

OpenSCADA 0.8.0 LTS

(Замечания к релизу)

Оглавление

Введение	1
1 Реализация плановых задач	4
2 Оптимизация, повышение стабильности, устойчивости и производительности системы	5
3 Усовершенствование и стабилизация графической подсистемы	8
4 Формирование, расширение и стабилизация API пользовательского программирования	11
5 Общесистемные расширения	12
6 Публикация решений OpenSCADA	15
Заключение	15
Ссылки	15

Введение

Релиз открытой SCADA(Supervisory control and data acquisition) системы версии 0.8.0 является стабильным промышленным релизом продолжительной поддержки (LTS).

Основной целью данного релиза является предоставление сообществу пользователей и разработчиков свободного программного обеспечения (ПО) стабильной платформы для построения решений комплексных систем автоматизации и других смежных решений, а также предоставления коммерческих услуг на основе проекта OpenSCADA.

Данный релиз является следующим стабильным релизом, для которого предоставляется техническая поддержка от разработчиков и для которого планируется выпуск исправлений в течение продолжительного времени. Жизненный цикл предыдущего стабильного релиза 0.7.0 продолжительной поддержки (LTS) будет прекращён в течении месяца после выпуска 0.8.0 последним обновлением.

Данный документ является обработкой(компиляцией) документа "ChangeLog" системы OpenSCADA версии 0.8.0, включая промежуточные рабочие релизы 0.7.1 и 0.7.2, который призван вкратце и наглядно осветить новые возможности системы OpenSCADA. Детально ознакомиться с изменениями можно в файле "ChangeLog" из дистрибутива системы или здесь: <http://wiki.oscada.org/Works/ChangeLog>.

Ключевыми особенностями данной версии являются:

- Реализация плановых задач.
- Оптимизация, повышение стабильности, устойчивости и производительности системы.
- Усовершенствование и стабилизация графической подсистемы.
- Расширение API пользовательского программирования.
- Формирование, расширение и стабилизация API пользовательского программирования
- Общесистемные расширения.
- Публикация решений OpenSCADA.

Новые и обновленные модули:

- *Archive.DBArch (0.9.5)* — Добавлены лимиты запроса данных по времени запроса и ещё некоторые расширения.
- *Archive.FSArch (1.5.1)* — Значительная стабилизация. Множество расширений, в том числе:
 - Повышение производительности чтения индексов файлов архивов значений.
 - Адаптация хранения и чтения вещественного к ARM FPA.
 - Добавлены лимиты запроса данных по времени запроса.
 - Добавлен лимит на общий размер файлов архиватора значений.
- *DB.MySQL (1.7.1)* — Стабилизация. Добавлена возможность указания таймаутов подключения.
- *DB.SQLite (1.6.4)* — Стабилизация.
- *DB.FireBird (0.9.7)* — Стабилизация.
- *DB.PostgreSQL (0.9.2)* — Стабилизация.
- *DAQ.DiamondBoards (1.2.5)* — Добавлена возможность смены типа параметра. Стабилизация.
- *DAQ.System (1.7.5)* — Стабилизация. Зависимость от библиотеки "Libsensors" сделана опциональной. Добавлено планирование вызовов по CRON.
- *DAQ.BlockCalc (1.6.0)* — Стабилизация. Множество расширений, в том числе:
 - Добавлена подсветка синтаксиса.
 - Добавлено планирование вызовов по CRON.
 - Добавлен запускающий и останавливающий вызов блоков.
 - Добавлена поддержка объектного типа параметров.
- *DAQ.JavaLikeCalc (2.0.0)* — Значительная стабилизация. Множество расширений, в том числе:
 - Добавлена подсветка синтаксиса.
 - Выполнена адаптация к аппаратной архитектуре ARM.
 - Добавлена реализация регулярных выражений и ряда расширений API пользователя.
 - Добавлена возможность записи символов строки с помощью восьмеричного "\041" и шестнадцатеричного "\x21" числа.
 - Добавлен прямой, динамический вызов библиотечных функций.
 - Добавлено прямое объединение строковых констант.
 - Добавлен условный вызов аргументов выражения (второго аргумента) с логическими операциями ||(OR) и &&(AND).
 - Добавлена поддержка объектного типа параметров источника данных.
- *DAQ.LogicLev (1.3.0)* — Стабилизация. Множество расширений, в том числе:
 - Добавлена возможность смены типа параметра.
 - Добавлено планирование вызовов по CRON.
 - Добавлена поддержка объектного типа параметров.
- *DAQ.SNMP (0.7.0)* — Значительная стабилизация. Множество расширений, в том числе:
 - Реализация полной поддержки клиентской части протокола.
 - Добавлена поддержка всех типов значений.
 - Добавлена поддержка записи значений.
 - Улучшена обработка ошибок.
 - Добавлены дополнительные параметры конфигурации подключения.
 - Добавлена возможность прямого запроса скаляров.
- *DAQ.Siemens (1.4.0)* — Значительная стабилизация. Множество расширений, в том числе:

