence of the previous session is added.
    - Window background for the editable widget is set to QT::Dense7Pattern template to eliminate the possible of overlapping by color.
    - The event handling of the widget's selecting when press in the scroll area, but outside of the widget is added.
    - "Elementary figure" primitive:
        - The reorganization of the operations of the Properties dialog and context menu with dynamic and static properties is completed.
        - In the process of addition the figure its rendering when moving the mouse cursor is implemented.
- *UI.WebVision*:
    - The handling of nested frames scale is fixed, as well as the mechanism of formation-enabling of scroll is revised.
    - The function of the images conversion on the server's side is added. Conversion function is used for resizing and discoloration the images of inactive buttons.
    - The opening of small windows of the nested pages as the DIV-blocks is implemented, it solves the problem of lock and the long opening of the external windows in many browsers.
    - The main page of the interface is centered in the browser's window.
- *UI.QTCfg*:
    - The restriction on the height of the rows of tables at half the height of the table is added.
    - The limit on the size of the input dialogue is added, and the adaptive determination of its size on its contents is added.
    - For the elements available to adjust their height by the user, for the work elsewhere, the fixed style "StyledPanel" is set.
    - The function of full text editing of table's cells is added.
    - Playback engine of the syntax highlight rules is completely rewritten to simplify it and to avoid hidden errors.
- *UI.WebCfgD*:
    - The brightness of decolorized-passive buttons is reduced.
    - All dialogs windows are implemented as DIV-blocks in order to avoid problems and delays in opening external windows in different browsers.
    - The group request the page's contents is used for the significant improvement of the performance on slow and high latency channels.
    - The information in the status bar about the current user and the possibility of its change are added.
    - The adaptation to work in full screen width is made.

4 Extension of the user programming API

It was continued the development of object API of the user programming, which provides the integration of user's functions into the object's tree of OpenSCADA. In addition, a number of changes was made to the existing library of the user's API functions.

In particular the following changes were made:

- *SYS*:
    - The functions to work with files are added: SYS.fileRead() and SYS.fileWrite().
    - The getElementBy() function is added to the XMLNodeObj object's API to search for embedded nodes by attribute's value.
    - The function of encoding the text between different character encodings is added.
    - The NodeObj.nodePath() function is added.
- *TConfig, TBD, TTransportIn, TTransportOut, TUser, TGroup, TPrmTmplLib, TPrmTempl, TParamContr, TController, TVArchive, TVArchivator, TMArchivator*: The following functions of the user API are added: cfg() and cfgSet() to access the configuration of the objects (stored in the database).
- *DAQ*:
    - The TController::alarmSet() function is added to generate the typical alarms in the controller's object of the "Data acquisition" subsystem's modules.
    - The SYS.DAQ["Modul"]["Controller"]["Parameter"]["Attribute"].arch() function is added to directly access the object's archive associated with the attribute of the parameter.
    - The following functions of the user API are added: enable() and start() for the direct monitoring of the controller's object.
- *DAQ.JavaLikeCalc, Special.FLibComplex1, Special.FLibMath, Special.FLibSYS*: The function for the dynamic call of the library functions is added: SYS.DAQ.JavaLikeCalc["lib_{Lib}"].funcId(prms, ...).
- *Archive*: The following functions of the user API are added: status(), end() and begin() for access to the state and properties of an object of the messages' archiver.
- *DAQ.JavaLikeCalc*:
    - The isEVal() function for the "null" data type is added, which always returns "true".
    - The call() function of the user API is added to the objects of the library functions.
- *DAQ.ModBus*: The messIO() function of the user API is added in the "ModBus" controller's object in order to provide an opportunity to send custom ModBus-queries directly from the template of the parameter.

5 System-wide expansions

To the system-wide API of the OpenSCADA system there have been made the significant changes and extensions with the purpose of the general stabilization and expansion:

- *SYS*:
    - The storage object of the string with the "ResString" resource is significantly expanded for the clear conversion from/to std::string type.
    - The ability to save to the configuration file is added:
        - The database prefix "<cfg>" is added for presentation of the configuration file as a source of load/store configuration.
        - The TDBS::dataSeek() and TDBS::dataDel() functions are updated for the strict processing the configuration file.
        - The chkSelDB() function is adapted to support the record to the configuration file.
        - The storage of large and multi-line field values in the text box of the individual tag of the configuration file is added.
    - The "long long" type, usually a 64-bit integer is replaced everywhere by the "int64_t" on all architectures.
    - The objects of the OpenSCADA treads are rigidly linked with their thread. The load and periodic threads' calls statistics is added.

