



В.Д. Чертовской
Санкт-Петербургский государственный электротехнический
университет «ЛЭТИ» имени В.И. Ульянова (Ленина)

ПОСТРОЕНИЕ РАСПРЕДЕЛЕННЫХ СИСТЕМ С ПОМОЩЬЮ ВЕБ-ПОРТАЛОВ И ВЕБ-САЙТОВ

В статье освещены вопросы построения и использования поисковых систем. Представлена систематизация данных о таких инструментах цифровизации, как веб-порталы и веб-сайты. Приведены их определения и свойства, указана их связь. Рассмотрены классификация, структура, особенности, технология построения. Показаны прикладные примеры реализации поисковых систем с порталами и систем управления.

Веб-портал, веб-сайт, понятия, классификация, структура, методы, программирование.

Одним из эффективных направлений развития экономики является ее цифровизация, предполагающая построение распределенных систем. В реализации таких систем выделяются две технологии связей локальных систем: «клиент – сервер» и одноранговая. Технология «клиент – сервер» чаще используется в системах управления. Вопросам применения этой технологии с упорядочением информации о ней посвящена работа [2]. Однако у пользователей и, особенно, у разработчиков систем управления возникает потребность в обращении к поисковым системам, использующим одноранговую технологию.

Предметом данной публикации, служащей фактически развитием работы [2], является построение и использование поисковых систем с одноранговой технологией. Инструментами этой технологии являются веб-порталы и веб-сайты. В связи с этим акцент в данной работе сделан на исследовании свойств этих инструментов и возможностей их использования в распределенных системах. При этом одновременно проводится систематизация разбросанной в разных источниках многочисленной информации о названных инструментах.

Первоначально следует разобраться в терминологии поисковых систем. В последнее время в таких системах появилось много новых терминов: веб-портал, веб-сайт, веб-сервер, веб-клиент, веб-сервис. Как правило, при их использовании определения часто не даются, а если и приводятся, то в разных, порой плохо доступных источниках. Это затрудняет работу опытных специалистов, не говоря о начинающих пользователях. Требуется, прежде всего, систематизировать терминологию и свести необходимые термины в одном источнике. Эта цель и преследуется в настоящей публикации.

Рассмотрим специфику веб-порталов и веб-сайтов, а затем приведем примеры прикладной реализации. Изложение информации о порталах и сайтах приведем в одной последовательности:

1. Определение и назначение.
2. Классификация.
3. Структура.
4. Особенности.
5. Технология создания и стандарты.

Веб-портал [1] – набор сайтов в компьютерной сети, который предоставляет пользователю различные интерактивные сервисы и интернет-сервисы.

Порталы являются точками доступа к информации в интернете и помогают пользователям в поиске информации в поисковых системах. Информация

предоставляется из различных источников в единообразном виде, в одном месте, с привлечением наибольшего числа пользователей.

Порталы разделяются по нескольким классификационным признакам.

По специализации информации выделяют горизонтальные и вертикальные порталы.

Горизонтальный портал (универсальный, общего назначения) охватывает множество тематик, представляет собой набор сервисов и ориентирован на максимально широкую аудиторию с максимальным учетом интересов. Наиболее известные горизонтальные порталы: Yahoo!, Mail.ru (русскоязычный), Yandex.

Вертикальный портал (нишевый) характеризуется узкой тематической направленностью. Он предоставляет различные сервисы для пользователей сети по определенным интересам и ориентирован на определенную тематику или область деятельности.

По направленности пользователя выделены корпоративный и публичный порталы.

Корпоративный портал – совокупность информационных систем и баз данных предприятия, организации, представленных в интернете.

Корпоративный портал предоставляет сотрудникам компании строго определенные права доступа к автоматизированной системе управления (системе подготовки принятия решений, экспертной системе, системе совместной работы).

Публичный портал является диаметральной противоположностью корпоративного портала. Он обеспечивает доступ для информации неограниченного круга лиц.

В структуре портала выделяют три составляющих.

Базовая инфраструктура обеспечивает работу системы безопасности, управление порталом, транзакциями. В нее входят веб-сервер, сервер баз данных и приложения.

Интеграция приложений отвечает за взаимодействие портала с различными системами компании, СУБД, CRM, ERP.

Интерфейс предназначен для управления наполнением информацией. Включает инструменты для работы с беспроводными и мобильными устройствами, служб обмена данными с проектами бизнес-партнеров, портлеты (программные модули).

Особенностями веб-порталов являются:

- 1) прямой доступ ко всей размещенной информации,
- 2) единый доступ ко всем приложениям,

- 3) наличие средств поиска данных,
- 4) публикация документов, поддержка документооборота, наличие каталогов документации,
- 5) поддержка групповой работы пользователей с персонализацией доступа,
- 6) гибкие и мощные инструменты для разработки приложений,
- 7) соответствие требованиям, предъявляемым к информационным системам предприятий,
- 8) поддержка беспроводного и мобильного доступа,
- 9) возможность интеграции с информационными системами партнеров и другими приложениями.

Рассмотрим наиболее популярные технологии создания порталов.

InfoExchange Portal. Здесь можно создавать порталы B2E и B2C для обмена информацией с поставщиками, для управления неструктурированным и структурированным информационным наполнением, для поддержки внутренних транзакций, для удаленного администрирования и персонализации. Каждый пользователь может изменять вид сайта для удобного доступа к приложениям. Работа с документами и данными основывается на ролях.

WebSphere Portal Server. Приложение позволяет строить корпоративные и горизонтальные порталы. Портал предоставляет данные, охватывающие информацию новостных агентств, неструктурированную информацию, пакеты приложений, созданные независимыми разработчиками, офисные пакеты, системы управления информационным наполнением ресурсов, файловые системы.

SharePoint Server. Приложение позволяет управлять содержимым порталов, обеспечивать совместную работу пользователей, предоставлять доступ к информации. Система предлагает различные средства для быстрого создания узлов публикации.

