

## ИСПОЛЬЗОВАНИЕ СОВРЕМЕННЫХ ВЕБ-ТЕХНОЛОГИЙ ДЛЯ СОЗДАНИЯ ГЛОБАЛЬНЫХ СИСТЕМ ЭЛЕКТРОННОГО ДОКУМЕНТООБОРОТА

Баринов Евгений Андреевич  
ОАО «Центр информатизации РМ»,  
Российская Федерация, г. Саранск  
E-mail: [f248zb@gmail.com](mailto:f248zb@gmail.com), тел. +7 (927) 1874661,  
430005, г. Саранск, ул. Богдана Хмельницкого, д. 39, ком. 503.

**Аннотация.** В статье рассматривается подход к построению системы электронного документооборота, основанный на использовании веб-технологий. Описываются различные технологии для разработки.

**Ключевые слова:** СЭД; веб-интерфейс; веб-сервер; электронный документооборот; PHP; ASP.NET.

### Введение

Система электронного документооборота (СЭД) — организационно-техническая система, обеспечивающая процесс создания, управления доступом и распространения электронных документов в компьютерных сетях, а также обеспечивающая контроль над потоками документов в организации.

В современной организации системы электронного документооборота (СЭД) становятся обязательным элементом ИТ-инфраструктуры. С их помощью повышают эффективность деятельности коммерческие компании и промышленные предприятия, а в государственных учреждениях на базе технологий электронного документооборота решаются задачи внутреннего управления, межведомственного взаимодействия и взаимодействия с населением.

Изначально системы этого класса рассматривались лишь как инструмент автоматизации задач классического делопроизводства, но со временем стали охватывать все более широкий спектр задач. Сегодня разработчики СЭД ориентируют свои продукты на работу не только с корреспонденцией и ОРД (организационно-распорядительными документами), но и с различными внутренними документами (договорами, нормативной, справочной и проектной документацией, документами по кадровой деятельности и др.). СЭД также используются для решения прикладных задач, в которых важной составляющей является работа с электронными документами: управление взаимодействием с клиентами, обработка обращений граждан, автоматизация работы сервисной службы, организация проектного документооборота и др. Фактически системой электронного документооборота называют любую информационную систему, обеспечивающую работу с электронными документами.

### Обзор функций и требований к СЭД.

На сегодняшний день существует несколько систем электронного документооборота, сочетающих в себе различные функции:

- обеспечение более эффективного управления за счет автоматического контроля выполнения, прозрачности деятельности всей организации на всех уровнях.
- поддержка системы контроля качества, соответствующей международным нормам.
- поддержка эффективного накопления, управления и доступа к информации и знаниям. Обеспечение кадровой гибкости за счет большей формализации деятельности каждого сотрудника и возможности хранения всей предыстории его деятельности.
- протоколирование деятельности предприятия в целом (внутренние служебные расследования, анализ деятельности подразделений, выявление "горячих точек" в деятельности).

- оптимизация бизнес-процессов и автоматизация механизма их выполнения и контроля.
- исключение или максимально возможное сокращение оборота бумажных документов на предприятии. Экономия ресурсов за счет сокращения издержек на управление потоками документов в организации.
- Исключение необходимости или существенное упрощение и удешевление хранения бумажных документов за счет наличия оперативного электронного архива.

Типовые требования к СЭД.

Система электронного документооборота должна:

- обеспечивать надежное хранение документов и их описаний;
- обеспечивать жизненный цикл документа (его создание, хранение версий, публикация, блокировка доступа к изъятому документу, передача документа для хранения в архиве);
- допускать задание пользователем различных типов документов, создания и редактирования карточек для них;
- поддерживать иерархию категорий для эффективного поиска документа;
- осуществлять поиск документов на основе информации из карточки, а также полного текста;
- обеспечивать разделение доступа к документам на уровне отдельных пользователей, по ролевому принципу, и на основе иерархической структуры организации;
- поддерживать технологию HSM;
- протоколировать все события, связанные с работой пользователей и самой системы; необходимо наличие развитых средств администрирования;
- поддерживать удаленный доступ к информации.

