

*OpenSCADA 0.9 LTS*

Савоченко Р. О.

OpenSCADA Team <oscada@oscada.org>

Анонс нової LTS версії 0.9 проекту OpenSCADA — відкрита Система Диспетчерського Контролю та Збору Даних. Нова стабільна версія є результатом шести років розробки, впровадження до різних рішень, стабілізації та експлуатації OpenSCADA у її робочій гілці. Та весь цей час продовжували випускатися оновлення до 0.8.0 LTS та що буде відбуватися для анонсованої 0.9 LTS.

## ***1 Вступ***

Реліз OpenSCADA, відкритої SCADA(Supervisory control and data acquisition) системи, версії 0.9 є стабільним промисловим релізом тривалої підтримки (LTS).

Основною метою релізу є надання оновленої та стабільної платформи побудови рішень комплексних систем автоматизації та інших суміжних рішень для співтовариства користувачів та розробників вільного програмного забезпечення. Також, реліз призначено стати актуальним та міцним підґрунтям у побудові комерційних рішень.

Реліз є наступною версією стабільної гілки для якого, протягом тривалого часу, надається технічна підтримка від розробників та забезпечується випуск виправлень у вигляді публічних збірок для основних та стабільних оточень Linux, а також оперативних збірок для власників пакетів технічної підтримки. Життєвий цикл попереднього релізу 0.8.0 LTS припиняється його останнім поновленням, безпосередньо перед першими збірками пакетів 0.9 LTS.

Саме на цій версії відбувся остаточний перехід на схему розробки Work/LTS, тобто, розробка здійснюється у межах робочої версії та на її основі періодично випускаються стабільні релізи, які, своєю чергою та паралельно розробці, оновлюються шляхом зворотнього портування сумісних змін із робочої версії. Ініціюючий перехід відбувся у 2013 році, коли поточною стабільною версією визнано 0.8.0 LTS, а робочою 0.9 Work. Наразі, стабільною версією стає анонсована 0.9 LTS, а робочою 1 Work.

Загалом, нова стабільна версія є результатом шести років роботи з розробки, впровадження у різні рішення, стабілізації та експлуатації OpenSCADA у її робочій гілці, яка наразі випускається поточною стабільною. Весь цей час оновлення для 0.8.0 LTS продовжували випускатися та що буде здійснюватися для анонсованої 0.9 LTS.

## 1.1 Загальна інформація

Проект OpenSCADA засновано Савоченко Романом у 2003 році як вільну реалізація Системи Диспетчерського Контролю та Збору Даних (SCADA) або Людино Машинний Інтерфейс (HMI), на основі її ґрунтовного проектування протягом 2002 року та досвіду використання і розробки комерційної SCADA-системи до цього.

SCADA або HMI системи загалом призначені та використовуються для здійснення людиною оперативного контролю за роботою складного та відповідального технологічного обладнання та процесів різноманітних виробничих підприємств.

На час анонсу 0.9 LTS, OpenSCADA є розвинутою SCADA/HMI системою, яка доволі широко застосовується як за прямим призначенням, так, завдяки своїй гнучкості, і у багатьох суміжних галузях, та яка із впевненістю може називатися більш загально — динамічна система роботи із даними реального часу.

Номер стабільної версії OpenSCADA менше за одиницю тільки через те, що її метою була заявлена багатоплатформність, досягти якої у запланованому об'ємі планується наступним стабільним релізом з номером 1. Загалом, цілями OpenSCADA є:

- **відкритість** — переважно GPLv2;
- **масштабованість, гнучкість, розширюваність** — модульність та внутрішня динамічність;
- **виконавча надмірність** — резервування;
- **доступність** — відкриті вихідні тексти; багатомовність; динамічна багатомовність; автоматичне складання архівів, пакетів, живих дисків, ...; складання та виконання на Linux оточеннях у широкому віковому діапазоні, від 2002 року (2009-[ALTLinux 6](#), [LP8x81](#), [Fedora 12](#); 2012-[Debian 7](#)) по сучасні;
- **надійність** — практичне застосування, швидке вирішення проблем;
- **безпе́чність** — розподіл прав, SSL;
- **багатоплатформність** — x86\_32, x86\_64, ARM, Web, Android, QNX ([преадаптовано](#)), MS Windows ([заплановано](#));
- **єдиний, уніфікований, зручний та розвинутий інтерфейс користувача** — Qt, Web, прозоре та багаторівневе віддалене керування;
- **широкий спектр джерел даних** — універсальні, плати DAQ та низькорівневі шини, створені у оточенні OpenSCADA.