- Версия библиотеки LibnoDave обновлена до 0.8.4.6.
- Добавлено планирование вызовов по CRON.
- Добавлена функция переподключения для "Industrial Ethernet" соединений.
- Добавлен вариант протокола ISO_TCP для S7-200.
- Добавлена поддержка объектного типа параметров.
- Добавлена реализация протокола "ADS".
- Несколько унифицированы ошибки подключения.
- Добавлена возможность установки номера БД и смещения в шестнадцатеричном виде.
- *DAQ.ModBus (1.3.0)* — Значительная стабилизация. Множество расширений, в том числе:
 - Добавлена подсветка синтаксиса списка атрибутов.
 - Добавлена поддержка типа параметра "Логический" для работы по шаблону параметра, а также функции пользовательского API для отправки произвольных нестандартных ModBus-запросов из шаблонов.
 - Добавлена и использована функция формирования сообщений о нарушениях в контроллере.
 - Добавлена поддержка функций групповой записи (0x0F, 0x10).
 - Добавлен параметр установки ограничения размера блока групповых запросов.
 - Добавлена поддержка объектного типа параметров.
 - Размер максимального блока запроса сделан опциональным.
 - Добавлена возможность комментирования элементов списка атрибутов стандартного типа параметра посредством символа '#' в начале.
- *DAQ.DCON (0.5.1)* — Модуль полностью переписан на предмет поддержки смешанных модулей посредством выбора команд запроса. Стабилизация.
- *DAQ.ICP_DAS (0.8.0)* — Добавлена библиотека API "ICP DAS" libi8k.a для архитектуры ARM и включена возможность сборки модуля для ARM. Сборка по умолчанию отключена для предотвращения попыток сборки на неподходящем ARM окружении. Стабилизация.
- *DAQ.DAQGate (0.9.5)* — Добавлен кеш параметров для их инициализации при автономной работе. Добавлено планирование вызовов по CRON. Стабилизация.
- *DAQ.SoundCard (0.6.2)* — Стабилизация.
- *DAQ.OPC-UA (0.6.2)* — Стабилизация. Добавлена подсветка синтаксиса списка атрибутов. Адаптация хранения и чтения вещественного к ARM FPA.
- *DAQ.BFN (0.5.1)* — Новый модуль.
- *Protocol.ModBus (0.6.4)* — Стабилизация. Некоторые улучшения.
- *Protocol.OPC-UA (0.6.2)* — Стабилизация.
- *Transport.Sockets (1.5.1)* — Значительная стабилизация. Добавлен таймаут повтора запроса исходящего транспорта.
- *Transport.SSL (1.0.1)* — Стабилизация. Некоторые улучшения.
- *Transport.Serial (0.8.0)* — Стабилизация. При инициализации порта использованы предыдущие настройки и реализована возможность опускать указание некоторых настроек. Добавлено управление потоком с помощью сигнала RTS для простых преобразователей сигналов RS232->RS485.
- *Protocol.HTTP (1.6.0)* — Стабилизация. Множество расширений, в том числе:
 - Добавлена поддержка переменной HTTP "Transfer-Encoding=chunked" в исходящем транспорте.
 - Добавлена поддержка пользовательских шаблонов для внутреннего содержимого модуля.
 - Добавлена генерация сообщений аутентификации пользователей.
 - Добавлена поддержка всех основных вариантов завершения строки при разборе HTTP-запроса.
- *Protocol.SelfSystem (0.9.5)* — Стабилизация.
- *Protocol.UserProtocol (0.6.2)* — Добавлена подсветка синтаксиса. Стабилизация.
- *Special.FLibComplex1 (1.1.0)* — Добавлен прямой, динамический вызов библиотечных функций.
- *Special.FLibMath (0.6.0)* — Добавлен прямой, динамический вызов библиотечных функций.

- *Special.FLibSYS (1.0.0)* — Добавлен прямой, динамический вызов библиотечных функций.
- *Special.SystemTests (1.5.1)* — Небольшие улучшения.
- *UI.QTStarter (1.7.0)* — Стабилизация. Значительные расширения и улучшения.
- *UI.QTCfg (2.1.1)* — Стабилизация. Значительные расширения и улучшения.
- *UI.WebCfg (1.5.6)* — Стабилизация.
- *UI.WebCfgD (0.8.1)* — Стабилизация. Значительные расширения и улучшения.
- *UI.VCAEngine (1.3.0)* — Стабилизация. Значительные расширения и улучшения.
- *UI.Vision (1.3.0)* — Стабилизация. Значительные расширения и улучшения.
- *UI.WebVision (1.0.1)* — Значительная стабилизация. Значительные расширения и улучшения.
- *UI.WebUser (0.6.2)* — Стабилизация. Добавлена подсветка синтаксиса процедур.

1 Реализация плановых задач

В соответствии с [планом](#) релиза были выполнены следующие задачи:

- *Формирование предоставления коммерческих услуг на основе OpenSCADA.* — Выстроена концепция и созданы механизмы предоставления коммерческих услуг разработчиками на основе OpenSCADA (<http://oscada.org/ru/uslugi>).
- *Адаптация системы OpenSCADA для работы на аппаратной платформе ARM.* — Осуществлена сборка, адаптация и полномасштабное тестирование OpenSCADA на архитектуре ARM; адаптация и тестирование проводились на интернет планшете фирмы [Nokia](#) — [N800](#) (<http://wiki.oscada.org/Works/Tests/ARM>). Выполнена сборка и адаптация OpenSCADA для сложного (очень старого) программного окружения контроллера [LP-5451](#), а также сборка для контроллера [SMH2Gi](#) и смартфонов фирма [Nokia](#): [N900](#), [N950](#), [N9](#). В рамках сборки и адаптации на различные мобильные устройства были выполнены следующие задачи:
 - *SYS:*
 - Прототипы функций `TMess::put()`, `TMess::get()`, `TCntrNode::grpSize()`, `TCntrNode::grpId()`, `TCntrNode::grpAt()` изменены для исправления использования беззнакового типа "char" на ARM как знакового.
 - Выполнена адаптация для сборки с Glibc версии меньше 2.5 (2.3.2).
 - *TVariant, DAQ.{AMRDevs, OPC-UA}*: Исправлен невыравненный доступ к данным.
 - *DAQ.OPC-UA*: Добавлены системные функции `floatLE()`, `floatLErev()`, `doubleLE()`, `doubleLErev()` для преобразования формата хранения вещественного числа на различных архитектурах.
 - Добавлена адаптация к uClibc. Сборка начинается с версии 0.9.32, которая имеет поддержку функций реального времени вроде `clock_nanosleep()`.
 - Добавлена проверка и отключение использования "Iconv" и "Intl" (I18N) API.
 - Добавлено автоматическое обнаружение и включение параметра конфигурации `--enable-CrossCompile` для окружения кросс-компиляции.
 - Добавлен параметр конфигурации `--enable-CoreLibStatic` для отключения сборки разделяемой библиотеки ядра OpenSCADA и статической линковки программы вызова-запуска.
 - Функция `usleep()` везде заменена на собственную функцию-обёртку `TSYS::sysSleep()` по причине отсутствия функции `usleep()` в uClibc и доступности её только при специальной конфигурации сборки uClibc. В `TSYS::sysSleep()` использована `nanosleep()`.
 - Функция `pthread_yield()` заменена на `sched_yield()`.
 - *DAQ:*
 - *JavaLikeCalc*: Виртуальная машина переписана для прямого использования структур команд с целью исключения проблемы выравнивания, а также для некоторого повышения производительности.
 - *Siemens*: Исправлена проблема библиотеки NoDave на предмет глобального

вызова команды упаковки структур "#pragma pack(1)".