Можно отметить также следующие средства: Oracle 9iAS Portal, iPlanet Portal Server, Enterprise Portal, Sybase Enterprise Portal, CleverPath Portal (Computer Associates, CA), Enterprise Information Portal (Hummingbird), PeopleSoft Enterprise Portal (PeopleSoft), Plumtree Portal Server (Plumtree Software), SAP Portals Enterprise Portal (SAP Portals).

Web Services for Remote Portlets (WSRP) – сетевой протокольный стандарт для связи с удаленными портлетами.

JSR 168 (Java Portlet Definition Standard) определяет контракт между портлетами и контейнером портлетов и предоставляет удобную программную модель разработчикам портлетов.

Веб-сайт – информация компьютерной сети, объединенная под одним адресом (доменным именем или IP-адресом). Для доступа к данным сайта можно использовать меню, «картинки», метки. Это совокупность документов частного лица или организации. По умолчанию подразумевается, что сайт располагается в сети Интернет. Для прямого доступа клиентов к веб-сайтам на серверах был специально разработан протокол HTTP.

Выражение «своя страничка в Интернет» подразумевает целый веб-сайт или личную страницу в составе чужого сайта. Кроме веб-сайтов, в сети Интернет доступны WAP-сайты для мобильных телефонов.

Изначально веб-сайты представляли собой совокупность статических документов. В настоящее время для них характерна динамичность и интерактивность.

Для таких случаев используют термин «приложение» – готовый программный комплекс для решения задач веб-сайта. Приложение входит в состав сайта, однако без данных работать не может.

Для повышения безопасности протоколов создано защищенное HTTPS-соединение. Оно применяется в личных кабинетах и интернет-магазинах.

Чаще всего в Интернете одному веб-сайту соответствует одно доменное имя. По доменным именам сайты идентифицируются в глобальной сети. Возможны и варианты: один сайт на нескольких доменах или несколько сайтов под одним доменом. Обычно несколько доменов используют крупные сайты (веб-порталы), чтобы логически отделить разные виды предоставляемых услуг (mail.google.com, news.google.com, maps.google.com). Нередки случаи выделения отдельных доменов для разных стран или языков. Например, google.ru и google.fr логически являются сайтом Google на разных языках, но технически это разные сайты. Объединение нескольких сайтов под одним доменом характерно для бесплатных хостингов.

Обычный сайт все равно посвящен одной тематике. **Форумы** являются местом для общения и обсуждения определенной темы. **Сайт-визитка** представляет основные данные о предприятии. **Блог** – это авторский дневник.

Задача перечисленных ресурсов – предоставить материалы в полном объеме по одному определенному предмету.

Задача интернет-портала в том, чтобы помочь посетителю найти ответ на свой вопрос: с помощью ссылок направить его на внешние сервисы и нужный ресурс с запрашиваемой информацией. Интернет-портал является площадкой со сложной структурой и большими функциями, чем у сайта.

Имеется несколько классификационных признаков веб-сайтов.

По доступности сервисов: открытые, полуоткрытые, закрытые.

По природе содержимого: статические, динамические (содержимое генерируется специальными скриптами).

По физическому расположению: внешние сайты сети Интернет, локальные сайты.

По схеме представления информации, ее объему и решаемым задачам: сайт-визитка (визитные карточки), корпоративный сайт, каталог продукции, интернет-магазин, промо-сайт (торговая марка), сайт-квест (интернет-ресурс с разгадыванием последовательности взаимосвязанных логических загадок).

По услугам: веб-сервис, доска объявлений, каталог сайтов, поисковые сервисы, почтовые сервисы, веб-форумы, блог-сервис, файлообменный пиринговый сервис, дата-хостинговый сервис, дата-эдиторинговый сервис, фото-хостинг, хранение видео.

Страницы сайтов – файлы с текстом, размеченным на языке HTML. Эти файлы обрабатываются браузером и выводятся на его средство отображения (монитор, экран, принтер или синтезатор речи). Язык HTML позволяет форматировать текст, создавать гипертекстовые ссылки и вставлять в отображаемую страницу изображения, звукозаписи, мультимедийные элементы. Отображение страницы можно изменить добавлением в нее таблицы стилей на языке CSS или сценариев на языке JavaScript.

Страницы могут быть простым статичным набором файлов или создаваться специальной компьютерной программой на сервере, называемой движком сайта. Движок может быть сделан на заказ для некоторого сайта, либо являться готовым продуктом, рассчитанным на выбранный класс сайтов. Ряд движков может обеспечить владельцу сайта гибкие настройки структурирования и выводов информации на веб-сайт. Эти движки называют системами управления содержанием.

Сайт предполагает наличие двух субъектов: источник данных и пользователь. Следовательно, между ними должна быть связь. Методы связи определяются применяемой технологией [2, 3]: клиент-сервер, одноранговые. Перечень методов представлен в таблице 1.

Таблица 1

Методы связей в веб-сайтах

Технология клиент-сервер	Одноранговая технология
Web	Сервис-ориентированная архитектура
Модель «клиент – сервер»	Веб-сервисы
Объектные системы	Технологии одноранговых сетей
Агентные технологии	Технологии Grid
Компонентные системы	Облачные вычисления

Данные по назначению и использованию могут быть двух видов: информационные и управляющие. Источником информационных данных может быть любой запрашиваемый сайт. Полученная из него информация может быть использована, как правило, в других системах сайтов или в ручном режиме. Управляющая информация получается из целенаправленных сайтов и используется в системе пользователя для принятия решений.

Все данные делят на структурированные (таблицы) и неструктурированные (текст, графика). Управляющие системы используют преимущественно структурированную информацию, что предполагает применение баз данных (БД). Поскольку управляющие системы все чаще имеют многоуровневую структуру, возникает проблема взаимодействия БД. БД может быть единой для системы или представлять собой набор связанных локальных БД [4–6].

Чтобы упростить выбор конкретного метода, в таблицах 2 и 3 приведены сравнительные характеристики перечисленных методов.

