

OpenSCADA 0.7.1

(Зауваження до релізу)

Показник

Вступ	2
1 Реалізація запланованих завдань	3
2 Оптимізація, підвищення стабільності, стійкості та продуктивності системи	4
3 Удосконалення та стабілізація графічної підсистеми	5
4 Формування та стабілізація API програмування користувача	6
5 Загальносистемні розширення	6
Заключення	7

Вступ

Цей реліз є запланованим випуском робочої версії системи OpenSCADA, які зазвичай відбуваються з періодичністю 3-4 місяці. Основною метою цього релізу є реалізація комерційного сервісу на основі OpenSCADA та адаптація до апаратної платформи ARM. Крім основних завдань, у межах цього релізу виконано велику роботу по стабілізації, очищенню вихідного коду, а також всебічного розширення можливостей.

Цей документ є обробкою (компіляцією) документа <ChangeLog> системи OpenSCADA версії 0.7.1, який призначено коротко та наочно освітлити нові можливості системи OpenSCADA. Детально ознайомитися зі змінами у системі OpenSCADA можна у файлі ChangeLog з дистрибутива системи або тут: <http://wiki.oscada.org/Works/ChangeLog> (RU).

Ключовими особливостями цієї версії є:

- Реалізація запланованих завдань.
- Оптимізація, підвищення стабільності, стійкості та продуктивності системи.
- Удосконалення та стабілізація графічної підсистеми.
- Формування та стабілізація API програмування користувача.
- Загальносистемні розширення.

Нові та оновлені модулі:

- *Archive.FSArch (1.4.5)* - Значна стабілізація.
- *DB.MySQL (1.7.0)* - Стабілізація. Додана можливість встановлення таймаутів підключення.
- *DB.SQLite (1.6.3)* - Стабілізація.
- *DB.FireBird (0.9.6)* - Стабілізація.
- *DB.PostgreSQL (0.9.1)* - Стабілізація.
- *DAQ.DiamondBoards (1.2.2)* - Стабілізація.
- *DAQ.System (1.7.3)* - Стабілізація. Залежність від бібліотеки "Libsensors" зроблена необов'язковою.
- *DAQ.BlockCalc (1.4.1)* - Стабілізація. Додано підсвічування синтаксису.
- *DAQ.JavaLikeCalc (1.9.0)* - Стабілізація. Додано підсвічування синтаксису. Виконано адаптацію до апаратної архітектури ARM. Додано реалізацію регулярних виразів та низки розширень API користувача. Додано можливість запису символів рядка за допомогою вісімкового "\041" та шістнадцяткового "\x21" числа.
- *DAQ.LogicLev (1.1.3)* - Стабілізація. Деякі розширення.
- *DAQ.SNMP (0.6.0)* - Стабілізація. Реалізація повної підтримки клієнтської частини протоколу. Додано підтримку всіх типів значень. Додано підтримку запису значень. Покращено обробку помилок. Додано додаткові параметри конфігурації підключення.
- *DAQ.Siemens (1.2.4)* - Стабілізація.
- *DAQ.ModBus (1.1.2)* - Стабілізація. Додано підсвічування синтаксису переліку атрибутів. Деякі удосконалення.
- *DAQ.DCON (0.5.0)* - Модуль повністю переписано на предмет підтримки змішаних модулів, за посередництвом обрання команд запити.
- *DAQ.ICP_DAS (0.7.3)* - Стабілізація.
- *DAQ.DAQGate (0.9.2)* - Стабілізація. Додано кеш параметрів для їх ініціалізації під час автономної роботи.
- *DAQ.SoundCard (0.6.2)* - Стабілізація.
- *DAQ.OPC-UA (0.6.1)* - Стабілізація. Додано підсвічування синтаксису переліку атрибутів атрибутів.
- *DAQ.BFN (0.5.0)* - Новий модуль.
- *Transport.Sockets (1.4.6)* - Стабілізація.
- *Transport.SSL (0.9.6)* - Стабілізація.
- *Transport.Serial (0.7.2)* - Стабілізація. Під час ініціалізації порту використано попередні налаштування та реалізовано можливість опустити встановлення окремих налаштувань.
- *Protocol.ModBus (0.6.2)* - Стабілізація. Деякі покращення.
- *Protocol.OPC-UA (0.6.1)* - Стабілізація.