## 1.2 Застосування

До актуальних галузей застосування, де відоме хоча-б одне впровадження та із коротким переліком яких від учасників проекту можна ознайомитися за [цим посиланням](#), відносяться:

- [АСУ ТП \(SCADA/HMI\)](#) або системи телемеханіки — основний напрямок та тут існує багато впроваджень;
- [динамічні моделі, імітатори та тренажери](#) технологічних процесів у реальному часі;
- [станки та промислові роботи](#);
- [системи сільсько-господарчої диспетчеризації та керування, пташники](#);
- [вбудовані та мобільні системи](#) — оточення виконання Програмованого Логічного Контролеру (ПЛК), роботи, ...;
- [моніторинг обладнання серверів](#);
- [розумні будинки та домашня автоматика](#).

Із деякими обмеженнями та доробками, значним чином у внутрішньому оточенні користувача, OpenSCADA може використовуватися й у таких галузях:

- керування ресурсами підприємства (ERP);
- Гео-локація та відстеження положення;
- торговельні системи;
- медичні діагностичні системи;
- бухгалтерський облік;
- білінгові системи.

## 2 *Результати попереднього релізу 0.8.0 LTS*

Реліз 0.8.0 LTS випущено у квітні 2012 року та протягом цих шести років він отримав 20 оновлень, якими, загалом, виправлено більше 500 помилок та додано багато вдосконалень, які не порушують сумісності бібліотечних БД та конфігурації.

Конфігурація та бібліотечні БД 0.8.0 LTS були загалом заморожені, через великі несумісні зміни у 0.9 Work, для методів запуску OpenSCADA, та недосконалість розповсюдження бібліотечних БД на момент випуску 0.8.0 LTS. Тому, оновлення та перехід на версію 0.9 LTS буде нетривіальним, хоча у 0.9 LTS передбачено все можливе для спрощення [цієї процедури](#). Оновлення же 0.9 LTS до запланованої у майбутньому 1 LTS вже не буде таким складним і фактично може перетворитися виключно у формальну зміну версії робочої

гілки, оскільки тепер планується оновлення всього, включно з бібліотечними БД.

### **3 Планові завдання релізу**

Розвиток OpenSCADA, після попередньої LTS версії та у межах робочої гілки, переважно здійснювався шляхом глибокої стабілізації та через практичну адаптацію з елементами розширення існуючого функціоналу, націлені на надання стабільного та надійного оточення автоматизації промислових та суміжних завдань, а відтак і не існувало чітко визначеного плану. Та за три роки до цього релізу такий план з'явився і склав наступні завдання, що видно із [загального плану розвитку](#):

- Повна ревізія основної документації та підготовка анонсу цього релізу.
- [Адаптація до роботи на програмній платформі "Android"](#).
- Перенос бази знань OpenSCADA, та Wiki-ресурсу документування, на новий рушій із уніфікацією структури під багатомовність із пріоритетом: Англійська, Українська, Російська; та генерації автономної (offline) документації за її зміни.
- Ревізія, актуалізація та деяке розширення Web-модулів OpenSCADA.
- Розширення та адаптація OpenSCADA до прямої роботи із низькорівневими шинами та пристроями на кшталт 1Wire, I2C та реалізація проекту розумного будинку у власному помешканні.
- Формальне створення та документування [проекту дистрибутиву Linux автоматизації із OpenSCADA](#).
- Створення автоматизованої системи складання пакетів OpenSCADA.
- Перенос серверної інфраструктури OpenSCADA на [власне обладнання та канал у інтернет](#).
- [DAQ.OPC-UA](#): Спрощення, збільшення функціональності та перенесення протокольного коду у окрему LGPL v3 бібліотеку.

Ці завдання виконано та деякі деталі щодо них наведено далі.

### **4 Загальносистемні властивості**

Нова стабільна версія OpenSCADA отримала помітні загальносистемні розширення, набула більшої стабільності у роботі та збільшення продуктивності, а також отримала помітні удосконалення графічного інтерфейсу та оточення довкола неї на кшталт: розширеного налаштування, документації, доступної прямо з програми (offline та online), та яку значним чином переглянуто і актуалізовано.