Таблица 2

Сравнительные характеристики технологий распределенной структуры «клиент – сервер»

Способ	Достоинства	Недостатки
Web	Простота структуры	Сложность обеспечения безопасности данных, применение в основном для поисковых систем
Модель «клиент – сервер»	- Простота структуры - Возможность применения в системах управления	- Трудность определения безопасности - Применение для систем относительно небольшой размерности, при росте количества клиентов повышаются требования к мощности сервера и пропускной способности канала - Устойчивая работа всех клиентов зависит от загрузки и функционирования одного сервера
Объектные системы	- Простота разработки распределенных приложений в сравнении с клиент-серверным подходом - Возможность разработки приложений для гетерогенных вычислительных сред с помощью виртуальных машин и независимого описания интерфейсов взаимодействующих компонентов	- Реализация удаленных вызовов гораздо сложнее реализации вызовов локальных процедур - Проблема передачи данных с адресного пространства с одной машины на другую - Неоднородность языков программирования и операционных сред: структуры данных и структуры вызова процедур, строгая ограниченность данной технологии платформой Java, ограниченность масштабируемости
Агентные технологии	Возможность доступа к разным точкам источника данных	- Ряд серьезных проблем с безопасностью агентных платформ - Трудности определяют принцип работы технологии: любой администратор вычислительной сети не позволит агентам путешествовать по компьютерам пользователей
Компонентные системы	- Пригодность для разработки крупных систем - Простая быстрая процедура инсталляции - Увеличение доли повторного применения кода - Снижение стоимости программного обеспечения - Унификация обработки объектов различной природы - Процесс создания программного обеспечения меньше зависит от человека	Высокие требования при разработке компонент, ограниченность технологии платформой Java

Сравнительные характеристики технологий распределенной структуры одноранговых систем

Способ	Достоинства	Недостатки
Сервис-ориентированная архитектура	• Пригодность для разработки средних по размерам систем • Увеличение доли повторного применения кода • Снижение стоимости программного обеспечения • Унификация обработки сервисов различной природы	• Недостаточная апробация программного обеспечения • Нет возможности работы с состояниями
Веб-сервисы	• Пригодность для разработки средних по размерам систем • Увеличение доли повторного применения кода • Снижение стоимости программного обеспечения • Унификация обработки сервисов различной природы • Обеспечение безопасности данных	• Достаточно сложная архитектура
Технологии одноранговых сетей	• Упрощается поддержка масштабируемости при значительном росте количества узлов в вычислительной сети • Повышается отказоустойчивость сети • Возможность объединения ресурсов отдельных участников сети при возможности предоставления услуг другим участникам • Отсутствие зависимости от централизованных сервисов и ресурсов • Пригодность для систем с большим объемом данных и обменом информацией между пользователями	• Рост требований к производительности каждого компьютера • Низкая степень защищенности машин из-за открытого доступа • Трудность учета гетерогенности аппаратного и программного обеспечения потенциальных участников • Сложность поиска доступных ресурсов без использования централизованной точки управления • Затруднения в вопросах безопасности предоставления ресурсов в основном индивидуальным пользователям
Технологии Grid	• Объединение крупных комплексов обработки и хранения данных, благодаря свойствам гетерогенности, масштабируемости, приспособляемости	• Сложная инфраструктура, которую целесообразно применять для сверхбольших систем
Облачные вычисления	• Сокращение в 3–5 раз стоимости бизнес-приложений и более чем в пять раз стоимости приложений для конечных пользователей • Оплата по мере использования ресурса • Масштабируемость и гибкость • Высокий уровень загрузки оборудования и снижение затрат на его эксплуатацию	• Сложности организации репликации данных между сервисами • Сложность администрирования в процедуре функционирования • Сложность архитектуры • Высокие риски в защите информации

В качестве программных средств могут быть использованы СУБД MySQL и SQLite, языки программирования Java и HTML. Возможные сочетания программных инструментов реализации приведены в таблице 4.

Таблица 4

Возможные варианты использования программных средств

Технология «клиент – сервер»		Одноранговая технология
СУБД	Язык программирования	
MySQL	PHP	Java
MySQL	Java	HTML
SQLite	Java	DHTML

Работа с технологией «клиент – сервер», чаще используемая в системах управления, описана в публикациях [7–8]. Для информационных систем с неструктурированной информацией предпочтительно приме-

нение языка HTML [9–12], в том числе с применением графики [13].

Вопросам технологии создания сайтов посвящены работы [14, 15].

Изготовление сайтов может вестись своими силами либо обращением к специализированным разработчикам (фрилансерам, студиям, бюро, конторам). Отношения между заказчиком и исполнителем регулируются с помощью договоров, технических заданий.

Технология создания корпоративных и публичных порталов несколько отличается.

Первоначально, независимо от типа портала, определяется цель реализации и реализуемый объект (система или процедура), выявляется структура объекта. Далее начинаются различия.

При применении корпоративного портала выявляются особенности системы, требования к методам математического описания и выбор метода. Им может быть однородный метод [2]. Осуществляется матема-

тическое описание и проводится переход к программированию. Выбирается метод связи и программные инструменты. Ими являются СУБД (MySQL, SQLite), промежуточный слой (например, Apache) и языки программирования (PHP, в последнее время все чаще Java, обладающий большими возможностями).

При применении публичного портала процедура создания гораздо сложнее.

Веб-дизайнеры разрабатывают макеты шаблонов страниц, определяют интерфейс доступа потребителя к информации и услугам сайта. Страницы могут включать графические элементы, подготовкой которых занимаются технические дизайнеры, шрифтовики. Готовые шаблоны предъявляют заказчику. Чтобы макеты выглядели более наглядно, в них помещается дополнительное содержимое, называемое на сленге дизайнеров рыбой. Если заказчик удовлетворен внешним видом шаблонов, то проводится верстка страниц сайта.

Верстка гораздо сложнее: макеты шаблонов представляют либо в виде изображений (например, в формате JPEG или PNG), либо разбитых по слоям (например, в PSD или AI). Надо получить из графических макетов гипертекстовые веб-страницы с подготовленными для интернета изображениями.

