

OpenSCADA 0.9 LTS

Савоченко Р. А.

OpenSCADA Team <oscada@oscada.org>

Анонс новой LTS версии 0.9 проекта OpenSCADA — открытая Система Диспетчерского Контроля и Сбора Данных. Новая стабильная версия является результатом шести лет разработки, внедрения в различных решениях, стабилизации и эксплуатации OpenSCADA в её рабочей ветви. И всё это время продолжали выпускаться обновления к 0.8.0 LTS и что будет осуществляться для анонсированной 0.9 LTS.

1 Введение

Релиз OpenSCADA, открытой SCADA(Supervisory control and data acquisition) системы, версии 0.9 является стабильным промышленным релизом продолжительной поддержки (LTS).

Основной целью релиза является предоставление обновленной и стабильной платформы построения решений комплексных систем автоматизации и других смежных решений для сообщества пользователей и разработчиков свободного программного обеспечения. Также, релиз предназначен стать актуальным и прочным фундаментом в построении коммерческих решений.

Релиз является следующей версией стабильной ветви для которого, в течение продолжительного времени, предоставляется техническая поддержка от разработчиков и обеспечивается выпуск исправлений в виде публичных сборок для основных и стабильных окружений Linux, а также оперативных сборок для владельцев пакетов технической поддержки. Жизненный цикл предыдущего релизу 0.8.0 LTS прекращается его последним обновлением, непосредственно перед первыми сборками пакетов 0.9 LTS.

Именно на этой версии осуществляется окончательный переход на схему разработки Work/LTS, т.е., разработка осуществляется в рамках рабочей версии и на её основе периодически выпускаются стабильные релизы, которые, в свою очередь и параллельно разработке, обновляются путём обратного портирования совместимых изменений из рабочей версии. Иницирующий переход произошёл в 2013 году, когда текущей стабильной версией определено 0.8.0 LTS, а рабочей 0.9 Work. На данный момент, стабильной версией становится анонсированная 0.9 LTS, а рабочей 1 Work.

В целом, новая стабильная версия является результатом шести лет работы с разработки, внедрения в различные решения, стабилизации и эксплуатации OpenSCADA в её рабочей ветви, которая на данный момент выпускается текущей стабильной. Всё это время обновления для 0.8.0 LTS продолжали выпускаться и что будет осуществляться для анонсированной 0.9 LTS.

1.1 Общая информация

Проект OpenSCADA основан Савоченко Романом в 2003 году как свободную реализацию Системы Диспетчерского Контроля и Сбора Данных (SCADA) или Человеко Машинный Интерфейс (HMI), на основе её фундаментального проектирования в течении 2002 года и опыта использования и разработки коммерческой SCADA-системы до этого.

SCADA или HMI системы в целом предназначены и используются для осуществления человеком оперативного контроля за работой сложного и ответственного технологического оборудования и процессов различных производственных предприятий.

На время анонса 0.9 LTS, OpenSCADA является развитой SCADA/HMI системой, которая довольно широко используется как по прямому назначению, так, благодаря своей гибкости, и во многих смежных областях, и которая с уверенностью может называться более обобщённо — динамическая система работы с данными реального времени.

Номер стабильной версии OpenSCADA меньше единицы только из-за того, что её целью была заявлена многоплатформенность, достичь которой в запланированном объёме планируется следующим стабильным релизом с номером 1. В целом, целями OpenSCADA являются:

- **открытость** — в основном GPLv2;
- **масштабируемость, гибкость, расширяемость** — модульность и внутренняя динамичность;
- **исполняемая избыточность** — резервирование;
- **доступность** — открытые исходные тексты; автоматическая сборка архивов, пакетов, живых дисков, ...; многоязычность; динамическая многоязычность; сборка и исполнение на Linux окружениях широкого возрастного диапазона, от 2002 года (2009-[ALTLinux 6](#), [LP8x81](#), [Fedora 12](#); 2012-[Debian 7](#)) по современные;
- **надёжность** — практическое применение, быстрое решение проблем;
- **безопасность** — распределение прав, SSL;
- **многоплатформенность** — x86_32, x86_64, ARM, Web, Android, QNX ([предадаптировано](#)), MS Windows ([запланировано](#));
- **единый, унифицированных, удобный и развитый интерфейс пользователя** — Qt, Web, прозрачное и многоуровневое удалённое управление;
- **широкий спектр источников данных** — универсальные, платы DAQ и низкоуровневые шины, созданные в окружении OpenSCADA.

