

УДК 658.5+669

*д.т.н., проф.. Заблодский Н.Н.,
к.т.н., доц. Бизянов Е.Е.,
к.э.н., доц. Зайцев И.С.
(ДонГТУ, г. Алчевск, Украина)*

СОВРЕМЕННАЯ КОНЦЕПЦИЯ ПОСТРОЕНИЯ ИНФОРМАЦИОННОЙ СИСТЕМЫ УПРАВЛЕНИЯ КРУПНЫМ МЕТАЛЛУРГИЧЕСКИМ ПРЕДПРИЯТИЕМ

Розглянуто концепцію побудови інформаційної системи крупного металургійного підприємства з урахуванням сучасних вимог і архітектури, а також специфіки діяльності підприємства.

В настоящее время металлургические предприятия при внедрении новых информационных систем выдвигают требования интеграции программных продуктов, желая сохранить "работающие" специализированные системы и информационно объединить их с вновь приобретаемыми. Как правило, характерным для таких предприятий является наличие парка персональных компьютеров (ПК) разной производительности и с различными ОС (UNIX, MS DOS, Windows), СУБД (ORACLE, FOXPRO, ACCESS, MS SQL Server), офисными пакетами и иным прикладным программным обеспечением (ПО). Кроме того, в большинстве случаев пользователи устанавливают на свои ПК дополнительные программы, не входящие в их функциональные обязанности, зато вполне способные привести систему в неработоспособное состояние или уничтожить коммерческую информацию. Поддержание однотипными парка ПК, системного и прикладного ПО требует немалых финансовых затрат и далеко не всем под силу. Затруднительным является регулярное резервное копирование пользовательских данных. В случае выхода ПК из строя зачастую происходит частичная или полная потеря документов пользователя. Многообразие используемых программных средств накладывает жесткие ограничения на дальнейшее проектирование и разработку комплексов задач и подсистем. Понятно, что применение более чем одной операционной системы, СУБД и средств разработки должно накладывать на разработанное программное обеспечение требование переносимости с одной платформы на другие без какого-либо вмешательства разработчиков. Сегодня это требование не может быть выполнено, поскольку существуют локальные разработки на разных платформах. Мало этого, до сих пор не существует программных соглашений,

устанавливающих необходимость и обязательность в новых разработках использовать только кросс-платформенные приложения.

Усугубляет это положение то, что использование различных средств управления базами данных крайне недопустимо, потому что в этом случае затраты на разработку, совмещение информации в базах данных и сопровождение продукта возрастают более чем вдвое. Для поддержки актуального состояния классификаторов и справочников необходима служба слежения за их состоянием в различных базах данных. Поиск и исправление ошибок, возникающих на этом этапе, весьма трудоемкий процесс, а именно от этой информации требуется полная достоверность, поскольку она ложится в основу бухгалтерского учета.

Для большинства металлургических предприятий характерно наличие нескольких «точек концентрации» пользователей заводской ИС с большим территориальным разбросом. Объемы данных, курсирующих по сети, довольно существенны. Уровень помех, генерируемых мощным промышленным оборудованием, для любого сетевого интерфейса (за исключением оптического) довольно высок и, соответственно, большой процент потерь пакетов и низкие скорости передачи данных.

Поэтому независимо от применяемой технологии связи имеет смысл использовать терминальный вариант работы удаленных «точек доступа». Это разумно и по соображениям экономии пропускной способности протяженных сетевых интерфейсов, и с точки зрения сохранности данных – все они находятся и обрабатываются на сервере, а для работы одного пользователя достаточно канала в пару десятков килобит.

В связи с тем что физически вся работа пользователя выполняется на сервере с применением одних и тех же пакетов прикладного ПО и общего хранилища данных, вопрос контроля версий ПО решается автоматически, а создание резервных копий не вызывает никаких проблем. Уровень контроля прав пользователей соответствует возможностям серверной ОС, существенно снижая вероятность утечки конфиденциальной информации.

Современные подходы [1] требуют, чтобы системное и прикладное программное обеспечение реализовывало многоуровневую архитектуру, обеспечивая простоту развертывания и управления многозвенными, кросс-платформенными, распределенными объектными приложениями.

В архитектуре современных информационных систем необходимо выделять три уровня:

- уровень представления (реализующий функции ввода и отображения данных) – уровень клиента;
- прикладной уровень (реализующий универсальные сервисы, а также функции, специфичные для определенной предметной области) – средний уровень сервера приложений;

– уровень доступа к информационным ресурсам (реализующий фундаментальные функции хранения и управления информационно-вычислительными ресурсами) – уровень сервера базы данных.

Связь между уровнями обеспечивает менеджер транзакций и коммуникаций.

