

## ИСПОЛЬЗОВАНИЕ ТЕХНОЛОГИИ «СОМЕТ AJAX» ДЛЯ РЕАЛИЗАЦИИ ВЕРХНЕГО ФУНКЦИОНАЛЬНОГО УРОВНЯ СИСТЕМ МОНИТОРИНГА ЭНЕРГОПОТРЕБЛЕНИЯ

А.В. Белоусов, Д.Ю. Доценко, Ю.А. Кошлич

Белгородский государственный технологический университет им. В.Г. Шухова

Тел.: (4722) 309-965, e-mail: koshlich@yandex.ru

Современные тенденции развития клиент-серверного программного обеспечения сводят к минимуму различия возможностей «толстых клиентов» и Web-приложений, причем кроссплатформенность работы последних и минимальные аппаратные требования к клиенту делает их развитие все более перспективным. Касательно верхнего функционального уровня современных систем мониторинга растет популярность использования интерфейсов с Web-доступом, что предоставляет возможность наблюдения за состоянием объекта с любого, в том числе мобильного устройства, оснащённого средствами коммуникации, достаточными для подключения к Web-серверу. Сложность организации такого доступа заключается в специфичности взаимодействия с OPC-серверами через интернет, распределенность объектов, а также требование к оперативности, получаемых через интернет данных.

Требование оперативности получаемых данных означает, что отображение информации на стороне клиента должно происходить динамически, без необходимости полной перезагрузки страницы. Данное требование противоречит первоначальной концепции обмена информацией по протоколу HTTP, когда для каждого следующего запроса клиент открывает соединение, которое будет закрыто сразу же после получения ответа от сервера, что исключает возможность частичного обновления запрошенной страницы.

Подходы к решению данной задачи прошли несколько эволюционных этапов развития, каждый из которых нашёл отражение в системах мониторинга.

Среди наиболее распространённых в настоящее время схем функционирования веб-приложений можно выделить схемы, основанные на использовании средств Java-среды и технологий AJAX (рис. 1), в том или ином виде.

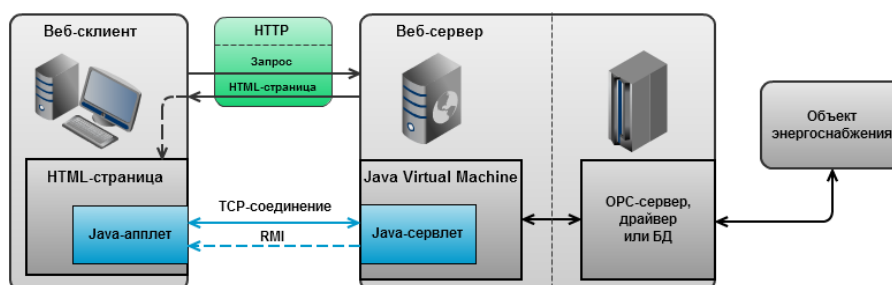


Рис. 1. Схема доступа к технологической информации посредством Java-апплета

Первые снискали свою популярность благодаря тому, что позволяют организовать событийно-ориентированный механизм взаимодействия с клиентом, когда данные передаются сервером непосредственно в момент изменения состояния объекта. При этом информация передаётся клиенту по отдельному TCP-каналу, открытому сервером на нестандартном порту, что создаёт трудности в работе клиентского приложения за сетевым экраном. Более того, при использовании протокола RMI (Remote Memory Invocation) данный порт выбирается случайным образом. Недостатком разумно считать и необходимость использования плагина Java для браузера – одной из самых популярных мишеней для сетевых атак.

AJAX (Asynchronous JavaScript And XML) этих недостатков лишен (рис. 2), однако в классической реализации данная технология не позволяет серверу отправлять обновления клиентам в произвольные моменты времени, определяемые самим сервером. Другие варианты включают в себя применение технологий .NET, таких как ASP.NET и Remote Scripting. В этих случаях, также как и с AJAX, используется только протокол HTTP, а потому невозможно добиться событийно-ориентированного обновления тонкого клиента.

Основной отрицательной чертой этих подходов является их жёсткая привязка к определённой программной платформе.