- *Protocol.HTTP (1.5.1)* - Стабілізація. Додано підтримку змінної HTTP "Transfer-Encoding=chunked" у вихідному транспорті.
- *Protocol.SelfSystem (0.9.4)* - Стабілізація.
- *Protocol.UserProtocol (0.6.1)* - Додано підсвічування синтаксису.
- *Special.SystemTests (1.5.1)* - Невеликі удосконалення.
- *UI.QTStarter (1.6.1)* - Стабілізація. Адаптація до ARM.
- *UI.QTCfg (2.0.0)* - Стабілізація. Використано груповий запит до інтерфейсу керування з метою оптимізації віддаленого доступу до OpenSCADA станції. Адаптація до ARM. Додано функцію ручної зміни висоти для елементів тексту та таблиць (схопив за нижній край та потягнув). Додано багаторівневе підсвічування синтаксису.
- *UI.WebCfg (1.5.5)* - Стабілізація.
- *UI.WebCfgD (0.6.6)* - Стабілізація.
- *UI.VCAEngine (1.1.0)* - Стабілізація. Додано контекстну підказку до атрибутів примітивів віджетів. Додано підсвічування синтаксису до атрибутів примітивів віджету.
- *UI.Vision (1.1.0)* - Значна стабілізація. Помітні удосконалення. Адаптація до ARM. Додано відображення та відновлення втрати зв'язку під час віддаленого виконання сеансу. Додано окремий повнотекстовий редактор для текстових атрибутів. Додано контекстну підказку до атрибутів примітивів віджетів. Додано підсвічування синтаксису до текстових атрибутів примітивів віджетів. Додано діалог для встановлення властивостей окремо обраних фігур.
- *UI.WebVision (0.9.1)* - Значна стабілізація. Невеликі покращення.
- *UI.WebUser (0.6.1)* - Стабілізація. Додано підсвічування синтаксису процедур.

1 Реалізація запланованих завдань

У відповідності до [плану](#) релізу було виконано наступні завдання:

- *Формування надання комерційних послуг на основі OpenSCADA.* - Побудовано концепцію та створено механізми надання комерційних послуг розробниками на основі OpenSCADA (<http://oscada.org/ua/poslugi>).
- *Адаптація системи OpenSCADA до роботи на апаратній платформі ARM.* - Здійснено збірку, адаптацію та повномасштабне тестування OpenSCADA до архітектури ARM; на інтернет планшеті фірми [Nokia](#) - [N800](#) (<http://wiki.oscada.org/Works/Tests/ARM> (RU)). А також приватні заходи по адаптації:
 - *SYS:* Прототипи функцій `TMess::put()`, `TMess::get()`, `TCntrNode::grpSize()`, `TCntrNode::grpId()`, `TCntrNode::grpAt()` змінено для виправлення використання беззнакового типу "char" на ARM як знакового.
 - *SYS.TVariant, DAQ.AMRDevs, DAQ.OPC-UA:* Виправлено невірний доступ до даних.
 - *SYS.TCntrNode:* Виправлено використання беззнакового "char" у ARM як знакового.
 - *Archive:* Прототипи функцій `TArchiveS::messPut()` та `TArchiveS::messGet()` змінено для виправлення змінної рівня повідомлення як знакового числа.
 - *DAQ.JavaLikeCalc:* Віртуальну машину переписано для прямого використання структур команд, з метою виключення проблеми вирівнювання, а також для деякого підвищення продуктивності.
 - *DAQ.Siemens:* Виправлено проблему бібліотеки NoDave на предмет глобального виклику команди пакування структур "#pragma pack(1)".
 - *UI.QTCfg:* Додано короткочасне засинання, після закриття всіх вікон, у функції виключення. З метою надати час остаточно зруйнуватися об'єктам всіх вікон та запобігти падінню.