1.2 Применение

К актуальным отраслям применения, где известно хотя-бы одно внедрение и с коротким перечнем которых от участников проекта можно ознакомиться по [этой ссылке](#), относятся:

- [АСУ ТП \(SCADA/HMI\)](#) или системы телемеханики — основное направление и тут существует множество внедрений;
- [динамические модели, имитаторы и тренажеры](#) технологических процессов в реальном времени;
- [станки и промышленные роботы](#);
- [системы сельско-хозяйственной диспетчеризации и управления, птичники](#);
- [встроенные и мобильные системы](#) — окружение исполнения Программируемого Логического Контролера (ПЛК), роботы, ...;
- [мониторинг оборудования серверов](#);
- [умные дома и автоматика зданий](#).

С некоторыми ограничениями и доработками, в основном во внутреннем окружении пользователя, OpenSCADA может использоваться и в таких отраслях:

- управление ресурсами предприятия (ERP);
- Гео-локация и отслеживание положения;
- торговые системы;
- медицинские диагностические системы;
- бухгалтерский учёт;
- билинговые системы.

2 Результаты предыдущего релизу 0.8.0 LTS

Релиз 0.8.0 LTS выпущен в апреле 2012 года и в течении этих шести лет он претерпел 20 обновлений, которыми, в целом, исправлено более 500 ошибок и добавлено множество усовершенствований, которые не нарушают совместимости библиотечных БД и конфигурации.

Конфигурация и библиотечные БД 0.8.0 LTS были в целом заморожены, ввиду больших несовместимых изменений в 0.9 Work, для методов запуска OpenSCADA, и несовершенство распространение библиотечных БД на момент выпуска 0.8.0 LTS. Поэтому, обновление и переход на версию 0.9 LTS будет нетривиальным, хотя в 0.9 LTS предусмотрено всё возможное для упрощения [этой процедуры](#). Обновление-же 0.9 LTS до запланированной в будущем 1 LTS уже не будет таким сложным и фактически может превратиться

исключительно в формальное изменение версии рабочей ветви, поскольку теперь планируется обновление всего, включая и библиотечные БД.

3 Плановые задачи релиза

Развитие OpenSCADA, после предыдущей LTS версии и в рамках рабочей ветви, в основном осуществлялось путём глубокой стабилизации и через практическую адаптацию с элементами расширения существующего функционала, нацеленные на предоставление стабильного и надёжного окружения автоматизации промышленных и смежных задач, а соответственно и не существовало чётко определённого плана. Но за три года до этого релиза такой план появился и составил следующие задачи, что видно из [общего плана развития](#):

- Полная ревизия основной документации и подготовка анонса этого релиза.
- [Адаптация к работе на программной платформе "Android"](#).
- Перенос базы знаний OpenSCADA и Wiki-ресурса документирования на новый движок с унификацией структуры под многоязычность с приоритетом: Английский, Украинский, Российский; и генерация автономной документации по её изменению.
- Ревизия, актуализация и некоторое расширение Web-модулей OpenSCADA.
- Расширение и адаптация OpenSCADA к прямой работе с низкоуровневыми шинами и устройствами вроде 1Wire, I2C и реализация проекта умного дома в собственной квартире.
- Формальное создание и документирование [проекта дистрибутива Linux автоматизации с OpenSCADA](#).
- Создание автоматизированной системы сборки пакетов OpenSCADA.
- Перенос серверной инфраструктуры OpenSCADA на [собственное оборудование и канал интернета](#).
- [DAQ.OPC-UA](#): Упрощение, увеличение функциональности и перенос протокольного кода в отдельную LGPL v3 библиотеку.

Эти задачи выполнены и некоторые детали около них приведены далее.

4 Общесистемные свойства

Новая стабильная версия OpenSCADA получила заметные общесистемные расширения, приобрела большей стабильности в работе и увеличение продуктивности, а также получила заметные усовершенствования графического интерфейса и окружения около неё, вроде: расширенной настройки, документации, доступной прямо из программы (offline и online), и которая значительным образом пересмотрена и актуализирована.

