

OpenSCADA. Практическое применение.

Савоченко Р. А. научный сотрудник НИП «ДІЯ», г.Днепродзержинск
тел. +380679859815, rom_as@diyaorg.dp.ua

Введение

Система OpenSCADA это открытая SCADA система разрабатываемая с 2003г. За этот период система прошла большой путь от проектирования и планомерной реализации до практического применения. На этом пути было достаточно много разговоров о системе на различных конференциях, три из которых происходили в этих стенах. Следуя традиции зачитаю доклад и в этот раз.

Данный доклад особый, в нём речь будет не столько о планах и новых возможностях сколько о первых шагах практического применения системы OpenSCADA. Хотя, о полноценном применении OpenSCADA в роли SCADA системы речь пока не идёт, тем не менее говорить о первых шагах практического применения нужно ввиду их важности для разработки.

В версии 0.5.0 (июнь 2006) система OpenSCADA приобрела функциональные возможности которых уже достаточно для решения определённого круга задач. В целом, из основных задач SCADA систем: сбора, обработки, архивирования, пользовательской визуализации и управления, осталась не реализованной пользовательская визуализация и управления, в то время как основной костяк, ядро OpenSCADA, реализован в полном объёме.

На основе версии 0.5.0 и рабочей версии системы OpenSCADA были реализованы следующие задачи:

- Построена подробная динамическая модель реального времени газо-лифтной компрессорной станции (ГЛКС).
- Реализован высокочастотный сбор, обработка и архивирование важных параметров компрессорной установки.

1 Построение динамической модели реального времени ГЛКС

Создание динамических моделей процессов реального времени является сложным и часто итерационным процессом, который может выполняться на различных уровнях детализации. Поэтому, наличие гибкого механизма создания таких моделей является важным фактором в повышении их эффективности и качества.

В практике создания автоматизированных систем управления технологическими процессами (АСУ-ТП) моделирование технологических процессов может использоваться для следующих целей:

- создание тренажеров для операторов и другого технологического

персонала;

- глубокого изучения и проработки механизмов управления технологическим процессом;
- создание устройств тестирования контроллеров управления технологическим процессом и их алгоритмов;
- использования для управления технологическим процессом (адаптивное управление, и управление по модели).

Как можно видеть, выше перечислены обычные подзадачи процесса создания АСУ-ТП, на разных стадиях которого нужна модель технологического процесса. Практически, модель технологического процесса сопровождает процес создания АСУ-ТП, а следовательно и процесс создания модели должен быть не отягощающим.

Учитывая эти задачи, система OpenSCADA была снабжена гибким механизмом пользовательского программирования, который позволяет достаточно просто создавать динамические модели, совмещая гибкость описания моделей аппаратов на языке высокого уровня и формализацию при формировании моделей целых технологических процессов на языке блоков.

Используя данный аппарат была создана подробная модель установки газолифтной компрессорной станции в составе двух компрессоров низкого, двух высокого давления и общестанционного оборудования. Технологическую схему моделируемой установки приведено на рис.1.

