

**ЯЗЫК ОПИСАНИЯ РАСПРЕДЕЛЕННЫХ ВЫЧИСЛЕНИЙ НА БАЗЕ
СОБЫТИЙНО-ОРИЕНТИРОВАННОЙ АРХИТЕКТУРЫ ВЗАИМОДЕЙСТВИЯ****Суховей Я. Ю. (ИТМО)****Научный руководитель – к. т. н., Быковский С. В.
(ИТМО)**

Введение. Сложность современных систем со временем только увеличивается, но так или иначе они представляют из себя объединенные различными каналами связи узко специализированные вычислительные системы или компьютеры. Каждая из составляющих системы способна функционировать автономно и выполнять различные задачи, которые от нее требуют, но именно объединение таких систем позволяет решать сложные задачи. Каждая из систем выполняет лишь какую-то часть вычислительного процесса, после чего передает данные по каналам связи дальше. Примеры таких систем можно найти повсюду: в космических аппаратах, на автоматических линиях производства или даже в компьютерных онлайн играх, где есть клиенты и серверы, постоянно обменивающиеся данными и выполняющие с ними различные действия. В таких случаях важно правильно описать поведение системы в целом.

Основная часть. При разработке программного обеспечения для составных систем в современных реалиях часто используются языки программирования, такие как C/C++, если речь идет о встраиваемых системах, или же языки более высокого уровня наподобие Java, Python или JavaScript, если это позволяют вычислительные мощности и аппаратные характеристики целевой машины. Каждый язык имеет свою сферу применения и особенности, но так или иначе все сводится к тому, что в программном коде описывается поведение каждой части системы по отдельности. Существуют и особые инструменты, предназначенные для параллельных вычислений, например, Erlang[1] и Elixir[2], компилируемые под beam машину, или OpenCL[3]. Однако все они так или иначе описывают поведение отдельных автономных частей, которые могут взаимодействовать друг с другом. Возможность абстракции от конкретных узлов, исполняющих те или иные действия, до уровня всей системы, описание ее поведения целиком с возможностью автоматически транслировать понятное человеку представление в исполняемые файлы для всех частей способно ускорить разработку составных систем, поскольку человеку будет легче освоить такой инструмент. Такой подход также может поспособствовать упрощению разработки программного обеспечения для составных встраиваемых систем, что позволит данной сфере развиваться активнее. Целью данной работы является разработка грамматики подобного высокоуровневого языка с учетом возможных конструкций, которые могут понадобиться разработчику, а также разработка компонентов транслятора этого языка в исполняемые файлы для целевых устройств. В работе описываются особенности грамматики разработанного языка и то, с какой целью была внедрена та или иная из них. Также описываются разработанные и разрабатываемые стадии трансляции языка, включающие сжатие абстрактного синтаксического дерева, импортирование частей и валидацию абстрактного синтаксического дерева, разметка система, трассировка системы, трансляция в промежуточное представление с учетом возможных оптимизаций и компиляция промежуточного представления под целевую платформу. Рассматривается, что и как должно происходить на каждом из этапов трансляции, и возникающие при этом конфликтные ситуации.

Выводы. Ввиду отсутствия прямых аналогов подобного универсального языка предлагается собственная грамматика и реализация компонентов транслятора языка описания распределенных вычислений на базе событийно-ориентированной архитектуры взаимодействия.

Список использованных источников:

1. Смирнов А. С., Выжигин А. Ю. Сравнительный анализ языков Elrang и C# // Вестник Московского государственного университета приборостроения и информатики. ISSN: 2079-8792, 2013, С. 101 - 106.
2. Branch Ph., Weinstock Ph., Functional programming for the internet of things: a comparative study of implementation of a LORA-MQRTT gateway written in Elixir and C++. Electronics. eISSN: 2079-9292, 2024, P. 3427.
3. Михелёв В. В. Исследование эффективности применения технологии параллельного программирования OpenCL в алгоритмах сортировки // Международная научно-техническая конференция молодых ученых БГТУ им. В. Г. Шухова. Белгород, 01–30 мая 2015 года.