- *ICP_DAS*: Добавлена библиотека API "ICP DAS" libi8k.a для архитектуры ARM и включена возможность сборки модуля для ARM.
- *Archive*:
 - Прототипы функций TArchiveS::messPut() и TArchiveS::messGet() изменены для исправления переменной уровня сообщения как знакового числа.
 - *FSArch*:
 - Исправлено хранение вещественного числа формата LE в архиве на архитектуре ARM.
 - В алгоритме быстрого подсчёта количества битов использовано функцию невыравненного чтения TSYS::getUnalign32().
 - Размер поля кодировки в плоском архиве сообщения расширен с 9 до 99 символов.
- *UI.QTCfg*: Добавлено кратковременное засыпание после закрытия всех окон в функции отключения, с целью предоставить время окончательно разрушиться объектам всех окон и предотвратить падение.
- *Реализация механизма откатов изменений редактирования в Vision*. — В рамках окна визуального редактирования виджетов реализован многоуровневый механизм отката изменений для всех основных операций: визуальное изменение геометрии, изменение значения атрибута виджета, добавление/удаление виджета, копирование виджета и редактирование виджетов на основе примитива "ElFigure".

2 Оптимизация, повышение стабильности, устойчивости и производительности системы

С момента последнего стабильного релиза в процессе работ над данной версией, а также её практической адаптации, было обнаружено и исправлено в общей сложности более 300 ошибок. В целом была проделана работа по чистке кода от предупреждающих сообщений компилятора, с флагом "-Wall":

- удаление неиспользованных переменных;
- добавление возврата во всех функциях;
- инициализация переменных с возможностью их использования неопределёнными;
- исключение неполной обработки перечислимых типов в операции выбора;
- сравнение знакового с беззнаковым целым.

Перечислим наиболее существенные ошибки, исправление которых значительно отразилось на повышении стабильности:

- *Общесистемные*:
 - Команда сборочной системы "\$ make dist" исправлена и адаптирована для корректного создания дистрибутивов OpenSCADA.
 - Функция TSYS::cron() для вызова по расписанию неоднократно исправлена.
 - Тип переменной для хранения результата функции find() строки везде изменён на корректный "size_t".
 - Исправлены внеплановые вызовы по расписанию, связанные с рассинхронизацией значений функций time() и clock_gettime(CLOCK_REALTIME, &sp_tm).
 - Исправлена проверка и ожидание потоков на доступность при перекрытии с остановкой предыдущего одноимённого потока.
 - Отключено ожидание инициализации для отсоединяемых задач с целью предотвращения зависания на ожидании быстро закрываемых задач.
 - Исправлена работа функций невыравненного чтения getUnalign*() на ARM. Проблема обнаружена на PXA270.
 - *Function*: Исключена возможность утечки памяти в случае попытки установить объект в необъектный параметр функции.
 - *DAQ.ModBus, Transport.Serial, UI.{WebCfg, WebCfgD, WebUser, WebVision}*: Исправлено ошибочное использование символа завершения строки '\n' для множества программных платформ UNIX, MAC, DOS/Windows, путём замены на "\x0A" .

- *ResString*: Для чтения-записи строки использован оборот "string(vl.data(), vl.size())", с целью предотвращения COW алгоритма.
- *Function*: Исправлена проверка на модификацию при установке значений разных типов IO (Вещественное в Целое).
- *Базы данных*:
 - *DB.PostgreSQL*: Исправлена обработка адреса БД на предмет определения пустых полей.
- *Транспорты и протоколы*:
 - *Transport*:
 - Исправлена загрузка всех транспортов из БД в случае наличия записей про транспорт, модуль которого отсутствует.
 - Исправлено использование беззнакового типа в возврате функций *read()* и *write()*. Добавлены ресурсы к счётчикам входных/выходных запросов.
 - Включена возможность прерывания исходящего транспорта сигналом на ожидании таймаута. Полезно для прерывания транспорта при выходе/останове.
 - *Sockets*:
 - Исправлен пропуск инициализации размера возвратного значения для функции *getsockopt()*.
 - Исправлено закрытие клиентского соединения по таймауту времени жизни, для подключений при отсутствии данных.
 - Unix-сокеты установлены в неблокирующий режим посредством флага `O_NONBLOCK`. Исправлено формирование сообщений об ошибках для исходящих транспортов.
 - *Serial*: Установлен флаг `O_NONBLOCK` для функции *open()* последовательного устройства с целью предотвращения блокирования открытия порта на некоторых USB->RS485 преобразователях.
 - *Sockets, SSL*: Добавлена очистка входного буфера для предотвращения повторного использования на высоколатентных каналах после таймаута.
 - *Transport, DAQ.{AMRDevs, DCON, ICP_DAS, ModBus, OPC-UA}, Protocol.HTTP*: Исправлено использование функции *TTransportOut::messIO()* при запросе хвоста и получении нулевого ответа.
 - *Protocol.HTTP, UI.{WebCfg, WebCfgD, WebVision, WebUser}*: Исправлен разбор элементов POST-запроса на предмет разбора всех атрибутов каждого элемента.
- *Сбор данных и архивы*:
 - *Archive*:
 - Исправлено использование буфера мягкой сетки с низким и высоким разрешением времени в архивах значений.
 - Исправлено вычисление начала архива в функции *begin()* при пустом буфере.
 - *FSArch*:
 - Исправлено создание файлов архива сообщений при помещении множества старых сообщений. Добавлен параметр предотвращения дубликатов.
 - Результат функции *fgetc()* записан в целочисленную переменную вместо символьной, что исправило обнаружение "EOF".
 - Исправлено чтение данных из буфера архива в соответствии с размером буфера в алгоритме быстрого вычисления количества битов.
 - Исправлено помещение смещения в кеш из алгоритма быстрого вычисления количества битов.
 - Исправлена некорректная обработка конца невыравненных блоков быстрого алгоритма подсчёта количества битов.
 - Добавлен монопольный ресурс к вызову функции *calcVIOff()* с целью предотвращения некорректной работы с кешем.
 - *DAQ*:
 - *System*: Исправлено падение при выходе для источника "Hddtemp".

