

ПЕРЕДОВОЙ ОПЫТ РЕАЛИЗАЦИИ ПРОЕКТОВ АСУ ТП НА МастерСКАДА



В статье приводится пример практической реализации проекта АСУ ТП с помощью российской SCADA-системы. Особенностью реализации проекта является использование не самых очевидных возможностей МастерСКАДА.

Ключевые слова: САПР, SCADA-системы, МастерСКАДА, АСУ ТП.

ОТКУДА НАЧИНАЕТСЯ ПРОЕКТ?

Всякая серьезная разработка начинается с проектирования. И не одним человеком, а коллективом, в котором части проекта, задачи распределены по разным исполнителям. Инженеры, проектирующие шкаф управления технологической линией, передают данные в том виде, в котором позволяет используемая САПР (система автоматизированного проектирования). Маловероятно, что разработчик SCADA-проекта сможет диктовать формат передачи исходных данных. В свою очередь инженеры, программирующие контроллеры, дополняют таблицы адресами и типами данных переменных. Вот с этой “до-SCADA-вской” задачи и начинается разработка проекта. Всем этим легко пренебречь при небольшом количестве переменных. Но для сложной АСУ ТП со многими тысячами параметров такой подход не работает. Понятно, что для автоматического формирования любого проекта “целиком”, волшебной палочки сделать не получится. Профессиональный разработчик SCADA-проекта должен досконально изучить технологический процесс, и, понимая логику его работы, создать оптимальные решения для управления объектом, “нарисовать” мнемосхемы. Объектно-ориентированная архитектура МастерСКАДА позволяет разбить сложный объект управления на более простые части, технологические узлы, блоки и поручить разработку разным исполнителям.

Но это – задача организационная и не для сегодняшней статьи. Так что, первая задача на сегодня: обеспечение возможности автоматизированного проектирования для создания связей проекта. Данный подход позволяет из-

бежать дублирования данных, снизить вероятность ошибки при выполнении рутинных операций, а также автоматически распространять неизбежные изменения проекта на всех его стадиях. С удовольствием заметим, что задача эта решалась не разработчиками MasterSCADA, а коллективом инженеров-программистов ООО Фирма “Калининград-газприборавтоматика”.

ВОЗМОЖНОСТИ АВТОМАТИЗИРОВАННОГО ПРОЕКТИРОВАНИЯ

В базовом наборе возможностей системы есть инструмент для установления и редактирования связей проекта (рис. 1). Импорт связей происходит через файл формата *.csv, который легко редактируется в Excel.

Но этот инструмент “такой, как есть”: он позволяет сделать только экспорт, импорт и редактирование связей между параметрами объектов и OPC-переменными. В рассматриваемом случае перед разработчиками была поставлена задача автоматизировать создание типовых объектов проекта без необходимости дополнительного ручного редактирования. Реализованный с помощью ФБ (функционального блока) “Скрипт С#” код загружает данные из конфигурационных таблиц БД, создает типовые объекты, автоматически заполняет все поля на основе данных из БД, создает OPC-переменные в дереве системы и устанавливает между ними связи. Скрипт выполняется только в режиме разработки по команде DoAction. Развитая объектная модель МастерСКАДА позволяет легко манипулировать элементами дерева системы и дерева объектов.