Одним из сложных моментов является обеспечение совместимости с множеством браузеров (программ для просмотра веб-страниц), которые могут одни и те же элементы разметки или правила CSS интерпретировать по-своему.

Когда получится, что большинство браузеров одинаково отображают готовые шаблоны, переходят к веб-программированию.

Программист создает программную основу сайта с нуля, используя фреймворк или CMS. Выбор языка программирования (HTML, CSS, Java) не принципиален. Полученный сайт заполняют данными.

В сайте может содержаться конфиденциальная информация, доступ к которой посторонним лицам

чреват финансовыми и моральными потерями. Мотивами вмешательства злоумышленников в сайт могут быть ревность, искажение информации.

Наиболее распространенными последствиями атаки на сайт могут быть:

- несанкционированное изменение злоумышленниками-хакерами содержания сайта (дефейсинг);
- подделка сайта (дизайн и содержимое сайта могут быть скопированы, а пароли – украдены).

Необходимо обеспечить таким сайтам должный уровень защиты, состав которой зависит от важности информации.

Обсудим вопросы реализации публичного и корпоративного порталов. Публичным порталам и соответствующим сайтам посвящено значительное количество литературы [8, 15, 16–20], в то время как по корпоративным порталам литературы мало [2]. В связи с этим первоначально обсудим особенности, связанные с корпоративным порталом.

Рассмотрим вопросы реализации на примере адаптивной автоматизированной системы управления производством [2].

Создание системы. Организационная структура системы представлена на рисунке 1.

В качестве метода математического описания системы принят однородный метод. Общая схема компьютерной реализации такой системы показана на рисунке 2.

В данной работе использована схема, отраженная на рисунке 3. В качестве СУБД принята SQLite. Языком программирования принят Java. Технологией реализации является клиент-сервер, а методом связи элементов – компонентный.

Схема связей БД приведена на рисунке 4.

Программная схема системы представлена на рисунке 5. Назначение программ показано в таблице 5.