- *ModBus*: Режим установки значений атрибутов параметров изменён на активный и улучшена обработка ошибок.
- *JavaLikeCalc*:
 - Добавлена проверка индекса при доступе к символу строки по индексу с целью предотвращения выхода за диапазон.
 - Исправлено падение функции *replace()* объекта "RegExp".
- *Siemens*:
 - Исправлено получение и запись значений вещественных типов.
 - Исправлено перепоключение в случае ошибки.
 - Исправлено использование свойства размера целого, вещественного и строкового типов.
 - Добавлена очистка связей параметра при его выключении.
- *Интерфейсы визуализации*:
 - *VCAEngine*:
 - Исправлено блокирование сеанса для динамически активных проектов при динамическом доступе к собственным атрибутам виджета.
 - Предотвращена возможность потери изменений в сервере визуализации при отображении в визуализаторах путём предварительного сохранения значения тактового счётчика в сервисном запросе "openList".
 - Ресурсы атрибутов виджетов сделаны более прозрачными (посредством использования рекурсивного мютекса) и использован один общий ресурс, для сеансов и другого, с целью предотвращения перекрёстного блокирования одной задачи другой.
 - *Vision, WebVision*:
 - Повышено разрешение координат всех виджетов до трёх знаков после запятой с целью обеспечения нормального позиционирования на больших масштабах.
 - Исправлен доступ за границу вектора при отрисовке графиков примитива "FormEls".
 - Исправлено обновление изображения примитива "Элементарная фигура" при изменении масштаба.
 - Исправлено обновление примитива "Протокол" по времени и размеру.
 - *Vision*:
 - Исправлено падение сеанса для проектов с периодом исполнения более 500 мс.
 - Реализована доводка координат фигур примитива "Элементарная фигура" к остатку положения виджета.
 - Масштабирование текста примитива "Текст" изменено на изменение размера шрифта вместо масштабирования конечного изображения, что повысило общее качество отображения шрифта.
 - *WebVision*:
 - Удалена проверка попадания точек фигур в область отображения примитива "Элементарная фигура". Данная функция переложена на библиотеку GD.
 - Исключена возможность падения для примитивов "Diagram" или "ElFigure" в случае передачи некорректных значений атрибутов изображения "xSc" или "ySc" из Web-браузера.
 - Исправлено открытие и замена корневой страницы.
 - Исправлена обработка масштаба вложенных страниц с учётом масштаба корневой.
 - Исправлена очистка хвоста примитива "Протокол".

Оптимизация и повышение производительности:

- *SYS*:
 - Механизм ресурса объекта "ResString" заменён с RW-блокировок на мютекс с целью повысить производительность и сократить потребление памяти.
 - *TConfig*: Объект элемента конфигурации (TCfg) основан на TVariant. Доступ к

объекту "TCfg" сделан прозрачным, что позволяет использовать связи без базовых типов.

- Механизм хранения дочерних узлов в карте по ключу "string" изменён на тип ключа "const char*". Все узлы адаптированы к константности идентификатора.
- *Archive.FSArch*: Обработка индексной таблицы архива значений значительно ускорена посредством использования быстрого алгоритма вычисления количества битов в 32-разрядном целом.
- *UI.WebCfgD*:
 - Добавлено изменение размера иконок на стороне сервера с целью снижения трафика.
 - Использован групповой запрос содержимого страниц для значительного повышения производительности на медленных и высоколатентных каналах.
- *UI.QTCfg*:
 - Использован групповой запрос к интерфейсу управления для повышения производительности на медленных и высоколатентных соединениях.
 - Исключён многократный запрос элементов выборных колонок таблиц.
- *UI.Vision*: Повышена производительность отрисовки заполнений примитива "ElFigure".
- *UI.VCAEngine*:
 - Выполнены мероприятия по повышению скорости запуска сеанса проекта в числе:
 - Пропущена инициализация базовых атрибутов ввиду их последующего наследования.
 - Удален перевод некоторых сообщений времени исполнения сеанса.
 - Реализовано включение только нужных страниц при запуске сеанса. Остальные страницы исполняющегося сеанса включаются по мере обращения к ним. Это позволило значительно повысить скорость запуска сеанса, а также уменьшить зависимость скорости запуска от сложности проекта визуализации.
 - Реализация архивного режима примитива "Документ" изменена на прямую работу с БД, что позволило расширить глубину архива до 1000000 документов и без ущерба для оперативной памяти.

3 Усовершенствование и стабилизация графической подсистемы

Заметная работа была выполнена в рамках графической подсистемы, а именно в модулях движка СВУ *UI.VCAEngine*, визуализаторов *UI.Vision* и *UI.WebVision*, а также конфигуураторах. Внесённые изменения были направлены на стабилизацию, оптимизацию потребления памяти и улучшение пользовательских свойств.