Продвинутые системы должны поддерживать:

- кластерные технологии для обеспечения бесперебойной работы;
- территориально распределенные организации;
- алгоритмы шифрования при хранении и передаче данных;
- цифровую подпись.

Требования к архитектуре:

- наличие выделенного сервера приложений;
- наличие тонкого клиента; поддержка доступа к документам с использованием браузера.
- кроссплатформенность для обеспечения масштабируемости;

Требования к открытости и интеграции с другими системами:

- интеграция со средствами потокового ввода документов;
- интеграция с офисными приложениями;
- интеграция с электронной почтой;
- наличие развитого программного интерфейса (API);
- интеграция со стандартными службами каталогов (к примеру, LDAP) для ведения и синхронизации списка пользователей системы;
- возможность адаптации пользовательского интерфейса под конкретные задачи;
- возможность дополнения системы собственными специализированными компонентами;

### **Обозначение, методы и средства решения задачи. Принципиальная схема функционирования.**

В настоящее время существует множество СЭД. Написаны они для множества современных платформ: Microsoft Windows, Unix-подобных, BSD. Имеют различные виды клиентских приложений от толстых и тонких клиентов, также написанных в нескольких различных вариациях под те же платформы до веб-клиентов, поддерживающих «правильную» кроссплатформенность. Но у всех существующих СЭД имеется существенный недостаток, который не позволяет выйти электронному документообороту на глобальный уровень. Недостаток заключается в том, что компании не имеют возможности оперативно обмениваться между собой различными документами в электронном виде. Решение этой проблемы определено выведет электронный документооборот на качественно новый уровень.

Решение заключается в создании единой СЭД, организованной по принципу веб-сайта. Организация СЭД в виде веб-приложения имеет ряд преимуществ, присущих всем веб-приложениям:

- кроссплатформенность;
- глобальный доступ;
- централизованное администрирование;
- высокая степень защищенности от сбоев.

Кроме того, компании, используя такие глобальные СЭД, получают большие организационные и экономические преимущества:

- отсутствие необходимости в квалифицированных специалистах для развертывания и поддержки сложных и громоздких СЭД на рабочих станциях;
- отсутствие необходимости дорогостоящего оборудования и программного обеспечения для функционирования серверных частей;

Архитектуру веб-ориентированной СЭД можно разделить на три функциональных блока:

- Центральное веб-сервер.
- Сервер баз данных;
- Клиентское приложение (веб-браузер).

Центральное веб-сервер является сердцем системы и реализует логику работы, механизмы аутентификации, разграничения прав доступа, резервирование данных, правила работы с СУБД и файлами. Сервер баз данных осуществляет хранение, резервирование, в определенной мере контроль доступа, защиту данных.

Архитектура предложенного метода показана на рисунке 1.

