

# OpenSCADA — целевая стабилизация

Савоченко Роман Алексеевич

OpenSCADA Team

<http://oscada.org>

## Аннотация

Разработка проекта OpenSCADA перешла в фазу глубокой стабилизации с элементами расширения существующего функционала нацеленную на предоставления стабильного и надёжного окружения автоматизации промышленных и смежных задач. В докладе освещены вопросы текущей политики разработки решений продолжительной поддержки OpenSCADA в сочетании с системным окружением платформы, а также общие достижения проекта за прошедшее время.

## Введение

В текущем году проект OpenSCADA тихо отметил десятилетие основания переходом в фазу глубокой стабилизации. Данное решение в значительной мере связано с активизацией работ по адаптации и, как следствие, всесторонним тестированием с выявлением и устранением специфических проблем.

Часто под стабилизацией многие пользователи и разработчики свободного ПО подразумевают фактическую остановку разработки, а полную стабилизацию — смерть. Это мнение во многом связано с сутью СПО — разработка ради удовольствия «Just For Fun». Поскольку процесс стабилизации, или поддержки, в течении продолжительного времени привязывает собственное детище к разработчику, а в случае с непоступательным развитием и расплывает силы, то удовольствия и времени остаётся мало. Такое положение дел, во многих случаях, породило состояние непрерывного процесса разработки, гонки за новыми сомнительными функциями, с фактической смертью старых версий на этапе достижения максимального уровня их стабильности и началом работ над новой, нестабильной, версией, а иногда и разработкой нового решения прямо перечёркивающее старое. Это явление очень часто характеризуется наличием множества отчётов о проблемах, что указывает на востребованность, но полное отсутствие реакции разработчиков.

На этапе сборки СПО в дистрибутив, дистрибьюторы фактически вынуждены заниматься поверхностной стабилизацией последних, нестабильных версий, а иногда и поддержкой предыдущих стабильных версий, что часто можно сравнить с доработкой незаконченного изделия напильником. Как результат решения-дистрибутивы получают недостаточного уровня стабильности, а пользовательской аудиторией их остаётся класс продвинутых пользователей, способных настроить, обойти, исправить или смириться с проблемами. Если на рабочем столе пользователя такие решения всё-же допустимы, то на компьютерах ответственного назначения, например, в пультах операторов и диспетчеров систем промышленного контроля это недопустимо.

Очень ярко проблема отсутствия стабильности и продолжительной поддержки видна на окружениях рабочего стола (DE), где многие производители решений для конечных пользователей крайне редко берут за основу существующий DE, часто берут за основу существующий графический Toolkit (GTK: Maemo, QT: MeGoo) с графическим стеком, а порой даже создают совершенно новое окружение с собственной поддержкой (Android). И это когда доступны исходные тексты решений и в рамках них уже проделана огромная работа!

Хорошим правилом и эффективной практикой является естественный путь стабилизации ПО, совмещаемый с поступательным развитием. А именно участие целевой аудитории в стабилизации, путём всестороннего-естественного использования, обратной связи и оперативной реакции разработчиков. Однако мотивации для последнего в СПО отсутствует и необходимо совмещение СПО с внешним финансированием, обычно через пожертвования и/или с предоставлением коммерческого сервиса. По пути коммерческого сервиса, а также организации сбора пожертвований и целевого финансирования разработки пошёл и проект OpenSCADA.

## Стабилизация и развитие OpenSCADA

Командой разработчиков уделяется значительное внимание стабилизации OpenSCADA, как через оперативность реакции на уведомления, так и в рамках целевого сервиса. В целом хотелось бы с гордостью отметить отсутствие, на данный момент, открытых уведомлений и оперативность устранения ошибок непосредственно в коде проекта.

В рамках стабильного релиза продолжительной поддержки 0.8.0 LTS было исправлено порядка 150 ошибок и выпущено 8 обновлений с циклом два месяца. Кроме непосредственно исправлений в стабильную ветвь внесено множество изменений по упрощению и расширению, которые не приводят к нарушению обратной совместимости API и ABI модулей.