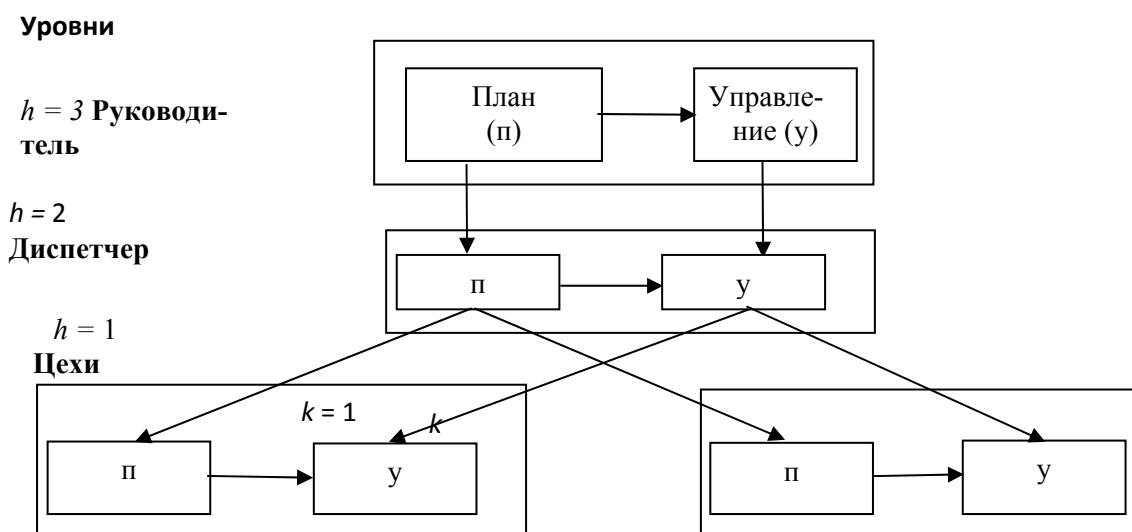


Рис. 1. Структура системы управления производством

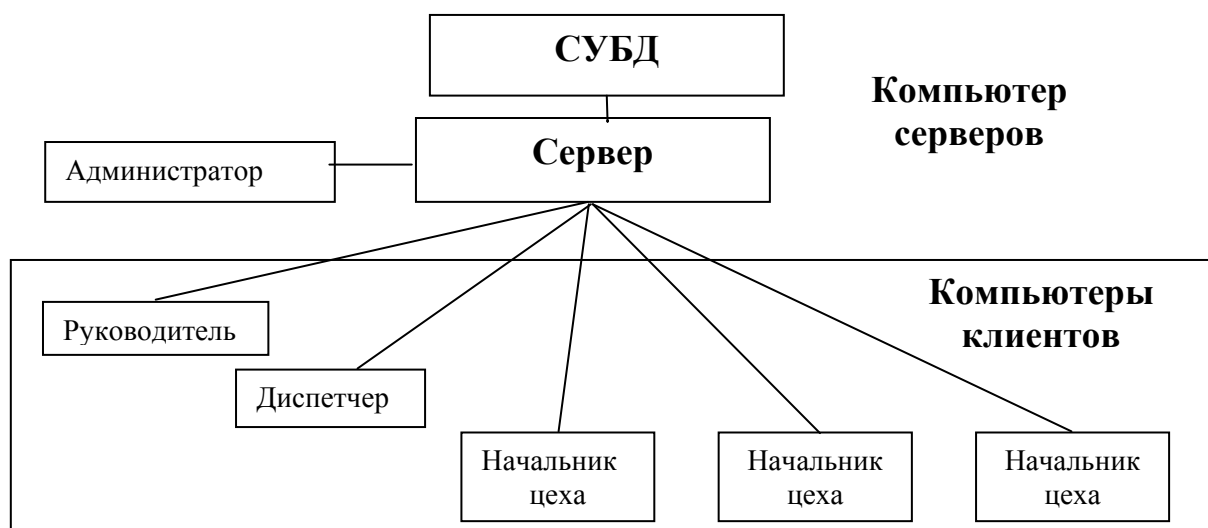


Рис. 2. Схема компьютерной реализации

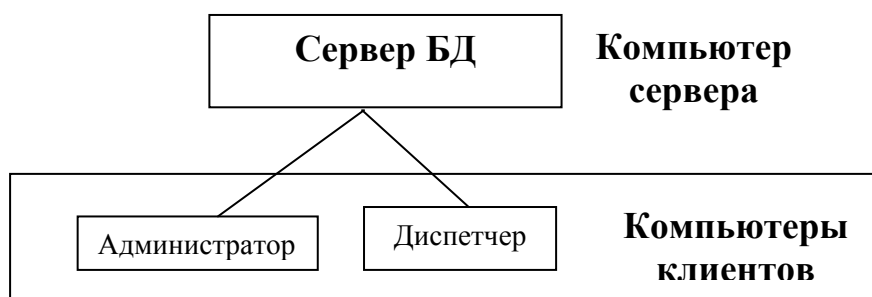


Рис. 3. Схема компьютерной реализации

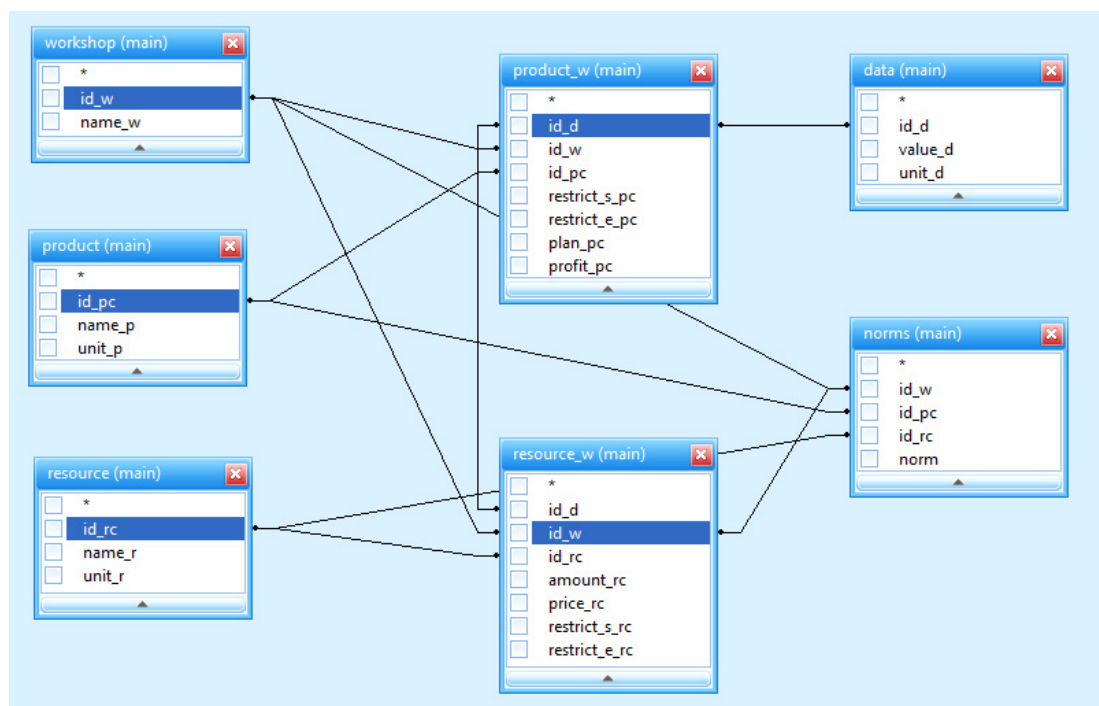


Рис. 4. Схема связей базы программ данных

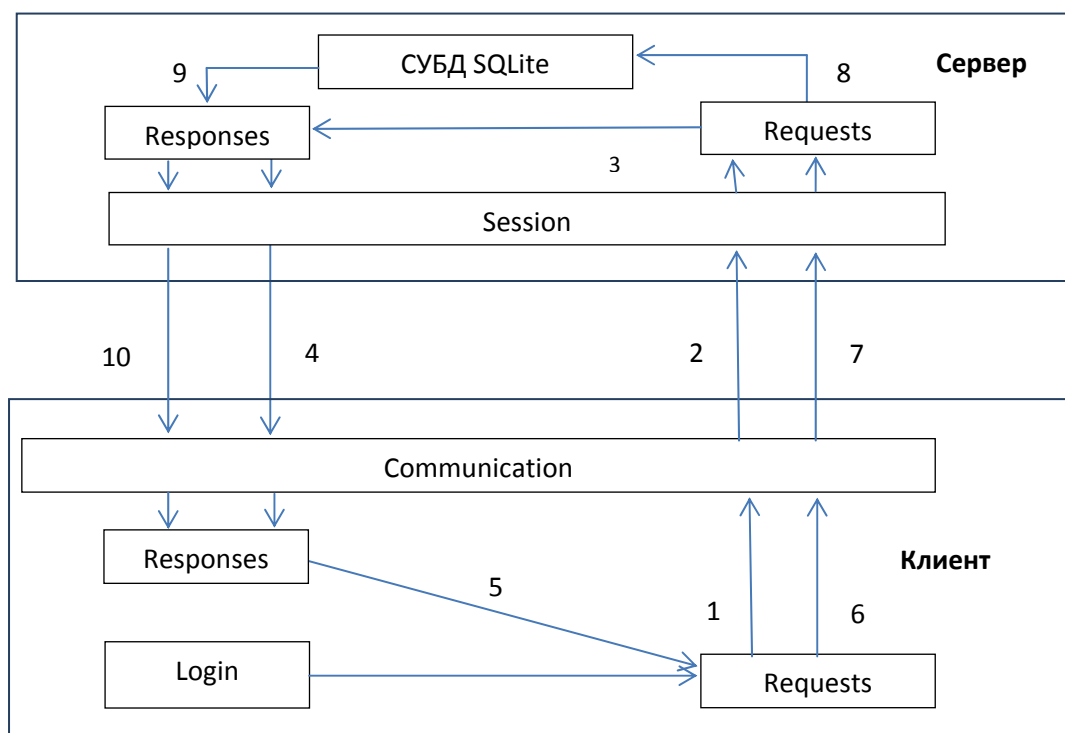


Рис. 5. Программная схема системы:
1–4 – запрос и ответ на установление связи; 5–10 – запрос и ответ на выдачу данных