2 Оптимізація, підвищення стабільності, стійкості та продуктивності системи

У процесі робіт над цією версією, а також її практичної адаптації, було виявлено та виправлено в загальній складності біля 90 помилок. В цілому було виконано роботу по чищенню коду від попереджувальних повідомлень компілятора, з прапорцем "-Wall":

- видалення невикористаних змінних;
- додання повернень до всіх функцій;
- ініціалізація змінних з можливістю їх використання невизначеними;
- виключення неповної обробки перелічуваних типів у операції обрання;
- порівняння знакового з беззнаковим цілим.

Перелічимо більш істотні помилки, виправлення яких значним чином відбилося на підвищені стабільності:

- *Загальносистемні:*
 - Команду складальної системи "\$ make dist" виправлено та адаптовано до коректного створення дистрибутивів OpenSCADA.
 - Функцію TSYS::cron(), для виклику за розкладом, неодноразово виправлено.
 - Тип змінної для збереження результату функції find(), рядка, всюди змінено на коректний "size_t".
 - *Function:* Виключено можливість витоку пам'яті у випадку спроби встановити об'єкт до необ'єктного параметру функції.
- *Бази даних:*
 - *DB.PostgreSQL:* Виправлено обробку адреси БД на предмет визначення порожніх ділянок.
- *Транспорти та протоколи:*
 - *Transport:* Виправлено завантаження всіх транспортів з БД, у випадку наявності записів про транспорт, модуль якого відсутній.
 - *Transport.Serial:* Встановлено прапор O_NONBLOCK для функції open() послідовного пристрою з метою попередження блокування відкриття порту на деяких USB->RS485 перетворювачах.
 - *Transport.Sockets, Transport.SSL:* Додано очищення вхідного буферу для попередження повторного використання на високолатентних каналах, після таймауту.
 - *Transport.Sockets:* Виправлено закриття клієнтського з'єднання за таймаутом часу життя, для підключень з відсутністю даних.
 - *Transport.Sockets:* Unix-сокет встановлено у неблокуючий режим за посередництвом прапору O_NONBLOCK. Виправлено формування повідомлень про помилки для вихідних транспортів.
 - *Protocol.HTTP, UI.WebCfg, UI.WebCfgD, UI.WebVision, UI.WebUser:* Виправлено розбір елементів POST-запиту на предмет розбору всіх атрибутів кожного елементу.
- *Збір даних та архіву:*
 - *Archive:* Виправлено використання буферу м'якої сітки з низькою роздільною здатністю часу, у архівах значень.
 - *Archive.FSArch:* Виправлено створення файлів архіву повідомлень при розташуванні безлічі старих повідомлень. Додано параметр попередження дублікатів.
 - *Archive.FSArch:* Результат функції fgetc() записано у цілочисельну змінну, замість символної, що виправило виявлення "EOF".
 - *DAQ.System:* Виправлено падіння при виході, для джерела "Hddtemp".
 - *DAQ.ModBus:* Режим встановлення значень атрибутів параметрів змінено на активний та покращено обробку помилок.
 - *DAQ.JavaLikeCalc:* Додано перевірку індексу при доступі до символу рядка за індексом, з метою попередження виходу за діапазон.
- *Інтерфейси візуалізації:*
 - *UI.VCAEngine:* Виправлено блокування сеансу для динамічно активних проектів,

під час динамічного доступу до власних атрибутів віджету.