Основная часть документации проекта перенесена на новую Wiki, на основе MediaWiki, и значительный объём этой части был отревизирован и переведен на три языка — Английский, Украинский, Русский. Для переноса была создана [процедура конвертации статей из диалекта движка WackoWiki на MediaWiki](#), которая была полностью написана на внутреннем языке программирования OpenSCADA и которую многократно использовано для переноса большого объёма старой Wiki. Формат автономной (offline) документации был изменён со статичных PDF-файлов, которые не обновлялись после предыдущей LTS версии, на HTML-файлы, которые динамически генерируются из актуальной базы знаний проекта (Wiki), а также имеют актуальные кросс-ссылки между страницами и ссылки на online-документацию, для исключения внешних материалов. Генерация offline-документации также производится [специально написанной процедурой](#) на внутреннем языке OpenSCADA, что, вместе с процедурой конвертации Wiki-диалекта и [комплексного тестирования релиза OpenSCADA](#), является ярким показателем мощности и текущего уровня развитости внутреннего языка OpenSCADA.

В рамках старой Wiki, после выпуска 0.8.0 LTS были добавлены и обновлены некоторые документы, а с переносом на новую Wiki, кроме ревизии и перевода основной части документации, была осуществлена общая унификацию её структуры из соображений: многоязычности (Английского как первичного языка), логичности организации, удобства перевода и возможности убрать дублирование статей с официальным сайтом, которые сейчас просто используются из Wiki. Отдельно отметим значительно-расширенные главные документы:

- [Быстрый старт](#);
- [Руководство по программе](#);
- [API пользователя OpenSCADA](#);
- [ЧаВо](#) и [Как сделать ...](#);
- [Создание модуля OpenSCADA](#).

Первичный язык (Английский) был полностью отревизирован в оригинальных сообщениях программы и во многом для основных статей документации-Wiki OpenSCADA, что на данное время делает OpenSCADA адекватно воспринимаемой для аудитории, которая не понимает исходных языков — Украинского или Русского. И, вместе с реализацией в полном объёме механизма динамического перевода, делает возможным построение на основе OpenSCADA динамично многоязыковых интерфейсов пользователя, что Вы можете кроме всего прочего посмотреть на публично доступном Web-интерфейсе динамических моделей ТП: [АГЛКС](#), [Котёл](#).

Рабочей версией, в основе этой стабильной, впервые было определено по-

нятие проекту OpenSCADA и реализован сценарий командной строки для запуска и создания проектов OpenSCADA. Непосредственно перед выпуском этой версии, понятие проекта OpenSCADA было окончательно закреплено за директорией с данными отдельного проекта и конфигурационным файлом OpenSCADA, а реализация менеджера проекта была интегрировано непосредственно в OpenSCADA. Соответственно, эта LTS версия имеет развитое понятие менеджера проектов, что позволяет гибко с ними работать и исключить опасную возможность многократного запуска с общими данными одного проекту.

Рабочей версией, также, внедрено изменение версий модулей по изменению в коде модуля и непосредственно перед их выгрузкой в репозиторию исходных текстов, а соответственно, версии модулей OpenSCADA этого релиза наглядно отражают общий уровень их развитости и стабильности.

В процессе осуществления работ над рабочей ветвью, в основе этой стабильной, было внедрено формирование репозитория пакетов дистрибутивов Linux со сборками OpenSCADA, которые до этого предоставлялись только как отдельные пакеты. Что заметно упростило развёртывание OpenSCADA и содержание её актуальной. Затем был создан [автоматических сборщик этих пакетов](#), который на данный момент насчитывает до 100 целей, и что значительно упростило выпуск обновлений как для рабочей ветви так и для этой стабильной. Соответственно, к этой LTS версии предоставляются сборки пакетов для основных окружений Linux и вся историю публичных обновлений будет сохранена.

К пакетам сборок OpenSCADA предоставляются также и сборки [живых дисков](#) быстрого ознакомления и развёртывания OpenSCADA вместе с системным окружением. Которые до этого момента получили формальное основание в виде [Linux дистрибутива автоматизации проекта OpenSCADA](#).

Сборки и пакеты OpenSCADA сопровождаются рядом открытого и свободного материала внутреннего окружения программы, а именно, это: наработки слоя сбора и обработки данных, элементы графического представления и целые-комплексные проекты моделей ТП. Эти материалы предоставляются в виде файлов БД SQLite и включают:

- Библиотеки функций (OscadaLibs), "LibsDB/OscadaLibs.db" — содержит все наработки проекта OpenSCADA в слое сбора и обработки данных включая [элементы источников данных пользовательского протокола](#).
- СВУ: Главные библиотеки (vcaBase), "LibsDB/vcaBase.db" — содержит [основные элементы графического представления](#) и [элементы мнемосхем](#) проекта OpenSCADA.
- СВУ: Тесты (vcaTest), "LibsDB/vcaTest.db" — содержит элементы тестов примитивов графического представления.

