УДК 004.6, 004.02

ПРИНЦИПЫ И ПОДХОДЫ К ПРОЕКТИРОВАНИЮ АРХИТЕКТУРЫ  
СИСТЕМЫ C-COMMERCE

Андрианова Е.Г., Морнев А.А. Трохаченкова Н.Н., Полторак А.В.

МИРЭА - Российский технологический университет, 119454, Россия, г. Москва, проспект Вернадского, 78, e-mail: andrianova@mirea.ru

В качестве базового подхода при концептуальном проектировании архитектуры системы C-Commerce использовался процессный подход. Также учитывались принципы открытости, масштабируемости, интероперабельности, глобализации (доступности), кастомизации (гибкости), аутсорсинга. На основе процессного подхода был выделен и принят SCM подход как принцип совместного использования ресурсов. Для технической реализации выбран сервисно-ориентированный подход. Отмечена важность концепции открытых систем при организации взаимодействие систем. В заключение рассмотрены ключевые требования к крупномасштабным открытым информационным системам.

Ключевые слова: C-Commerce, E-Comemrce, сервис-ориентированный подход (SOA), открытость, масштабируемость, интероперабельность, глобализация, кастомизация, SCM подход.

**PRINCIPLES AND APPROACHES TO ARCHITECTURE DESIGN**

**C-COMMERCE SYSTEMS**Andrianova E.G., Mornev A.A. Trokhachenkova N.N., Poltorak A.V.

*MIREA - Russian Technological University, 119454, Russia, Moscow, 78 Vernadsky Avenue, e-mail: andrianova@mirea.ru.*

**A process-based approach was used as a basic approach in the conceptual design of the C-Commerce system architecture. The principles of openness, scalability, interoperability, globalisation (availability), customisation (flexibility) and outsourcing were also considered. Based on the process approach, the SCM approach was identified and adopted as a principle of resource sharing. A service-oriented approach has been selected for technical implementation. The importance of the concept of open systems in organising the interaction of systems was emphasised. In conclusion, the key requirements for large-scale open information systems were considered.**

Keywords: C-Commerce, E-Comemrce, service oriented approach (SOA), openness, scalability, interoperability, globalization, customization, SCM approach.

**Введение**

В настоящее время в экономике преобладает концепция E-Commerce (электронная коммерция), т.е. деятельность по покупке или продаже товаров на онлайн-сервисах или через интернет. C-Commerce - это совокупность взаимодействий между предприятием, его поставщиками, торговыми партнерами, клиентами и сотрудниками на основе электронных средств. Если концепция лежит в основе автоматизации конкурентной экономики, то C-Commerce лежит в основе автоматизации совместной экономики. Разработка принципов и подходов к проектированию систем C-Commerce является актуальной задачей. Также интересна экстраполяция указанных принципов и подходов на проектирование крупномасштабных открытых информационных систем (КОИС).

**Система C-Commerce как открытая информационная система**

C-Commerce система нацелена на повышение эффективности использования материалов при снижении дополнительных расходов, связанных с издержками взаимодействия с поставщиками на рынке. Идеальным является решение, в котором одновременно будут работать все участники бизнес цепочки. В реальности переход к такой концепции моментально не возможен, так как, каждое предприятие, участвующее в бизнес цепочке имеет свою учетную систему с разной степенью продуманности. Следовательно, системе C-Commerce необходимо иметь механизм взаимодействия с различными уже существующими на рынке учетными системами, т.е. быть открытой системой. Под открытой системой, понимается «программная и информационная система, построенная на базе исчерпывающего и согласованного набора международных стандартов на информационные технологии и профилях функциональных стандартов, которые реализуют открытые спецификации на интерфейсы, службы и поддерживающие их форматы, чтобы обеспечить взаимодействие (интероперабельность) и мобильность программных приложений, данных и персонала» [1]. Обеспечение интероперабельности достигается за счет использования специально разработанных стандартов и моделей, утвержденных различными организациями и комитетами по изучению открытых систем. Стандарты могут включать в себя различные спецификации, технологии, протоколы и алгоритмы.