Технология Интранет наложила свой отпечаток на эту классическую схему, расположив на уровне представления универсальный клиент – Web-навигатор (возможно, пополненный прикладными апплетами) и возложив функции информационного концентратора (которые естественно объединить с функциями менеджера транзакций и коммуникаций) на Web-сервер.

Опираясь на эту архитектуру, корпорация Oracle предлагает три ключевых элемента информационных систем (рисунок 1):

- объектно-реляционный сервер СУБД Oracle 8i/9i;
- универсальный сервер приложений Oracle Application Server, а также ряд специализированных прикладных серверов (Oracle Payment Server, Internet Commerce Server, Video Server и т.д.). Подобные средства (Sun Java 2 Enterprise Edition, BEA Weblogic, IBM Websphere, Bluestone Sapphire/Web), отличающиеся производительностью, стоимостью, набором выполняемых функций и качеством сопровождения предлагают и другие производители ПО;
- набор драйверов в стандарте JDBC, специально оптимизированных для доступа из Java к СУБД Oracle.

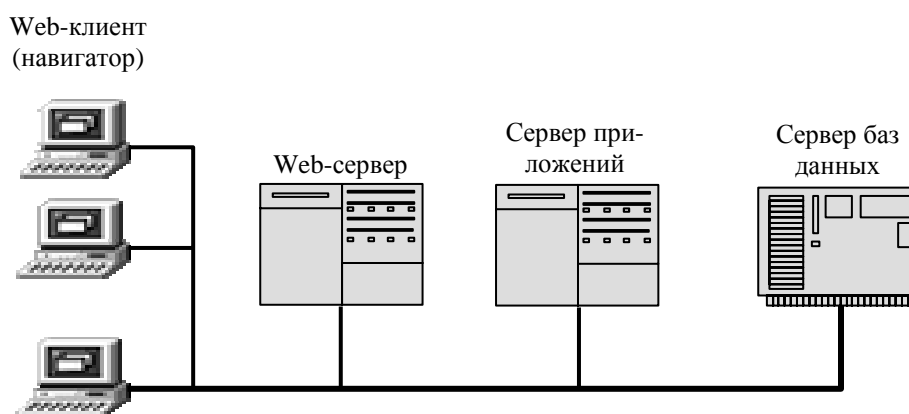


Рисунок 1 – Трехуровневая архитектура в технологии Интранет

Основная идея, заложенная в основу трехуровневой архитектуры – перенос реализации бизнес-правил (формализованных и воплощенных в программном коде правил работы организации) как с серверной, так и с клиентской части приложения в некоторое промежуточное звено –

сервер приложений. Это промежуточное звено можно рассматривать как набор компонентов, объектов, реализующих требуемые бизнес-правила.

Преимущества этого подхода очень существенны. Во-первых, клиентская часть приложения (Web-браузер) сильно упрощается и удешевляется. Во-вторых, функциональность системы не связана ни с конкретным языком программирования, используемым для создания клиентских частей приложений, ни с особенностями используемой СУБД.

Для обеспечения доступа к базам данных на стороне Web-клиента сегодня в основном применяется Java-технология. Java – это современный объектно-ориентированный язык программирования для разработки приложений, созданный специально для распределенных сред.

Одно из важных свойств Java-технологии – это мобильность, суть которой заключается в том, что написанный на Java код может исполняться на любой компьютерной платформе.

Для взаимодействия Java-приложений с внешним сервером баз данных разработан специализированный протокол JDBC, который фактически сочетает функции шлюзования между интерпретатором мобильных Java-кодов и интерфейсом ODBC (Open Data Base Connectivity).

Исходя из вышесказанного основные требования, предъявляемые к корпоративным приложениям и реализующему их программному обеспечению, как системному, так и прикладному формулируются следующим образом:

Переносимость и гибкость – независимость от платформы и способность сохранять соответствие непрерывно растущим требованиям и возможность нормального функционирования при замене операционных систем и аппаратных средств. Никакие вновь разрабатываемые программные средства не должны зависеть от платформы реализации;

Наращиваемость – способность справляться с постоянно возрастающим объемом хранимой, обрабатываемой и передаваемой информации в условиях возможного роста количества работающих с этой информацией пользователей.

Высокая надежность, – обеспечение непрерывного и качественного доступа к информации в любой момент, когда это может потребоваться. Высокая готовность и отказоустойчивость для всех аспектов распределенной вычислительной системы является требованием корпоративных приложений. Когда происходит сбой в серверном процессе, его клиенты должны быть способны продолжить работу либо используя тот же экземпляр (после перезагрузки), либо посредством связывания с другим работающим экземпляром.