Улучшения графической подсистемы:

- *Добавлено API подсветки синтаксиса*. На основе данного API подсветка синтаксиса добавлена в:
 - *DAQ*:
 - *BlockCalc*: Подсветка синтаксиса в описании атрибутов параметров.
 - *JavaLikeCalc*: Подсветка синтаксиса к компилятору языка *JavaLikeCalc.JavaScript*, а также ко всем текстовым полям, использующим этот язык внутри модуля.
 - *ModBus, OPC-UA*: Подсветка синтаксиса списка атрибутов.
 - *Protocol.UserProtocol*: Подсветка синтаксиса к текстам процедур протокола.
 - *UI*:
 - *QTCfg*: Отображение многоуровневой подсветки синтаксиса, передаваемой из модели данных в виде правил регулярных выражений.
 - *VCAEngine*: Подсветка синтаксиса к текстовым атрибутам примитивов виджета, для которых это нужно.
 - *Vision*: Подсветка синтаксиса к текстовым атрибутам примитивов виджета, а также текстам программ виджетов.
 - *WebUser*: Подсветка синтаксиса процедур формирования Web-интерфейса.

- *UI.QTStarter*:
 - Адаптация для ARM.
 - Шрифт сообщения в "сплеше" зафиксирован в размере 10 пикселей для обеспечения единообразного отображения.
 - Реализован останов OpenSCADA в случае закрытия последнего окна диалога выбора QT UI-модулей.
 - Реализован проброс параметров командной строки QT4 из OpenSCADA (например, `--qws` и `--style` в `-qws` и `-style`).
- *UI.QTCfg*:
 - Добавлена возможность ручного изменения высоты текстовых и табличных полей. Изменение осуществляется схватыванием и перетаскиванием нижнего края поля.
 - Адаптация для ARM.
 - Групповой запрос "CntReq" использован для запроса контекста страницы.
 - Добавлено ограничение на высоту строк таблиц в половину высоты таблицы.
 - Введено ограничение на размер диалога ввода и включено адаптивное определение его размера по содержимому.
 - У элементов доступных для изменения высоты пользователем для обеспечения работы везде установлен фиксированный стиль "StyledPanel".
 - Добавлена функция полнотекстового редактирования текста ячеек таблиц.
 - Движок воспроизведения правил подсветки синтаксиса полностью переписан для упрощения и предотвращения скрытых ошибок.
- *UI.{VCAEngine, Vision, WebVision}*:
 - Исправлено обращение за границы массива (вектора) при построении трендов примитива "Диаграмма".
 - Добавлено и реализовано свойство ширины графика примитива "Диаграмма".
 - Добавлена периодическая проверка дерева виджетов страницы с целью обнаружения факта удаления виджетов путём запроса полного перечня виджетов.
 - Реализовано отображение графиков группы графиков примитива "Диаграмма" в шкале значения в случае отличия шкал всех графиков не более чем на 20%.
 - Добавлена и реализована возможность выбора режима подтверждения для элементов редактирования строки и текста примитива "Элементы формы".
 - Исправлена возможность потери некоторых изменений в модели при отображении их в визуализаторах.
 - Добавлено и реализовано свойство количества значений на пиксел в графиках примитива "Диаграмма" с целью управления детализацией экспорта в CSV и т.д.
 - Реализована логарифмическая шкала значений для графиков примитива "Диаграмма".
 - Добавлено и реализовано свойство сохранения соотношения сторон главной страницы при разворачивании главного окна исполнения сеанса проекта.
 - Добавлен новый тип "Полное Видео" в примитиве "Медиа". Для реализации в *UI.Vision* использован медиа-движок "Phonon".
 - Добавлено поле свойств архива "prn{X}rgr" в примитиве "Диаграмма", которое заполняется значением доступной глубины и периодичности архива.
- *UI.VCAEngine*:
 - Добавлены специфические свойства к атрибутам примитивов вроде помощи и подсветки синтаксиса.
 - Реализовано включение страниц сеанса проекта по надобности с целью повышения скорости запуска и оптимизации использования оперативной памяти.
 - Примитив "Документ" переделан для ведения архивов, архивного режима полностью в БД, а также добавлена функция пользовательского API для доступа к элементам архива.
 - Добавлена периодическая (30 минут) проверка и закрытие уже неиспользованных (потерянных) сеансов проектов.
 - Удалён предыдущий (противоречивый) механизм отката удаления вложенных виджетов и заменён полноценным механизмом откатов в визуализаторе *UI.Vision*.
 - Выполнена оптимизация использования памяти элементами сеанса до 20%.

- Удалена поддержка первой, устаревшей, версии структуры БД среды визуализации.
- Добавлено ограничение на время генерации документов в примитиве "Документ" в 5 секунд.
- Добавлена поддержка типа атрибутов "Объект" и "Текст".
- Отключена принудительная инициализация новых пользовательских атрибутов в EVAL.
- Добавлена контекстная подсказка к атрибутам примитивов виджетов.
- Примитив "Документ" переключен в режим полного разбора XML с целью полноценного сохранения разметки XHTML.
- Добавлена индикация состояния корректности ссылок в виде "(+)" в конце.
- Добавлено перенаправление необработанных событий к странице выше. Это позволяет глобально обрабатывать события на самой верхней странице.
- Добавлена возможность установки связи между виджетами типа "wdg:" как в виде абсолютной, так и относительной формы.
- Добавлена поддержка типа связи "arh:".
- Добавлена поддержка объектного типа параметров источника данных.
- *UI.Vision:*
 - Добавлено отображение помощи по атрибутам виджета.
 - Адаптация для ARM.
 - Добавлена индикация сообщения о потере соединения с сервером визуализации, а также восстановление при удалённой визуализации.
 - Добавлен отдельный полнотекстовый редактор для текстовых атрибутов.
 - Добавлена реализация полноценных откатов и повторов изменений при визуальном редактировании виджетов.
 - Улучшения в отзывчивости и обновлении изменений в сеансе исполнения проекта: оптимизирована производительность отрисовки примитива "ElFigure", обработка изменения уровня виджета "geomZ".
 - Добавлен экспорт в CSV формат из примитивов "Диаграмма" и "Документ".
 - Введено ограничение на размер диалогов ввода и включено адаптивное определение их размеров по содержимому.
 - Полностью переписан механизм исполнения правил подсветки синтаксиса.
 - Добавлено создание нового сеанса после восстановления удалённого подключения и отсутствии предыдущего сеанса.
 - Фон окна редактируемого виджета установлен в шаблон QT::Dense7Pattern с целью устранения возможного перекрытия по цвету.
 - Добавлена обработка событий выбора виджета при нажатии в области скроллинга, но за пределами виджета.
 - Примитив "Элементарная фигура":
 - Добавлен диалог установки свойств отдельно выбранных фигур.
 - Выполнена реорганизация операций из диалога свойств и контекстного меню с динамическими и статическими свойствами.
 - В процессе добавления фигуры реализована её отрисовка при перемещении курсора мыши.
- *UI.WebVision:*
 - Добавлена поддержка gif и jpg изображений в элементах примитива "Элементарная фигура".
 - Исправлена обработка масштаба вложенных кадров, а также пересмотрен механизм формирования-включения скрола.
 - Добавлена функция преобразования изображений на стороне сервера. Функцию преобразования использовано для изменения размера и обесцвечивания изображений неактивных кнопок.
 - Реализовано открытие малых окон вложенных страниц в виде DIV-блоков, что решает проблему блокировки и продолжительного открытия внешних окон многими браузерами.
 - Главная страница интерфейса отцентрована в окне браузера.