**Обоснованность применения сервис-ориентированного подхода**

Проектируемый концепт системы воспринимается внешними предприятиями как некий сервис, предоставляемый для достижения определенного результата. Не у всех предприятий есть потребность в оптимизации всех внешних бизнес-процессов. Для этого каждый процесс необходимо рассматривать атомарно и как «черный ящик» относительно другого предприятия. Следовательно, необходимо использовать сервис-ориентированный подход (SOA) при проектировании архитектуры системы C-Commerce.

При этом достигаются три цели:

1. Структурирование процедур или программных компонентов в качестве услуг, которые используются только при необходимости.

2. Обеспечение механизма публикации доступных услуг, который включает их функциональность и требования ввода-вывода. Службы публикуются таким образом, чтобы разработчики могли легко включать их в приложения.

3. Контроль использования этих услуг во избежание проблем безопасности и управления. Безопасность в значительной степени зависит от безопасности отдельных компонентов в рамках архитектуры, процедур идентификации и аутентификации, связанных с этими компонентами, а также защиты фактических соединений между компонентами архитектуры.

**Структура системы C-Commerce**

Рассмотрим систему C-Commerce на трех уровнях (рис. 1) концептуальном, функциональном и локальном/предметном.

На концептуальном уровне обозначены технологии, модели и принципы пользовательского уровня системы. На этом уровне представлены две определяющих базисных модели: OSE RM (ISO/IEC DTR 14252, Portable Operating System Interface for Computer Environments - POSIX IEEE, P1003.0, Draft Guide to the POSIX Open System Environment) и OSI RM (ISO 7498:1996, Information processing systems - Open Systems Interconnection, Basic Reference Model; (ITU-T Rec. X.200)).

Обе модели, разработанные по стандартам POSIX, являются определяющими для выбора и построения архитектуры системы. Для построения системы выбрана архитектура Web-приложений и использован подход SOA (Service-Oriented Architecture), что упрощает интеграцию приложений.

Разработчики могут получать доступ к сервисам, используя стандартные языки и протоколы. На функциональном уровне предоставлены функции системы C-Commerce для взаимодействия с другими системами, реализованные через собственный API-интерфейс (Application Program Interface) системы. API-интерфейс может предоставляться на различных уровнях взаимодействия, а также использовать один из стандартизованных подходов REST или SOAP.