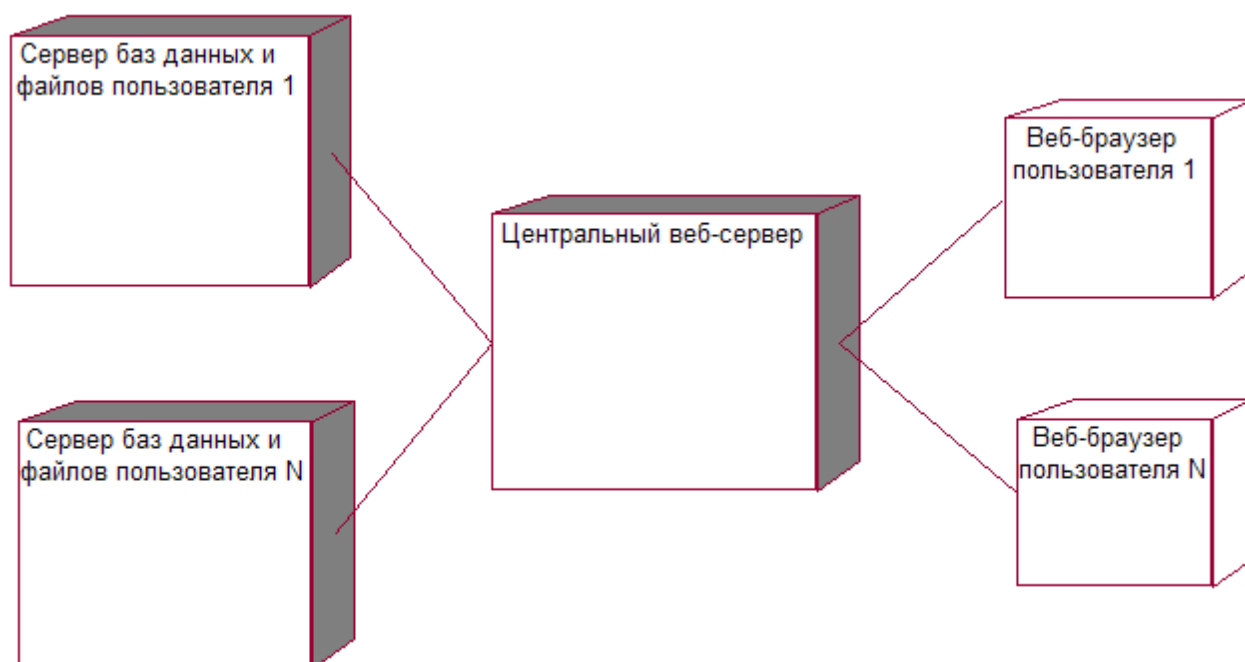


Рисунок 1.

Для построения такой архитектуры можно выбрать несколько подходов и средств.

**Подход 1. Использование популярного средства создания веб-приложений скриптового языка PHP, технологии AJAX, СУБД MySQL и веб-сервера Apache.**

Из преимуществ языка PHP следует выделить следующие:

- автоматическое извлечение POST и GET-параметров, а также переменных окружения веб-сервера в predefined массивы;
- взаимодействие с большим количеством различных систем управления базами данных (MySQL, MySQLi, SQLite, PostgreSQL, Oracle (OCI8), Oracle, Microsoft SQL Server, Sybase, ODBC, mSQL, IBM DB2, Cloudscape и Apache Derby, Informix, Ovrimos SQL, Lotus Notes, DB++, DBM, dBase, DBX, FrontBase, FilePro, Ingres II, SESAM, Firebird /InterBase, Paradox File Access, MaxDB, Интерфейс PDO);
- широкие возможности использования ООП (начиная с версии PHP5)
- наличие фреймворков, облегчающих разработку крупномасштабного веб-приложения, такого как глобальная СЭД.
- кроссплатформенность.

Для обеспечения полной интерактивности, дружелюбности к пользователю интерфейса веб-приложения является предпочтительным использовать технологию AJAX (Asynchronous JavaScript and XML). AJAX - подход к построению интерактивных пользовательских интерфейсов веб-приложений, заключающийся в «фоновом» обмене данными браузера с веб-сервером. В результате, при обновлении данных, веб-страница не перезагружается полностью, и веб-приложения становятся более быстрыми и удобными. AJAX базируется на двух главных принципах:

- использование технологии динамического обращения к серверу «на лету», без перезагрузки всей страницы полностью, например:
  - с использованием XMLHttpRequest (основной объект);
  - через динамическое создание дочерних фреймов;
  - через динамическое создание тега <script>;
- использование DHTML для динамического изменения содержания страницы;

Преимущества AJAX:

- Экономия трафика;
- Уменьшение нагрузки на сервер;
- Ускорение реакции интерфейса.

Недостатки AJAX:

- Отсутствие интеграции со стандартными инструментами браузера;
- Динамически загружаемое содержимое недоступно поисковикам (если не проверять запрос, обычный он или XMLHttpRequest);
- Старые методы учёта статистики сайтов становятся неактуальными;
- Усложнение проекта;
- Требуется включенный JavaScript в браузере.

