

РАЗРАБОТКА И ОЦЕНКА ЭФФЕКТИВНОСТИ ПРИМЕНЕНИЯ СЦЕНАРНОЙ ПАРАДИГМЫ ОПИСАНИЯ РАСПРЕДЕЛЕННЫХ СИСТЕМ

Суховаей Я. Ю. (ИТМО)

**Научный руководитель – к. т. н., Быковский С. В.
(ИТМО)**

Введение. В современных реалиях большинство вычислительных систем являются распределенными. Это означает, что они представлены комбинацией систем меньшего размера, взаимодействующих между собой через каналы связи. Примером можно считать умные дома, производства и города, являющиеся частями “Интернета вещей”, или системы меньшего масштаба вроде автомобилей или космических аппаратов. В любом случае сложное поведение распределенной системы, которое обычно представляется в виде бизнес-процессов или диаграмм последовательности, приходится дробить и перегруппировывать, чтобы сформировать наборы реакций/поведения, привязанных к каждому конкретному вычислительному узлу системы, что требует дополнительных трудозатрат и является причиной возникновения сложностей и проблем во всем процессе. Решение этих трудностей требует разработки нового представления и подходов к проектированию и разработке программного обеспечения распределенных систем.

Основная часть. Используемый сейчас подход к разработке программного обеспечения распределенных систем обычно предполагает описание поведения в виде диаграмм последовательности или бизнес-процессов на этапе проектирования с последующим разбиением этого представления на части и их распределение по модулям, выполняющим свои наборы задач. Одному аппаратному узлу обычно соответствует один или несколько программных модулей-акторов. Разработка всех выделенных частей как правило происходит независимо, в виде отдельных проектов, которыми могут заниматься разные команды программистов. После разработки следует этап интеграции всех этих частей с проверкой их взаимодействия. Так или иначе проектирование и разработка сводится к рассмотрению распределенной системы через нечто похожее на модель акторов, где основной единицей является узел-актор. У такого подхода есть ряд недостатков. Так, например, при масштабировании системы или добавлении в систему нового поведения, затрагивающего несколько узлов, требуется переписывание исходного кода нескольких разных проектов с проведением повторной интеграции во избежание появления новых ошибок взаимодействия. Такой процесс трудозатратен и может занимать значительное время.

Из существующих на данный момент альтернатив для распределенных систем наиболее перспективной можно считать модель реакторов, описанную в статьях [1] и [2]. В ней каждый реактор является дополнением актора, содержащим входы, на изменения которых могут быть настроены реакции, и выходы. За счет привязки к дискретным событиям достигается детерминизм поведения всей системы. Для этой модели был разработан инструмент – язык координации Lingua Franca, упоминаемый в работе [3]. Он дает возможность работать с логикой на привычных языках программирования, помещая ее в модули-реакторы, которые выполняют эту логику в качестве реакций на изменения входов. Однако такое представление предполагает такое же разделение поведения, как было описано ранее, а это означает, что проблемы с рассинхронизацией версий проектов и трудоемким этапом интеграции реакторная модель не решает.

Целью данной работы является снижение трудоемкости разработки программного обеспечения распределенных систем за счет использования сценарной парадигмы, позволяющей автоматизировать реализацию механизмов интеграции и синхронизации распределенных компонентов. В работе также проводится оценка эффективности сценарной парадигмы в сравнении с используемым в данный момент разделением на узлы-акторы. Приводится описание самой сценарной парадигмы для распределенных систем, принципы,

которые возникают в ней, и примеры ее использования, а также описание парадигмы в виде математических сущностей. В работе приводится эксперимент с описанием 10 различных распределенных систем при помощи сценарной парадигмы, диаграмм последовательности и применяемого в современной разработке разделения на акторы. Этот эксперимент доказывает эффективность сценарной парадигмы по сравнению с разделением поведения на узлы-акторы в плане метрик трудоемкости и сложности программного кода, основанных на метриках кода Холстеда.

Выводы. Существующие подходы для работы с распределенными системами вроде акторной и реакторной модели обладают недостатками, вызванными необходимостью разделять поведение на узлы-акторы. Это приводит к увеличению трудоемкости разработки, сложностям с масштабированием и синхронизацией версий частей, а также к возможным нарушениям детерминизма поведения всей системы. Разработанная сценарная парадигма описания распределенных систем решает эти проблемы за счет альтернативного разделения поведения на сценарии – простые распределенные программы, а не узлы-акторы. Принципы этой парадигмы позволяют сохранять детерминизм при совмещении сценариев друг с другом для формирования сложного поведения. Для оценки эффективности применения такой парадигмы был проведен эксперимент со снятием метрик, основанных на метриках кода Холстеда для разделения поведения на узлы-акторы в рамках существующих подходов и для предлагаемого новой парадигмой сценарного разделения.

Список использованных источников:

1. Marten Lohstroh, Christian Menard, Soroush Bateni, Edward A. Lee. Toward a Lingua Franca for Deterministic Concurrent Systems // ACM Transactions on Embedded Computing Systems (TECS), 2021. V. 20. I. 4 N. 36. P. 1-27. <https://doi.org/10.1145/34448128>
2. Marten Lohstroh, Í. Í. Romeo, Andrés Goens, P. Derler, J. Castrillón, Edward A. Lee, A. Sangiovanni-Vincentelli. Reactors: A Deterministic Model for Composable Reactive Systems // International Workshop on Design, Modeling, and Evaluation of Cyber Physical Systems, 2019. P. 59-85. https://doi.org/10.1007/978-3-030-41131-2_4
3. Edward A. Lee, Soroush Bateni, Shaokai Lin, Marten Lohstroh, Christian Menard. Quantifying and Generalizing the CAP Theorem // ArXiv, 2021. V. abs/2109.07771. <https://doi.org/10.48550/arXiv.2109.07771>