

**ПРОЕКТИРОВАНИЕ СИСТЕМ «БАНК – КЛИЕНТ»
НА ОСНОВЕ ВЕБ-СЕРВИСА**

М.А. Кузнецов, Д.Х. Чан

Статья посвящена использованию веб-сервиса при разработке систем дистанционного обслуживания клиентов банка. Веб-сервисы позволяют организовывать обмен данными между банком и удаленными клиентами через интернет-сеть при допустимом уровне надежности и безопасности связи. Данная технология осуществляет межплатформенное взаимодействие с возможностью менять архитектуру подсистем как на стороне сервера, так и на стороне клиента. Содержит основные сведения об объекте разработки, требования к системам «Банк – клиент», а также возможные архитектурные решения, удовлетворяющие поставленным требованиям.

Ключевые слова: веб-сервис, «Банк – клиент», распределенное приложение, программная архитектура, асинхронное взаимодействие.

Key words: web-service, “Bank – client”, distributed application, software architecture, asynchronous communication, WCF.

На сегодняшний день электронные банковские услуги являются одной из самых быстро развивающихся областей банковского бизнеса. Предпосылки для быстрого прогресса создаются за счет удешевления технологий, лежащих в основе электронного сервиса. Электронные банковские услуги – это новый способ осуществления банковских бизнес-процессов, суть которого состоит в проведении транзакций с помощью электронных сетей [1].

Взаимодействие нескольких географически удаленных элементов банковской человеко-машинной системы можно увидеть на различных этапах прохождения электронного документопотока. Это межфилиальный документооборот в рамках одного юридического лица, межбанковский обмен внутри государства, а также проведение международных платежей. Но наибольшая гибкость в вопросах обработки данных на стороне клиента, а также жесткость в системах обеспечения надежности и безопасности проявляются в документообороте между банком и клиентами. Это обусловлено как различными требованиями со стороны разных клиентов к обработке данных, так и существенными отличиями в программном и аппаратном обеспечении. В одном случае клиент желает получать и посылать платежные документы, используя мобильные устройства. В другом – ему необходимо генерировать и обрабатывать документы автоматически, на основе взаимодействия с несколькими подсистемами в рамках вычислительных сетей организации. Причем структура сети в каждом конкретном случае тоже может существенно различаться, что накладывает ограничения на технологии обработки данных.

Сферой наиболее сложной области проектирования электронных систем в силу выше-названных причин следует считать систему «Банк – клиент». В классическом понимании она позволяет клиенту без выезда в офис кредитной организации получать банковские услуги с помощью взаимодействия через интернет-сеть [5]. Например, получение информации о состоянии счета, перевод денег между счетами как внутри банка, так и между банками, отправление платежных поручений, получение выписок об операциях по счетам, конверсионные безналичные операции. Список можно значительно расширить, если учитывать воз-

ИНФОРМАЦИОННО-ТЕЛЕКОММУНИКАЦИОННЫЕ СИСТЕМЫ И ТЕХНОЛОГИИ

возможные безналичные операции, начиная от кредитования и кончая безналичной оплатой покупок в магазине.

Работа с подобными системами требует использования следующего сценария:

- 1) аутентификация пользователя в системе на основе сертификата (для юридических лиц) или пароля (для физических лиц);
- 2) просмотр состояния счета и получение выписки из банка;
- 3) создание платежных документов;
- 4) отсылка документов в банк;
- 5) верификация успешности операции и корректировка документов в случае ошибок.

Описанные действия могут повторяться несколько раз в течение дня в произвольном порядке.

Для выполнения операций можно использовать браузер или специальное программное обеспечение, установленное на компьютерах клиента банка (рабочие станции). Браузер наиболее удобен для физических и мелких юридических лиц. Как правило, этим категориям клиентов банка не требуется специальных действий для создания документов. Достаточно заполнить вручную формы в веб-браузере и передать содержимое на обработку в банк. Криптозащита современных браузеров достаточна для проведения большинства безналичных операций.

Веб-технология позволяет предоставить клиентам довольно широкий спектр унифицированных услуг в любой сфере безналичного обслуживания. Она не требует установки дополнительного программного обеспечения и позволяет осуществлять программирование полностью на стороне сервера, т.е. сопровождение и обновление системы осуществляется, минуя большинство операций на стороне клиента.

Однако крупные клиенты, имеющие собственные автоматизированные системы документооборота, требуют обеспечения электронного взаимодействия с системой «Банк – клиент». Естественно, программирование таких возможностей становится более узкоспециализированным и ведет к привлечению прикладных специалистов. В этом случае необходимо дополнительная обработка данных на стороне клиента. Кроме того, подобный процесс может осуществляться распределенной системой.