Основну частину документації проекту перенесено на нову Wiki, на основі MediaWiki, та значний об'єм цієї частини було відревізовано та пере-

кладено на три мови — Англійська, Українська, Російська. Для перенесення було створено [процедуру конвертації статей з діалекту рунія WascoWiki на MediaWiki](#), яку було повністю написано на внутрішній мові OpenSCADA та яку повсякчасно використано для переносу великого об'єму старої Wiki. Формат автономної (offline) документації було змінено зі статичних PDF-файлів, які не оновлювалися після попередньої LTS версії, на HTML-файли, які динамічно генеруються з актуальної бази знань проекту (Wiki), а також мають актуальні крос-посилання між сторінками та посилання на online-документацію, для виключно зовнішніх матеріалів. Генерація offline-документації також здійснюється [спеціально написаною процедурою](#) на внутрішній мові OpenSCADA, що, разом із процедурою конвертації Wiki-діалекту та [комплексного тестування релізу OpenSCADA](#), є яскравою ознакою потужності та точного рівня розвиненості внутрішньої мови OpenSCADA.

У межах старої Wiki, після випуску 0.8.0 LTS було додано та оновлено деякі документи, а з перенесенням на нову Wiki, окрім ревізії та перекладу основної частини документації, було здійснено загальну уніфікацію її структури з міркувань: багатомовності (Англійської як первинної мови), логічності організації, зручності перекладу та можливості прибрати дублювання статей з офіційним сайтом, які на разі просто використовуються із Wiki. Окремо відзначимо значно-розширені головні документи:

- [Швидкий старт](#);
- [Посібник по програмі](#);
- [API користувача OpenSCADA](#);
- [ЧАП та Як зробити ...](#);
- [Створення модуля OpenSCADA](#).

Первинну мову (Англійську) було повністю відревізовано у оригінальних повідомленнях програми та багато у основних статтях документації-Wiki OpenSCADA, що наразі робить OpenSCADA адекватно сприятливою для аудиторії, що не розуміє вихідних мов — Української або Російської. Та, разом із реалізація у повному обсязі механізму динамічного перекладу, робить можливим побудову на основі OpenSCADA динамічно багатомовних інтерфейсів користувача, що Ви можете, зокрема, побачити на публічно доступному Web-інтерфейсі динамічних моделей ТП: [АГЛКС](#), [Котел](#).

Робочою версією, у основі цієї стабільної, вперше було визначено поняття проекту OpenSCADA та реалізовано сценарій командного рядку для запуску та створення проектів OpenSCADA. Безпосередньо перед випуском цієї версії, поняття проекту OpenSCADA було остаточно закріплено за текою з даними окремого проекту та конфігураційним файлом OpenSCADA, а реалізацію менеджера проекту було інтегровано безпосередньо у OpenSCADA. Відтак, ця LTS версія має розвинуте поняття менеджера проектів, що до-

звляє гнучко з ними працювати та виключити небезпечну можливість багаторазового запуску із загальними даними одного проекту.

Робочою версією, також, запроваджено зміну версій модулів за змінами у кодї модуля та безпосередньо перед їх вивантаженням до репозиторію вихідних текстів, а відтак, версії модулів OpenSCADA цього релізу наочно відзеркалюють загальний рівень їх розвинутості та стабільності.

У процесі здійснення робіт над робочою гілкою, у основі цієї стабільної, було запроваджено формування репозиторіїв пакетів дистрибутивів Linux зі збірками OpenSCADA, які до цього надавалися лише як окремі пакети. Що помітно спростило розгортання OpenSCADA та утримання її актуальною. Потім було створено [автоматичний складальник цих пакетів](#), який наразі нараховує до 100 цілей, та що значно спростило випуск оновлень як для робочої гілки так і для цієї стабільної. Відтак, до цієї LTS версії надаються збірки пакетів для основних оточень Linux та всю історію публічних оновлень буде збережено.

До пакетів збірок OpenSCADA надаються також і збірки [живих дисків](#) швидкого ознайомлення та розгортання OpenSCADA разом із системним оточенням. Якї до цього моменту отримали формальне підґрунтя у вигляді [Linux дистрибутиву автоматизації проекту OpenSCADA](#).