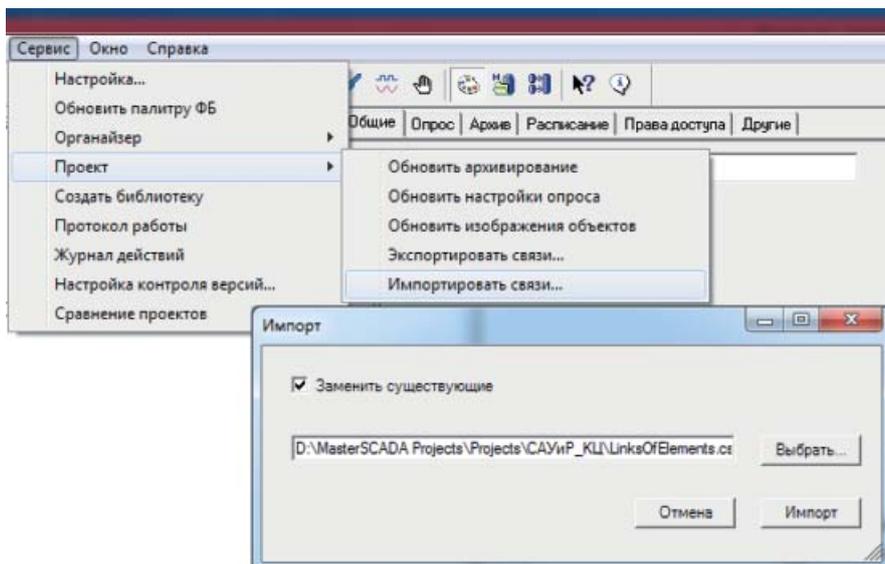


Рис. 1. Импорт-экспорт связей

СОЗДАНИЕ НА C# СОБСТВЕННЫХ ФУНКЦИОНАЛЬНЫХ БЛОКОВ С ПОМОЩЬЮ MasterSCADAToolkit

В каждой SCADA-системе есть библиотека функциональных блоков, в какой-то больше, в какой-то меньше. Но в каждой реальной системе найдется нечто особенное, чего в библиотеке нет или есть, но работает по другому алгоритму. Можно потребовать от разработчика дополнить библиотеку, а можно разработать собственные ФБ и ВФБ (визуальные функциональные блоки) и считать их своим конкурентным преимуществом при создании систем этого типа (рис. 2).

Если планируется не единичное внедрение системы, то трудозатраты на разработку ФБ окупятся уже на втором объекте. В данном случае была прямая выгода разработки ФБ для типовых обработок (измерение, дискретная паковка, уставка, командная паковка и т.д.), поскольку их количество в проекте исчисляется сотнями. Такие примитивы необходимо выполнять в виде ФБ и ВФБ на основе беззаконных классов. При нестандартной обработке с использованием штатных элементов МастерСКАДА в таком количестве структура дерева объектов становится очень перегруженной, а разработка оказывается вовсе невозможной ввиду очень долгого времени загрузки как в режиме разработки, так и в режиме исполнения. Кроме увеличения скорости работы применение типовых ФБ на порядок сокращает размер проекта на жестком диске и в оперативной памяти, а также снижает загрузку процессора в 2-3 раза.

Для реализации компонентов со сложной графикой и нестандартным поведением были сделаны ВФБ на основе оконных классов. Такой подход позволил использовать все преимущества разработки под .Net. В частности, была решена задача создания элемента управления “Кнопка”, с поведением, схожим с обычной кнопкой Windows. Особенностью работы ВФБ (class CButton) является задержка выходного сигнала нажатия на время цикла объекта, внутри которого она находится, для обеспечения уверенного срабатывания связанных параметров.

Следует отметить, что возможности разработки собственных компонентов и средства доступа к внутренним объектам не требуют дополнительного лицензирования и входят в бесплатную среду разработки МастерСКАДА.

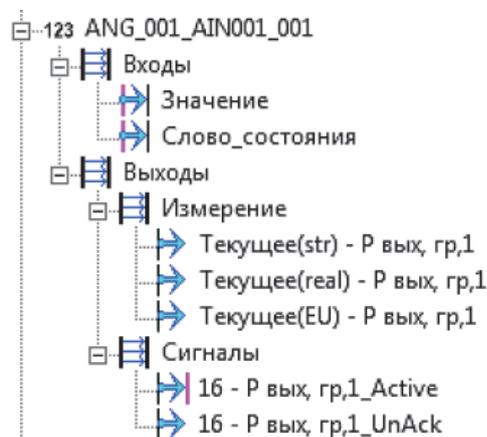


Рис. 2. Вид ФБ “Измерение” в дереве объектов

Для каждого из множества объектов проекта, по клику на мнемосхеме должно выводиться окно управления, вид которого зависит от типа объекта (окно паспорта, окно ввода настроек, уставок или управления исполнительным механизмом). При использовании традиционного для МастерСКАДА объектного подхода, такое окно находится в объекте, т.е. в проекте создается огромное число одинаковых окон, что существенно увеличивает размер проекта и сказывается на производительности. Выходом стало создание по одному окну каждого типа. Для вывода в это окно переменных от выбранного объекта (ВФБ) применяется скрипт С#, который в режиме исполнения устанавливает динамические связи между окнами и логическими объектами.

Для решения данной задачи в МастерСКАДА предусмотрен механизм тиражирования объектов, который называется “типизация”. При использовании этого метода в дереве объектов существует только один объект каждого типа, связи с которым динамически устанавливаются в режиме исполнения. Типизированные объекты имеют ряд ограничений (например, не допускается использование вложенных скриптов), которые не позволили применить их в данном проекте.