- *UI.WebCfgD*:
 - Уменьшена яркость обесцвеченных-пассивных кнопок.
 - Все окна диалогов реализованы в виде DIV-блоков с целью исключить проблемы и задержки в открытии внешних окон различными браузерами.
 - Использован групповой запрос содержимого страниц для значительного повышения производительности на медленных и высоколатентных каналах.
 - Добавлена информация в строке статуса о текущем пользователе и возможность его смены.
 - Выполнена адаптация для работы в полную ширину экрана.

4 Формирование, расширение и стабилизация API пользовательского программирования

Было продолжено формирование объектного API пользовательского программирования, которое предусматривает интеграцию пользовательских функций в дерево объектов системы OpenSCADA. Кроме этого был внесен ряд изменений в существующие библиотеки функций пользовательского API.

В частности были осуществлены следующие изменения:

- *SYS*:
 - Пользовательское API для объекта *TArrayObj* расширено на предмет возможности использования объектных свойств, если свойства - не числа.
 - В дерево объектов OpenSCADA пользовательского API добавлена функция *SYS.Security.access()* для контроля доступа пользователя OpenSCADA к ресурсам с нужными правами.
 - Добавлены функции работы с файлами: *SYS.fileRead()* и *SYS.fileWrite()*.
 - К API объекта "*XMLNodeObj*" добавлена функция *getElementBy()* для поиска вложенных узлов по значению атрибута.
 - Добавлена функция осуществления кодирования текста между различными символьными кодировками.
 - Добавлена функция *NodeObj.nodePath()*.
- *SYS, DAQ.JavaLikeCalc*: Добавлена реализация поддержки пользовательского API регулярных выражений, основанной на "PCRE", посредством объекта "*TRegExp*". Реализован интерфейс функций *RegExp.exec()* и *RegExp.test()* в стандарте для языка "JavaScript". Для *DAQ.JavaLikeCalc* реализована "**new RegExp()**" инициализация для объекта "*TRegExp*". Для *DAQ.JavaLikeCalc* реализованы дополнительные функции с использованием "*RegExp*": *int search()*; *Array match()*; *Array split(RegExp pat, int limit)*; *string replace(string substr, string str)*; *string replace(RegExp pat, string str)*.
- *TConfig, TBD, TTransportIn, TTransportOut, TUser, TGroup, TPrmTplLib, TPrmTempl, TParamContr, TController, TVArchive, TVArchivator, TMArchivator*: Добавлены функции пользовательского API *cfg()* и *cfgSet()* для доступа к конфигурации объектов (хранящейся в БД).
- *TFunction, DAQ.JavaLikeCalc*: Добавлен объект пользовательского API "*TFuncArgObj*" для получения аргументов функции.
- *DB*: Добавлены функции пользовательского API *fieldStruct()*, *fieldSeek()*, *fieldGet()*, *fieldSet()* и *fieldDel()* для объекта таблицы, с целью предоставить пользовательский доступ к БД посредством интерфейса БД OpenSCADA.
- *DAQ*:
 - Добавлена функция *TController::alarmSet()* для генерации типовых нарушений в объекте контроллера модулей подсистемы "Сбор данных".
 - Добавлена функция *SYS.DAQ["Modul"]["Controller"]["Parameter"]["Attribute"].arch()* для прямого обращения к объекту архива, связанного с атрибутом параметра.
 - Добавлены функции пользовательского API *enable()* и *start()* для прямого контроля за состоянием объекта контроллера.
 - *JavaLikeCalc*:

- Добавлена встроенная функция *typeof()* для проверки типа значения.
- Для типа данных "null" добавлена функция *isEval()*, которая всегда возвращает "true".
- К объектам библиотечных функций добавлена функция пользовательского API *call()*.
- *ModBus*: Добавлена функция пользовательского API *messIO()* в объект контроллера "ModBus" с целью предоставления возможности отправки нестандартных ModBus-запросов прямо из шаблона параметра.
- *DAQ.JavaLikeCalc*, *Special.{FLibComplex1, FLibMath, FLibSYS}*: Добавлена функция динамического вызова библиотечных функций *SYS.DAQ.JavaLikeCalc["lib_{Lib}"].funcId(prms, ...)*.
- *Archive*: Добавлены функции пользовательского API *status()*, *end()* и *begin()* для доступа к состоянию и свойствам объекта архиватора сообщений.
- *UI.VCAEngine*: Добавлена функция пользовательского API *wdgAt()* для простого подключения к виджету как по абсолютному, так и относительному пути.