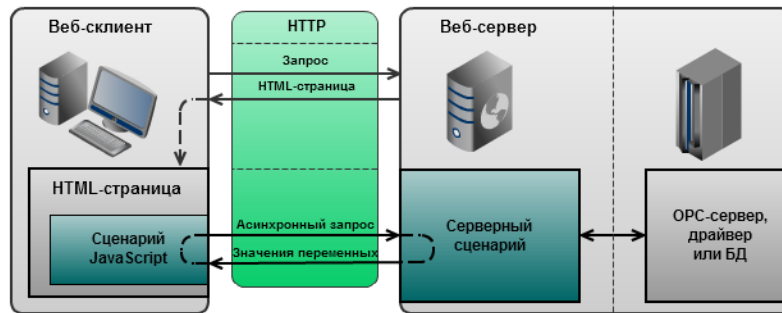


Рис. 2. Схема доступа к технологической информации с использованием AJAX

Результатом изысканий путей устранения указанных недостатков явилась разработка метода, подразумевающего использование технологии реверсивного AJAX и long poll (рис. 3) или Comet AJAX. Общая черта таких моделей состоит в том, что все они основаны на технологиях, непосредственно поддерживаемых браузером, а не на проприетарных плагинах. Теоретически, модель Comet отличается от изначальной концепции всемирной паутины, при которой для обновления страницы браузер запрашивает её полностью или частично.

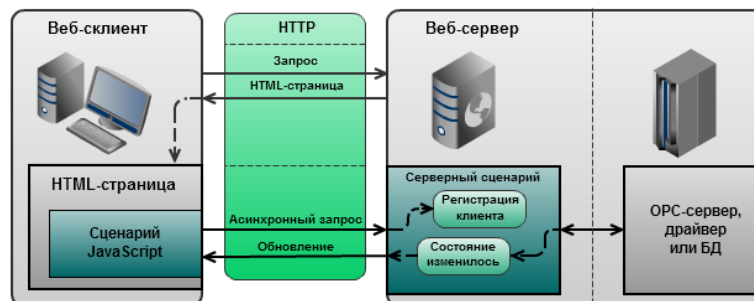


Рис. 3. Структура событийно-ориентированного доступа на основе Comet AJAX

В простейшем виде схема такого клиент-серверного взаимодействия может быть описана следующим образом. После загрузки статической информации с веб-сервера клиент посылает асинхронный HTTP-запрос с информацией, определяющей его текущее состояние. Web-сервер фиксирует это состояние, оставляя HTTP-соединения открытыми. Серверное приложение, осуществляющее опрос датчиков объекта, уведомляет веб-сервер об очередном изменении состояния. Информация о данном изменении отсылается веб-сервером клиенту, после чего клиент закрывает HTTP-соединение.

Организация коммуникации по протоколу OPC реализуется с помощью модуля OpenOPC для Python. Связь с OPC-сервером осуществляется по протоколу OPC-DA 2.05a, который осуществляет взаимодействие с оборудованием нижнего уровня по протоколу Modbus. В качестве веб-сервера выступает распространённый HTTP-сервер, наделённый поддержкой спецификаций WSGI (Web Server Gateway Interface) – Apache 2.2 с модулем mod\_wsgi.

В рамках демонстрационной зоны по энергосбережению БГТУ им. В.Г. Шухова внедрена система мониторинга распределенных объектов электропотребления на основе web-базируемого доступа. Электронный ресурс расположен по адресу <http://ntk.intbel.ru/energo>.

Система мониторинга и оперативной визуализации технологических параметров распределенных объектов электропотребления на основе web-базируемого доступа в полной мере соответствует требованиям современных автоматизированных систем контроля и учета электроэнергии и обеспечивает.

Кроме того, web-интерфейс системы диспетчеризации позволяет выводить графики распределения электрической нагрузки объектов, а также производить оценку энергоэффективности.

В итоге, оператор получает обновления состояния по протоколу HTTP, используя веб-браузер в качестве клиента.

#### Литература

1. Тимирбаев А., Лангманн Р. Web-базируемый доступ к технологической информации Мир компьютерной автоматизации, 5, 2002.
2. Белоусов, А.В. Web-базируемый доступ к технологическим параметрам распределенных объектов энергоснабжения и энергораспределения зданий / А.В. Белоусов, Ю.А. Кошлич, А.Б. Быстров // Инновационные материалы и технологии: сб. докл. Междунар. науч. – практ. конф., Белгород, 11-12 окт. 2011 г. / Белгор. гос. технол. ун-т. – Белгород: Изд-во БГТУ, 2011. – Ч. 1. – С. 45-51.

3. Григорьев А.Б. Взаимодействие с OPC-серверами через Internet Промышленные АСУ и контроллеры, 11, 2002
4. OPC Data Access Automation Interface Specification, Version 2.02. Instead of version 2.01; released 03.02.99. OPC Foundation, 1999.