Збірки та пакети OpenSCADA супроводжуються низкою відкритого та вільного матеріалу внутрішнього оточення програми, а саме, це: напрацьованя шару збору та обробки даних, елементи графічного представлення та цілі-комплексні проекти моделей ТП. Ці матеріали надаються у вигляді файлів БД SQLite та включають:

- Бібліотеки функцій (OscadaLibs), "LibsDB/OscadaLibs.db" — містить всі напрацьовання проекту OpenSCADA у шарі збору та обробки даних включно з [елементами джерел даних користувачького протоколу](#).
- СВУ: Головні бібліотеки (vcaBase), "LibsDB/vcaBase.db" — містить [основні елементи графічного представлення](#) та [елементи мнемосхем](#) проекту OpenSCADA.
- СВУ: Тести (vcaTest), "LibsDB/vcaTest.db" — містить елементи тестів примітивів графічного представлення.
- СВУ: Бібліотека електро-елементів мнемосхем користувачького інтерфейсу (vcaElectroEls), "LibsDB/vcaElectroEls.db" — містить елементи графічного представлення компонентів електричних схем.
- Хостінг проекту загалом та матеріалів 0.9 LTS зокрема, було перенесено на [власний сервер проекту](#), де додатково було розгорнуто: демонстраційні Web-інтерфейси моделей OpenSCADA, проект моніторингу серверу та розумного будинку на основі OpenSCADA та

складальник пакетів репозиторію OpenSCADA.

## 4.1 Внутрішні

Роздільну здатність внутрішніх даних цілого типу оточення OpenSCADA збільшено до 64 розрядів. Загалом-же, внутрішні дані OpenSCADA, з резервуванням значення помилки (EVAL) для кожного, уніфіковано загальними типами: логічне, ціле, реальне, рядок та об'єкт. Що в першу чергу стосується джерел даних.

До ядра OpenSCADA, [власного протоколу](#) та всіх вузлів що працюють з віддаленими станціями OpenSCADA; додано можливість "підняття" вузлів OpenSCADA, що знаходяться за іншими вузлами та, як правило, у іншій мережі. Що загалом дозволяє централізовано керувати мережею станцій OpenSCADA на будь-якому рівні ієрархії.

## 4.2 Вдосконалення та адаптація до різних платформ

Ця версія OpenSCADA набула поглибленої підтримки та здібності до адаптації до різноманітних платформ. Це переважно сталося завдяки адаптації до роботи на програмній платформі Android та відновлення збірки і роботи з ucLibC, та що планується використати у подальшій адаптації до роботи на програмних платформах QNX та MS Windows.

Роботу на одноплатних ПК було розширено платами [Raspberry Pi](#) та [Orange Pi](#).

Підтримку Linux смартфонів Nokia доповнено останнім, заснованим на MeeGo 1.2, це — Nokia N9. Або це було оновлення підтримки Nokia N950.

Окрім безпосередньої роботи (нативно) на різних платформах, було значно розширено Web-інтерфейс, який наразі реалізує всі загальні можливості концепції середовища візуалізації та управління (СВУ).

## 4.3 Оптимізація, стабілізація та продуктивність

Значної стабілізації ядра OpenSCADA, та програми загалом, досягнуто через уніфікацію контролю внутрішніх ресурсів та розширення [можливостей налагодження користувачем](#). Загалом, користувацьку діагностику та налагодження розширено:

- загальним ввімкненням-вимкненням налагодження та цільовим контролем вузлів налагодження;
- особливим налагодженням об'єктів контролерів джерел даних;
- особливим налагодженням виконання проектів СВУ;
- формуванням поглибленої статистики виконання динамічних об'єктів OpenSCADA, як то: об'єкти контролерів джерел даних та їх параметрів, вхідних та вихідних транспортів, сеансів СВУ з деталіза-

цією до віджетів;

- [протоколюванням вхідного та вихідного трафіку транспортів](#).

Глибокої та всебічної стабілізації притерпіли майже всі модулі OpenSCADA та багато було оптимізовано, з яких особливо треба відзначити:

- [Всі модулі БД](#) — додано обробку та перевірку помилок БД або СУБД, та видачу повідомлень про ці помилки у випадку користувацького завантаження та запису; збільшено продуктивність (до порядку) БД що підтримують SQL, через реалізацію режиму предзавантаження сканувального запиту;
- Обчислювач на основі Java-подібної мови ([DAQ.JavaLikeCalc](#)) — збільшено продуктивність через: збереження контексту виконання функцій, предзавантаження постійних та прямий доступ до рядку.
- Архіватор на БД ([Archive.DBArch](#)) — значно оптимізовано запис та читання БД, а саме це: гуртовий запис декількох архівів до однієї таблиці та читання блоками (кратними десяти) у одному запиті.
- [Всі транспорти](#) — загально підвищено продуктивність.
- Робочий користувацький інтерфейс (WEB) ([UI.WebVision](#)) — збільшено чутливість інтерфейсу використанням асинхронних запитів у загальному циклі оновлення.

Надійність резервованої станції у масштабі комплексного рішення, точніше збереження даних історії, було вдосконалено розширенням механізму резервування, який потенційно передбачає резервування будь-якої підсистеми та реалізує наразі резервування підсистем "Збір даних" та "Архіви-Історія".