На рисунке 2 показано отличие в функционировании обычного веб-приложения (слева) и веб-приложения, «обогащенного» AJAX(справа).

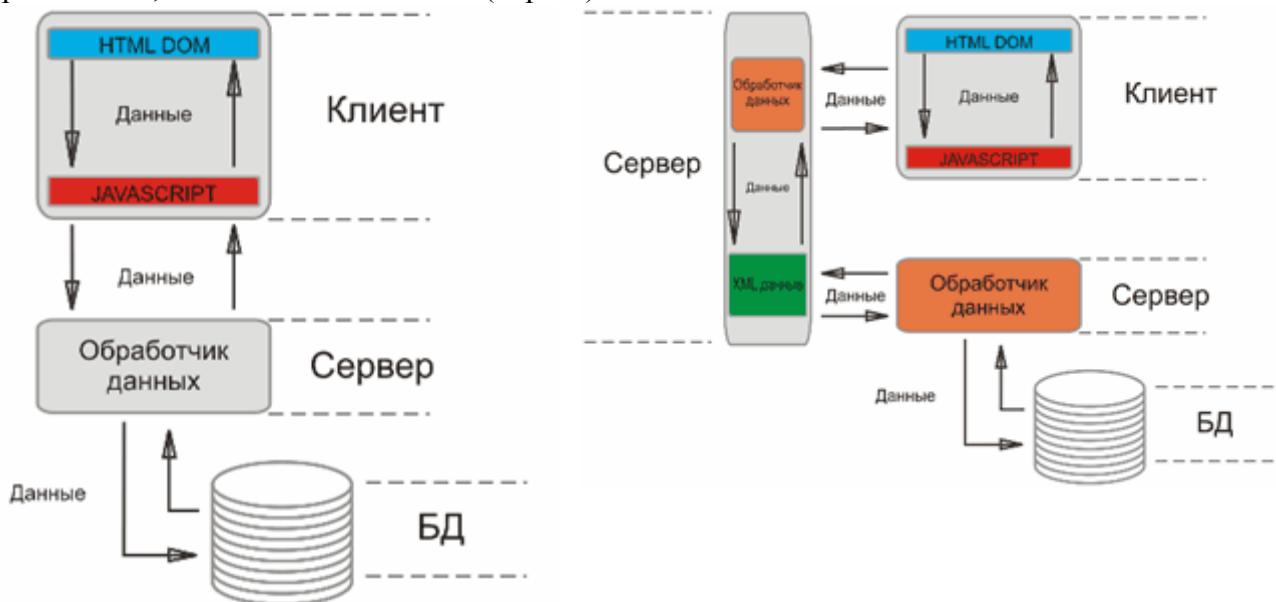


Рисунок 2.

Выбор СУБД и веб-сервера не является однозначным в данном вопросе, но наиболее предпочтительно использовать связку MySQL - Apache, так как этим средствам схожи общие преимущества, а именно надежность, гибкость, высокий уровень безопасности, быстрое действие, масштабируемость.

### Подход 2. Использование технологии Microsoft ASP.NET и фреймворка ASP.NET MVC Framework.

ASP.NET является объединенной платформой веб-приложений, обеспечивающей выполнение служб, необходимых для сборки и развертывания веб-приложений на уровне предприятий. ASP.NET предоставляет новую модель и инфраструктуру программирования для создания более защищенных, масштабируемых и устойчивых в работе приложений, предназначенных для любых средств просмотра или устройств. ASP.NET является частью платформы Microsoft .NET Framework, компьютерной среды, упрощающей разработку приложений для исключительно разветвленной среды Интернета. .NET Framework включает среду CLR, которая обеспечивает выполнение основных служб, например управления памятью, управления потоками, защиты кода программ. Среда также включает библиотеку классов .NET Framework, которая представляет обширное объектно-ориентированное собрание типов, используемых разработчиками для создания приложений.

