

# РАЗРАБОТКА ФРЕЙМВОРКА ДЛЯ СИСТЕМЫ КОНТРОЛЯ ТЕХНОЛОГИЧЕСКИХ ПРОЦЕССОВ В МАСШТАБЕ РЕАЛЬНОГО ВРЕМЕНИ

Речкунов Д.И., Боровцов Е.Г.

Алтайский Государственный Технический Университет им. И. И. Ползунова  
(г. Барнаул)

В настоящее время активно предпринимаются меры по автоматизации различных технологических процессов на предприятиях: будь то производственные предприятия или коммунальные службы. Такие организации располагают объектами, к примеру, скважинами, водонапорными башнями, тепловыми пунктами, автоматизированными станками и конвейерами. Подобные объекты требуют постоянного внимания и контроля, но постоянное присутствие человека на них нежелательно. Отсутствие систем удаленной диспетчеризации объектов ведет к излишним экономическим и трудовым затратам, что сказывается на себестоимости производства или предоставления каких-либо услуг. К тому же при использовании системы контроля технологических процессов становится проще сделать прогноз по износу оборудования или проанализировать его эффективность.

Класс таких систем называют аббревиатурой SCADA (SCADA, англ. Supervisory Control And Data Acquisition), что означает «Диспетчерский контроль и сбор данных». SCADA-системы обычно состоят из следующих компонентов:

- Человеко-машинный интерфейс (HMI, англ. Human Machine Interface) – инструмент, который представляет данные о ходе процесса человеку-оператору, что позволяет ему контролировать процесс и управлять им.
- Диспетчерская система – собирает данные о процессе и отправляет команды процессору (управление).
- Удаленный терминальный блок (RTU, англ. Remote Terminal Unit), подсоединяемый к датчикам процесса, преобразует сигнал с датчика в цифровой код и отправляет данные в диспетчерскую систему.
- Программируемый Логический Контроллер (PLC, англ. Programmable Logic Controller) используется как полевое устройство из-за экономичности, универсальности и гибкости, нежели RTU специального назначения. Обладает логикой автоматического управления объектом.

- Коммуникационная инфраструктура для реализации технологической сети.

Рассмотрим ситуацию, когда при разработке SCADA-системы уже имеются все аппаратные решения (RTU, PLC) и диспетчерская система, которая имеет возможность записи всех изменений состояния объекта в указанную базу данных, а также принимает команды управления объектом из сторонних систем. На этом этапе нам необходим человеко-машинный интерфейс, который будет визуализировать данные диспетчерской службы и предоставлять пользователю возможность отправлять этой же службе команды управления объектом. Зачастую, создание такого интерфейса ведет к большим временным затратам в силу серьезного различия объектов наблюдения между собой. Фактически, интерфейс создается заново для каждого объекта, учитывая определенную специфику, что влечет за собой повторную разработку или адаптацию модулей, имеющих одинаковую для всех объектов задачу. Такой процесс станет весомым препятствием при внедрении SCADA-системы на новые объекты, однако, при использовании некоего программного каркаса (фреймворка), в котором реализованы основные моменты работы человеко-машинного интерфейса SCADA-системы, можно серьезно сократить временные и трудовые затраты на процесс внедрения.

С целью решения такой задачи разработан фреймворк для создания человеко-машинного интерфейса SCADA-системы, решающий проблему большого объема работ по созданию визуального представления параметров объекта диспетчеризации и средств его контроля. Фреймворк представляет собой веб-приложение и имеет следующие возможности:

- Выполнение процесса идентификации, аутентификации и авторизации пользователей в системе, а также контроль доступа к функциям системы и объектам контроля через ролевые политики безопасности.
- Ведение журнала работы каждого пользователя в системе, а именно отправки

## РАЗРАБОТКА ФРЕЙМВОРКА ДЛЯ СИСТЕМЫ КОНТРОЛЯ ТЕХНОЛОГИЧЕСКИХ ПРОЦЕССОВ В МАСШТАБЕ РЕАЛЬНОГО ВРЕМЕНИ

команд управления на объект контроля, редактирования пользователями любой информации в базе данных системы в соответствии с политиками безопасности.

- Контроль статуса пользователя (состояние подключения: в системе/не в системе, операционная система пользователя, его браузер, IP адрес, время последнего сеанса, время последнего обращения к системе). Ведение контроля в реальном времени средствами, предназначенными для администраторов системы.

- Контроль входных данных при редактировании пользователями информации, хранимой в базе данных системы.

- Обеспечение одновременного мониторинга группы объектов различного рода с визуальными и звуковыми оповещениями при критических значениях параметров объекта.