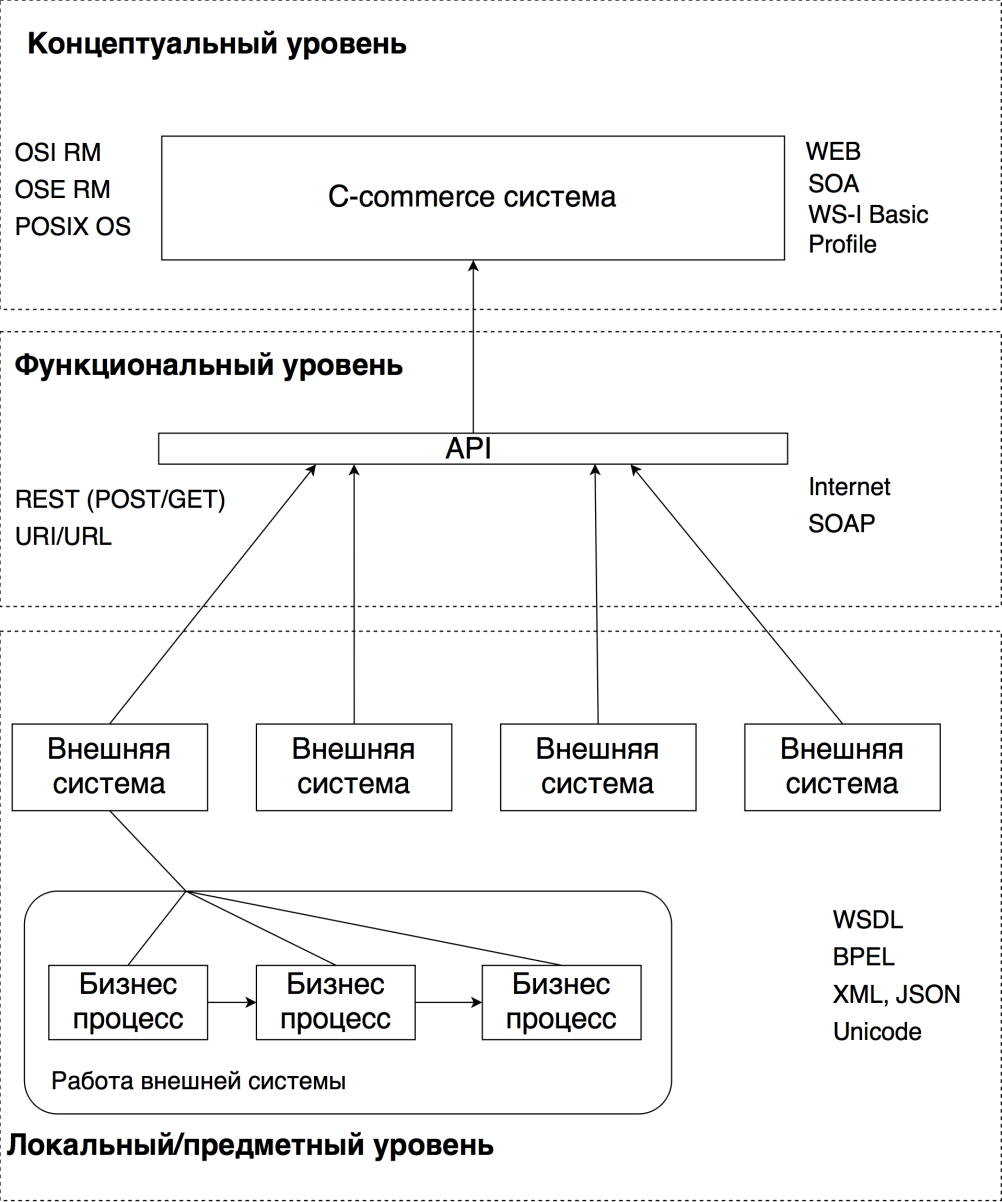


Рис. 1. Трехуровневая модель системы C-Commerce.

Локальный уровень - уровень работы сторонних приложений. На этом уровне раскрывается предметная область организации взаимодействия систем. На рис. 1 видно, что общие данные, полученные из внешних источников, состоят из данных, получаемых на отдельных бизнес процессах внешней системы. Это говорит о том, что с точки зрения C-Commerce системы данные бизнес процессы могут служить «черным ящиком», имеющим только входные и выходные данные. Для передачи этих данных необходимо во внешней системе использовать предоставляемые функции (API), в которых указываются соответствующие параметры и данные. Для передачи данных необходимо использовать также утвержденный формат разметки данных - XML или JSON. Так как передаваемые данные носят абсолютно разнородный характер (название товаров, производителей и другие коммерческие данные) необходимо учитывать кодирование печатных символов. Для решения этой проблемы используется необходимый стандарт кодирования символов Unicode.

**Принципы построения архитектуры системы C-Commerce**

Предложенная архитектура системы C-Commerce описывает основные процессы, которые являются базисными, и все данные, которые будут передаваться по вызову удаленных процедур из внешних систем предприятий, и не зависят от реального бизнес-процесса предприятия, так как цепочка процессов, приводящая к получению этих данных, с точки зрения C-Commerce системы, является «черным ящиком». Масштабируемость в этом случае приобретает характер детализации бизнес-процессов и погружение в этот «черный ящик».

