

**Микляева Татьяна Вячеславовна**

Магистрант НАЧОУ ВПО СГА

**Направление:** Информатика и вычислительная техника

**Магистерская программа:** Распределенные автоматизированные системы

### **Применение Java-технологий для построения распределенных компонентных приложений**

**Аннотация.** На сегодняшний день язык программирования Java не уступает другим объектно-ориентированным языкам по популярности. Для создания распределенных компонентных приложений, использующих Java-технологии необходимо понимать основные требования, предъявляемые к данным системам. В данной статье рассматриваются основные требования к распределенным компонентным приложениям, основные Java-технологии и архитектурные стили для создания приложений.

**Ключевые слова:** распределенные системы, компоненты программного обеспечения, компонент, Java 2 Mobile Edition (J2ME), Java 2 Standard Edition (J2SE), Java 2 Enterprise Edition (J2EE), Java Beans, Java Database Connectivity (JDBC), Java Native Interface (JNI), Remote Method Invocation (RMI), Swing и Advanced Window Toolkit (AWT), Java Server Pages (JSP), многоуровневая система, независимые компоненты, данные–представление–обработка (MVC).

На сегодняшний день платформа Java привлекла более 9 миллионов разработчиков программного обеспечения. Она используется во всех главных сегментах индустрии, а также в широком диапазоне устройств, компьютеров и сетей.

Основная причина успеха языка Java – его быстрая изменчивость. Начиная с его первого выпуска 1.0, этот язык непрерывно адаптируется к изменениям в среде программирования и подходов к программированию.

Самое главное – он не просто следует тенденциям, а помогает их создавать. Способность адаптации языка к высокой скорости изменений в компьютерном мире – основная причина, по которой он остается столь успешным языком [2].

По данным рейтинговой компании Tiobe Programming Community в 2014 году Java занимает второе место среди языков программирования, при этом его опережает язык Си всего на 2 процента. Tiobe оценивают популярность языков по количеству программистов, курсов обучения и поставщиков инструментальных средств. Компания PyPL Popularity of Language, которая оценивает языки программирования по количеству поисковых запросов, касающихся руководств по языкам программирования ставит Java на первое место.

В отличие от некоторых других языков программирования, влияние которых со временем уменьшилось, язык Java стал только сильнее. Со времен своего первого выпуска он выдвинулся на передний край программирования для Интернета. Каждая последующая версия укрепляла эту позицию. Сегодня он все еще первый и является наилучшим выбором для разработки web-ориентированных приложений [2].

Для создания распределенных компонентных приложений необходимо понимать основные требования предъявляемые к данным системам.

Распределенные системы должны относительно легко поддаваться расширению, или масштабированию. Эта характеристика является прямым следствием наличия независимых компьютеров, но в то же время не указывает, каким образом эти компьютеры на самом деле объединяются в единую систему.

Распределенные системы обычно существуют постоянно, однако некоторые их части могут временно выходить из строя. Пользователи и приложения не уведомляются о том, что эти части заменены или починены, или что добавлены новые части для поддержки дополнительных пользователей или приложений.

Интернет является одной из самых известных в настоящее время распределенных систем. С самого начала эта сеть строилась как распределенная система, способная продолжать функционировать при уничтожении части (возможно – большей части) составляющих ее узлов.

Компоненты программного обеспечения – это простейшие структурные элементы, которые можно повторно использовать при построении программных систем. Они реализуют какие-либо прикладные функции информационной системы, представляя семантически значимые услуги прикладного или технического характера, и могут быть модифицированы в процессе разработки на уровне двоичных исполняемых кодов.

При этом компонент (component) – это независимая программная единица уровня приложения, разрабатываемая не для конкретного приложения, а с определенной целью. К программированию компонентов можно привлечь группу разработчиков, наделив каждого своими обязанностями, что, несомненно, сэкономит время на разработку. Проектирование особенно крупных приложений нередко поручают даже нескольким организациям.

Компонентные технологии разработки Web-приложений, ориентированы на повышение их гибкости, удобства их создания и поддержки, а также на снижение трудоемкости внесения изменений в приложения такого рода.

Использование тех или иных Java-технологий в общем случае зависит от платформы программирования Java: Java 2 Mobile Edition (J2ME), Java 2 Standard Edition (J2SE), Java 2 Enterprise Edition (J2EE), которые представляют собой разработку мобильных, стандартных и корпоративных приложений соответственно [2].

© Основными базовыми технологиями Java разработки распределенных Web-приложений. являются:

- Java Beans, программные компоненты, написанные на языке Java, и предусматривающие многократное использование программного кода;
- Java Database Connectivity (JDBC), средства для связи с базами данных, позволяющие обращаться к SQL базам данных от различных разработчиков;

– Java Native Interface (JNI), интерфейс «родных» программ, возможность взаимодействия Java программ с библиотеками кода, написанными на других языках;

– Remote Method Invocation (RMI), вызов удаленных методов, позволяет объекту Java вызывать методы иного Java-объекта, расположенного на другой машине. С помощью этого средства выполняется построение распределенных систем;

– Swing и Advanced Window Toolkit (AWT), базовые компоненты для построения графического интерфейса пользователя в Java приложениях;

– Java Server Pages (JSP), серверные страницы для создания CGI-интерфейса.

Приложения на базе J2EE строятся с использованием трех основных архитектурных стилей: многоуровневая система, независимые компоненты, данные–представление–обработка (MVC).

Многоуровневая система. Самые крупные подсистемы организованы как уровни, решающие различные задачи. Интерфейс взаимодействия с внешней средой, включая пользователей, реализуется при помощи Web-компонентов.

Уровень бизнес-логики и модели данных реализуется при помощи EJB-компонентов. Уровень управления ресурсами строится на основе коммерческих систем управления базами данных (СУБД). Можно также подключать другие виды ресурсов, для которых имеется реализация интерфейса поставщика служб J2EE (J2EE service provider interface, J2EE SPI).

Независимые компоненты. Первые два уровня построены из отдельных компонентов, каждый из которых имеет собственную область ответственности, но может привлекать для решения частных задач другие компоненты.

Данные–представление–обработка (MVC). Работа компонентов в рамках обработки группы тесно связанных запросов организуется по образцу MVC. При этом сервлеты и обработчики Web-событий служат обработчиками, компоненты JSP – представлением, а компоненты EJB – моделью данных.

## Литература

1. Беседа А.Б., Аноприенко А.Я. Технологии Java и особенности их применения в учебном процессе: Сборник научных трудов II-й международной научной конференции. ДонНТУ, май 2006. Донецк, 2007.
2. Шилдт Г. Java. Полное руководство 8-е изд. М.: Вильямс, 2012.
3. Эванс Б., Вербург М. Java. Новое поколение разработки. СПб.: Питер, 2014.

© Бюллетень магистранта 2014 год № 6