Значительное внимание в стабилизации и укреплении было уделено через анализ производительности, использования оперативной памяти, ревизии и упрощения избыточного кода и общей эффективности работы. Для фиксации результатов этих мероприятий в базе знаний проекта создан специальный раздел расширенного тестирования, анализа и оптимизации.

Отметим наиболее важные исправления и изменения в LTS ветке:

- *Документация*: Завершен документ "Создание модулей OpenSCADA". Значительно обновлен и дополнен документ "Быстрый старт OpenSCADA" и создано видео-дополнение к нему. Создано руководство оператора-пользователя, на основе модели «АГЛКС».
- *БД*: Значительные расширения, исправления и оптимизация генерации документов по времени, а так-же реализация новой концепции движка рецептов.
- *Системные*:
  - Реальное не создание задачи из высокоприоритетного потока на некоторых Linux ядрах, для примера 2.6.29-rt, исправлено заменой ожидания вызова sched\_yield() на TSYS::sysSleep(1e-3).
  - Заменена немногочисленных функций: Функция вычисления хеша пароля crypt() заменена на crypt\_r(). Функция "getopt" полностью заменена на собственную функцию TSYS::getCmdOpt(). Скрыто-проблемная функция readdir() заменена на многопоточный вариант readdir\_r().
  - Системные атрибуты MessLev, LogTarget, Lang, Lang2CodeBase, WorkDB, Workdir, IcoDir, ModDir разрешены для чтения и записи только в конфигурационный файл.
  - В функцию создания задач добавлен контроль и освобождение завершённых, но с присутствующим объектом, неотсоединённых задач.
  - Адаптация к утраченной поддержке односимвольных кодировок, проверено на KOI8-R.
  - Добавлен контроль на возможность редактирования для конфигурационных полей различных объектов, зависимо от состояний "Включен" и "Запущен".
  - Пользовательские объекты: Исправлена бесконечная рекурсия при установке значения строки в тип Объект. Хранение ссылки в XmlNodeObj на родителя изменено на прямой указатель для предотвращения самоблокировки и утечки памяти. Добавлен объект TEValObj для подобной к скалярам обработки EVAL.
  - Добавлена функция пользовательского API SYS.mktime() для получения времени с эпохи 1.1.1970 с конвертацией из разделённого времени.
  - Функция пользовательского API XmlNodeObj::childIns() исправлена для обнаружения аргумента 1 типа XmlNodeObj.
  - DAQ.Templates,Protocol.ModBus: Добавлено использование предыдущей строки как шаблона при создании новой. Добавлена нумерация строковых меток посредством strLabEnum().
  - SYS.{WorkDir,IcoDir,ModDir},Archive.FSArch,DB.SQLite,Protocol.HTTP,Transport.Serial: Функция ctrlListFS() добавлена и использована для навигации по директориям и файлам в конфигурационных полях.
- Подсистема «БД»:
  - DB.SQLite: Добавлено закрытие открытой транзакции при остановке модуля с целью

предотвращения потери данных при падении некоторых модулей на отключении.