Следующий принцип, который ложится в основу построения концепции C-Commerce - принцип аутсорсинга. Аутсорсинг - это один из способов оптимизации всех процессов на предприятии, при помощи концентрации усилий на основном виде деятельности и делегировании непрофильных функций сторонним компаниям. В отношении C-Commerce принцип аутсорсинга применяется как передача часть бизнес функций C-Commerce системе для повышения качества анализа потребности и освобождения ресурсов в контексте общей цепочки бизнес-процессов.

Глобализация - это тенденция к расширению взаимодействия между людьми или компаниями в мировом масштабе благодаря прогрессу в области информационных и коммуникационных технологий. С усилением взаимодействия между национальными государствами и отдельными людьми произошел рост международной торговли, идей и культуры.

Глобализация представляет собой рост числа компаний, вовлеченных в одну и ту же деятельность, но находящихся в разных географических условиях. Для общей интеграции информационных систем партнеров, находящихся на большом расстоянии друг о друга, должен действовать принцип доступности взаимодействия с C-Commerce системой.

Гибкость используется как атрибут различных типов систем. В контексте инженерного проектирования можно определить гибкость как способность системы реагировать на потенциальные внутренние или внешние изменения, влияющие на ее стоимость, своевременно и экономически эффективным образом. Таким образом, гибкость для инженерной системы - это простота, с которой система может реагировать на неопределенность таким образом, чтобы поддерживать или увеличивать ее стоимость. Неопределенность является ключевым элементом определения гибкости. Неопределенность может создавать как риски, так и возможности в системе, и именно с наличием неопределенности гибкость становится ценной.

**Реализация архитектуры системы C-Commerce**

Проектируемый концепт системы C-Commerce служит посредником между внутренними, не зависящими друг от друга, учетными системами отдельных предприятий. Это означает, что в основу разработки такой системы должен обязательно лечь принцип промежуточного программного обеспечения (ППО). ППО можно сгруппировать по следующим категориям:

• удаленный вызов процедур или промежуточное ПО на основе RPC, которое позволяет процедурам в одном приложении вызывать процедуры в удаленных приложениях, как если бы они были локальными вызовами;

• посредник запросов объектов или промежуточное программное обеспечение на основе ORB, которое позволяет распределять и совместно использовать объекты приложения в гетерогенных сетях;

• Message Oriented Middleware или mom-based middleware, которая позволяет распределенным приложениям обмениваться данными путем отправки и получения Сообщений.

Все эти модели позволяют одному программному компоненту влиять на поведение другого компонента в сети. Они отличаются тем, что промежуточное программное обеспечение на основе RPC и ORB создает системы тесно связанные компоненты, в то время как системы на основе MOM допускают более свободную связь компонентов. В системе, основанной на RPC или ORB, когда одна процедура вызывает другую, она должна ждать вызываемой процедуры для возврата, прежде чем она сможет сделать что - либо еще. В этих моделях синхронного обмена сообщениями ППО частично функционирует как суперлинкер, размещая вызываемую процедуру в сети и используя сетевые службы для передачи в процедуру параметров функции или метода, а затем для возврата результатов.

Учитывая, что проектируемая система нацелена на обмен большим количеством информации, то разумно предположить, что самым удобным классом будет являться ППО направленное на рассылку и получения сообщения от внешних источников. Message - oriented middleware (MOM) – это программная и/или аппаратная инфраструктура, поддерживающая отправку и получение сообщений между распределенными системами. MOM позволяет распределять модули приложений по разнородным платформам и снижает сложность разработки приложений, которые охватывают несколько операционных систем и сетевых протоколов. ППО создает уровень распределенной связи, который изолирует разработчика приложения от деталей различных операционных систем и сетевых интерфейсов. Интерфейсы API, которые распространяются на различные платформы и сети, обычно предоставляются MOM.

Поскольку предприятия и технологии постоянно изменяются, программные системы, которые их обслуживают, должны быть способны адаптироваться к таким изменениям. После слияния, добавления услуги или расширения доступных услуг бизнес не может позволить себе воссоздать свои информационные системы. Именно в этот самый критический момент ей необходимо интегрировать новые компоненты или максимально эффективно масштабировать существующие.