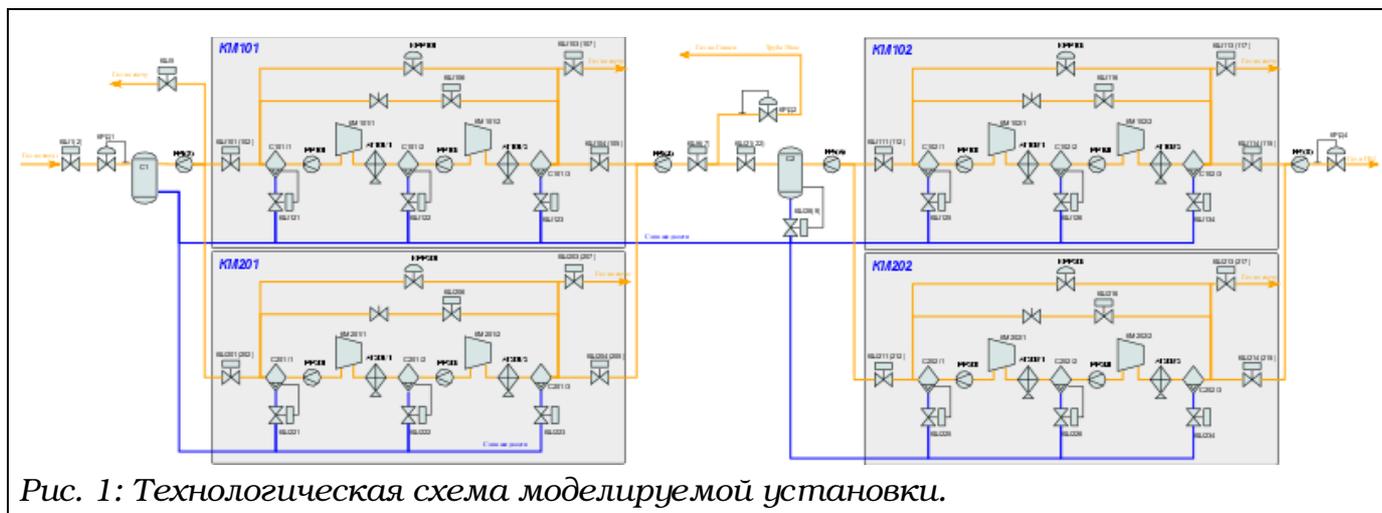


Рис. 1: Технологическая схема моделируемой установки.

Технологическая схема моделируемой установки изображённая на рис.1 практически прямо отображается на блочную схему модели рис.2. Такой подход значительно упрощает процесс создания, модификации и дополнения модели. В целом, моделирование состоит из двух стадий:

- создание моделей аппаратов технологического процесса (удобно описывать на языке высокого уровня);
- создание модели технологического процесса в целом, путём объединения моделей отдельных аппаратов в принципиальную схему (удобно формализовать до блоков – аппаратов ТП).