Таким образом, важными особенностями системы «Банк – клиент» являются гибкость, кроссплатформенность, возможность масштабирования как на стороне сервера (банка), так и на стороне клиента, высокая надежность и безопасность. Можно детализировать эти основные требования:

- реализация бизнес-процессов с помощью веб-браузера (один из вариантов кроссплатформенности и одновременно обеспечение унификации операций для нескольких клиентов);
- высокая производительность, надежность и масштабируемость. Это важный и обязательный критерий для обслуживания большого количества клиентов;
- возможность взаимодействия с бухгалтерскими автоматизированными системами на стороне клиента;
- возможность интеграции с автоматизированными банковскими системами;
- безопасность при передаче информации по линиям связи и защита от возможных подделок со стороны клиентов или третьих лиц.

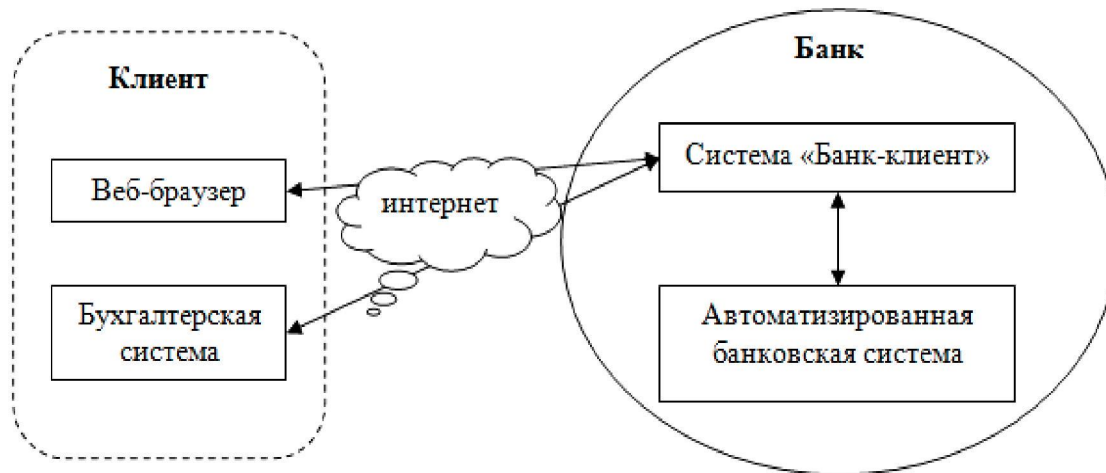


Рис. 1. Схема требуемой системы «Банк – клиент»

Таким образом, архитектура системы должна соответствовать схеме, приведенной на рис. 1. При построении системы «Банк – клиент» только на основе классической веб-технологии возникает ряд проблем, основной из которых является отсутствие удобной возможности обмена данными с бухгалтерской системой, используемой на предприятии.

На помощь в решении проблемы приходят веб-сервисы. Они обеспечивают взаимодействие программных систем независимо от платформы и языков программирования на высоком уровне абстракции. Это обеспечивается использованием стандартного протокола обмена сообщениями SOAP. SOAP позволяет равноценно использовать себя в приложениях, написанных на широком спектре универсальных языков (к примеру: Java, C#, VB.NET), а также на предметно-ориентированных (например, 1С).

В настоящее время имеется достаточное количество инструментов, как платных, так и открытых. Они значительно облегчают написание как самих веб-сервисов, так и приложений-клиентов, использующих возможности веб-сервисов [2, с. 22].

Потребителем веб-сервиса может быть как само веб-приложение (например, JavaScript, AJAX или Java-applet), так и любое другое программное обеспечение, установленное на рабочих станциях клиента. Такие программы обращаются к веб-сервису через специальную библиотеку, входящую в комплект разработчика потребителя веб-сервиса. Для наглядности рассмотрим общую схему построения системы и ее использования на примере Windows ориентированных систем (см. рис. 2).

Веб-сервис предоставляет собой, с точки зрения потребителя, унифицированный интерфейс для того, чтобы пользовательские приложения могли обратиться к нему за получением необходимой информации или передать ему какие-либо данные.

Взаимодействие между клиентами и веб-сервисом обеспечивается с помощью сообщений. Для избежания потерь сообщений необходимо привлечение промежуточного слоя – специального программного обеспечения для транспортировки и обработки сообщений (Message Oriented Middleware, MOM).

