

Шаблон архитектуры распределенной системы, основанной на технологии блокчейн

С.А. Кучеров, Ю.Ю. Липко, Ю.И. Rogozov, А.С. Свиридов

Южный федеральный университет, Ростов-на-Дону

Аннотация: Тенденцией развития архитектур проектирования информационных систем является их усложнение и движение в сторону все большего распределения функций и компонентов системы. Одним из вариантов разработки децентрализованных систем являются системы, основанные на блокчейн-технологиях. В статье рассматривается архитектурный шаблон для приложений, разрабатываемых на базе технологии блокчейн. Рассматриваются основные элементы многоуровневой архитектуры, являющейся основой при построении приложений такого типа. Выделены шесть слоев, которые описывают инфраструктурный уровень, уровни данных, сетевой, консенсуса, прикладной и представления. Проводится краткий анализ основных особенностей и отличий реализации каждого из уровней архитектурного шаблона. По результатам анализа, наиболее специфичные для систем на основе блокчейн компоненты, реализуются на уровне слоев консенсуса и данных.

Ключевые слова: децентрализованное приложение, блокчейн-платформа, информационная система, многоуровневая архитектура, шаблон.

Введение

Выбор архитектуры при проектировании информационной системы является ключевым решением для создания будущей системы. Тенденцией развития архитектур является их усложнение, начиная от клиент-серверной архитектуры и заканчивая движением в сторону все большего разделения функций и компонентов по различным узлам системы.

Для решения задач, требующих децентрализации ответственности, хранения данных и выполнения бизнес-логики (например, задач управления транспортными системами [1], документооборотом [2], управления активами [3]) появились новые подходы к построению систем, основанные на технологиях блокчейн. Системы такого класса имеют свои отличия в строении, диктуемые особенностями реализации. Технологии блокчейн (распределенного реестра) дают возможность хранить данные пользователя в распределенном виде, обеспечивая при этом прослеживаемость, неизменность и достоверность [4].

Также распространение получил подход по построению распределённых приложений – DApps (Distributed Application). Децентрализованные приложения [5] имеют архитектуру системы, в которой отсутствуют выделенные узлы, контролирующие деятельность других узлов. Также особенностями децентрализованной модели является [6]:

- механизм консенсуса, который гарантирует синхронизацию состояния всей системы без наличия управляющих узлов;
- open-source реализации;
- системы саморегуляции при распределении ресурсов системы.

Многоуровневая архитектура децентрализованных приложений

Архитектура распределённых приложений отличается от интернет-приложений, построенных на web-архитектуре. Основные отличия содержатся на уровне хранения данных и уровне прикладной логики, а также имеются компоненты, специфичные для данной технологии.

Основное отличие находится в компонентах, отвечающих за хранение данных: в отличие от традиционной базы данных - блокчейны хранят данные в блоках, которые объединяются в цепочку [5]. После заполнения блока данными, он присоединяется к предшествующим блокам, в результате чего данные объединяются в цепочку в хронологическом порядке. Новые данные формируют новые блоки в цепочке, причем старые блоки, а, следовательно, и информация в них изменению не подлежит.

Для разработки приложений на основе блокчейн технологии может быть использован архитектурный шаблон «n-уровневая архитектура». С учетом особенностей данного типа приложений она будет состоять из 6 уровней (рис. 1). Данный шаблон используется для организации программ, которые можно разложить на группы подзадач, относящихся к определенным уровням абстракции. Каждый слой предоставляет свои службы через открытый интерфейс для следующего сверху уровня.