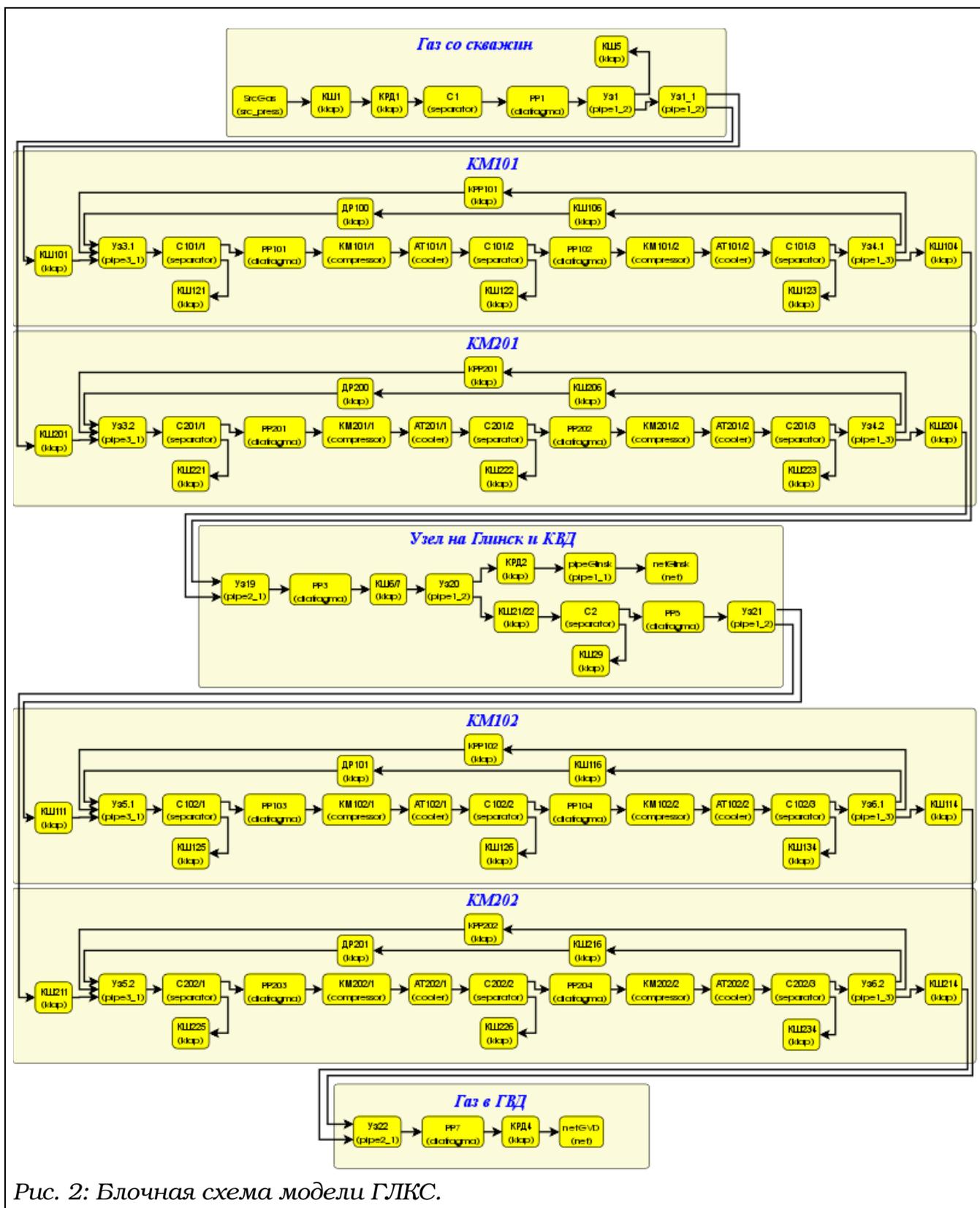


Рис. 2: Блочная схема модели ГЛКС.

Разделение процесса разработки динамической модели на две стадии позволяет органично распределить его между разработчиками различной сферы деятельности. Так, библиотеку аппаратов может создавать математик со знанием аппаратов ТП, а создавать модели ТП в целом может инженер технолог.

В общей сложности моделью (рис.1) описано около 86 технологических

аппаратов, 2 контура регулирования и связи между ними.

Модели аппаратов построены в соответствии с физическими принципами моделирования технологических процессов используя описание разностными уравнениями во времени для дискретных машин.

Если оценивать производительность модели, то можно отметить что, вычисление компонентов модели на системе Athlon 64 3000+ составляет:

- компрессор: 150 мкс;
- общестанционное оборудование: 176 мкс.

Учитывая масштаб времени динамической модели в 5 мс можно определить что, вычисления одного компрессора требует три процента нагрузки процессора, а вычисление модели в целом около 21 процента.

Практически, разработанная модель АГЛКС предназначалась для проверки контроллера управления ТП АГЛКС и его алгоритмов, включая антипомпажную защиту. Устройство моделирования объекта управления, на основе данной модели, было собрано на одноплатном компьютере АТН400-128 фирмы Diamond Systems Corporation.

2 Высокочастотный сбор и архивирование важных параметров компрессорной станции

Сбор данных является ответственной задачей для любой SCADA системы, однако их сфера применения в АСУ-ТП довольно инерционных процессов привело к тому, что редко встретишь SCADA систему позволяющую работать с потоком данных периодичностью менее секунды. В тоже время, встречаются технологические установки требующие значительно меньших интервалов.

Одним из классов таких установок являются центробежные компрессора. Функционирование установок этого класса сопровождается таким явлением как помпаж. Явление помпажа сопровождается колебаниями от единиц герц до нескольких килогерц. При этом, контролировать важно именно высокочастотные колебания, поскольку они являются предвестником эффекта помпажа, в то время как низкочастотные колебания практически являются его индикатором.

Система OpenSCADA оснащена достаточно мощным механизмом архивирования непрерывных процессов и оптимизированным трактом прохождения больших массивов значений. В целом, система OpenSCADA позволяет выполнять сбор, обработку и архивирование потоков данных с частотой в десятки килогерц на десятках каналов.