Таблица 5

Клиентская часть

Название	Функция
Tcp.Core	Основной пакет, содержащий клиентскую и серверную часть
Request	Объект описывает данные для запроса, наследует интерфейс IMessage.
Response	Объект описывает необходимые данные для ответа, наследует интерфейс IMessage
DataSetObject	Объект содержит в себе объекты данных (матрицы, имена, наследует Serializable)
LoginObject	Содержит объекты необходимые для входа в систему (логин и пароль, роль, наследует Serializable)
WorkshopObject	Объект содержит данные изменяемые в ходе работы, наследует Serializable
DataSetRequest	Реализует поток данных запросов с применением пространства имен DataSetObject
GetAllDataSetRequest	Заносит в запрос данные в массиве
GetAllWorkshopRequest	Заносит в запрос рабочие (измененные) данные в массиве
LoginRequest	Запрос на вход в систему
DataSetResponse	Реализует поток данных ответов с применением пространства имен DataSetObject
GetAllDataSetResponse	Заносит в ответ данные в массиве
GetAllWorkshopResponse	Заносит в ответ все рабочие (измененные) данные в массив
LoginResponse	Ответ на вход в систему
MessageWriter	Инструмент для написания сообщений
MessageReader	Инструмент для чтения созданных сообщений
MessageFactory	Преобразует, структурирует данные на основе запросов и ответов, формируя их в один пакет-сообщение для удобной работы системы (преобразует данные в байт-вид)

Пакет Client реализует всю клиентскую базу, содержит метод входа в систему, получение и отправки данных под видом сообщений. Пакет Gui содержит FXML-файлы для описания интерфейса программы. Пакет Controllers осуществляет правильный процесс объединения элементов интерфейса и программных методов для обеспечения полного функционала программы.

Серверная часть тоже содержит пакет Tcp.core. Tcp.server – пакет-основа серверной части. Client Session реализует сессионный механизм пользователя

и организует работу сервера, т.е. обмен сообщениями, хранение информации. Context – реализует механизм соединения. Server – сервер, содержит серверную информацию, отображает подключившихся. Server Launcher – механизм запуска сервера. SessionManager – создает сессию.

Сервер приложений и клиент расположены на разных компьютерах. Все программные модули скомпонованы в две папки – server и client. Установить пакет server на сервере, а пакет client на клиенте.

Папки server и client устанавливаются на соответствующих компьютерах в доступных для пользователя местах. Например, таким местом может быть «Рабочий стол» или «Documents». В данном случае будет находиться на «Рабочем столе». Папка client будет выступать клиентской машиной, папка server будет выступать серверной машиной. Обе машины должны находиться в одной сети и иметь разные IP-адреса. При желании можно использовать виртуальную локальную сеть, если получится настроить. Обмен между сервером и клиентом будет идти по протоколу TCP/IP.

Работа программной системы осуществляется через меню компьютеров таким образом.

Предварительная стадия.

1. Администратор со своего компьютера генерирует числовые данные для диспетчера с учетом инерционности процесса планирования.

2. Данные (кроме планов) передаются в БД со схемой связей, в которую добавляются данные динамики.

Рабочая стадия.

1. По запросу диспетчера с его компьютера данные передаются для решения задач динамического линейного программирования.

2. Результаты поступают в БД и отражаются на экране компьютера клиента.

Если производство территориально распределено, то можно использовать Интернет. Для этого базу данных и промежуточное программное обеспечение следует установить на сайте в интернете и задать его IP-адрес.

Размещение сайта в интернете подробно описано в работах [11, 21]. Если сайт подготовлен, то его необходимо:

- зарегистрировать и получить доменное имя;
- загрузить сайт на интернетовский веб-сервер, который может управляться хостинговой компанией.

Перейдем к описанию публичного портала. В его представлении принял участие студент В. М. Харламов.

Построение и использование такого портала в простейшем случае удобно провести на примере фрагмента HTML-файла doc3.html, заимствованного из работы [10] и показанного в листинге 1.

Листинг 1 (фрагмент файла doc3.html)

```
<OL type=1>
<LI><A href=doc4.html#1> Жил да был ...</A>
</OL>
```

Нетрудно видеть, что файл doc3.html служит клиентом, который вызывает по ссылке данные с сервера doc4.html. В файлах могут использоваться и IP-адреса.

Рассмотрим построение поисковой системы, целью создания которой является повышение эффективности работы и увеличение производительности труда работников организации. Система, которую назовем «Соланж», должна обладать высокой информативностью и иметь минимум затрат на проектирование.

Прикладной целью выпускной работы является реализация в рамках задач системы «Соланж» следующих основных разделов для пользователя:

Общая структура системы представлена на рисунке 6.

Система состоит из сервера (хостинга) и веб-портала (клиента). Портал в данном случае представлен одним сайтом с несколькими страницами.

К веб-порталу как визуальному продукту предъявляются следующие требования:

1) портал должен иметь интуитивно понятный веб-интерфейс, чтобы пользователь мог прочесть представленную информацию;



Рис. 6. Структура поисковой системы

2) на сайте должна быть представлена актуальная динамически меняющаяся и достоверная информация;

3) на сайте должна использоваться нейтральная цветовая гамма;

4) все гиперссылки должны иметь описание и быть рабочими;

5) на сайте должна быть организована удобная навигационная система;

6) контент не должен противоречить требованиям действующего законодательства и организационным документам организации, нести сведения негативного характера и информацию, порочащую честь и достоинство сотрудников.