- *UI.Vision*: виправлено падіння сеансу для проектів з періодом виконання більш 500 мс.
- *UI.Vision*: Реалізовано доведення координат фігур примітиву "Елементарна фігура" до залишку положення віджету.
- *UI.Vision*: Масштабування тексту, примітиву "Текст", змінено на зміну розміру шрифту замість масштабування кінцевого зображення, що підвищило загальну якість відображення шрифту.
- *UI.Vision, UI.WebVision*: Підвищено роздільну здатність координат всіх віджетів до трьох знаків після коми, з метою забезпечення нормального позиціонування на великих масштабах.
- *UI.WebVision, UI.Vision*: виправлено оновлення зображення примітиву "Елементарна фігура" під час зміни масштабу.
- *UI.WebVision*: Видалена перевірка влучення точок фігур у межі відображення примітиву "Елементарна фігура". Цю функцію перекладено на бібліотеку GD.
- *UI.WebVision*: Виключено можливість падіння для примітивів "Diagram" та "ElFigure", у випадку передавання некоректних значень атрибутів зображення "xSc" або "ySc" з Web-браузера.

Оптимізація та підвищення продуктивності:

- *UI.WebCfgD*: Додано зміну розміру іконок на боці сервера, з метою зниження трафіку.
- *UI.QTCfg*: Використано груповий запит до інтерфейсу керування, для підвищення продуктивності на повільних високолатентних підключеннях.
- *UI.QTCfg*: Виключено багаторазовий запит елементів обраних рядків таблиць.

3 Удосконалення та стабілізація графічної підсистеми

Помітну роботу було виконано у межах графічної підсистеми, а саме у модулях рушія СВУ *UI.VCAEngine*, візуалізаторів *UI.Vision* та *UI.WebVision*, а також конфігураторах. Внесені зміни було спрямовано на стабілізацію, оптимізацію споживання пам'яті та поліпшення користувацьких властивостей.

Удосконалення графічної підсистеми:

- *SYS, DAQ.JavaLikeCalc, UI.QTCfg, UI.VCAEngine, UI.Vision*: Додано API підсвічування синтаксису. На основі цього API підсвічування синтаксису додано:
 - *DAQ.BlockCalc*: підсвічування синтаксису у опису атрибутів параметрів.
 - *DAQ.JavaLikeCalc*: підсвічування синтаксису до компілятора мови *JavaLikeCalc.JavaScript*, а також до всіх текстових полів, які використовують цю мову всередині модуля.
 - *DAQ.ModBus, DAQ.OPC-UA*: підсвічування синтаксису переліку атрибутів.
 - *Protocol.UserProtocol*: підсвічування синтаксису до текстів процедур протоколу.
 - *UI.QTCfg*: відображення багаторівневого підсвічування синтаксису, яке передається з моделі даних у вигляді правил регулярних виразів.
 - *UI.VCAEngine*: підсвічування синтаксису до текстових атрибутів примітивів віджету, для яких це потрібно.
 - *UI.Vision*: підсвічування синтаксису до текстових атрибутів примітивів віджету, а також текстам програм віджетів.
 - *UI.WebUser*: підсвічування синтаксису процедур формування Web-інтерфейсу.
- *UI.QTCfg*:
 - Додано можливість ручної зміни текстових та табличних полів. Зміни відбуваються захопленням та пересуванням нижнього краю поля.
 - Груповий запит "CntReqs" використано для запити контексту сторінки.
- *UI.VCAEngine*: Додано специфічні властивості до примітивів, а саме допомога та підсвічування синтаксису.
- *UI.Vision*:
 - Додано відображення допомоги за атрибутами віджету.
 - Додано індикацію повідомлення про втрату з'єднання з сервером візуалізації, а

також відновлення, при віддаленій візуалізації.

- Для примітиву "Елементарна фігура" додано діалог встановлення властивостей окремо обраних фігур.
- Додано окремий повнотекстовий редактор для текстових атрибутів.
- *UI:WebVision*: Додано підтримку gif та jpg зображень у елементах примітиву "Елементарна фігура".

4 Формування та стабілізація API програмування користувача

Було продовжено формування об'єктного API програмування користувача, яке передбачає інтеграцію функцій користувача до дерева об'єктів системи OpenSCADA. Крім того, було внесено низку змін до існуючих бібліотек функцій API користувача.