ИНФОРМАЦИОННО-ТЕЛЕКОММУНИКАЦИОННЫЕ СИСТЕМЫ И ТЕХНОЛОГИИ

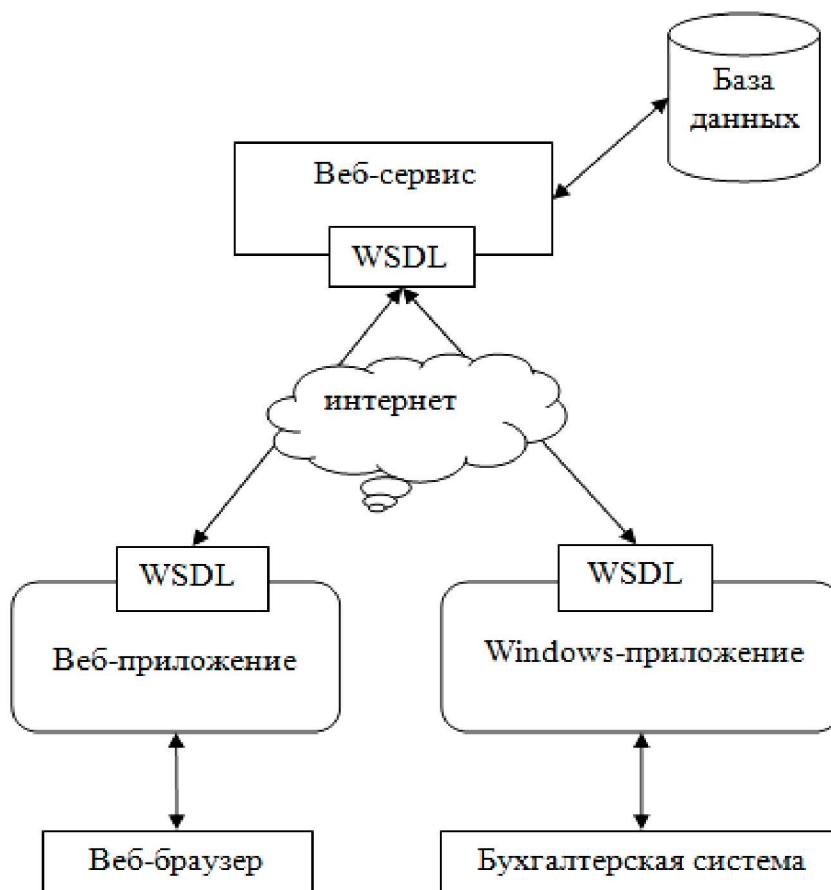


Рис. 2. Применение веб-сервиса в распределенной системе

МOM – это библиотека, которая находится в обеих частях клиент-серверной архитектуры и поддерживает асинхронные вызовы между клиентским и серверным приложением [4]. Наличие очереди сообщений позволяет обеспечивать их временное хранение в случае, если основные процессы (как приложения-серверы, так и приложения-клиенты) заняты или в конкретный момент времени отключены от сети.

Структура системы «Банк – клиент» с использованием MOM выглядит следующим образом (рис. 3).

Использование веб-службы позволяет удобно и эффективно обеспечить доступ всех клиентов к банковскому серверу. Систему можно считать эффективной в том случае, если каждый подключенный к системе дистанционного обслуживания клиент банка в любой момент времени будет иметь информацию о своих средствах и может управлять ими, независимо от аппаратной платформы и установленного у него программного обеспечения.

Конечная реализация системы «Банк – клиент» требует выбора средств программирования и библиотек, обеспечивающих работу веб-сервиса. За счет кроссплатформенности веб-сервисов такой выбор можно сделать независимо друг от друга на стороне банка и его клиентов.