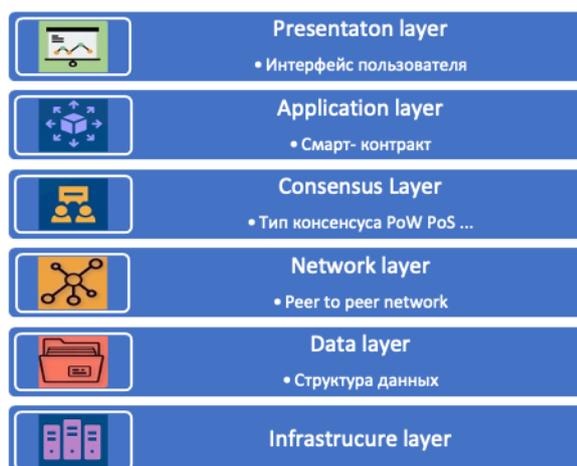


Рис. 1. – Уровни распределённого приложения

На каждом слое системы реализуются набор различных компонентов. Эти компоненты реализуются в виде совокупности классов, объектов и пакетов, которые выполняют общую задачу.

Presentation layer (уровень представления) содержит компоненты пользовательского интерфейса приложения - web-страницы, формы интерфейса пользователя или программные протоколы взаимодействия, которые обеспечивают взаимодействие с конечными пользователями. Данный уровень не содержит значительных отличий от приложений других типов [7].

Логика работы приложения определяется на уровне прикладной логики и реализуется в виде смарт контрактов, которые программируются с использованием специализированных языков или языков высокого уровня общего назначения. В каждой блокчейн платформе используются различный набор поддерживаемых в смарт-контрактах языков. Смарт-контракт является программным кодом, который может быть встроен в блокчейн. Смарт-контракты работают по определённому алгоритму, и активизируются при выполнении заданных условий.

Уровень консенсуса содержит реализацию одного из алгоритмов консенсуса - механизма, который отвечает за координацию действий узлов

системы в распределенной сети. Он обеспечивает достижение согласия между всеми участниками сети о текущем состоянии данных в блокчейне.

Сетевой уровень реализует peer-to-peer взаимодействие узлами, обслуживающими блокчейн. На нем реализуется обнаружение транзакций и распространение блоков. Задачей этого уровня является обеспечение возможности узлам находить друг друга и взаимодействовать, распространять и синхронизировать информацию.

На уровне данных осуществляется хранение информации в виде цепочки блоков. Разработка компонентов по работе с данными имеет сходные черты с разработкой с использованием СУБД с точки зрения используемого набора типовых компонентов по отношению к уровню операций по работе с данными [8]. Так как операции с данными являются типовыми, они могут быть унифицированы и реализованы в виде фреймворка или аналога технологии ORM (Object-Relational Mapping) для того, чтобы обеспечить работу с данными в терминах классов [9].

Инфраструктурный уровень представлен аппаратными узлами сети, реализующими хранение данных и обработку смарт-контрактов.

Многоуровневый архитектурный шаблон, описанный выше, является универсальным для систем, построенным на основе блокчейн-технологии. Компоненты системы, создаваемые на каждом уровне могут иметь различные технологии реализации, учитывающие особенности функционирования блокчейн-платформы. Например, могут быть использованы различные алгоритмы консенсуса и хранения данных, такие, как Proof of Work, Proof of Stake или их разновидности. Также могут отличаться механизмы работы с данными, позволяя сохранять в блокчейне все пользовательские данные или же их ключевые элементы/метаданные (off-chain), что повышает эффективность работы приложений.

Заключение

Сегодня в мире и России активно развиваются проекты с использованием блокчейн – как в корпоративных, так и в государственных системах, что говорит о готовности технологии к широкому применению. Поэтому формирование шаблонов проектирования является важным процессом, который позволяет выделить эффективные способы решения типовых задач проектирования. Развитие методов проектирования и выделение архитектурных шаблонов для блокчейн-систем будет способствовать снижению сложности разработки таких систем. Одним из возможных способов снижения трудоёмкости [10] станет использование выделенных шаблонов в low/zero code подходах к созданию систем, что сделает процесс разработки максимально приближенным к конечному пользователю.

Благодарности

Исследование выполнено при финансовой поддержке РФФИ в рамках научного проекта 20-07-00995А.