Та, для програми загалом, було виконано [низку формальних всебічних тестів](#), які було попередньо розширено у [внутрішній процедурі комплексного тестування релізу OpenSCADA](#). За підсумком цих тестів було виявлено декілька помилок, які виправлено.

## 4.4 Збір даних

Враховуючи ключову роль збору даних у програмах такого роду, ця функція, у особі підсистеми "Збір даних" та її модулів, отримала значні вдосконалення, з яких особливо треба відзначити: зміщення акценту у розширені підтримуваних джерел даних з реалізації окремих модулів підсистеми "Збір даних" на системній мові "C/C++", на їх реалізацію у оточені OpenSCADA та на її внутрішній мові — [логічний рівень OpenSCADA](#). Тобто, на логічному рівні OpenSCADA, може та реалізується все, що: для доступу до даних використовує мережу, не потребує використання специфічних бібліотек та функцій та не є дуже складним. Наразі, таким чином реалізовано:

- Відправлення SMS повідомлень (SMS) та електронних листів



(SMTP).

- Джерело Безперервного Живлення (UPS), як об'єкт даних із атрибутами зі значеннями.
- Прості сенсори та давачі:
  - Елемер TM510x;
  - EDWARDS TURBOMOLECULAR PUMPS (SCU750);
  - Sycon Multi Drop Protocol (SMDP);
  - Блок живлення турбо-молекулярного насосу (TMP-xx03);
  - Вимірювання температури ІТ-3 (ІТ3);
  - IVE-452HS-02;
  - OPTRIS CT/CTL;
  - CTR 100, 101.
- Комп'ютер теплового лічильника VKT7.
- IEC-60870-104.
- Тестові реалізації та приклади: DCON, OWEN.
- Шина "One Wire" за допомогою {DS9097,DS9097U} (1W\_{DS9097,DS9097}) для чипів: DS1820, DS1820/DS18S20/DS1920, DS1822, DS2413, DS2408, DS2450, DS2438.
- Шина I2C: PCF8591, PCF8574, BMP180, DS3231, AT24C{32|64}.
- Загальні порти ВВ (GPIO): DHT11,22 (AM23XX). GPIO|I2C: 1602A(HD44780).

Враховуючи збільшення ролі логічного рівня OpenSCADA, зросли і вимоги до гнучкості моделі даних джерела даних та загалом, що було задоволено:

- завершенням покриття внутрішньої моделі даних всіма функціями, характерними до галузей застосування;
- доданням ієрархічності до параметрів об'єктів контролерів джерел даних;
- доданням можливості довільного та динамічного формування моделі даних — набору атрибутів параметрів.

Внутрішня мова програмування OpenSCADA набула значних удосконалень та, наразі, задовольняє всім вимогам галузей застосування OpenSCADA. Багато з цих удосконалень потрапили до попередньої версії 0.8.0 LTS, та низка не могла туди потрапити через порушення зворотної сумісності або недостатньої стабільності на той час. З цих, суттєвих

вдосконалень, треба ще раз відзначити: збільшення роздільної здатності даних внутрішнього оточення OpenSCADA цілого типу до 64 розрядів та збереження контексту виконання внутрішніх процедур.

## 4.5 Графічне оточення

Загалом, графічне оточення отримало значний розвиток та головним з них стало розширення [модуля запуску Qt-інтерфейсу](#), як підгрунття для решти локальних. В першу чергу, цей модуль та ядро OpenSCADA адаптовано для можливості запуску бібліотеки графічного середовища Qt у основному потоці програми, що виключило багато проблем, пов'язаних із виконанням у неголовному потоці, а також забезпечено роботу з версією 5 цієї бібліотеки. По друге, цей модуль взяв на себе роль інтерфейсу обрання проєктів OpenSCADA при її запуску та перемиканні, а також створення нових. І по третє, через його первинність до запуску Qt, він отримав функцію керування виглядом програми незалежно від графічного оточення та можливість його запуску-закриття у системний лоток. Що загалом зробило можливим налаштування OpenSCADA під персональні вимоги користувача та адаптувати її до дуже специфічних оточень на кшталт [Android](#).

Помітних удосконалень отримали модулі конфігурації Qt та Web, де особливо треба відзначити:

- Конфігуратор-Qt ([UI.OTCfg](#)): здійснення запитів до віддалених станцій у окремому потоці від потоку Qt, що зробило його зручнішим та більш прогнозованим та що навіть, разом з одним із оновленням, потрапило до попередньої стабільної версії 0.8.0 LTS.
- Конфігуратор-WEB ([UI.WebCfgD](#)): загалом, був повністю оновлений на предмет інтерфейсу, який наразі є більш динамічним, зручним та може бути розширений темами.