Самый простой способ интеграции гетерогенных компонентов - не воссоздать их как однородные элементы, а создать слой, позволяющий им взаимодействовать, несмотря на различия. Этот уровень, называемый промежуточным программным обеспечением, позволяет программным компонентам, которые были разработаны независимо и которые работают на различных сетевых платформах, взаимодействовать друг с другом. Приложения, распространяемые на разных сетевых узлах, используют интерфейс приложения для взаимодействия, не заботясь о деталях операционных сред, в которых размещаются другие приложения, и службах, которые подключают их к этим приложениям. Кроме того, предоставляя административный интерфейс, эта новая виртуальная система взаимосвязанных приложений может стать надежной и безопасной. Свое представление можно измерить и настроить, и его можно вычислить по масштабу.

Основными элементами системы MOM являются клиенты, сообщения и поставщик MOM, который включает API и средства администрирования. Поставщик MOM использует различные архитектуры для маршрутизации и доставки Сообщений: он может использовать централизованный сервер сообщений или распределять функции маршрутизации и доставки на каждый клиентский компьютер.

Некоторые продукты MOM сочетают эти два подхода. С помощью системы MOM клиент выполняет вызов API для отправки сообщения в пункт назначения, управляемый поставщиком.

Вызов вызывает службы поставщика для маршрутизации и доставки сообщения. После отправки сообщения клиент может продолжать выполнять другие действия, будучи уверенным, что поставщик сохранит сообщение до тех пор, пока получающий клиент не получит его. Модель на основе сообщений в сочетании с посредничеством поставщика позволяет создать систему слабосвязанных компонентов (рис. 2).