- Визуализация подробных мнемонических схем объектов наблюдения, реализованная по принципу интерактивных шаблонов, разрабатываемых с помощью HTML, CSS и JavaScript. Средства контроля наблюдаемых объектов также реализуются в рамках шаблона визуализации.

- Генерация отчетов по параметрам объектов наблюдения за заданный период времени. Просмотр, печать, экспорт в HTML и CSV файлы. Структура отчета также определяется шаблоном, аналогично с шаблонами визуализации.

- Предоставление API для использования возможностей системы сторонними приложениями.

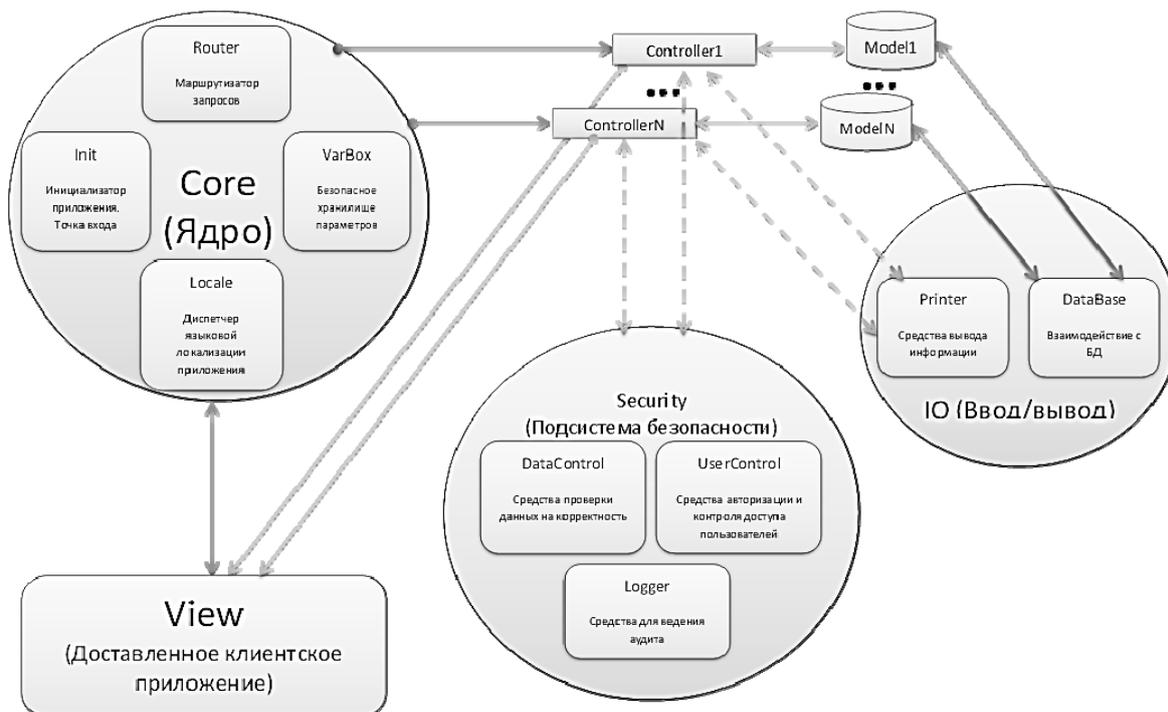


Рисунок 1- Принципиальная схема работы системы

Иными словами, фреймворк представляет собой некую информационную систему, которая является заготовкой для разработчика человеко-машинного интерфейса SCADA-системы. Вероятнее всего, понадобится лишь разработать шаблоны мнемонических схем наблюдаемых объектов и добавить информацию об удаленном PLC объекта в систему, и уже после перечисленных действий будет возможным наблюдение за объектом и формирование отчетов по накопленным данным.

Особое внимание стоит уделить процессу создания мнемонических схем, которые представляют собой специальные HTML-шаблоны состоящие из блоков. Для создания схемы объекта, разработчику достаточно иметь навыки блочной верстки HTML-страниц и, в случае создания более сложных интерактивных схем, знание языка JavaScript. Язык разметки HTML и блочная верстка страниц обладают низким порогом обучаемости и, по сути, не требуют высокой квалификации специалиста и даже навыков в программировании.

нии. Требования, предъявляемые к разрабатываемому шаблону не так объемны: достаточно использовать в качестве идентификатора каждого блока HTML-страницы текст, содержащий в себе название датчика (или команды управления) и условное обозначение типа блока (индикатор, подпись, значение, кнопка). Подробная информация о процессе разработки мнемонических схем объектов представлена в документации системы. Также стоит заметить, что такой подход дает большой уровень гибкости в процессе разработки мнемонических схем и позволяет учесть практически любую специфику наблюдаемых объектов. В дальнейших версиях системы планируется создание графического инструментария для еще более упрощенного процесса создания шаблонов, что предоставит возможность обычным пользователям справляться с процессом создания вышеописанных шаблонов.