ASP.NET предлагает следующие возможности:

- Управляемость. ASP.NET использует текстовую иерархическую систему конфигурации, упрощающую применение параметров к среде сервера и веб-приложений. Поскольку сведения о конфигурации сохраняются в виде простого текста, новые параметры могут применяться без помощи локальных средств

администрирования. Все изменения файлов конфигурации распознаются автоматически и применяются к приложению.

- Безопасность. ASP.NET предоставляет стандартные схемы авторизации и проверки подлинности для веб-приложений. Разработчики могут без труда добавлять, удалять или изменять данные схемы в зависимости от требований приложения.
- Простота развертывания. Приложение ASP.NET развертывается на сервере простым копированием необходимых файлов на сервер. Перезапуск сервера не требуется даже для развертывания и замены выполняющегося скомпилированного кода.
- Повышенная производительность. ASP.NET является скомпилированным кодом, выполняющимся на сервере. Возможности раннего связывания, компиляции по требованию JIT (just-in-time), оптимизации кода для выполнения на конкретном компьютере и кэширования служб позволяют повысить производительность страниц ASP.NET по сравнению с производительностью обычных страниц ASP.
- Гибкое выходное кэширование. В ASP.NET возможно кэширование данных страницы, частей страницы или целых страниц, в зависимости от требований приложения.
- Многоязыковая поддержка. ASP.NET использует кодировку Юникод на внутреннем уровне для представления данных запросов и ответов. Параметры многоязыковой поддержки можно настроить для каждого компьютера, каталога и для каждой отдельной страницы.
- Поддержка мобильных устройств. ASP.NET поддерживает все виды обозревателей на всех видах устройств. Для новых мобильных устройств разработчики используют те же технологии программирования, что и для традиционных обозревателей рабочего стола.
- Масштабируемость и доступность. При создании ASP.NET учитывались возможности масштабируемости, специально разработанные для улучшения производительности в разветвленных многопроцессорных средах. Более того, процессы подвергаются пристальному наблюдению и управлению со стороны служб IIS и среды ASP.NET CLR, поэтому при сбое одного из процессов вместо него создается новый процесс, что помогает поддерживать приложения в рабочем состоянии для обработки запросов.
- Трассировка и отладка. ASP.NET содержит службы трассировки, которые могут быть включены во время отладки на уровне приложения и на уровне страницы. Сведения можно просматривать на странице или с помощью средства просмотра трассировки на уровне приложения. ASP.NET поддерживает локальную и удаленную отладку с помощью средств отладки .NET Framework как в процессе разработки, так и в процессе работы приложения. При внедрении приложения инструкции трассировки могут быть оставлены в коде производства, что не отражается на производительности.
- Интеграция с .NET Framework. Поскольку ASP.NET является частью .NET Framework, мощность и гибкость всей платформы доступны веб-приложениям. К библиотеке классов .NET, как и к решениям службы сообщений и доступа к данным, без труда можно получить доступ через сеть. ASP.NET не зависит от языка программирования, поэтому разработчики могут выбрать язык, наилучшим образом подходящий для приложения. Кроме того, совместимость среды CLR сохраняет существующую функциональность в разработке на основе COM.
- Совместимость с существующими приложениями ASP. ASP и ASP.NET могут параллельно выполняться на веб-сервере IIS, не вступая в конфликт.

Повреждение существующего приложения ASP при установке ASP.NET невозможно. ASP.NET обрабатывает только файлы с расширением .aspx. Файлы с расширением .asp продолжают обрабатываться программами ASP. Однако следует иметь в виду, что состояние сеанса и состояние приложения не являются общими для страниц ASP и ASP.NET.

ASP.NET включает две возможности для распределенных приложений: веб-формы и веб-службы XML. Эти возможности поддерживаются одной инфраструктурой конфигурации и отладки.