- Подсистема «Транспорты»:
  - Transport.{Serial,Sockets}: Добавлена обработка номера ошибки EAGAIN в функции write() для исходящих транспортов с целью предотвращения ошибок на передаче больших блоков данных.
  - Transport.Serial: Добавлено конфигурационное поле "Приоритет" для задачи входного транспорта.
- Подсистема «Сбор данных»: Исправлена утечка памяти в объектах деструктора Tval::~TVal(). Ограничение исходного текста процедур расширено до 1000000 символов. Для модулей по шаблону предотвращена перегрузка исполнительного контекста из БД на запущенных объектах контролеров.
  - DAQ.System: значительно обновлен и улучшен для создания новых объектов источников при их появлении.
  - DAQ.SoundCard: Переписан сбор для обеспечения реального времени, а также контроля частоты выборки и потери кадров. Добавлена возможность установки приоритета задачи сбора, из PortAudio. Добавлены некоторые сообщения об ошибках. Несколько увеличена производительность.
  - DAQ.DCON: Значительные изменения от Алмаза Каримова: добавлено чтение аналоговых входов одним методом; исправлена работа метода 1AO (#AA) (блоки NL-1AO, I-7021, ADAM-4021); добавлен метод 4AO (#AACN) для работы с блоками ADAM-4024; метод чтения дискретных входов @AA заменён на \$AA6 для совместимости с блоками ADAM; методы записи дискретных выходов 4DO (#AA000F), 8DO (#AA00FF), 12DO (#AA00FFF) добавлены для работы с блоками ADAM.
  - DAQ.JavaLikeCalc: В язык добавлено разделение назначений символом ';' для отделения операций внутри оператора «for». Удаление ошибочного правила грамматики "expr OR CONST{". Условная логическая команда '||' и '&&' исправлена для постоянных.
  - DAQ.ModBus: Исправлена ошибочная установка EVAL для атрибута по идентификатору ссылки, в случае ошибки записи по этой ссылке. Добавлена поддержка комплексного типа "Строка", блок регистров, в DAQ посредством суффикса "\_s" к регистру "R[I]". Добавлена поддержка комплексных типов "Целое" (32 бит, знаковый), "Вещественное" и "Строка" (блок регистров) в объект узла входящего протокола.
  - DAQ.{BlockCalc,LogicLev,ModBus},Archive.TArchVal,UI.VCAEngine: Навигация по атрибутам параметров и самим параметрам перемещена в общую функцию TDAQ::ctrlListPrmAttr() и использовано везде.
  - DAQ.OPC-UA,{Lib,Protocol}.OPC-UA: Часть протокола OPC UA для клиента и сервера перемещена в LGPL v3 библиотеку для упрощения использования другими проектами. Объект контроллера упрощён на предмет выбора Конечных Узлов (EndPoints) и транспортов; Вы можете просто установить Конечную Точку, а транспорт будет создан из неё. Для включенного объекта контроллера Вы получите перечень доступных Конечных Узлов. TokenId и SessionId адаптированы к комплексному типу NodeId и проверено на Ignition OPC UA сервере. OPC UA сервер исправлен на предмет обработки объединённых входных сообщений. Обработка основных атрибутов узлов в значительной степени улучшена и расширена.
  - TVal: Добавлена остановка архива при отключении объекта атрибута для разблокирования и завершения действия.
- Подсистема «Архивы»: Исправлено включение архива из DAQ параметров и падения архива значений, значительно улучшена обработка архивов. Добавлен запрос по нужной периодичности для обеспечения быстрого доступа к архивам значений высокого разрешения, посредством указанного размера кадра и периодичности. Предотвращено в Archive.FSArch разрушение архивных данных посредством перекрытия памяти буфера индексов, исправлено помещение больших сообщений в архив плоского текста, при достижении лимита на размер буфера.