К программным инструментам веб-портала (сайта) предъявляются следующие требования:

1) учет наличия в организации других программно-аппаратных средств, внесение изменений в которые должно отсутствовать;

2) интеграция нового программного обеспечения с уже существующим;

3) совместимость с распространенными браузерами;

4) защищенность от вредоносного кода;

5) обеспечение одновременной работы нескольких пользователей.

6) сайт должен быть кросс-браузерным.

Этим требованиям удовлетворяют охарактеризованные ранее языки HTML, CSS (Cascading Style Sheets – каскадные таблицы стилей), JavaScript [22] (комментарии), PHP [23–26] для навигационных операций. Программы на языке JavaScript называют скриптами или сценариями. Дополнениями к ним могут быть программы (плагины), не имеющие самостоятельного значения и улучшающие основную программу. Для навигации язык Java [26–29] или апплеты в данной работе не используются. Для хранения данных использована СУБД MySQL.

В последнее время создание и поддержка данных на сайте серьезно автоматизирована [30, 31] с помощью специальной управляющей программы CMS (Content Management Software), устанавливаемой на хостинге. CMS имеет две цели:

- показ страниц сайта для пользователей, генерируя их содержимое с использованием заранее заданных шаблонов, дизайна и контента, хранящихся в базе данных;
- помощь владельцу сайта без специальных навыков и умений управлять сайтом, публикуя новые страницы или новости, видео, размещая ссылки как на внешние, так и на внутренние ресурсы, для чего администратор использует визуальный редактор. В качестве CMS используется вариант WordPress.

В этом случае процедура создания сайта может проводиться в таком порядке.

1. Выбрать и оплатить хостинг, получая некое дисковое пространство для размещения своих сайтов.
2. Хостинг предоставляет возможность входа в свой личный кабинет. В нем и происходит создание сайтов.
3. Для создания сайта нужно указать домен (если такой есть) или использовать домен, предоставляе-

мый хостингом. Затем указывается адрес сайта и хостинг сам разместит сайт в интернете.

4. Начинается создание уже самого сайта. В указанном личном кабинете надо выбрать нужную CSM. Обычно предоставлен выбор из самых распространенных CSM. После выбора хостинг сам установит выбранный CSM.

5. Использовать внешнюю базу данных, тогда в специальной панели нужно будет указать адрес и данные подключения. Возможно создание БД посредством самого хостинга. В этом случае БД будет расположена на хостинге, хостинг сам создает все нужное для работы БД.

6. На почту приходит письмо со ссылкой на личный кабинет администратора CSM, где можно создавать сам сайт, наполнять его контентом, добавлять пользователей.

7. Наконец, надо провести тестирование и апробацию сайта, его внедрение в рабочий процесс организации.

Окончательный вариант построенного сайта представлен на рисунке 7.

В работе осуществлена систематизация информации о поисковых системах, в том числе о порталах и сайтах. Рассмотрены соответствующие понятия, классификация, структура, методы описания, применяемые программные продукты и методы программной реализации, в том числе с помощью системы управления контентом. Показано отличие порталов от сайтов, корпоративных порталов от публичных порталов. Приведены примеры программной реализации системы управления и поисковой системы с построением сайта портала.