Помітних вдосконалень також отримало середовище візуалізації та управління (СВУ), у складі всіх його модулів:

- Рушій СВУ та модулі візуалізаторів загалом ([UI.VCAEngine](#)):
  - дозволяє здійснювати повноцінну-гарячу розробку, тобто — редагування проєкту СВУ на момент його виконання сеансами;
  - оригінальні повідомлення інтерфейсу є повністю відкоригованими, граматично та орфографічно коректними для Англійської мови;
  - надано низку додаткових видів примітиву "Елементи форми", "Діаграма" та розширено примітив "Text" відображенням "HTML";
  - надано механізм специфічних до візуалізатору атрибутів віджетів, що дозволило розкрити та використовувати їх інди-

відуальні властивості;

- новий-гнучкий механізм повідомлень користувача про позаштатні події у контрольованому системою процесі, який передбачає можливість вільного формування користувацьких повідомлювачів з потрібними властивостями, як то: моно-тоновий сигнал, синтез мови.
- Візуалізатор-Qt ([UI.Vision](#)):
  - помітно покращено роботу з виконання віддалених інтерфейсів та забезпечено здійснення запитів до віддаленої станції у окремому потоці від потоку Qt, що спростило віддалену розробку із паралельним виконанням проекту;
  - формування примітиву "Документ" може здійснюватися за допомогою WebKit.
- Візуалізатор-WEB ([UI.WebVision](#)):
  - уніфіковано, оптимізовано та розширено використанням CSS3;
  - надано реалізацію всіх елементів-примітивів уніфікованого інтерфейсу;
  - здійснюється масштабування до доступного простору вікна браузеру;
  - збільшено продуктивність роботи та чутливість інтерфейсу, через використання виключно асинхронного механізму при оновленні.

Всі Web-модулі, загалом, набули можливість зміни вигляду за посередництва тем, через загальносистемні інтерфейси [модуля протоколу HTTP](#) та його механізми обробки запитів і формування відповідей. Також, всі вони отримали підтримку динамічного перекладу інтерфейсу, що особливо актуально для багатокористувацьких Web-інтерфейсів та з чим Ви можете ознайомитися на демонстраційних Web-інтерфейсах моделей OpenSCADA: [АГЛКС](#), [Котел](#). Додано також механізм розподілу доступу до сторінок, який, зокрема, дозволяє обмежувати доступ до Web-модулів загалом.

## ***5 Нові та значно оновлені модулі***

У новій версії було додано нові модулі та значним чином оновлено низку чинних:

- Додано нові модулі:
  - MMS(IEC-9506) ([DAQ.MMS](#)) — модуль підтримки обміну даних за протоколом "Manufacturing Message Specification (MMS, IEC-9506)".
  - Comedi ([DAQ.Comedi](#)) — модуль підтримки джерел даних

реального часу (бібліотека "Comedi"), які основані на платах збору даних різних виробників та встановлюються на шинах: ISA, PCI, PCMCIA та USB.

- SMH2Gi ([DAQ.SMH2Gi](#)) — модуль реалізації доступу до апаратних модулів джерел даних ПЛК Segnetics SMH2Gi, SMH4 для "MC", "MR", а також взаємодії із оригінальним оточенням "SMLogix".
- Fastwel IO ([DAQ.Fastwel](#)) — модуль обміну даними з апаратними модулями Fastwel IO.
- FT3 (АПСТМ) ([DAQ.FT3](#)) — модуль обміну даними з контролерами АПСТМ, АСДКУ, СУАП.
- GPIO ([DAQ.GPIO](#)) — модуль доступу до GPIO одноплатних ПК на кшталт Raspberry Pi, Orange Pi та інші.
- Обчислювач на основі Java-подібної мови ([DAQ.JavaLikeCalc](#)) розширено: підтримкою внутрішніх функцій, динамічним перекладом повідомлень та багатьма іншими функціями інтерфейсу користувачького програмування загалом.
- Шлюз джерел даних ([DAQ.DAOGate](#)) розширено шлюзуванням повідомлень, пов'язаних із обраним джерелом даних.
- Збір даних ОС ([DAQ.System](#)) розширено джерелами даних: "Файлова Система", "ДБЖ", "QSensor" та можливістю відокремлення повільних джерел від швидких.
- ModBus ([DAQ.Protocol](#)}\_{ModBus}) розширено підтримкою рядка, як послідовності значень регістрів.
- Клієнт DCON ([DAQ.DCON](#)) значно розширено підтримкою специфічних модулів.
- OPC-UA ([DAQ.OPC-UA](#)) значно розширено публікацією та підтримкою "шматків" у клієнтській частині сервісу, специфічний код протоколу відокремлено у бібліотеку.
- Обладнання ICP\_DAS ([DAQ.ICP\\_DAS](#)) значно розширено, а фактично повністю переписано, підтримкою всіх наявних плат збору даних від фірми ICP\_DAS для серії I8k, I-87k, на шині ISA та уніфіковані типи для стандартних модулів серії I7k.
- Siemens S7 ПЛК ([DAQ.Siemens](#)) значно розширено власною реалізацією ISO-TSAP.
- Diamond плати збору даних ([DAQ.DiamondBoards](#)) значно розширено, а фактично повністю переписано, підтримкою всіх наявних плат збору даних від фірми Diamond Systems.