- Подсистема «Пользовательские интерфейсы»:
  - UI.{VCAEngine, Vision, WebVision}: Расширен верхний предел количества параметров примитива "Диаграмма" до 20, реализовано разделение шкал отдельных параметров, добавлено формирование трендов из прямо указанных данных посредством префикса "data:" и "line:". В примитивах "FormEl", "ElFigure", "Media", "Box" использованы определения индексов из "types.h".
  - UI.{Vision, WebVision}: Исправлен завал буфера данных примитива "Диаграмма" текущими значениями и потеря старых видимых значений при достижении ограничения буфера. Исправлена загрузка конфигурации элементов примитива "ElFigure" с учётом требования присутствия всех точек. Функция attr{s}Set() примитива "ElFigure", и "FormEl" в UI.Vision, полностью пересмотрена и оптимизирована.
  - UI.{Vision, VCAEngine}: Добавлен вариант обзора элементов "Дерево" в примитиве "FormEl". Добавлена одно-строчная навигация ресурсов и файлов для image/media атрибутов. Таблица "Mime" контейнеров разрешает прямую загрузку, а не только для выбранных элементов, при этом "mime" берётся из расширения; "mime" также использован при выгрузке файлов. Установлен лимит на размер загружаемых файлов.
  - UI.VCAEngine: Добавлена загрузка/включение виджетов библиотек и проектов по требованию. Лимит на архив примитива "Документ" исправлен на расширение до 1000000. Экспортные символы перемещены в файл types.h и предназначен для включения визуализаторами. Добавлено изменение части адреса родителя вложенных виджетов при изменении родителя библиотечного виджета или страницы, при включении.
  - UI.WebVision: Вычисление времени периода обновления переписано для обеспечения большей отзывчивости и адаптации к реальным свойствам сетевого канала. Добавлена попытка загрузки внешнего JS файла "WebVisionVCA.js" из текущей директории. Изменения в JS коде: значительная очистка и исправления; добавлена проверка на нулевую прозрачность для отключения фона; попытка использование асинхронных запросов в основном цикле; примитивы "ElFigure", "Media", "FormEl" в значительной степени адаптированы для предотвращения полной замены контекста на динамических изменениях одиночных свойств; отключено схватывания для изображений в FireFox. ID объекта сеанса переименован в PATH и nodeName() установлен в строку с разделителем от пути. Добавлена опция уровня сжатия PNG для некоторого контроля производительности формирования изображений, а значений по умолчанию установлено в "1".
  - UI.Vision: Добавлена проверка на реальную видимость части виджета (по непрозрачности пиксела). UI.Vision: Добавлена функция WdgView::attrsSet() и использована для установки группы атрибутов в одном запросе для предотвращения потери некоторых связанных событий при обработке, а также повышения производительности в сети. Изменения и исправления примитива "ElFigure": добавлено и использовано определение диапазона значений; добавлено перечисление ShapeItem::Type для фигур "Линия", "Дуга" и "Безье"; добавлено копирование индексов "Ширин", "Стилей" и "Цветов" при копировании фигур. исправлено разделение точек 5 нескольких дуг при перемещении клавиатурой и мышью; добавлено перемещение курсора мыши с перемещением клавиатурой; добавлен разрыв контура окружности перемещением с клавиатуры; исправлено падение примитива "ElFigure" при нажатии на общие точки 1,2 дуги и точку 1 линии; значительная очистка кода. Исправлено переформирование вектора index\_argau для точек 1 и 2 для помещения только общих фигур единой (исключая Дугу, в случае объединения точек 1 и 2). Исправлено падение на Линиях и Дугах (предыдущее ошибочное исправление). Примитив "ElFigure" исправлен на предмет масштабирования и размасштабирования толщин путём перемещения масштабирования прямо на момент отрисовки фигур.
- UI.WebCfg: Исправлено отображение поля "Дата+Время" в таблице и других местах.
- Подсистема «Специальные»: Spec.ModSched: Поля "ModAllow" и "ModDeny" открыты для редактирования и сохранения в конфигурационном файле.

Хотя с момента выпуска релиза продолжительной поддержки 0.8.0 LTS прошло более года и с того момента ещё не было ни одного релиза рабочей ветки, работа над версией 0.8.1 активно ведётся и в

неё включено ряд знаковых изменений, которые по причине нарушения совместимости не могут быть перенесены в LTS версию. В целом нужно отметить, что рабочая ветвь проекта также подверглась духу стабилизации и разработка осуществляется в ключе улучшения и расширения старых модулей и функций, создания новых модулей, а сборки рабочей ветки используются в создании новых ответственных решений.