Обеспечение необходимой безопасности для корпоративных данных – предприятию необходимо надежное решение, которое обеспечивает конфиденциальность, целостность и аутентификацию. Программное обеспечение должно использовать ресурсы Web, в то же время, обеспечить безопасный, надежный метод выполнения транзакций в распределенной среде.

Обеспечение доступа к корпоративным данным на основе широко используемых стандартов – программное обеспечение должно интегрироваться с источниками данных предприятия. Решение, которое использует доступ к базам данных на основе промышленного стандарта взаимодействия, позволяет воспользоваться преимуществами новых технологий. Необходимо использовать промышленный стандарт сетевого взаимодействия между Java и разнообразными типами баз данных (JDBC) для предоставления доступа к корпоративным базам данных.

Открытость кода – программное обеспечение должно поставляться с открытым кодом.

Перечень этих требований формирует определенные ограничения при выборе технического и системного программного обеспечения, удовлетворяющих поставленным ограничениям. На стороне клиента программное обеспечение должно формироваться на базе открытых систем (Linux RedHat, Linux ASP), не требующих никакого клиентского программного обеспечения, кроме браузера, поставляемого со всеми модификациями операционной системы Linux. На среднем уровне – сервере приложений необходим Java Application Server и Web Server, поставляемые многими производителями открытых систем. Такая конфигурация, отвечая всем требованиям обеспечения функциональности клиента, значительно удешевляет стоимость системных программных средств, не требуя затрат на лицензионное ПО клиента. Одновременно с этим снижаются и требования к техническому обеспечению – для нормальной работы клиента достаточно удовлетворить минимальным техническим требованиям (Pentium III/128Mb/20Gb).

Таким образом, разрабатываемая система должна соответствовать следующим общим техническим требованиям:

- возможность интеграции с большим числом программных продуктов с минимальным уровнем интеграции – на уровне открытых кодов или командной строки;

- обеспечение безопасности с помощью различных методов контроля и разграничения доступа к информационным ресурсам. Наличие в составе ERP-системы сертифицированных программно-аппаратных средств защиты информации (позволяющих шифровать данные, поддерживающих электронную цифровую подпись и аутентифицирующих на ее основе пользователей);

- масштабирование: модульный принцип построения системы из оперативно-независимых функциональных блоков с расширением за счет открытых стандартов (API, J2EE и др.);

- применение 3-звенной архитектуры (сервер базы данных, сервер приложений, клиент) с возможностью использования терминалов. Клиент может быть "толстым" или "тонким";

- система должна иметь возможность миграции с платформы на платформу. Обязательно должны быть версии для ОС MS Windows и UNIX (и их клонов);

- в набор СУБД, поддерживаемых информационной системой, обязательно должно входить распространенное ПО (например, Oracle, Sybase, MS SQL Server, Informix и др);

- поддержка технологий распределенной обработки информации, технологий Интернет/интранет с возможностью работы через "тонкого клиента". Такое техническое решение позволяет использовать стандартные хранилища данных (библиотеки документов, базы данных) из локальных, корпоративных и глобальных сетей, не требуя существенных затрат на дополнительное администрирование и поддержание целостности, надежности и безопасности хранения данных;

- поддержка технологий многоуровневого электронного архивирования информации на различных носителях (дисковых массивах, DVD-ROM, DVD-RW, магнитооптических дисках и библиотеках, ленточных библиотеках и др.);

- наличие аналитических возможностей и встроенных инструментальных средств (позволяющих самостоятельно наращивать функциональность установленной информационной системы).

- удовлетворительные эксплуатационные характеристики (легкость администрирования, эргономичность, русскоязычный интерфейс и др.);

- инструментальные программные средства разработки информационных приложений должны обеспечивать следующие важные свойства: поддержку многоплатформенности; независимость от производителя; унификацию средств разработки; создание надежного и качественного программного обеспечения; поддержку разработанного ПО на протяжении всего времени жизни; проектирование с использованием современных методик; ведение версий; поддержку Web-технологии.

В условиях быстрой эволюции вычислительной техники, смены операционных систем, постоянного обновления технологий разработки устойчивость информационно-вычислительной среды может быть обеспечена разумным консерватизмом в сочетании с трезвым анализом тенденций современного рынка.

Рассмотрена концепция построения информационной системы крупного металлургического предприятия с учетом современных требований и архитектур, а также специфики деятельности предприятия.

The concept of construction of information system of the large metallurgical enterprise is considered in view of modern requirements and architectures, and also specificity of activity of the enterprise.

Библиографический список

1. Автоматизация управления предприятием. / Баронов В.В. и др. – М.: ИНФРА-М, 2000. – 239 с.