ОСОБЕННОСТИ ИСПОЛЬЗОВАНИЯ МОДУЛЯ ТРЕНДОВ

Гордостью компании “ИнСАТ” является функциональность появившегося в версии 3 нового модуля трендов с расширенными настройками и развитыми аналитическими возможностями, включающими одновременный просмотр разных трендов, синхронизацию с журналом, задающие графики, математические обработки перьев (рис. 3)

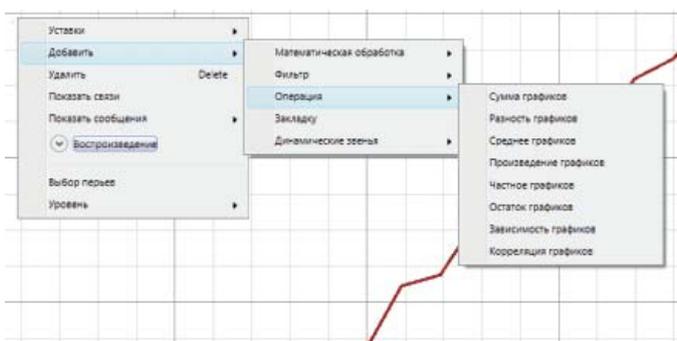


Рис. 3. Пример добавления обработок на тренд параметра

и т.п. В большинстве применений используется лишь небольшая часть имеющейся функциональности, по умолчанию даже загружается “базовый” уровень работы, а не “расширенный”.

Оказалось, здесь есть, что совершенствовать. Реализовать собственные приемы работы позволило использование скрипта С# для взаимодействия с ActiveX компонентом тренда: был реализован контекстный переход к трендам из паспорта сигнала, добавление, настройка отображения графиков (границы диапазона, единицы измерения и т.д.).

Кроме того, для хранения архивных переменных был создан специальный объект, отделенный от объектов контроля и управления. Структура такого объекта соответствует иерархии логических объектов, но имеет более простой вид для оптимизации поиска нужного пера в дереве объектов. Такое решение позволяет избежать потери данных при удалении объекта, отвечающего за визуализацию, сделать быстрой и удобной настройку архива. Стоит заметить, что это решение эффективно в случае разработки типовых проектов, предполагающих постоянное развитие и доработку, изменение алгоритмов обработки внутренней структуры. Для одновременного просмотра такого дерева архива и трендов был создан специальный визуальный компонент (рис. 4).

Для ускорения последующих открытий окно трендов вызывается при старте проекта, но за пределами видимости мониторов. При поступлении команды открытия трендов, окно перемещается в зону экрана с помощью штатного ФБ “Управление документом”.

ПОСТРОЕНИЕ РЕЗЕРВИРОВАННЫХ СИСТЕМ

Любая полнофункциональная SCADA-система имеет возможность создания систем с резервированием серверов опроса, контроллеров, ОРС серверов. Стандартная схема резервирования — это использование двух серверов опроса, один из которых ведет опрос, а второй находится в “горячем резерве”, получает от первого данные синхронизации, а сам начинает опрос в том случае, если основной вышел из строя, или по команде оператора. Некоторые вариации основного алгоритма определяются настройками резервирования