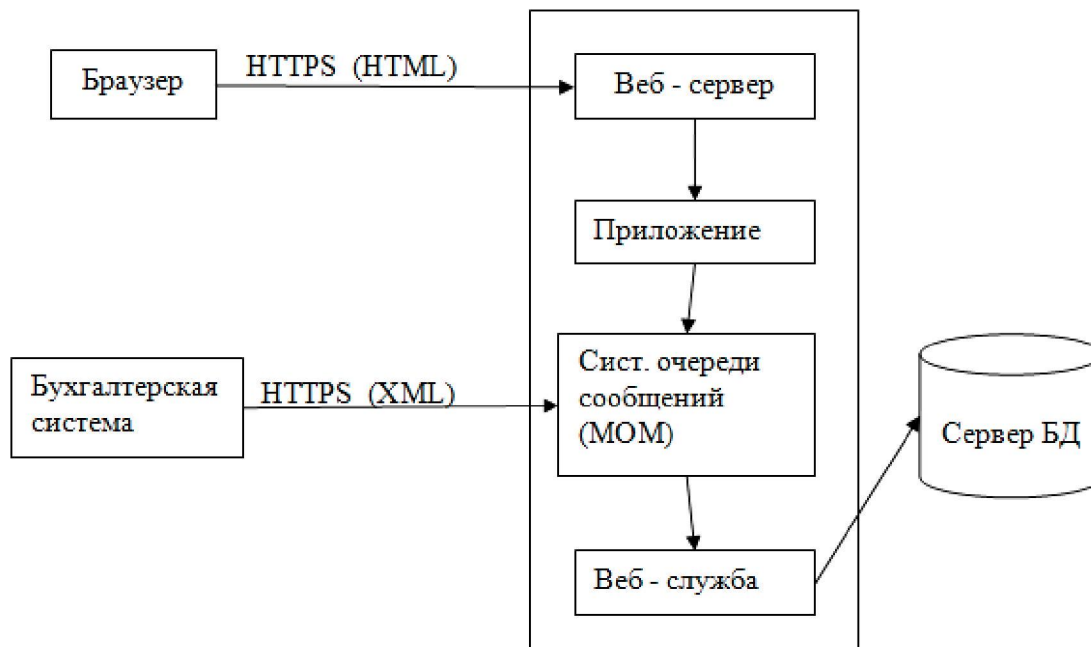


Рис. 3. Схема взаимодействия модулей системы «Банк – клиент» через MOM

Данной системе требуются эффективные коммуникации одновременно с широкими возможностями по обеспечению обработки данных. Комбинация требований заставляет разработчиков либо отказываться от удовлетворения некоторых требований, либо создавать собственную технологию построения распределенных приложений.

Рассмотрим особенности проектирования поставщика веб-сервиса (т.е. банковского сервера). Учитывая распространение в коммерческой среде операционных систем Windows, роль продукции Microsoft в данной области становится весомой, если не главенствующей. Рассмотрим одну из последних технологий для обеспечения коммуникации между различными узлами распределенной системы, активно рекламируемой Microsoft. Это Windows Communication Foundation.

Windows Communication Foundation (WCF) – это унифицированная интегрированная среда для создания защищенных, надежных, транзакционных и интероперабельных распределенных приложений.

До появления WCF Microsoft поддерживала следующие основные технологии для распределенных вычислений: PRC, WSE, ASMX, .NET Remoting, COM+, MSMQ. В прошлом выбор наиболее подходящей технологии для распределенного приложения зависел от требований, предъявляемых к этому приложению. Если требовалось обеспечить совместный доступ к информации от разных платформ, использовалась веб-служба ASMX. Если достаточно было только обеспечить передачу данных между клиентом и сервером, использующим ОС Windows, то использовалось удаленное взаимодействие .NET Remoting. Если требовалось обеспечить транзактные коммуникации, то нужно было использовать Enterprise Services (DCOM), а если требовалось реализовать модель с организацией очереди, то нужно было использовать Message Queuing (или MSMQ).

WCF включает в себе функциональные возможности всех этих технологий в унифицированную модель программирования, как показано в таблице [3, с. 29].

ИНФОРМАЦИОННО-ТЕЛЕКОММУНИКАЦИОННЫЕ СИСТЕМЫ И ТЕХНОЛОГИИ

Справедливости ради стоит заметить, что WCF не предоставляет средств для неограниченной комбинации функциональных требований, однако количество возможных комбинаций гораздо больше, чем было до этого в других технологиях.

Таблица

Сравнение возможностей WCF [3, с. 30]

Возможность	WSE	ASMX	Remot-ing	COM+	MSMQ	WCF
Поддержка WS-*	+			+		+
Взаимодействие с Web-сервисами		+		+		+
Коммуникация .NET к NET			+			+
Распределенные транзакции				+	+	+
Гарантированная доставка сообщений					+	+

Изначально WCF проектировалась для сетевого взаимодействия с другими системами. Такими системами могут быть автоматизированная банковская система или бухгалтерская система, работающие под разными операционными системами и/или на других платформах. Построенная на базе технологии WCF система «Банк – клиент» может взаимодействовать с приложениями, которые принимают сообщения WS-* и XML. В свою очередь, они могут передаваться по TCP, HTTP, именованным каналам и MSMQ. С точки зрения сетевого взаимодействия, WCF является единой технологией, обладающей возможностями других существующих технологий. Это позволяет вести речь об унификации разных подходов в рамках одной технологии.