- Технология веб-форм разрешает сборку мощных веб-страниц на основе форм. Страницы веб-форм используют повторно используемые встроенные или специальные компоненты для упрощения кода страницы.
- Веб-службы XML, создаваемые с помощью ASP.NET, разрешают удаленный доступ к серверам. С помощью веб-служб XML предприятия могут создавать программные интерфейсы для данных и основных рабочих задач, к которым, в свою очередь, приложения клиентов и серверов могут получить доступ для работы с ними. Веб-службы XML обеспечивают обмен данных с помощью таких стандартов, как служба сообщений XML и параллельный брандмауэр HTTP в сценариях клиент-сервер и сервер-сервер. Веб-службы XML можно вызвать из программы, написанной на любом языке и выполняющейся на любой операционной системе.

Изначально ASP.NET спроектирована на использование в качестве СУБД Microsoft SQL Server. Это достаточно мощная и тяжелая СУБД, требующая больших капитальных вложений, но куда более функциональная, чем MySQL. С другой стороны, мы можем использовать и MySQL в паре с ASP.NET. Для этого необходимо использовать MySQL ODBC Driver на машине, с которой будет осуществляться подключение к базе данных.

### **Выводы.**

В результате были проанализированы достижения, перспективы развития в области систем электронного документооборота. Было рассмотрено два подхода к проектированию и созданию веб-ориентированных СЭД. Плюсами первого подхода являются: полная поддержка кроссплатформенности, бесплатность, простота перехода на новую версию среды. Минусы заключаются в отсутствии четко ориентированной парадигмы программирования, низкая производительность по сравнению с ASP.NET. Плюсы второго подхода: высокая производительность, наличие компилятора, широкая реализация объектной модели, встроенная поддержка AJAX. Основным минус ASP.NET – это необходимость использования на стороне сервера лицензионного платного программного обеспечения Microsoft. Основываясь на этих исследованиях, можно сделать вывод о том, что выбор того или иного подхода зависит от конкретной ситуации даже в рамках одной и той же задачи. Но именно использование веб-технологий наиболее оправданно для построения глобальных СЭД.

## Список литературы

1. СЭД // Портал ITeam [Электронный ресурс]. Режим доступа: [http://www.iteam.ru/publications/it/section\\_64/article\\_2688/](http://www.iteam.ru/publications/it/section_64/article_2688/). (дата обращения: 02.12.2011)
2. Исследование российских СЭД // Портал ITeam [Электронный ресурс]. Режим доступа: [http://www.iteam.ru/publications/it/section\\_64/article\\_2886/](http://www.iteam.ru/publications/it/section_64/article_2886/). (дата обращения: 02.12.2011)
3. AJAX // Портал Wikipedia [Электронный ресурс]. Режим доступа: <http://ru.wikipedia.org/wiki/Ajax>. (дата обращения: 03.12.2011)
4. Кристиан Дари, Богдан Бринзаре, Филип Черчез-Тоза AJAX и PHP. Разработка динамических веб-приложений – Москва: Изд-во Символ-Плюс, 2009. – 336 с.
5. ASP.NET // Microsoft Technet [Электронный ресурс]. Режим доступа: [http://technet.microsoft.com/ru-ru/library/cc783228\(WS.10\).aspx](http://technet.microsoft.com/ru-ru/library/cc783228(WS.10).aspx) (дата обращения: 04.12.2011)
6. Dino Esposito Programming Microsoft ASP.NET 4 – Redmond, Washington: Microsoft Press, 2011 – 992 с.

## USE OF THE MODERN WEB TECHNOLOGIES TO CREATE GLOBAL ELECTRONIC DOCUMENT MANAGEMENT SYSTEMS

Barinov A. Eugene  
JSC "Center of Informatization of RM"  
The Russian Federation, Saransk  
E-mail: [f248zb@gmail.com](mailto:f248zb@gmail.com), tel. +7 (927) 1874661,  
39 Bohdan Khmelnytsky str., room 503, 430005, Saransk, Mordovia, Russia

**Abstract.** The paper considers the approach to the construction of electronic document management system, based on the use of Web technologies. Describes the various technologies for development.

**Keywords:** EFD; the web interface; web server; electronic document management, PHP, ASP.NET.