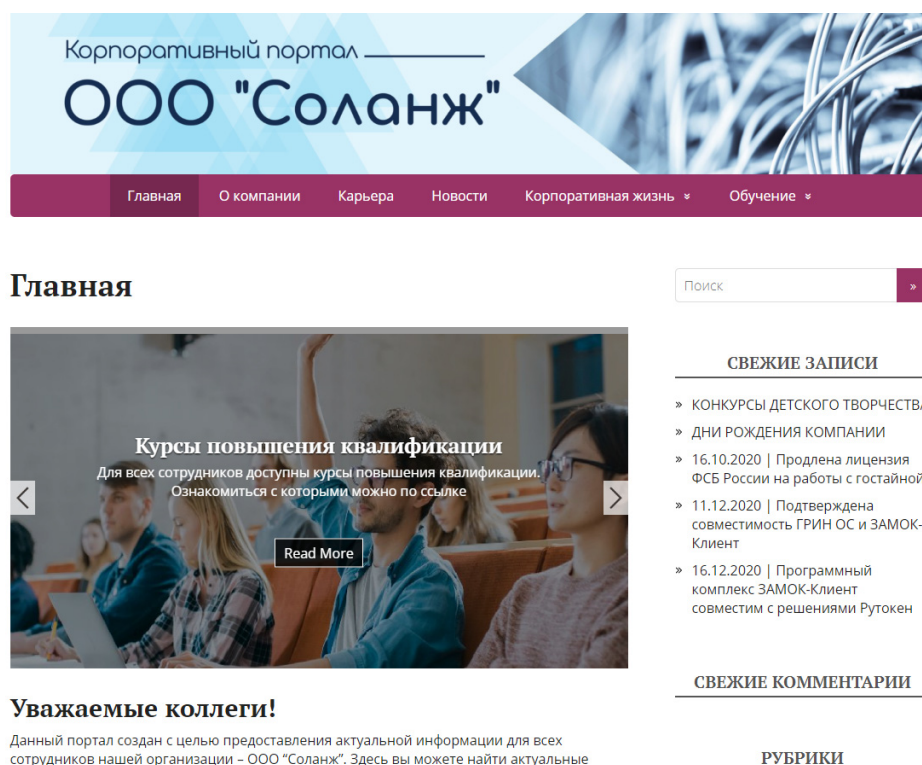


Рис. 7. Главная страница портала

Литература

1. Web-графика. – URL: <https://topref.ru/referat/50547.html> (дата обращения: 10.06.2021). – Текст : электронный.
2. Цехановский, В. В. Распределенные информационные системы / В. В. Цехановский, В. Д. Чертовской – Санкт-Петербург : Лань, 2020. – 240 с.
3. Механизмы интеграции баз данных и программ анализа. – URL: <https://cyberleninka.ru/article/n/mechanizmy-integratsii-baz-dannyh-i-programm-analiza> (дата обращения: 10.06.2021). – Текст : электронный.
4. Додян, Э. Г. Методы, модели, средства хранения / Э. Г. Додян, Ю. А. Зеленков – Москва : Инфра, 2017. – 167 с.
5. Дубова, Н. Краткий курс интеграции данных / Н. Дубова // Открытые системы. – 2016. – № 4.
6. URL: <http://qaru.site/questions/126239/how-to-insert-table-values-from-one-database-to-another-database> (дата обращения: 10.06.2021). – Текст : электронный.
7. Никсон, Р. Создаем Web-сайты с PHP, MySQL, JavaScript? CSS, HTML / Р. Никсон. – Москва : Питер, 2013. – 560 с.
8. Веллинг, Л. Разработка Web-приложений с помощью PHP и MySQL / Л. Веллинг, Л. Томсон. – Москва : Вильямс, 2010. – 320 с.
9. Хоган, Б. HTML5 и CSS3. Веб-разработка по стандартам нового поколения / Б. Хоган. – 2-е изд. – Санкт-Петербург : Питер, 2014. – 320 с.
10. Дуванов, А. А. Web-конструирование HTML / А. А. Дуванов. – Санкт-Петербург : БХВ, 2001. – 336 с.
11. Дакетт HTML и CSS. Разработка и создание вебсайтов. – Москва : Эксмо, 2013. – 475 с.
12. Хабибуллин, И. Ш. Разработка Web-служб средствами Java / И. Ш. Хабибуллин. – Санкт-Петербург : БХВ-Петербург, 2003. – 400 с.
13. Дунаев, В. В. Самоучитель. Сценарии для Web-сайта. PHP и JavaScript / В. В. Дунаев. – Санкт-Петербург : БХВ-Петербург, 2019. – 279 с.
14. Ставрова, О. Б. Введение в Интернет с электронным пособием по созданию web-страниц / О. Б. Ставрова. – Москва : Интеллект-Центр, 2005. – 80 с.
15. Большев, А. К. Администрирование систем сервисно-ориентированной архитектуры [Комплект] : учебное пособие / А. К. Большев, А. А. Лавров, В. В. Яновский ; Санкт-Петербургский государственный электротехнический университет им. В.И. Ульянова (Ленина) ЛЭТИ. – Санкт-Петербург : СПбГЭТУ ЛЭТИ, 2011. – 96 с.
16. Гоше, Х. Д. HTML5. Для профессионалов / Х. Д. Гоше. – Санкт-Петербург : Питер, 2013. – 496 с.
17. Мак-Дональд, М. Создание Web-сайта. Недоступное руководство / М. Мак-Дональд. – Санкт-Петербург : БХВ-Петербург, 2013. – 624 с.
18. Роббинс, Дж. HTML5, CSS3 и JavaScript. Исчерпывающее руководство / Дж. Роббинс. – Москва : Эксмо, 2014. – 528 с.
19. Фрейн, Б. HTML5 и CSS3. Разработка сайтов для любых браузеров и устройств / Б. Фрейн. – Санкт-Петербург : Питер, 2014. – 304 с.
20. Фрейн, Б. HTML5 и CSS3. Разработка сайтов для любых браузеров и устройств / Б. Фрейн. – 2-е изд. – Санкт-Петербург : Питер, 2017. – 272 с.
21. Интернет-Технологии.ру. – URL: <https://www.internet-technologies.ru> (дата обращения: 10.06.2021). – Текст : электронный.
22. Дуванов, А. А. Web-конструирование DHTML / А. А. Дуванов. – Санкт-Петербург : БХВ, 2003. – 512 с.
23. Ташков, П. А. Веб-мастеринг на 100 %: HTML, CSS, JavaScript, PHP, CMS, AJAX, раскрутка / П. А. Ташков. – Москва : АСТ-Пресс, 2021. – 365 с.
24. Мейнджер, Д. JavaScript: основы программирования / Д. Мейнджер. – Киев : БХВ, 1997. – 510 с.
25. Бенкен, Е. PHP, MySQL, XML: программирование для Интернета / Е. Бенкен. – Санкт-Петербург : БХВ, 2007. – 336 с.
26. Хабибуллин, И. Ш. Разработка Web-служб средствами Java / И. Ш. Хабибуллин. – Санкт-Петербург : БХВ-Петербург, 2003. – 400 с.
27. Шилдт Герберт. Java 8. Полное руководство : перевод с английского / Шилдт Герберт. – 9-е изд. – Москва : ООО Вильямс, 2015. – 1 376 с.
28. Перроун, В. Д. Создание корпоративных систем на основе Java 2 Enterprise Edition. Руководство разработчика : перевод с английского / В. Д. Перроун, В. С. Р. Чаганти – Москва : Вильямс, 2001. – 1184 с.
29. Разработка Web-служб средствами Java. – Санкт-Петербург : БХВ, 2003. – 400 с.
30. Демченко Д. Обзор систем управления сайтом (CMS) / Д. Демченко. – URL: <http://www.white-windows.ru/obzor-sistem-upravleniya-sajtom-cms/> (дата обращения: 10.06.2021). – Текст : электронный.
31. Хилтер У. Этапы разработки web-сайта / У. Хилтер. – URL: <http://www.mistli.ru/etapi-razrabotki-web-sayta> (дата обращения: 10.06.2021). – Текст : электронный.

V.D. Chertovskoy

Uljanov (Lenin) State Electrotechnical University Saint-Petersburg

CONSTRUCTION OF DISTRIBUTED SYSTEMS VIA WEBPORTALS AND WEBSITES

The article deals with the issues of formation and use of search systems. The systematization of the data on such digitalization tools as webportals and websites is presented. Their definitions and properties are given. Their link is pointed out. The classification, structure, peculiarities, construction technology are discussed. The applied-oriented examples of implementation of the search systems with portals and management systems.

Webportal, website concepts, classification, structure, programming methods.