Имея такие возможности уже можно производить сбор, обработку и архивирование данных компрессорных установок. И недавно такая необходимость возникла. Фирмой НИП «ДІЯ» выполняются работы по автоматизации компрессорной станции АГЛКС(Анастасиевская Газо-Лифтная Компрессорная Станция) в процессе которой возникла необходимость записать и, в последствии, проанализировать спектр сигналов.

Для решения этой задачи было собрано устройство сбора и архивирования на основе одноплатного компьютера АТН400-128 формфактора РС104 фирмы Diamond Systems Corporation. Плата АТН400-128 содержит вычислительную систему на основе процессора фирмы АТІ - Eden400, оперативную память 128Мб, множество портов (Ethernet, 4xUSB, 4xCom и т.д.), а также порты сбора данных (DAQ) в составе: 16АІ, 4АО, 24DІО. Для хранения рабочей системы и архивов, к плате подключен жёсткий диск 2.5" объёмом 120 Гб. Программное окружение построено на основе дистрибутива ОС Linux "ALTlinux" и системы OpenSCADA.

К полученному устройству было подключено 8 каналов в составе 6 датчиков давлений и 2 датчиков дифференциального давления. Все данные собирались и архивировались с частотой дискретизации 1кГц. Данные собирались как за время пуска так и в течении суток. Примеры графиков данных дифференциального давления по пуску компрессора приведены на рис.3, рис.4 и рис.5.

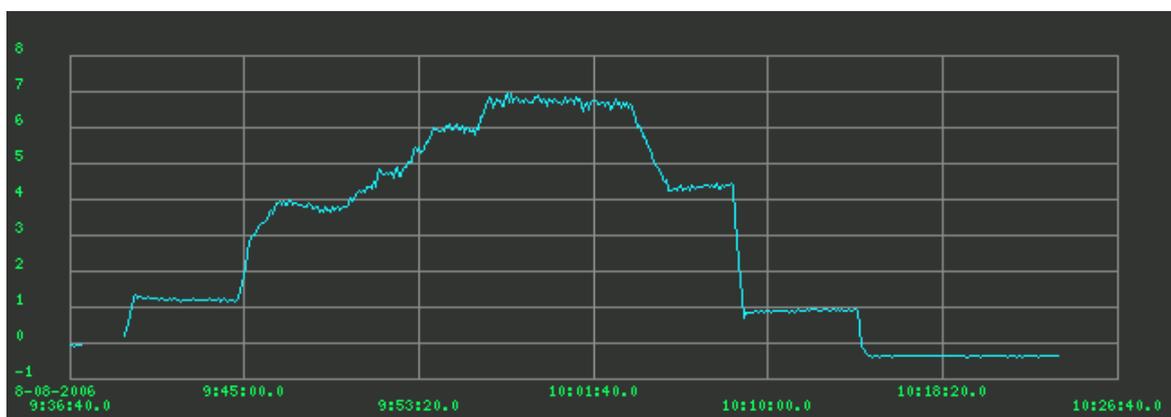


Рис. 3: Полный график пуска и останова компрессора.