5 Общесистемные расширения

В общесистемное API системы OpenSCADA были внесены значительные изменения и расширения с целью общей стабилизации и ввиду общих мероприятий по чистке кода, а также реорганизации дерева исходных текстов:

- *SYS*:
 - Реализована новая структура дерева исходных текстов.
 - Все ресурсы проекта перемещены в отдельный архив.
 - Для создания архивов исходных текстов и ресурсов можно использовать команды "\$ make dist-lzma" и "\$ make dist-reslzma".
 - Документация выполнена статичной для модулей и поделена по языкам. Для каждого языка документации формируется отдельный пакет. В документации каждого языка создан индексный файл по всему перечню локальной и сетевой документации. Общий перечень локальной документации значительно расширен новыми документами, а также вспомогательной сетевой документацией.
 - Добавлены пакеты библиотечных БД "*-LibDB.Main" и "*-LibDB.VCA", помещены в "data/LibsDB".
 - Вместо "DemoDB" пакета добавлены пакеты моделей "AGLKS" и "Boiler" помещены в "data/ModelsDB".
 - Конфигурационные файлы для построения RPM и DEB пакетов обновлены.
 - Зависимость на библиотеку PCRECPP глобально заменена на PCRE.
 - Добавлена новая функция *TSYS::time2str()* для преобразования интервалов времени в строку вроде "1 час 23 мин 10 сек". Эта функция использована везде.
 - Добавлена реализация собственного XML-парсера вместо Expat. Собственный XML-парсер решает проблему поддержки различных кодировок, отличных от UTF, а также безопасной обработки ситуации наличия недопустимых для текущей кодировки символов.
 - *TMess*: Общие сообщения интерфейса управления перемещены в одну функцию *TMess::lab**.
 - Объект хранения строки с ресурсом "ResString" значительно расширен на предмет прозрачного преобразования из/в тип "std::string".
 - Добавлена возможность сохранения в конфигурационный файл:
 - Добавлен префикс БД "<cfg>" для представления конфигурационного файла в роли источника загрузки/сохранения конфигурации.
 - Функции *TDBS::dataSeek()* и *TDBS::dataDel()* обновлены для строгой обработки конфигурационного файла.
 - Функция *chkSelDB()* адаптирована для поддержки записи в конфигурационный файл.
 - Добавлено сохранение больших и многострочных значений полей конфигурации в текстовом поле отдельного тега конфигурационного файла.

- Тип "long long", обычно 64-рядное целое, заменён везде на более определённый "int64_t" на всех архитектурах.
- Объекты потоков OpenSCADA жёстко слинкованы со своим потоком. Добавлена статистика загрузки и вызовов периодических потоков.
- Реализована возможность проверки загруженных объектов на их отсутствие в БД и удаление. Функция проверки активируется только в случае прямой загрузки из БД.
- Добавлена системная периодическая функция *perSYS*Call() вызова объектов подсистем и их модулей. Используется для сервисных целей ненагруженных и редких задач модулей и подсистем.
- Функция тестирования выражения по шаблону перемещена в объект "TRegExp".
- *XMLNode*:
 - Полностью удалён код использования XML-парсера "Expat".
 - Добавлена возможность загрузки текстовых частей тега в отдельные специальные теги в полном режиме. Предназначено для полного сохранения конфигурации тега.
 - Добавлена возможность загрузки и сохранения блоков комментариев в отдельные специальные теги, в полном режиме.
- Добавлена встроенная функция включения генерации дампа памяти (снятия ограничения на размер файла) посредством параметра запуска OpenSCADA *--CoreDumpAllow*.
- Во все скрипты запуска OpenSCADA добавлена процедура автоматической обработки файла дампа памяти (формирование файла разворота стека — backtrace). Файл дампа памяти ожидается в рабочей директории в виде "core".
- Добавлена возможность включения (встраивания) модулей в библиотеку ядра OpenSCADA.
- Версия модульного API всех подсистем увеличена до 6.
- *TConfig*, *TValFunc*, *TVal*: Хранение строковых конфигурационных полей переведено в объект ресурсной строки "ResString", а затем возвращено с использованием внутреннего ресурса.
- *TVariant*: Механизм хранения для строки изменён в хранение указателя на массив в "union" с адаптивным механизмом выделения памяти (простая, вплоть до 7 символов хранится статически, а под более память выделяется динамически).
- Сборочная система была значительно обновлена на предмет:
 - Добавлена проверка и отключение использования "Iconv" и "Intl" (I18N) API.
 - Добавлено автоматическое обнаружение и включение параметра конфигурации *--enable-CrossCompile* для окружения кросс-компиляции.
 - Добавлен параметр конфигурации *--enable-CoreLibStatic* для отключения сборки разделяемой библиотеки ядра OpenSCADA и статической линковки программы вызова-запуска.
 - Добавлена сводная таблица конфигурации, в конце.
 - Перемещение кода обработки I18N в файл I18N.mk и включение его непосредственно в Make-файлы, по надобности, на последней стадии конфигурации.
 - Добавлен общий макрос AX_MOD_EN() для унифицированного контроля модулей.
 - Добавлена установка включаемых файлов (заголовки).
 - Добавлен файл openscada.pc для проверки конфигурации OpenSCADA при построении внешних модулей.
 - Глобальная версия исходных файлов OpenSCADA изменена в 0.8.0 и версия библиотеки ядра OpenSCADA изменена в 2.0.0.
- Реализация манипуляции объектами через атрибуты параметров:
 - Реализована общая поточная безопасность (threadsafe) для объектов пользовательского API: "TVarObj", "TArrayObj" и "XMLNodeObj". Добавлено автоматическое удаление объекта в "AutoHD" при возврате функцией AHDDisConnect() — "true".