Литература

1. Лапшин В.С., Елькин Д.М., Кучеров С.А., Рогозов Ю.И. Обзор методов проектирования архитектур интеллектуальных транспортных систем // Инженерный вестник Дона. 2018, № 4. URL: ivdon.ru/ru/magazine/archive/n4y2018/5422
2. Пескова О.Ю., Половко И.Ю., Захарченко А.Д. Применение блокчейн-технологий в системах электронного документооборота: анализ и программная реализация // Инженерный вестник Дона, 2019, №3. URL: ivdon.ru/ru/magazine/archive/n3y2019/5801
3. Тихонова К. В., Гаранова М.В., Бурдова Д.В., Тихонов Д.А. Оптимизация системы управления объектами недвижимости на основе

внедрения технологии блокчейн в учетно-регистрационную процедуру
Инженерный вестник Дона, 2019, № 7. URL:
ivdon.ru/ru/magazine/archive/n7y2019/6078

4. Buterin V. Ethereum: a next generation smart contract and decentralized application platform. 2013. URL: ethereum.org/en/whitepaper (дата обращения: 01.11.2022).

5. Равал С. Децентрализованные приложения. Технология Blockchain в действии. СПб.: Питер, 2017. 240 с.

6. Фаулер М. Архитектура корпоративных программных приложений. М.: Издательский дом "Вильямс", 2006. 544 с.

7. Belikova S.A., Rogozov Y.I., Sviridov A.S. [et al.] Approach to user interfaces development based on semantic model of user activity // Journal of Physics: Conference Series, 1457 (1), 2020, art. no. 012012.

8. Романенко К. С., Свиридов А. С. Анализ преимуществ использования блокчейн-платформ при проектировании распределённых информационных систем // Информатизация и связь. 2021. № 3. С. 19-22.

9. Candel C. J. F., Sevilla Ruiz D., García-Molina J. J. (2022). A unified metamodel for NoSQL and relational databases. Information Systems, 104 doi:10.1016/j.is.2021.101898

10. Рогозов Ю.И., Лапшин В.С., Кучеров С.А., Боровская М.А. Современные тенденции построения low-code платформ // Информатизация и связь. 2022. № 2. С. 16-20.

References

1. Lapshin V.S., El'kin D.M., Kucherov S.A., Rogozov Yu.I. Inzhenernyj vestnik Dona, 2018, № 4. URL: ivdon.ru/ru/magazine/archive/n4y2018/5422.

2. Peskova O.Yu., Polovko I.Yu., Zaxarchenko A.D. Inzhenernyj vestnik Dona, 2019, №3. URL: ivdon.ru/ru/magazine/archive/n3y2019/5801



3. Tixonova K. V., Garanova M.V., Burdova D.V., Tixonov D.A. Inzhenernyj vestnik Dona, 2019, № 7. URL: ivdon.ru/ru/magazine/archive/n7y2019/6078
4. Buterin V. Ethereum: a next generation smart contract and decentralized application platform. 2013. URL: ethereum.org/en/whitepaper (accessed: 01.11.2022).
5. Raval S. Decentralizovanny`e prilozheniya. Texnologiya Blockchain v dejstvii [Decentralized applications. Blockchain technology in action]. SPb.: Piter, 2017. 240 p.
6. Fauler M. Arxitektura korporativny`x programmny`x prilozhenij [Architecture of corporate software applications]. M.: Izdatel`skij dom "Vil`yams", 2006. 544 p.
7. Belikova S.A., Rogozov Y.I., Sviridov A.S. [et al.] Journal of Physics: Conference Series, 1457 (1), 2020, art. no. 012012.
8. Romanenko K. S., Sviridov A. S. Informatizaciya i svyaz`. 2021. № 3. pp. 19-22.
9. Candel, C. J. F., Sevilla Ruiz, D., & García-Molina, J. J. (2022). A unified metamodel for NoSQL and relational databases. Information Systems, 104 doi:10.1016/j.is.2021.101898
10. Rogozov Yu.I., Lapshin V.S., Kucherov S.A., Borovskaya M.A. Informatizaciya i svyaz`. 2022. № 2. pp. 16-20.