Зокрема були здійснені наступні зміни:

- *SYS*: API користувача для об'єкта TArrayObj розширено на предмет можливості використання об'єктних властивостей, якщо властивості не числа.
- *SYS*: До дерева об'єктів OpenSCADA API користувача додано функцію SYS.Security.access() з метою контролю доступу користувача OpenSCADA до ресурсів з потрібними правами.
- *SYS, DAQ.JavaLikeCalc*: Додано реалізацію підтримки користувацького API регулярних виразів, заснованих на PCRE, за посередництвом об'єкту TRegExp. Реалізовано інтерфейс функцій RegExp.exec() та RegExp.test() у стандарті для мови JavaScript. Для DAQ.JavaLikeCalc реалізовано "new RegExp()" ініціалізацію для об'єкта TRegExp. Для DAQ.JavaLikeCalc реалізовані додаткові функції з використанням RegExp: int search(); Array match(); Array split(RegExp pat, int limit); string replace(string substr, string str); string replace(RegExp pat, string str).
- *TFunction, DAQ.JavaLikeCalc*: Додано об'єкт API користувача "TFuncArgObj" для отримання аргументів функції.
- *DB*: Додано функції користувацького API fieldStruct(), fieldSeek(), fieldGet(), fieldSet() та fieldDel() для об'єкта таблиці, з метою надати користувацький доступ до БД за посередництвом інтерфейсу БД OpenSCADA.
- *DAQ.JavaLikeCalc*: Додано вбудовану функцію typeof() для перевірки типу значення.

5 Загальносистемні розширення

До загальносистемного API системи OpenSCADA були внесені значні зміни та розширення з метою загальної стабілізації та у зв'язку із загальними заходами по очищенню коду, а також реорганізації дерева вихідних текстів:

- *SYS*:
 - Реалізовано нову структуру дерева вихідних текстів.
 - Всі ресурси проекту переміщено до окремих архівів.
 - Для створення архівів вихідних текстів та ресурсів можна використовувати команди "\$ make dist-lzma" та "\$ make dist-reslzma".
 - Документація виконана статичною, для модулів, та поділена за мовами. Для кожної мови документації формується окремий пакет. В документації кожної мови створено індексний файл по всьому переліку локальної та мережевої документації. Загальний перелік локальної документації помітно розширено новими документами, а також допоміжною мережевою документацією.
 - Додано пакети бібліотечних БД *-LibDB.Main та *-LibDB.VCA, розташовано у data/LibsDB.
 - Замість DemoDB пакета додано пакети моделей "AGLKS" та "Boiler", розташовано у data/ModelsDB.
 - Конфігураційні файли для побудови RPM та DEB пакетів оновлено.
 - Залежність на бібліотеку PCRECPP глобально замінено на PCRE.
 - Додано нову функцію TSYS::time2str() для перетворення інтервалів часу в строку

типу "1год 23хвил 10сек". Цю функцію всюди використано.

- Додано реалізацію власного XML-парсеру, замість Expat. Власний XML-парсер вирішує проблему підтримки різних кодувань, відмінних від UTF, а також безпечної обробки ситуації наявності недозволених, для поточного кодування, символів.
- *SYS.TMess*: Загальні повідомлення інтерфейсу керування переміщено до однієї функції *TMess::lab**.
- *Transport, Transport.**: Додано функцію *TTransportS::traf2str()* для перетворення значень лічильника трафіку у рядок виглядом "12.5KiB"

Заключення

Система OpenSCADA отримала поштовх для здійснення реальної апаратної багатоплатформності, за посередництвом чого код проекту було значним чином очищено та зроблено багатоплатформним. Ці напрацювання дозволять продовжити рух проекту у бік апаратної багатоплатформності, а також реалізації програмної, що заплановано після наступного LTS релізу 0.8.0.

Також продовжено роботу по вдосконаленню загальної функціональності, простоти та зручності використання із загальним підвищенням якості надання послуг на основі системи OpenSCADA.