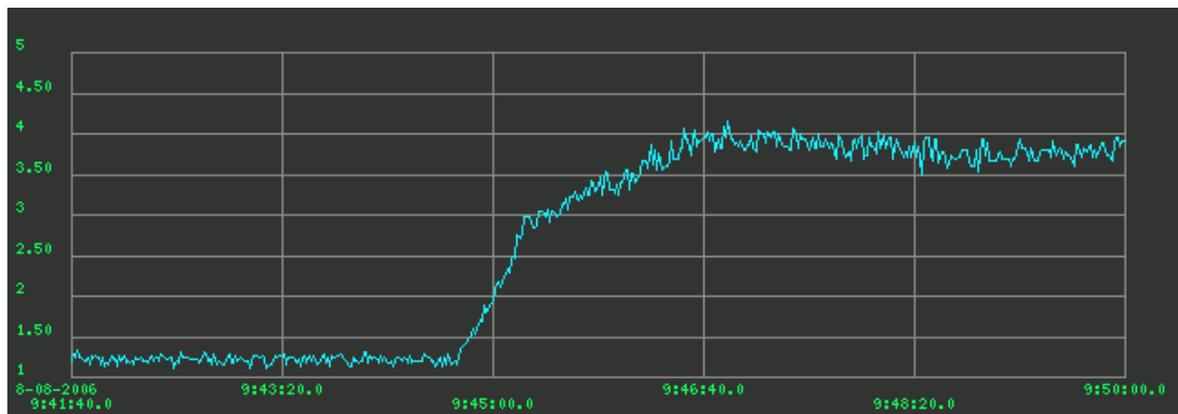


Рис. 4: График набора оборотов компрессором.

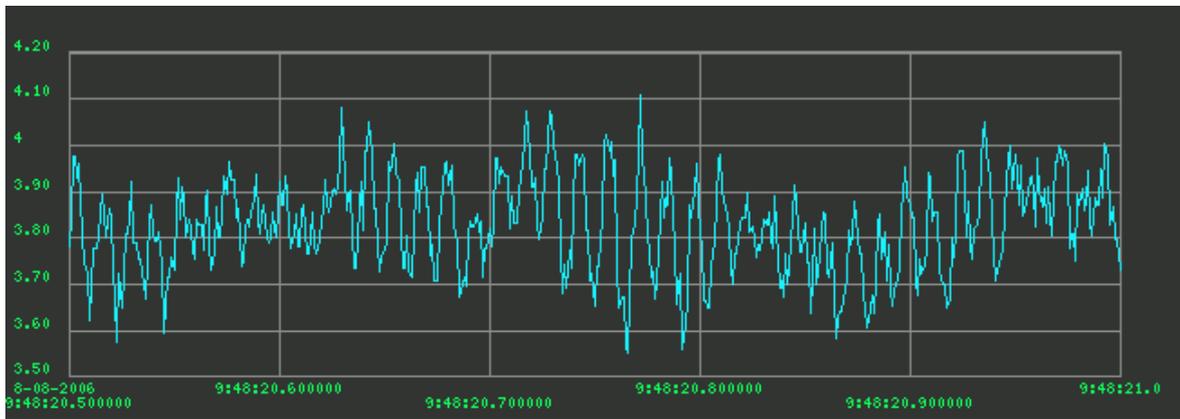


Рис. 5: График рабочего шума компрессора.

В процессе использования этого устройство была подтверждена высокая эффективность тракта обслуживания потоковых данных и алгоритмов модуля архивирования на файловую систему. Так, степень упаковки на реальных зашумленных данных составила 10% последовательным упаковщиком и 71% полной упаковкой включающей последовательную упаковку и упаковку стандартным архиватором *gzip*.

Сама возможность изучать данные визуально важна, однако часто нужно выполнить определённую обработку и анализ, подразумевающий выполнение определённых вычислений над данными в архиве. Система *OpenSCADA* предусматривает два механизма выполнения таких операций.

Первый механизм это экспорт в один из известных форматов с последующей обработкой его сторонним приложением. Для этой цели модуль архивирования на файловую систему снабжён функцией экспорта в формат звукового файла *wav*. С помощью этой функции было выполнено экспортирование и последующий спектральный анализ полученных данных.

Второй механизм подразумевает обработку архива внутри самой системы *OpenSCADA* с помощью одного из языков пользовательского программирования и API объектной модели системы *OpenSCADA*. Функции API для доступа к архивам позволяют реализовать практически любые задачи обработки, начиная от обработки буфера архива в реальном времени с формированием производных архивов и заканчивая после-обработкой существующих архивов. С помощью этого механизма было выполнено приведение архива из кода к шкале реального прибора.

3 Планы дальнейшего развития

Разработка системы не останавливается на достигнутом и продолжает свой путь к совершенству. Всё ещё существуют критические для дальнейшего применения задачи. И они запланированы и уже реализуются.