Отметим наиболее знаковые улучшения в рабочей ветке, которые несовместимы с LTS версией и не включены в неё:

- *DB*: Значительные расширения и дополнения библиотек процедур и визуальных элементов OpenSCADA.
- *SYS*: Переписан скрипт запуска проектов рабочих станций OpenSCADA — «opencada\_start» для хранения всех данных проекта в отдельной директории; возможность запуска проекта указанного в переменной окружения «OSCADA\_ProjName»; возможность запуска проекта указанного в имени ссылки «opencada\_{ProjName}»; возможность выбора и создания нового проекта в диалоге от «dialog», «kdialog», «zenity» и «Xdialog».
- *SYS*: Сборочная система расширена для установки всех необходимых данных программы (прямо через «make install»). Путь установки данных изменён на «/usr/share/opencada». Из конфигурационного файла удалены все пути, а указанные при конфигурации и сборке установлены как по умолчанию прямо в библиотеке ядра OpenSCADA. Оставшиеся конфигурационные переменные устанавливаются в файлы конфигурации через подмену «\*.in». Унифицированы цели создания архивов дистрибутива. Упрощён и унифицирован механизм включения модулей в ядро программы.
- *SYS*: Изменён механизм формирования отладочных сообщений. Сообщения отладки включаются установкой уровня сообщений в системе — «Отладка (0)» и выбора во вкладке главной страницы «Отладка» генерации сообщений нужных подсистем. Из общего пути локального объекта OpenSCADA удалён идентификатор станции.
- *SYS*: Реализован новый механизм окончания исполнения системных задач, основанный на сигнале SIGUSR1 и проверке функцией TSYS::taskEndRun().
- *DAQ*: Добавлена функция «Диагностика», позволяющая в соответствующей вкладке источника данных выбирать уровень диагностики, а уровнем «Отладка (0)» включать режим отладки запросов и реакции на них.
- *DAQ.JavaLikeCalc*: Повышение производительности внутренних вычислений за счёт сохранения контекста вычисления и перевода алгоритма проверки таймаута безопасности на сравнения с внешней переменной времени.
- *DAQ.ICP\_DAS*: Фактически полностью переписан на предмет поддержки большинства модулей серии I8k и I87k, а также плат на шине ISA.
- *DAQ.Comedy*: Новый модуль поддержки различных DAQ-плат посредством библиотеки <http://comedy.org>, на шинах ISA, PCI, USB.
- *DAQ.DiamondBoards*: Полностью переписан для поддержки большинства плат DAQ фирмы «Diamond Systems», а также реализованы все три механизма сбора данных: синхронный, асинхронный и пакетный.
- *DAQ.OneWire*: Новый модуль первичной поддержки сенсоров на шине 1-Wire.
- *DAQ.SMH2Gi*: Новый, специализированный модуль реализации прямого опроса УСО контроллера SMH2Gi от Segnetics.
- *DAQ.DAQGate*: Добавлена возможность шлюзования сообщений, генерированных обслуживаемыми источниками данных.
- *UI*: Функции получения иконок и ресурсов графических интерфейсов icoPresent(), icoGet() и icoPath() объединены в одну функцию icoGet() и адаптировано к обработке нескольких директорий. Директории иконок по умолчанию установлены в "icons/;@oscd\_datadir\_full@/icons/".
- *UI.{VCAEngine,Vision}*: Количество возможных параметров примитива «Диаграмма» увеличено до 100. Вид «Кнопка» примитива «Элементы Формы» расширен режимами: «Стандарт-

ный», «Выбор», «Меню», «Загрузка», «Сохранение».