- СВУ: Библиотека электро-элементов мнемосхем пользовательского интерфейса (vcaElectroEls), "LibsDB/vcaElectroEls.db" — содержит элементы графического представления компонентов электрических схем.

Хостинг проекта в целом и материалов 0.9 LTS в частности, был перенесен на [собственный сервер проекта](#), где дополнительно были развёрнуты: демонстрационные Web-интерфейсы моделей OpenSCADA, проект мониторинга сервера и умного дома на основе OpenSCADA и сборщик пакетов репозитория OpenSCADA.

4.1 Внутренние

Разрешающая способность внутренних данных целого типа окружения OpenSCADA увеличена до 64 разрядов. В целом-же, внутренние данные OpenSCADA, с резервированием значения ошибки (EVAL) для каждого, унифицированы общими типами: логическое, целое, реальное, строка и объект. Что в первую очередь касается источников данных.

В ядро OpenSCADA, [собственный протокол](#) и все узлы что работают с удалёнными станциями OpenSCADA; добавлена возможность "поднятия" узлов OpenSCADA, которые находятся за другими узлами и, как правило, в другой сети. Что в целом позволяет централизованно управлять сетью станций OpenSCADA на любом уровне иерархии.

4.2 Усовершенствования и адаптация для различных платформ

Эта версия OpenSCADA получила углублённую поддержку и способность к адаптации для различных платформ. Что в основном случилось благодаря адаптации для работы на программной платформе Android и восстановления сборки и работы с ucLibC, и что планируется использовать в дальнейшей адаптации для работы на программных платформах QNX и MS Windows.

Работа на одноплатных ПК была расширена платами [Raspberry Pi](#) и [Orange Pi](#).

Поддержка Linux смартфонов Nokia дополнена последним, основанным на MeeGo 1.2, это — Nokia N9. Или это было обновление поддержки Nokia N950.

Кроме непосредственной работы (нативно) на различных платформах, был значительно расширен Web-интерфейс, который сейчас реализует все общие возможности концепции среды визуализации и управления (СВУ).

4.3 Оптимизация, стабилизация и продуктивность

Значительной стабилизации ядра OpenSCADA, и программы в целом, достигнуто через унификацию контроля внутренних ресурсов и расшире-

ния [возможностей отладки пользователем](#). В общем, пользовательская диагностика и отладка расширена:

- общим включением-выключением отладки и целевым контролем узлов отладки;
- особенной отладкой объектов контролеров источников данных;
- особенной отладкой исполнения проектов СВУ;
- формированием углублённой статистики исполнения динамических объектов OpenSCADA, таких как: объекты контроллеров источников данных и их параметров, входных и выходных транспортов, сеансов СВУ с детализацией до виджетов;
- [протоколирование входного и выходного трафика транспортов](#).

Глубокой и всесторонней стабилизации претерпели почти все модули OpenSCADA и многие были оптимизированы, из которых особенно нужно отметить:

- [Все модули БД](#) — добавлена обработка и проверка ошибок БД или СУБД, а также выдача сообщений про эти ошибки в случае пользовательской загрузки и записи; увеличена производительность (до порядка) БД поддерживающих SQL, через реализацию режима предзагрузки сканирующего запроса;
- Вычислитель на основе Java-подобного языка ([DAO.JavaLikeCalc](#)) — увеличена производительность через: сохранение контекста исполнения функции, предзагрузку постоянных и прямой доступ к строке.
- Архиватор на БД ([Archive.DBArch](#)) — значительно оптимизирована запись и чтение БД, а именно это: групповая запись нескольких архивов в одну таблицу и чтение блоками (кратными десяти) в одном запросе.
- [Все транспорты](#) — в целом увеличена производительность.
- Рабочий пользовательский интерфейс (WEB) ([UI.WebVision](#)) — увеличена чувствительность интерфейса использованием асинхронных запросов в общем цикле обновления.

Надёжность резервируемой станции в масштабе комплексного решения, точнее сохранение данных истории, была усовершенствована расширением механизма резервирования, который потенциально предусматривает резервирование любой подсистемы и реализует сейчас резервирование подсистем "Сбор данных" и "Архивы-История".