Система состоит из двух основных частей: серверной и клиентской. Полная схема работы системы представлена на рисунке 1. Схема взаимодействия клиентской и серверной частей системы представлена на рисунке 2. Основная задача серверной части системы состоит в том, чтобы доставить в браузер пользователя клиентскую часть системы и далее отвечать на ее запросы, при этом

обеспечивая всецелую безопасность работы в системе и механизмы обработки данных. Клиентская часть предоставляет пользователю удобный интерактивный графический интерфейс для работы в системе и взаимодействует с серверной частью посредством асинхронных запросов с использованием технологии AJAX. Ответом на запрос являются данные, предоставляемые в формате JSON. Помимо клиентского приложения с запросами к серверной части системы могут обращаться и сторонние приложения, которые используют особый механизм авторизации. В таком случае система работает со сторонним приложением также как и с обычным пользователем, а значит, ведет журнал всех его действий и запросов к системе.

Получая доступ к системе, пользователь, в зависимости от своих привилегий, может наблюдать за состоянием доступных ему объектов, формировать отчеты за заданный период времени, редактировать информацию об объектах, просматривать и редактировать информацию о других пользователях, просматривать журнал событий системы. Привилегии пользователей распределены в соответствии с ролевыми политиками безопасности. Роли и их привилегии задаются администраторами системы, каждый пользователь может иметь несколько ролей.

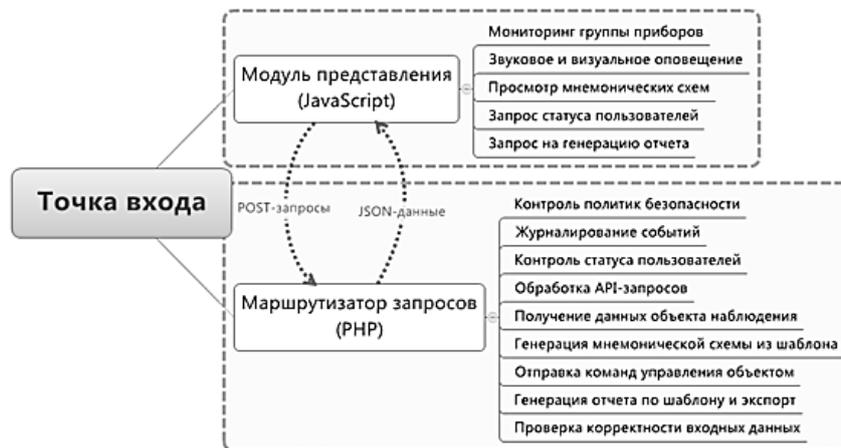


Рисунок 2 - Схема взаимодействия между серверной и клиентской частью системы

Серверная часть системы разработана с использованием языка PHP, под управлением веб-сервера Apache. Для хранения информации используется база данных под управлением любой СУБД из нижеприведенного списка:

- Cubrid

- FreeTDS / Microsoft SQL Server / Sybase
- Firebird/Interbase 6
- IBM DB2
- IBM Informix Dynamic Server
- MySQL 3.x/4.x/5.x
- Oracle Call Interface

## РАЗРАБОТКА ФРЕЙМВОРКА ДЛЯ СИСТЕМЫ КОНТРОЛЯ ТЕХНОЛОГИЧЕСКИХ ПРОЦЕССОВ В МАСШТАБЕ РЕАЛЬНОГО ВРЕМЕНИ

- ODBC v3 (IBM DB2, unixODBC and win32 ODBC)
- PostgreSQL
- SQLite 3 and SQLite 2
- 4D

Клиентская часть системы разработана с использованием библиотеки ExtJS, которая предоставляет разработчику средства для создания оконного интерфейса в веб-приложениях.

Приложение имеет архитектуру соответствующую шаблону проектирования MVC (Model-View-Controller). Добавление нового функционала в систему не является трудоемкой задачей и сводится к созданию классов модели и контроллера, а также модуля для клиентской части приложения на JavaScript (в качестве представления). Стоит отметить,

что в базовом классе модели уже реализован обширный список методов для работы с данными, а в базовом классе контроллера реализованы операции чтения, добавления, удаления, редактирования данных в модели и механизм проверки входных данных на корректность.

Применение данной системы в качестве программного каркаса для создания человеко-машинного интерфейса к SCADA-системе значительно облегчит труд разработчиков и позволит в короткие сроки внедрить систему контроля технологических процессов на промышленный объект и вместе с этим получить все преимущества современных веб-приложений применимо к диспетчеризации объектов.