- *UI.VCAEngine*: Генерация документа в примитиве «Документ» перемещена в отдельную задачу с целью предупредить блокирование основной на момент обработки. При обработке документа результат обновляется каждые 5 секунд. Атрибут "process" добавлен для контроля окончания.
- *UI.Vision*: Возможность использование WebKit для рендеринга примитива «Документ»

## **Общая стабилизация окружения исполнения — дистрибутив Linux с OpenSCADA**

OpenSCADA — среда работы с динамическими данными в реальном времени, сама работает в окружении ОС Linux, а также графического стека, рабочего стола и с использованием различных свободных библиотек. Следовательно общая стабильность конечного комплексного решения зависит от слаженной работы и стабильности программного комплекса в целом. По этой причине командой разработчиков OpenSCADA уделяется значительное внимание стабилизации окружения исполнения OpenSCADA путём создания стабильного репозитория пакетов, на основе ветви дистрибутива ALTLinux T6, а также непосредственного участия в стабилизации отдельных частей этого окружения.

Во введении говорилось об остроте проблемы стабилизации и продолжительной поддержки свободного окружения ОС Linux, особенно для ответственных задач. Однако наблюдаются положительные тенденции по переходу на схему выпуска LTS-версий как отдельных программ, так и дистрибутивов в целом:

- «RedHat Linux» (Centos, Scientific Linux) — выпускает стабильные LTS дистрибутивы, нацеленные на серверное применение.
- «Ядро Linux» — отчётливо выражено выделение LTS релизов и продолжительная их поддержка.
- «Ubuntu» — в ряду продуктов явно фигурируют, выпускаются и поддерживаются стабильные дистрибутивы.
- «Trinity DE» — поддержка и развитие кодовой базы KDE3, содержит явно выраженную LTS ветвь.
- «KDE4» — запланирован переход на LTS версии 4.11, хотя уже на горизонте просматривается 4.12.
- Web-браузеры и ряд другого прикладного ПО.

Однако в окружении ОС Linux всё ещё остался ряд компонентов, политика разработки которых не предусматривает LTS, что часто приводит к проблемам при согласовании с оборудованием. Особенно остро эта проблема стоит с XOrg.

Команда OpenSCADA непосредственно принимает участие в следующих проектах свободного окружения SCADA-системы:

- «Linux kernel» — для ряда задач программируемых логических контроллеров (ПЛК) осуществлялась тонкая адаптация и сборка ядра реального времени, на основе ядра 2.6.29 и PREEMPT\_RT патчей.
- «TDE» — для предоставления отлаженного, нетребовательного и развитого окружения рабочего стола TDE в репозитория ALTLinux был реанимирован, исправлены критические ошибки и осуществляется последующая поддержка и сборка LTS выпусков.

## **Заключение**

Разработка проекта OpenSCADA сейчас в значительной степени направлена на целевую стабилизацию и расширение, в следствии чего долгосрочные планы значительно пересмотрены. Дата выпуска следующего стабильного релиза продолжительной поддержки 0.9.0 LTS перенесена на Май 2014 года, а время жизни текущего стабильного релиза продолжительной поддержки 0.8.0 LTS продлена до этого срока.

Несмотря на приоритет в стабилизации, разработка также продолжается. О чём свидетельствуют значительные расширения и упрощения в рабочей ветке проекта, которые не могут быть перемещены в стабильную ветвь, но заметно улучшающие программу в целом. Разработка стала более направленной, а из востребованных задач был сформирован пул для желающих поучаствовать в проекте и в целом для формирования перспективы разработки. Подведение черты под разработкой запланировано выпуском первого релиза рабочей ветки 0.8.1 к концу этого года.

После выпуска релиза рабочей ветки 0.8.1 запланировано значительное обновление представительских материалов (логотип, дизайн, ...), открытие программ целевого и общего финансирования для привлечения в проект больше разработчиков, которых на данный момент немного, но растущая заинтересованность в проекте позволяет говорить о перспективности и значительном потенциале этих мероприятий по ускорению стабилизации, развитию и адаптации на различные платформы. Также, после выпуска 0.8.1, планируется приступить к запланированной адаптации и сборке OpenSCADA для альтернативных программных платформ.

Команда разработчиков OpenSCADA благодарна Вам за внимание к проекту и приглашает к участию и сотрудничеству в решении различных задач повсеместной автоматизации!