И, для программы в целом, был выполнен [ряд формальных всесторонних тестов](#), которые были предварительно расширены во [внутренней процедуре комплексного тестирования релиза OpenSCADA](#). В результате этих тестов

было выявлено несколько ошибок, которые исправлены.

4.4 Сбор данных

Учитывая ключевую роль сбора данных в программах такого рода, эта функция, в лице подсистемы "Сбор данных" и её модулей, получила значительные усовершенствования, из которых особенно нужно отметить: смещение акцента в расширении поддерживаемых источников данных из реализации отдельных модулей подсистемы "Сбор данных" на системном языке "C/C++", на их реализацию в окружении OpenSCADA и на её внутреннем языке — логический уровень OpenSCADA. Т.е., на логическом уровне OpenSCADA, может и реализуется все, что: для доступа к данным использует сеть, не требует использования специфических библиотек и функций и не является очень сложным. На данный момент, таким образом реализовано:

- Отправка SMS сообщений (SMS) и электронных писем (SMTP).
- Источник Бесперебойного Питания (UPS), как объект данных с атрибутами и значениями.
- Простые сенсоры и датчики:
 - Елемер TM510x;
 - EDWARDS TURBOMOLECULAR PUMPS (SCU750);
 - Sycon Multi Drop Protocol (SMDP);
 - Блок питания турбо-молекулярного насоса (TMP-xx03);
 - Измерение температуры IT-3 (IT3);
 - IVE-452HS-02;
 - OPTRIS CT/CTL;
 - CTR 100, 101.
- Компьютер теплового счётчика VKT7.
- IEC-60870-104.
- Тестовые реализации и примеры: DCON, OWEN.
- Шина "One Wire" с помощью {DS9097,DS9097U} (1W_{DS9097,DS9097}) для чипов: DS1820, DS1820/DS18S20/DS1920, DS1822, DS2413, DS2408, DS2450, DS2438.
- Шина I2C: PCF8591, PCF8574, BMP180, DS3231, AT24C{32|64}.
- Общие порты ВВ (GPIO): DHT11,22 (AM23XX). GPIO|I2C: 1602A(HD44780).

Учитывая рост роли логического уровня OpenSCADA, выросли и требова-

ния к гибкости модели данных источников данных и в целом, что было удовлетворено:

- завершением покрытия внутренней модели данных всеми функциями, характерными для области применения;
- добавлением иерархичности к параметрам объектов контроллеров источников данных;
- добавлением возможности произвольного и динамического формирования модели данных — набору атрибутов параметров.

Внутренний язык программирования OpenSCADA приобрёл значительные усовершенствования и, сейчас, удовлетворяет всем требованиям отраслей применения OpenSCADA. Многие из этих усовершенствований попали в предыдущую версию 0.8.0 LTS, а ряд из них не мог туда попасть из-за нарушения обратной совместимости или недостаточной стабильности на то время. Из этих, существенных усовершенствований, нужно ещё раз отметить: увеличение разрешающей способности данных внутреннего окружения OpenSCADA целого типа до 64 разрядов и сохранение контекста исполнения внутренних процедур.

4.5 Графическое окружение

В целом, графическое окружение получило значительное развитие и главным из них стало расширение [модуля запуска Qt-интерфейса](#), как основы для остальных локальных. В первую очередь, этот модуль и ядро OpenSCADA адаптированы для возможности запуска библиотеки графической среды Qt в основном потоке программы, что исключило много проблем, связанных с исполнением в неглавном потоке, а также обеспечена работа с версией 5 этой библиотеки. Во вторых, этот модуль взял на себя роль интерфейса выбора проектов OpenSCADA при её запуске и переключении, а также создание новых. И в третьих, ввиду его первичности к запуску Qt, он получил функцию управления видом программы независимо от графического окружения и возможность его запуска-закрытия в системный лоток. Что в целом сделало возможным настройку OpenSCADA под персональные требования пользователя и адаптировать её к очень специфическим окружениям вроде [Android](#).

Заметных усовершенствований получили модули конфигурации Qt и Web, где особенно нужно отметить:

- Конфигуратор-Qt ([UI.QTCfg](#)): осуществление запросов к удалённым станциям в отдельном потоке от потока Qt, что сделало его удобнее и более прогнозируемым и что даже, вместе с одним из обновлений, попало в предыдущую стабильную версию 0.8.0 LTS.
- Конфигуратор-WEB ([UI.WebCfgD](#)): в целом, был полностью обновлен на предмет интерфейса, который сейчас является более динамическим, удобным и может быть расширен темами.