Систему «Банк – клиент» можно считать безопасной при передаче информации только по защищенным линиям связи (требуется как защита от подделок в сообщениях, так и защита от разглашения информации). Это налагает на систему следующие требования: целостность, секретность, аутентификация. Они же, в свою очередь, приводят к необходимости предоставления возможности управления доступом (авторизации).

WCF поддерживает несколько механизмов, обеспечивающих безопасность передачи данных и авторизацию:

- транспортный способ безопасности использует протокол транспортного уровня, такой как HTTPS, чтобы достигнуть необходимый уровень безопасности. Транспортный способ имеет преимущество быть широко принятым и доступным на многих платформах. Он также менее сложен в вычислительном отношении. Однако у него есть недостаток обеспечения только двухточечного обмена данными;

- способ обеспечения безопасности сообщения использует WS-Security, чтобы осуществить безопасную передачу. Поскольку безопасность сообщения применена непосредственно к сообщениям SOAP, она имеет преимущество независимости от транспортного протокола. Этот механизм обеспечивает непрерывную безопасность в рамках целой системы.

Таким образом, целесообразно для реализации систем «Банк – клиент» на стороне банка использовать WCF. Ее функциональные возможности гибко интегрируются в существующие системы банковского документооборота. При этом высока кроссплатформенность WCF с точки зрения дистанционного подключения удаленных клиентов, использующих другие программные и аппаратные платформы. Это позволяет проектировать, строить, отлаживать и сопровождать распределенные системы существенно быстрее и обеспечивает большую функциональность, чем это было возможно до появления WCF.

ПРИКАСПИЙСКИЙ ЖУРНАЛ: управление и высокие технологии № 2 (10) 2010

Библиографический список

1. *Вагин, В. Н.* Кому и зачем нужен интернет-банкинг? / В. Н. Вагин // Мир интернет. – 1999. – № 9. – С. 20–22.
2. *Веб-сервис: XML, WSDL, SOAP и UDDI* : пер с англ. – М. : Питер, 2003. – 256 с.
3. *Основы Windows Communication Foundation*: Пер. с англ. – М. : Русская Редакция ; БХВ-Петербург, 2008. – 384 с.
4. *Программное* обеспечение промежуточного слоя для обработки сообщений. – Режим доступа: <http://www.cio-world.ru/infrastructure/system/29702/print.html>, свободный. – Заглавие с экрана. – Яз. рус.
5. *Система* «Банк – Интернет – Клиент». – Режим доступа: <http://www.moscow-bank.ru/ClientBank.htm>, свободный. – Заглавие с экрана. – Яз. рус.

УДК 004.428.4

ПРОГРАММНЫЙ КОМПЛЕКС ДЛЯ ДЕМОНСТРАЦИИ ПРИМЕНЕНИЯ КОРРЕКТИРУЮЩЕГО КОДИРОВАНИЯ

М. О. Смирнова

В статье представлен программный продукт, который демонстрирует основные этапы применения корректирующих кодов. Дано описание основных компонентов программного продукта и возможностей использования при изучении теории информации и кодирования.

Ключевые слова: информация, передача информации, демонстрационная программа, корректирующие коды, программный код.

Key word: information, transmission of information, the demonstration program, corrective code, the program code.

В процессе хранения данных и передачи информации по сетям связи неизбежно возникают ошибки. Контроль целостности данных и исправление ошибок – важные задачи на многих уровнях работы с информацией (в частности, физическом, канальном, транспортном уровнях модели OSI).

Корректирующие коды, помехоустойчивые коды, коды обнаружения и исправления ошибки, применяются при передаче и обработке информации в вычислительной технике, телеграфии, телемеханике и технике связи, где возможны искажения сигнала в результате действия различного рода помех. Кодовые слова корректирующих кодов содержат информационные и проверочные разряды (символы) [5].

В процессе кодирования при передаче информации из информационных разрядов в соответствии с определенными для каждого корректирующего кода правилами формируются дополнительные символы – проверочные разряды. При декодировании из принятых кодовых слов по тем же правилам вновь формируют проверочные разряды и сравнивают их с принятыми; если они не совпадают, при передаче произошла ошибка. Существуют коды, обнаруживающие факт искажения сообщения, и коды, исправляющие ошибки, т.е. такие, с помощью которых можно восстановить первичную информацию [8].