    - The possibility of checking the loaded objects for their absence in the database and deleting is added. Check function is activated only in the case of direct loading from the database.
    - The perSYSCall() system periodic function for the call of subsystems' and their modules' objects is added. It is used for service purposes of unloaded and rare task of the modules and subsystems.
    - The test function of the expression by the template is moved to the TRegExp object.
    - *SYS.XMLNode*:
        - The code of "Expat" XML-parser using is completely removed.
        - The ability to load text parts of the tag to the separate specific tags in full mode is added. It is designed for full storage the configuration of the tag.
        - The ability to load and store the comments blocks into the separate specific tags in full mode is added.
- *DOC*:
    - Check and repair the English texts in the OpenSCADA core is made. The translations for the OpenSCADA core to German, Russian and Ukrainian languages are updated.
    - Check and repair the English texts in the OpenSCADA modules is made. The translations for the OpenSCADA modules to German, Russian and Ukrainian languages are updated.
    - The all key documents in accessible languages are updated: openscada.pdf, build.pdf, properties.pdf, "OpenSCADA API", "About OpenSCADA", "TechApp Library", "Quick Start", "WLib Main", "WLib Mnemo Elements".
- *Libraries databases*:
    - The library of the main visual elements is appended by the frames of the implementation of editing and execution of "Prescriptions" - "User's programs".
    - The library of the electrical elements is added to the kit of libraries package.
    - The frame of the main page is appended by the mechanism of "playing" the models of technological processes (TP).
    - The industrial boiler unit(RUS) model is translated to English and Ukrainian languages.
- *DB*: The time limit of 5 seconds and the field of specifying the starting offset are added for the request of the large tables' contents.
- *DAQ*: The ability to select the type of the "object" template's attribute is added. It is made primarily to store internal data in the object and can not be used as the type of parameter's attribute of data source controller's object.
- *TArchives*: The ability to use regular expressions when searching by category using "/match/" is added.
- *Transport*: Types of formats for outgoing requests are expanded with versions of the text with a different end of line: LF, CR, CR/LF.
- *DAQ, DAQ.DiamondBoards, DAQ.ModBus, DAQ.LogicLev*: The ability to change the type of the parameter for the disabled parameters of multi-type modules of "Data acquisition" subsystem is added.
- *DAQ.LogicLev, DAQ.Siemens, DAQ.ModBus*: In the case of the scheduled execution the time after the last calculation in the negative value is written to the "f_frq" attribute. The initialization and stop calls are added during enabling/disabling of the parameter.
- *TArchives, Archive.DBArch, Archive.FSArch*: The time border when requesting messages from the archive is added.
- *SYS.TConfig, Archives, DAQ, Security, Transports, Archive.*, DAQ.*, Protocol.UserProtocol, Transport.*, UI.VCAEngine, UI.WebUser*: Keeping the string configuration fields is moved to the "ResString" object of the resource string.

6 Publication of OpenSCADA solutions

During the work on this version the solutions, based on the OpenSCADA, have been presented at various conferences and exhibitions:

- FOSS Sea 2011 — the report about the solutions, based on the OpenSCADA was read at the conference and in the hall there was made the mini-stand with the live solutions, based on OpenSCADA.
- International Innovation Forum 2011 — at the stand of "DIYA" Ltd and Dneprodzerzhinsk State University (DSTU) the equipment and materials on the solution, based on OpenSCADA were presented.
- 10 Ukrainian Conference for Free Software Developers and Users — the report about the solutions, based on the OpenSCADA was read at the conference and in the hall there was made the mini-stand with the live solutions, based on OpenSCADA.
- Qt Developer Days 2011 in Munich and San Francisco: there were made the demonstration of OpenSCADA solutions Dynamic model of the DMKD multifuel steam boiler unit №9 on the PC and The dynamic model of real-time of the Anastasievskaja GLKS on Nokia N9 in the Qt Demo Pavilion.

Conclusion

This release of OpenSCADA actually finishes the work on adaptation to the ARM hardware architecture, which will be verified in the provision and publication of the official release 0.7.2 builds for mobile devices.

Although this release was delayed and a few late but it still was carefully prepared and will be a good basis for finishing the project's code base for the upcoming responsible release 0.8.0 LTS. In addition, this release can be used for widespread implementation in critical applications, followed by a smooth transition to 0.8.0 LTS after its release.

Subsequent development efforts will be directed to the completion of planned tasks and preparation for release the OpenSCADA long term support (LTS) version 0.8.0.