Заметных усовершенствований также получила среда визуализации и управления (СВУ), в составе всех её модулей:

- Движок СВУ и модули визуализаторов вообще ([UI.VCAEngine](#)):
 - позволяет осуществлять полноценную-горячую разработку, т.е. — редактирование проекту СВУ на момент его исполнения сеансами;
 - оригинальные сообщения интерфейса являются полностью откорректированными, грамматически и орфографически корректными для Английского языка;
 - предоставлено ряд дополнительных видов примитива "Элементы формы", "Диаграмма" и расширено примитив "Text" отображением "HTML";
 - предоставлено механизм специфических для визуализатора атрибутов виджетов, что позволило раскрыть и использовать их индивидуальные свойства;
 - новый-гибкий механизм сообщений пользователя про внештатные события в контролируемом системой процессе, который предусматривает возможность свободного формирования пользовательских извещателей с нужными свойствами, как: моно-тонный сигнал, синтез речи.
- Визуализатор-Qt ([UI.Vision](#)):
 - заметно улучшена работа исполнения удалённых интерфейсов и обеспечено осуществление запросов к удалённой станции в отдельном потоке от потока Qt, что упростило удалённую разработку с параллельным исполнением проекта;
 - формирование примитива "Документ" может осуществляться с помощью WebKit.
- Визуализатор-WEB ([UI.WebVision](#)):
 - унифицирован, оптимизирован и расширен использованием CSS3;
 - предоставлена реализацию всех элементов-примитивов унифицированного интерфейса;
 - осуществляется масштабирование до доступного пространства окна браузера;
 - увеличена производительность работы и чувствительность интерфейса, через использование исключительно асинхронного механизма при обновлении.

Все Web-модули, в целом, получили возможность изменения вида посредством тем, через общесистемные интерфейсы [модуля протокола HTTP](#) и его механизмы обработки запросов и формирования ответов. Также, все они получили поддержку динамического перевода интерфейса, что особенно актуально для многопользовательских Web-интерфейсов и с чем Вы можете ознакомиться на демонстрационных Web-интерфейсах моделей OpenSCADA: [АГЛКС](#), [Котел](#). Добавлен также механизм разделения доступа по страницам, который, кроме всего прочего, позволяет разграничить доступ к Web-модулям вообще.

5 Новые и значительно обновленные модули

В новой версии были добавлены новые модули и значительным образом обновлен ряд существующих:

- Добавлены новые модули:
 - MMS(IEC-9506) ([DAQ.MMS](#)) — модуль поддержки обмена данных по протоколу "Manufacturing Message Specification (MMS, IEC-9506)".
 - Comedi ([DAQ.Comedi](#)) — модуль поддержки источников данных реального времени (библиотека "Comedi"), которые основаны на платах сбора данных различных производителей и устанавливаются на шинах: ISA, PCI, PCMCIA и USB.
 - SMH2Gi ([DAQ.SMH2Gi](#)) — модуль реализации доступа к аппаратным модулям источников данных ПЛК Segnetics SMH2Gi, SMH4 для "MC", "MR", а также взаимодействия с оригинальным окружением "SMLogix".
 - Fastwel IO ([DAQ.Fastwel](#)) — модуль обмена данными с аппаратными модулями Fastwel IO.
 - FT3 (АПСТМ) ([DAQ.FT3](#)) — модуль обмена данными с контроллерами АПСТМ, АСДКУ, СУАП.
 - GPIO ([DAQ.GPIO](#)) — модуль доступа к GPIO одноплатных ПК вроде Raspberry Pi, Orange Pi и другие.
- Вычислитель на основе Java-подобного языка ([DAQ.JavaLikeCalc](#)) расширен: поддержкой внутренних функций, динамическим переводом сообщений и многими другими функциями интерфейса пользовательского программирования в целом.
- Шлюз источников данных ([DAQ.DAQGate](#)) расширен шлюзованием сообщений, связанных с выбранным источником данных.
- Сбор данных ОС ([DAQ.System](#)) расширен источниками данных: "Файловая Система", "ИБП", "QSensor" и возможностью отделения медленных источников от быстрых.