- Добавлена обратная сериализация (разбор XML представления объекта) для объектов "TVarObj", "TArrayObj" и "XMLNodeObj". Включено для "TVariant", "TVal", "TValFunc", "UI.VCAEngine.Widget", что позволяет сохранять и загружать значения типа Объект в БД.
- *TVal*, *DAQ*.{*BlockCalc,JavaLikeCalc,LogicLev,ModBus,Siemens*}, *UI.VCAEngine*: Добавлена поддержка типа значения "Объект". Включено для повсеместного использования.
- *DOC*:
 - Выполнена проверка и исправление Английских текстов в ядре OpenSCADA. Обновлено переводы ядра OpenSCADA на Немецкий, Русский и Украинский языки.
 - Выполнена проверка и исправление Английских текстов модулей OpenSCADA. Обновлено переводы модулей OpenSCADA на Немецкий, Русский и Украинский языки.
 - Обновлено все основные документы на доступных языках: openscada.pdf, build.pdf, properties.pdf, "OpenSCADA API", "Про OpenSCADA", "Библиотека TechApp", "Quick Start", "WLib Основные", "WLib Элементы мнемосхем".
- *Базы данных библиотек*:
 - В библиотеку основных визуальных элементов добавлено:
 - Кадры реализации редактирования и исполнения "Рецептов" — "Пользовательских программ".
 - Диалог выбора и конфигурации параметров для кадра "Группа Графиков".
 - Добавлены некоторые улучшения в кадр "Группа Графиков" для выбора параметров пользователем и другого.
 - В кадр "Группа Графиков" добавлен горизонтальный скрол-бар обзора архивов.
 - К комплекту пакета библиотек добавлено [библиотеку элементов Электрических схем](#).
 - Кадр главной страницы дополнен механизмом "проигрывания" моделей технологических процессов (ТП).
 - Модель [промышленного котлоагрегата](#) переведено на Английский и Украинский языки.
- *DB*: Добавлено временное ограничение в 5 секунд и поле указания начального смещения для запроса содержимого больших таблиц.
- *Transport*:
 - Добавлена функция *TTransportS::traf2str()* для преобразования значения счётчика трафика в строку вроде "12.5КиБ".
 - Типы форматов исходящих запросов расширены вариантами текста с разным завершением строки: LF, CR, CR/LF.
- *DAQ*:
 - Добавлена возможность выбора типа атрибута шаблона "Объект".
 - *DiamondBoards, ModBus, LogicLev*: Добавлена возможность смены типа параметра для отключенных параметров многотиповых модулей подсистемы "Сбор данных".
 - *LogicLev, ModBus, Siemens*:
 - В случае выполнения по расписанию в атрибут "f_freq" записывается время после последнего вычисления в отрицательном значении.
 - Добавлены инициализирующие и останавливающие вызовы при включении/выключении параметра.
 - Добавлено обновление атрибутов параметра логического типа при изменении типа, флага "ТолькоЧтение" или имени в шаблоне.
- *TArchives*:
 - Добавлена возможность использования регулярных выражений при поиске по категории посредством "/match/"
 - *DBArch, FSArch*: Добавлена временная граница при запросе сообщений из архива.
 - Адрес ссылки на атрибут параметра упрощён к виду "DAQMod.Cntr.Prm.attr". Поддержка старых адресов сохранена.

6 Публикация решений OpenSCADA

В процессе осуществления работ над данным релизом на разных конференциях и выставках были представлены решения на основе проекта OpenSCADA:

- [FOSS Sea 2011](#) — на конференции был зачитан доклад о решениях на основе OpenSCADA, а в холле был организован мини-стенд с решениями на основе OpenSCADA вживую.
- [Международный инновационный форум 2011](#) — в составе стенда ООО НИП "ДІЯ" и Днепродзержинского Государственного Университета (ДГТУ) было представлено оборудование и материалы с решениями на основе OpenSCADA.
- [10 Всеукраинская Конференция разработчиков и пользователей свободного ПО](#) — на конференции был зачитан доклад о решениях на основе OpenSCADA, а в холле был организован мини-стенд с решениями на основе OpenSCADA вживую.
- Дни разработчиков QT [Qt Developer Days 2011](#) в Мюнхене и Сан-Франциско: в рамках этих мероприятий в Qt Demo Pavilion состоялась демонстрация OpenSCADA посредством представления решений [Динамическая модель парового котла №9 ДМК](#) на ПК и [Динамическая модель реального времени Анастасиевской ГЛКС](#) на Nokia N9.

Заключение

В процессе создания нового стабильного релиза 0.8.0 промышленного назначения с продолжительным сроком поддержки проделана большая работа по стабилизации, расширению функциональных возможностей, а также адаптации для работы на альтернативных аппаратных платформах (на архитектуре ARM). Всё это в целом позволило расширить рамки полноценного применения OpenSCADA на все уровни систем автоматизации от ПЛК, серверов SCADA-систем и до распределённых систем визуализации.

В появлении новой промышленной версии системы OpenSCADA 0.8.0 активное участие приняли:

- [Савоченко Роман](#): Основной объём работ по разработке, документированию и тестированию.
- [Лысенко Максим](#): Сопровождение и стабилизация примитива "Элементарная фигура" среды визуализации и управления (СВУ), а так-же перевод и обновление большей части документации на Английский язык.
- [Попкова Ирина](#): Перевод и обновление интерфейса системы OpenSCADA и её модулей на Немецкий язык.
- [Алмаз Каримов](#): Расширение модуля поддержки протокола DCON и активное тестирование.
- [Попков Алексей](#): Сборка пакетов для Fedora-based дистрибутивов.
- Многие другие пользователи системы OpenSCADA посредством всестороннего тестирования.

Последующие усилия разработки будут направлены на реализацию программной многоплатформенности, а также расширение функциональности путём создания модулей поддержки новых интерфейсов, в том числе и по схеме авансированной разработки, с распределением затрат реализации на проданные экземпляры (для модулей взаимодействующих с коммерческими системами и интерфейсами). Для реализации централизованного распространения решений, услуг и расширений на основе OpenSCADA планируется создание магазина в рамках главного информационного ресурса OpenSCADA (<http://oscada.org>).

Ссылки

Протокол тестирования промышленного релиза: <http://wiki.oscada.org/Works/Tests/release080>