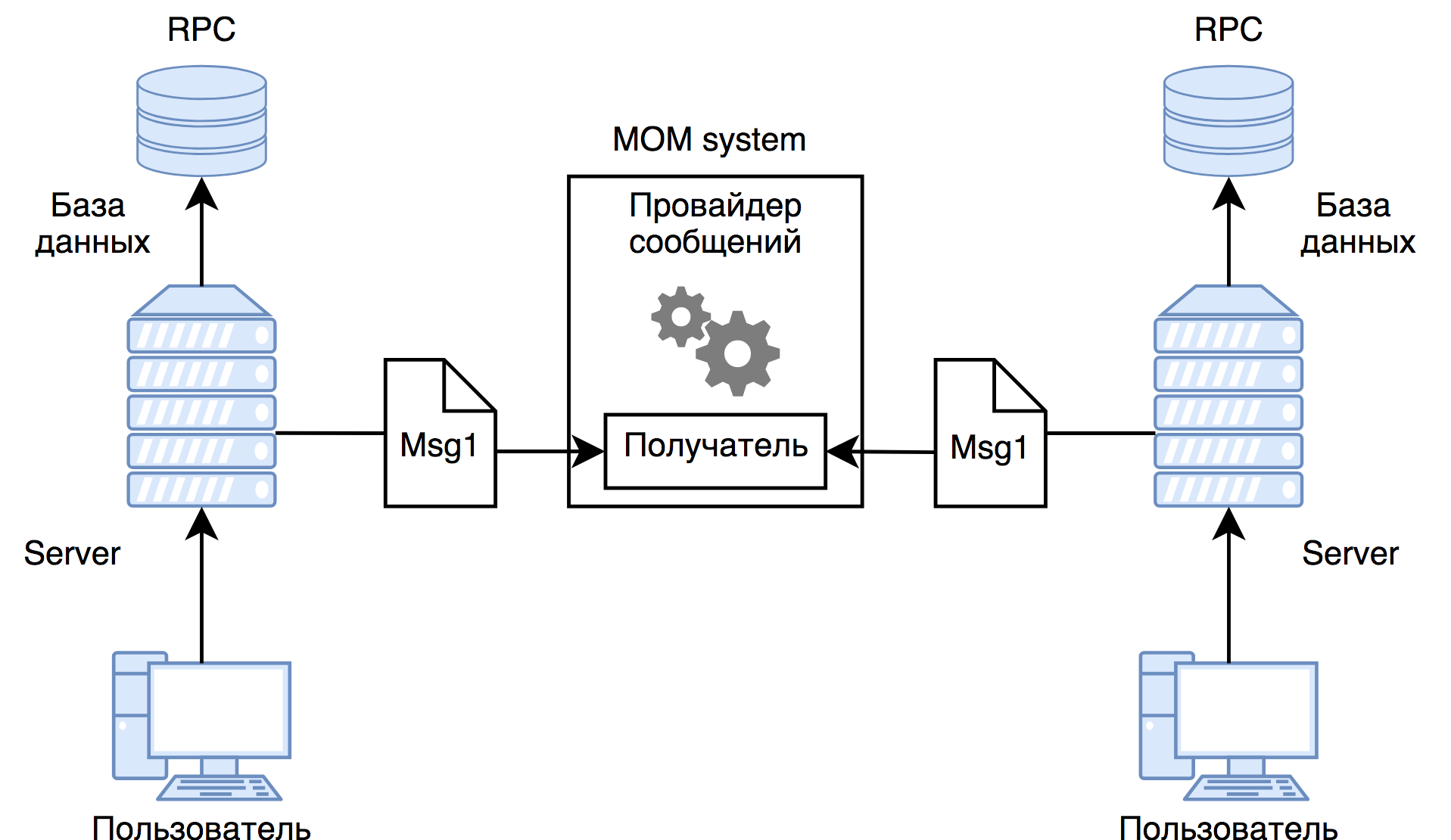


Рис. 2. Совмещение принципов RPC и MOM в архитектуре системы C-Commerce.

**Заключение**

В качестве базового подхода при концептуальном проектировании архитектуры системы C-Commerce использовался процессный подход, а также учитывались принципы открытости, масштабируемости, интероперабельности, глобализации (доступности), кастомизации (гибкости), аутсорсинга. На основе процессного подхода был выделен SCM подход и принят принцип совместного использования ресурсов. Для технической реализации выбран сервисно-ориентированный подход. Отмечена важность концепции открытых систем при организации взаимодействие систем.

Принципы глобализации, кастомизации и аутсорсинга являются отражением современных трендов развития ИТ-технологий.

Экстраполируя полученные результаты по построению системы C-Commerce на крупномасштабные открытые информационные системы, можно выделить следующие значимые характеристики концептуальной архитектуры КОИС:

• клиентоцентричность (наличие полной информации о клиенте, его потребностях, технических средствах, информационных ресурсах и т.п.);

• единое информационное пространство;

• гибкие механизмы настройки сложных продуктов и сквозная (непрерывная, полностью автоматизированная) обработка информационных ресурсов (STP –процессы);

• универсальный интерфейс для пользователей (клиентов) по взаимодействию с сервисами (серверами) – OpenAPI к платформе;

• наличие возможностей и механизмов машинного обучения;

• возможность глобального использования;

• максимальная надежность (24 часа, 7 дней в неделю);

• горизонтальное масштабирование на низкозатратном оборудовании;

• хранение и обработка данных в оперативной памяти (in memory) без использования дисковой памяти;

• использование программного обеспечения с открытым исходным кодом;

• кибербезопасность.

Список литературы

1. Д.Волков Цифровая платформа: вызовы и проблемы // Открытые системы. СУБД, 2018, № 2, https://www.osp.ru/os/2018/2/13054185/ (дата обращения 22.12.2020).

2. Свинарев С. Цифровая трансформация ИТ-дистрибьютора // itWeek №5 (941) 30 октября 2018, по материалам конференции MERLION IT Solutions Summit от 10 сентября 2018 https://www.itweek.ru/business/article/ detail.php?ID=203140 (дата обращения 22.12.2020).