- ModBus ([{DAQ.Protocol}.ModBus](#)) расширен поддержкой строки, как последовательности значений регистров.
- Клиент DCON ([DAQ.DCON](#)) значительно расширен поддержкой специфических модулей.
- OPC-UA ([DAQ.OPC-UA](#)) значительно расширен публикацией и поддержкой "кусков" в клиентской части сервиса, специфический код протокола выделен в библиотеку.
- Оборудование ICP_DAS ([DAQ.ICP_DAS](#)) значительно расширен, а фактически полностью переписан, поддержкой всех присутствующих плат сбора данных от фирмы ICP_DAS для серии I8k, I-87k, на шине ISA и унифицированные типы для стандартных модулей серии I7k.
- Siemens S7 ПЛК ([DAQ.Siemens](#)) значительно расширен собственной реализацией ISO-TSAP.
- Diamond платы сбора данных ([DAQ.DiamondBoards](#)) значительно расширен, а фактически полностью переписан, поддержкой всех присутствующих плат сбора данных от фирмы Diamond Systems.
- Устройства АСКУ ([DAQ.AMRDevs](#)) реализована поддержка счётчика Контар (МЗТА).
- [Все модули БД](#) значительно расширены динамическим переводом.
- Архиватор на БД ([Archive.DBArch](#)) расширен архивацией нескольких архивов в одну таблицу и восстановлением перечня архивов из информации в БД.
- Архиватор на ФС ([Archive.FSArch](#)) добавлена поддержка промежуточных типов "Int16", "Int32", "Int64", "Float", "Double" и дополнен абсолютным ограничением размера архива на диске.
- [Все модули транспортов](#) расширены режимом пулинга входных транспортов и протоколов.
- Сокеты ([DAQ.Sockets](#)) расширен поддержкой шины RAWCAN и инициативным подключением входного транспорта.
- Последовательный интерфейс ([Transport.Serial](#)) расширен поддержкой низкоуровневой шины I2C; специальными пользовательскими функциями последовательного интерфейса: "sendbreak", "TS", "DR", "DCD", "RI"; расширенным управлением RTS для RS-485.
- Слой безопасных сокетов ([Transport.SSL](#)) расширен поддержкой: TLSv1.1, TLSv1.2, DTLSv1.
- Собственный протокол программы ([Protocol.SelfSystem](#)) расширен иерархическим и многоуровневым направлением запросов на внеш-

ние хосты.

- HTTP ([Protocol.HTTP](#)) расширен адаптивностью к интерфейсам пользователя в системных диалогах и предоставлением общего API построения HTTP-интерфейсов как для пользователя, так и модулям за ним.
- Программный конфигурактор (Qt) ([UI.OTCfg](#)) запросы интерфейса управления OpenSCADA вынесены в отдельный поток и улучшена реализация элементов контроля.
- Программный конфигурактор (Динамический WEB) ([UI.WebCfgD](#)) расширен новым дизайном с использованием CSS3 и возможностей современных WEB-браузеров.
- Концепция и движок среды визуализации и управления ([UI.VCAEngine](#)) расширен: поддержкой специфических для визуализатора атрибутов виджетов, реализацией нового-гибкого механизма уведомления пользователя, фоновым (в отдельном потоке) исполнением задачи формирования документов, увеличением количества трендов на одном кадре до 100 и поддержкой логарифмической шкалы примитива "Диаграмма".
- Рабочий пользовательский интерфейс (Qt) ([UI.Vision](#)) расширен: для работы в сети через сервер визуализации; реализацией группы специфических этого визуализатора атрибутов; реализацией видов "Дерево", "Таблица" и расширением вида "Кнопка", примитива "Элементы формы"; реализацией вида "XY" примитива "Диаграмма"; значительным рефакторингом кода примитива "Элементарная фигура".
- Рабочий пользовательский интерфейс (WEB) ([UI.WebVision](#)) расширен использованием CSS3 и возможностей современных браузеров, а именно: добавлено масштабирование к доступному пространству окна браузера, реализованы все примитивы унифицированного интерфейса.
- Библиотека функций системного API среды пользовательского программирования ([Special.FLibSYS](#)) расширен функциями и объектами: "floatExtract", "md5", "tmSleep", объект "IO"; значительно расширены существующие функции и объекты: "dbReqSQL", "FFT", "strParse", "strDec4Bin".