- Пристрої АСКО ([DAQ.AMRDevs](#)) реалізовано підтримку лічильника Контар (МЗТА).
- [Всі модулі БД](#) значно розширено динамічним перекладом.
- Архіватор на БД ([Archive.DBArch](#)) розширено архівацією декількох архівів у одну таблицю та відновленням переліку архівів із інформації у БД.
- Архіватор на ФС ([Archive.FSArch](#)) додано підтримку проміжних типів "Int16", "Int32", "Int64", "Float", "Double" та доповнено абсолютним обмеженням розміру архіву на диску.
- [Всі модулі транспортів](#) розширено режимом пулінгу вхідних транспортів та протоколів.
- Сокети ([DAQ.Sockets](#)) розширено підтримкою шини RAWCAN та ініціативним підключенням вхідного транспорту.
- Послідовний інтерфейс ([Transport.Serial](#)) розширено підтримкою низькорівневої шини I2C; спеціальними користувацькими функціями послідовного інтерфейсу: "sendbreak", "TS", "DR", "DCD", "RI"; розширеним керуванням RTS для RS-485.
- Шар безпечних сокетів ([Transport.SSL](#)) розширено підтримкою: TLSv1.1, TLSv1.2, DTLSv1.
- Власний протокол програми ([Protocol.SelfSystem](#)) розширено ієрархічним та багаторівневим спрямуванням запитів до зовнішніх хостів.
- HTTP ([Protocol.HTTP](#)) розширено адаптивністю до інтерфейсів користувача у системних діалогах та наданням загального API побудови HTTP-інтерфейсів як для користувача, так і модулів за ним.
- Програмний конфігуратор (Qt) ([UI.OTCfg](#)) запити інтерфейсу керування OpenSCADA винесено до окремого потоку та покращено реалізацію елементів контролю.
- Програмний конфігуратор (Динамічний WEB) ([UI.WebCfgD](#)) розширено новим дизайном з використанням CSS3 та можливостей сучасних WEB-браузерів.
- Концепція і рушій середовища візуалізації та керування ([UI.VCAEngine](#)) розширено: підтримкою специфічних до візуалізатору атрибутів віджетів, реалізацією нового-гнучкого механізму повідомлення користувача, фоновим (у окремому потоці) виконанням задачі формування документів, збільшення кількості трендів на одному кадрі до 100 та підтримкою логарифмічної шкали примітиву "Діаграма".
- Робочий користувацький інтерфейс (Qt) ([UI.Vision](#)) розширено: для

роботи у мережі через сервер візуалізації; реалізацією групи специфічних до цього візуалізатору атрибутів; реалізацією видів "Дерево", "Таблиця" та розширенням виду "Кнопка", примітиву "Елементи форми"; реалізацією виду "XY" примітиву "Діаграма"; значним рефакторингом коду примітиву "Елементарна фігура".

- Робочий користувацький інтерфейс (WEB) ([UI.WebVision](#)) розширено використанням CSS3 та можливостей сучасних браузерів, зокрема: додано масштабування до доступного простору вікна браузера, реалізовано всі примітиви уніфікованого інтерфейсу.
- Бібліотека функцій системного API середовища користувацького програмування ([Special.FLibSYS](#)) розширено функціями та об'єктами: "floatExtract", "md5", "tmSleep", об'єкт "IO"; значно розширено чинні функції та об'єкти: "dbReqSQL", "FFT", "strParse", "strDec4Bin".