4. Хасин М. «Технологический тренды, влияющие на ИТ-стратегии компаний: видение Сбербанка». Материалы конференции TAdviser SummIT, Москва, 30 мая 2018 г., http://www.tadviser.ru/index.php/Статья:Тенденции\_ мирового\_ИТ-рынка (дата обращения: 14.11.2018).

5. Аналитический отчет «App Annie Ретроспектива 2017» / App Annie, 2018. URL: www.appannie.com/ru/insights/market-data/app-annie-2017-retrospective/ (обращение 05.03.2018)

6. Аналитический отчет «App Annie Forecast 2017–2022» / App Annie, 2018. URL: www.appannie.com/en/insights/market-data/app-annie-2017 -2022-forecast/ (обращение 05.04.2018)

7. Андрианова Е.Г, Морнев А.А., Томашевская В.С. Принципы реализации концепции C-Commerce // Современные информационные технологии в управлении и образовании: Семнадцатая науч. практ. конф.: Сборн. научн. труд./редкол.: Волков Н.В. и др. – М.: Издательство «Проспект», 2018. - 164 - 170 с., ISBN: 978-5-98597-372-3

8. Suman Sarkar: The Supply Chain Revolution: Innovative Sourcing and Logistics for a Fiercely Competitive World: AMACOM; Special ed. edition: 2017. - ISBN-10: 0814438784

10. Supply Chain Management Системы управления цепями поставок [Электронный ресурс]: Деловой портал TAdviser; URL: http://www.tadviser.ru/index.php/SCM (дата обращения: 14.11.2018)

11. Технология открытых систем под общей редакцией А.Я. Олейникова. –М.: Янус-К, 2004, 288 с.

References

1. D. Volkov Digital Platform: challenges and problems // Open systems. DBMS, 2018, No. 2, https://www.osp.ru/os/2018/2/13054185/ (circulation date 22.12.2020).

2. Svinarev S. Digital transformation of the IT distributor // itWeek No. 5 (941) 30 October 2018, based on the MERLION IT Solutions Summit conference of 10 September 2018, https://www.itweek.ru/business/article/ detail.php?ID=203140 (circulation date 22.12.2020).

4. Hasin M. "Technological trends affecting the IT strategies of companies: Sberbank's vision". Proceedings of the TAdviser SummIT conference, Moscow, 30 May 2018, http://www.tadviser.ru/index.php/Статья:Тенденции\_ world\_IT market (circulation date: 14.11.2018).

5. Analytical report "App Annie Retrospective 2017" / App Annie, 2018. URL: www.appannie.com/ru/insights/market-data/app-annie-2017-retrospective/ (address: 05.03.2018).

6. Analytical Report "App Annie Forecast 2017-2022" / App Annie, 2018. URL: www.appannie.com/en/insights/market-data/app-annie-2017 -2022-forecast/ (address 05.04.2018)

7. Andrianova, E.G.; Mornev, A.A.; Tomashevskaya, V.S. Principles of the C-Commerce concept realization (in Russian) // Modern information technologies in management and education: Seventeenth scientific conference: Collection of scientific papers / edited by: Volkov, N.V. et al. - Moscow: Prospect Publishing House, 2018. - 164 - 170 p., ISBN: 978-5-98597-372-3

8. Suman Sarkar: The Supply Chain Revolution: Innovative Sourcing and Logistics for a Fiercely Competitive World: AMACOM; Special ed. edition: 2017. - SUMAN SARKAR: THE SUPPLY CHAIN REVOLUTION: INNOVATIVE SOURCING AND LOGISTICS FOR A FIERCELY COMPETITIVE WORLD: AMACOM; SPECIAL ED.

10. Supply Chain Management Systems [Electronic Resource]: TAdviser Business Portal; URL: http://www.tadviser.ru/index.php/SCM (date of address: 14.11.2018)

11. Technology of open systems under the general editorship of A.Y. Oleynikov. -M.: Yanus-K, 2004, 288 p.