6 Выводы

По пути к новому релизу 0.9 LTS промышленного назначения осуществлена большая работа по стабилизации, расширению функциональных возможностей, а также расширению адаптационных возможностей к работе на альтернативных платформах. Всё это в целом ещё больше расширило границы

полноценного применения OpenSCADA на всех уровнях систем промышленной автоматизации и смежных направлениях автоматизации и автоматики.

LTS версии OpenSCADA не являются пустым звуком, они реально поддерживаются всё время, до следующей LTS версии, и поддержка этой версии ещё будет расширена сервисными обновлениями. Также, будет осуществлено смещение акцента и политики внедрения LTS версии, которая до этого рекомендовалась к обновлению конфигураций с замороженными библиотечными БД, на приоритет в использовании на широком спектре новых решений.

В появлении новой промышленной версии OpenSCADA 0.9 LTS приняли участие:

- [Роман Савоченко](#): основной объём работ по разработке, тестированию, сборке, документированию и переводу программы и документации тремя языками.
- Максим Кочетков: реализация модулей DAQ.Fastwel, DAQ.FT3; расширение модуля [Transport.Sockets](#) поддержкой RAWCAN и модуля [Transport.Serial](#) расширенным управлением RTS для RS-485.
- Арсен Закоян: реализация поддержки счётчиков электроэнергии "Меркурий 200" и "Меркурий 230".
- Руслан Ярмлюк: реализация поддержки счётчика электроэнергии NIK 2303.
- Алмаз Каримов: расширение модуля реализации протокола DCON.
- Аркадий Кисель: реализация поддержки I2C сенсора температуры, барометрического давления и влажности BME280.
- Константин (IrmIngeneer): поддержка сборки OpenSCADA на дистрибутиве Linux Gentoo.
- Сергей Дорошка: [предварительная адаптация OpenSCADA к работе на QNX](#).
- Организации, которые содействовали наиболее значительному усовершенствованию OpenSCADA через её интеграцию в собственные системы управления:
 - Proviron Holding NV: приобретение пакетов технической поддержки для общей поддержки, исправления и улучшения реализации протокола Siemens ISO-TSAP(ProfiNet), некоторых наработок около Raspberry Pi, 2014, 2016-2018.
 - Лаборатория вакуумных технологий: "Вакуумная технологическая установка", 2011-2018, и финансирование оборудования сервера OpenSCADA, 2014.
 - Оптима: "Система Автоматизации Метрополитена", 2016.

- Вектор: приобретение пакетов технической поддержки для исправления и улучшения реализации OPC-UA в работе с разными OPC-UA клиентами и серверами, 2015-2016.
- Краматорсктеплоэнерго+ДЛЯ: "САУ шаровых барабанных мельниц ШБМ 287\410 котлоагрегата №8 БКЗ 160–100 ПТ", 2015, та "САУ Фосфатирования, Амминирования и Гидразина котлоагрегатов БКЗ 160–100 ПТ", 2014. В качестве ПЛК использован ICP-DAS LP-8781 с OpenSCADA в роли среды исполнения ПЛК.
- Хартрон: "Система Контроля Реактивности (СКР) Ядерной Подкритической Установки (ЯПУ)", 2013-2015.
- Много других организаций и отдельных пользователей, которые пожелали остаться анонимными, через приобретение технической поддержки и услуг, конструктивные отзывы внедрения и эксплуатации.

Последующие усилия разработки будут направлены на:

- Завершение ревизии основной документации, в основном по модулям и библиотекам.
- Чёткое определение и обновление политики предоставления коммерческих услуг вокруг проекта OpenSCADA:
 - пересмотр и обновление условий пакетов технической поддержки;
 - политика и организация сборок обновлений к стабильной ветви: сохранение истории всех публичных обновлений и организация сервисных;
 - дистрибуция исключительно коммерческих сборок с публичными демонстрационными на примере пакетов для Android — разработка и формирование механизма контроля за сроком пакета технической поддержки и демонстрационного режима.
- Освоение и адаптацию к отрасли "Управление ресурсами предприятия (ERP)" с направленностью на:
 - создание интерфейса управления ресурсами на сервере проекта;
 - создание общего интерфейса контроля задач и их финансирования, с организацией разработчиков к привлечению исполнения этих задач и технической поддержки.
- Расширение функций применения в отраслях "Умный дом", "Домовая автоматика" и "Бытовые роботы".
- Адаптацию к работе в окружении операционных систем QNX и MS Windows.