## **6 Висновки**

На шляху до нового релізу 0.9 LTS промислового призначення здійснено велику роботу по стабілізації, розширенню функціональних можливостей, а також розширенню адаптаційних можливостей до роботи на альтернативних платформах. Все це загалом ще більше розширило межі повноцінного застосування OpenSCADA на всіх рівнях систем промислової автоматизації та суміжних напрямків автоматизації та автоматики.

LTS версії OpenSCADA не є порожнім звуком, вони реально підтримуються весь час, до наступної LTS версії, та підтримку цієї версії ще буде розширено сервісними оновленнями. Також, буде здійснено зміщення акценту та політики впровадження LTS версії, яка до цього рекомендувалась до оновлення конфігурацій із замороженими бібліотечними БД, на пріоритет у використанні на широкому спектрі нових рішень.

У появі нової промислової версії OpenSCADA 0.9 LTS прийняли участь:

- [Роман Савоченко](#): основний об'єм робіт по розробці, тестуванню, збірці, документуванню та перекладу програми і документації трьома мовами.
- Максим Кочетков: реалізація модулів DAQ.Fastwel, DAQ.FT3; розширення модуля [Transport.Sockets](#) підтримкою RAWCAN та модуля [Transport.Serial](#) розширеним керуванням RTS для RS-485.
- Арсен Закоян: реалізація підтримки лічильників електроенергії "Меркурій 200" та "Меркурій 230".
- Руслан Ярмолюк: реалізація підтримки лічильнику електроенергії NIK 2303.
- Алмаз Каримов: розширення модуля реалізації протоколу DCON.
- Аркадій Кисіль: реалізація підтримки I2C сенсору температури,

барометричного тиску та вологості BME280.

- Константин (IrmIngeener): підтримка складання OpenSCADA на дистрибутиві Linux Gentoo.
- Сергій Дорошка: [попередня адаптація OpenSCADA до роботи на QNX](#).
- Організації, що сприяли найбільш значущому вдосконаленню OpenSCADA через її інтеграцію до власних систем керування:
  - Proviron Holding NV: придбання пакетів технічної підтримки для загальної підтримки, виправлення та покращення реалізації протоколу Siemens ISO-TSAP(ProfiNet), деяких напрацювань довкола Raspberry Pi, 2014, 2016-2018.
  - Лабораторія вакуумних технологій: "Вакуумна технологічна установка", 2011-2018, та фінансування обладнання серверу OpenSCADA, 2014.
  - Оптима: "Система Автоматизації Метрополітену", 2016.
  - Вектор: придбання пакетів технічної підтримки для виправлення та покращення реалізації OPC-UA у роботі з різними OPC-UA клієнтами та серверами, 2015-2016.
  - Краматорськтеплоенерго+ДІЯ: "САУ кульових барабанних млинів ШБМ 287/410 котлоагрегату №8 БКЗ 160–100 ПТ", 2015, та "САУ Фосфатування, Амінування та Гідразину котлоагрегатів БКЗ 160–100 ПТ", 2014. У якості ПЛК використано ICP-DAS LP-8781 із OpenSCADA у ролі середовища виконання ПЛК.
  - Хартрон: "Система Контролю Реактивності (СКР) Ядерної Підкритичної Установки (ЯПУ)", 2013-2015.
- Багато інших організацій та окремих користувачів, що побажали залишитися анонімними, через придбання технічної підтримки та послуг, конструктивні відгуки впровадження та експлуатації.

Подальші зусилля розробки буде спрямовано на:

- Завершення ревізій основної документації, переважно по модулям та бібліотекам.
- Чітке визначення та оновлення політики надання комерційних послуг довкола проекту OpenSCADA:
  - перегляд та оновлення умов пакетів технічної підтримки;
  - політика та організація збірок оновлень до стабільної гілки: збереження історії всіх публічних оновлень та організація сервісних;

- дистрибуція виключно комерційних збірок із публічними демонстраційними на прикладі пакетів для Android — розробка та формування механізму контролю за терміном пакету технічної підтримки та демонстраційного режиму.
- Освоєння та адаптацію до галузі "Керування ресурсами підприємства (ERP)" із спрямуванням на:
  - створення інтерфейсу керування ресурсами на сервері проекту;
  - створення загального інтерфейсу контролю задач та їх фінансування, з організацією розробників до залучення виконання цих задач та технічної підтримки.
- Розширення функцій застосування у галузях "Розумний будинок", "Домова автоматика" та "Побутові роботи".
- Адаптацію до роботи в оточені операційних систем QNX та MS Windows.