3.1 Разработка концепции среды визуализации и управления (СВУ)

Как ранее отмечалось, система OpenSCADA ещё не имеет собственной, полноценной, среды визуализации и управления (СВУ). Однако, планом предусмотрена разработка проекта концепции СВУ и реализация её на двух основах. Первая основа это библиотека QT4, благодаря которой будет создана высоко-реактивная и достаточно многоплатформенная СВУ. Вторая основа это Web-технологии, благодаря которым будет создана предельно многоплатформенная и простая в развёртывании СВУ.

Концепцией предусмотрено три уровня сложности в использовании СВУ. Этими уровнями являются:

- 1) Простое назначение динамики готовых кадров из библиотеки виджетов. Может выполняться даже в конфигураторе системы OpenSCADA.
- 2) Создание новых кадров с использованием библиотечных виджетов визуализации.
- 3) Создание новых кадров или виджетов с описанием связей между элементами в сценарии на одном из языков пользовательского программирования OpenSCADA.

Использование такого подхода сведёт к минимуму возможное отторжение СВУ из-за чрезмерной сложности.

Базовым понятием концепции СВУ выступает виджет. Виджет это элемент графического интерфейса, который предоставляет информацию в какой-то форме и принимает события на управление визуализацией. Все виджеты собираются в библиотеки. Виджеты могут быть базовыми и производными. Базовые виджеты реализуются в строго определённом спецификацией наборе и формируются каждой реализацией СВУ отдельно. Производные виджеты формируются на основе базовых или других производных виджетов независимо от реализации СВУ. При описании связей внутри производного виджета может использоваться язык пользовательского программирования системы OpenSCADA. Отдельные виджеты могут выступать в роли конечных элементов визуализации формируя кадр или окно. Окна размещаются в проекте формируя интерфейс визуализации и управления.

Такая концепция позволяет создавать широкий спектр интерфейсов СВУ различных типов начиная от стандартных SCADA интерфейсов и заканчивая простыми фоновыми.

3.2 Реализация компонентов распределённой архитектуры.

Не менее важной задачей, запланированной к реализации, является создание компонентов распределённой архитектуры. Базовым элементом этих компонентов является собственный протокол системы OpenSCADA, используемый для доступа к системе. Протокол использует интерфейс управления системой в качестве носителя информации. Остальными компонентами распределённой архитектуры

выступают модули коммуникации с удалённой системой для каждой подсистемы. Задачей этих модулей является отражение объектов отдельной подсистемы удалённой системы на локальную систему создавая видимость локальности и маскируя реальную природу данных.

Кроме коммуникаций на уровне подсистем собственный протокол может использоваться для удалённого управления и формирования своеобразных центров конфигурации распределённых архитектур. Например, если снабдить конфигуратор поддержкой этого протокола можно получить центр конфигурирования.

На данный момент собственный протокол уже реализован и на его основе в модуль конфигурирования QTCfg добавлена поддержка удалённых станций. Таким образом в один конфигуратор можно собрать нити конфигурирования всех доступных хостов.

В ближайшее время планируется реализация первого модуля отражения объектов удалённой подсистемы на локальную.

4 Выводы

Настал тот момент когда уже можно говорить о практическом применении OpenSCADA. Особенно это важно если учесть сложность и общие сроки разработки системы. Однако, каким бы сладким не был первый опыт практического применения, нельзя замедлять скорости развития и нужно продолжать совершенствование системы. Как отражение этих мыслей уже реализуются дальнейшие, не менее амбициозные, планы призванные завершить реализацию базовых функций полноценной SCADA системы.

Полагаю следующий доклад на этой конференции уже будет освещать опыт намного более широкого использования системы OpenSCADA.

5 Литература

1. <http://oscada.diyaorg.dp.ua> – Основной ресурс системы OpenSCADA.
2. <http://diyaorg.dp.ua/oscadawiki> – Wiki-ресурс системы OpenSCADA.
3. http://www.linuxrsp.ru/artic/about_OpenSCADA.html – Интервью Сергея Мороза с автором системы OpenSCADA об OpenSCADA.