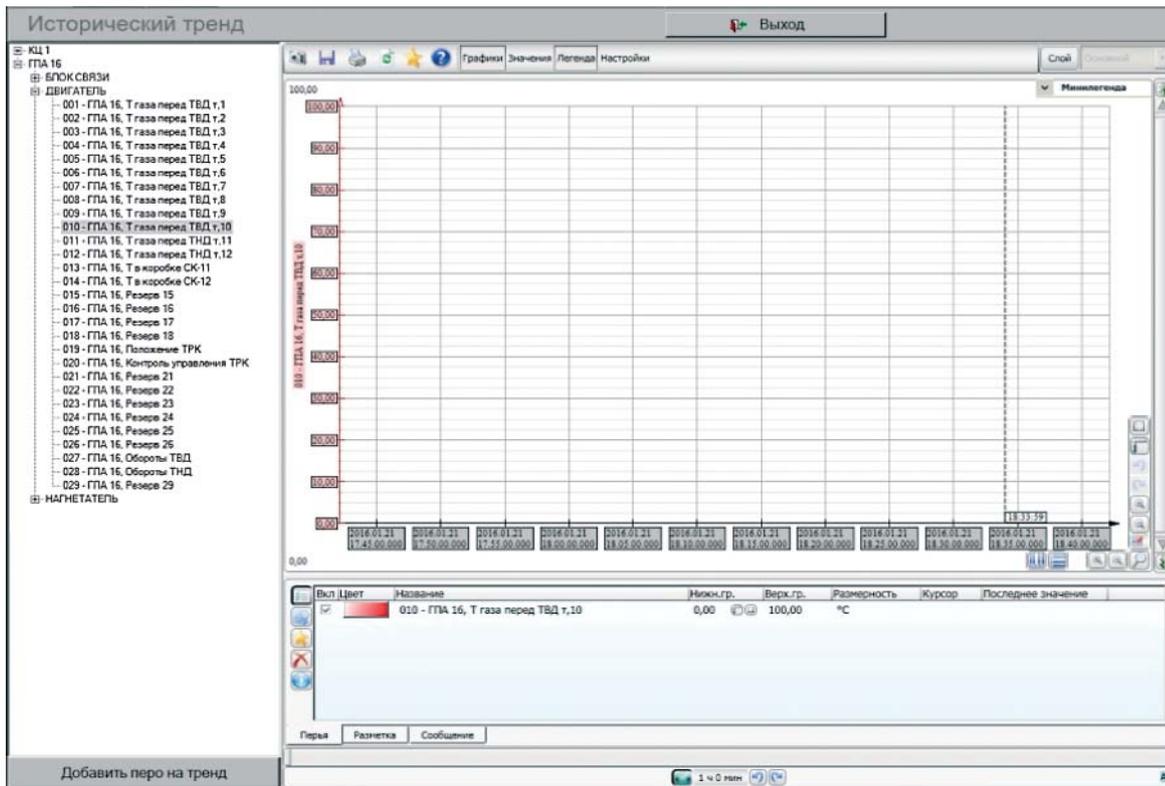


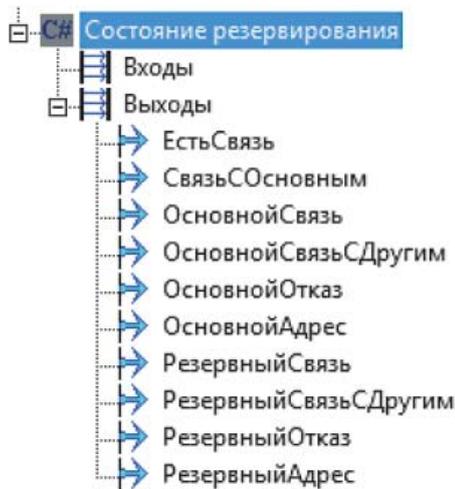
Рис. 4. Пользовательский визуальный функциональный блок

Общие	Опрос	Архив	Окна	Режим исполнения	Действия	Сообщения	Журналы	Права доступа	Операторы	Смены
Контроль системы										
Допустимый объем памяти приложения (МБ)										0
Таймаут выдачи окна о зависании										00:01:00
Межкомпьютерная связь										
Ручная настройка межкомпьютерных связей										Пожь
Настройки связи										
IP-адреса основного компьютера										10.1.108.100
IP-адреса резервного компьютера										10.1.108.200
Адрес сети										
Запрет синхронизации конфигурационных файлов										Истина
Запретить прямое подключение к сетевым БД										Истина
Запуск на нескольких компьютерах										Пожь
Период проверки подключения к сетевым компьютерам										00:00:30
Таймаут подключения к сетевым компьютерам										00:00:30
Резервирование задач										
Резервирование компьютера										
IP порт задачи резервирования										3676
Задержка переключения										00:00:10
Запрет переключения на основной										Пожь
Количество передаваемых архивных значений										50
Кол-во попыток повтора опроса состояния компьютера										3
Период синхронизации										00:00:01
Период синхронизации времени										01:00:00
Таймаут ожидания остановки										00:00:30
Таймаут опроса состояния компьютера										00:00:01
Таймаут перезапуска задачи резервирования										00:00:00
Формирование отчетов										

Рис. 5. Настройки резервирования

(рис. 5). А диагностика состояния основного и резервного серверов опроса – штатная задача МастерСКАДА.

При реализации проекта была решена задача разделения функций системы между автономными станциями визуализации и резер-



▲ Рис. 6. Вид скрипта в дереве объектов

вированным высоконагруженным сервером архивов. Данная архитектура отличается от стандартной архитектуры клиент-сервер, поэтому диагностика работы механизма резервирования серверов, а также контроль за подключением к базе данных доступен только при помощи скриптов С#. В результате работы

над данным проектом для реализации контроля работы сервера в стандартной библиотеке “Примеры скриптов” появился специальный объект (рис. 6).

РЕЗУЛЬТАТЫ

Данная АСУ ТП разрабатывалась в рамках программы импортозамещения, и, конечно, МастерСКАДА – не единственная “новинка”, с которой пришлось столкнуться проектировщикам, разработчикам, программистам и наладчикам. Система успешно прошла стендовые и заводские испытания и в настоящий момент внедряется на реальном объекте. Интегратор – ООО Фирма “Калининградгаз-приборавтоматика” – не только сам получил опыт разработки в новом инструментальном пакете, но и дал много полезной информации разработчикам SCADA-системы для совершенствования своего программного обеспечения, улучшения производительности системы и удобства работы инженеров и пользователей. Большинство доработок вошло в версию 3.7, вышедшую в конце 2015 года.

Веселуха Галина Леонидовна – заместитель генерального директора по проектам, Компания “ИнСАТ”,
Седов Роман – начальник отдела ПО ООО Фирма “КГПА”,
Кабыш Сергей – инженер-программист отдела ПО ООО Фирма “КГПА”.