

ОБ ИМИТАЦИОННОЙ МОДЕЛИ МУЛЬТИКЛИЕНТСКОГО КЛАСТЕРА БАЗ ДАННЫХ

Бойцов Е.А., Соколов В.А.

ЯрГУ им. П.Г. Демидова, факультет ИВТ, Россия, Ярославль, boytsovea@yandex.ru
ЯрГУ им. П.Г. Демидова, факультет ИВТ, Россия, Ярославль, valery-sokolov@yandex.ru

Одним из важных аспектов разработки облачного сервиса является организация подсистемы хранения данных. К сожалению, существующие реляционные СУБД не обладают достаточной степенью масштабируемости, чтобы быть легко интегрированными в инфраструктуру типичного облачного приложения. Новые подходы, такие как использование мультиклиентской (multi-tenant) архитектуры, с одной стороны, уменьшают затраты провайдера облачного сервиса, а с другой стороны - увеличивают сложность и стоимость разработки.

Одним из вариантов решения проблемы организации подсистемы хранения данных является концепция мультиклиентского кластера баз данных, которая ориентирована на приложения с большим числом (десятки и сотни тысяч) небольших и средних компаний-клиентов. Мультиклиентский кластер баз данных является дополнительным уровнем абстракции, предоставляющим программный интерфейс, который был бы максимально близок к интерфейсу традиционных СУБД. Основными преимуществами такого решения являются упрощение разработки приложений, связанное с возможностью использования разработчиками уже имеющихся знаний, а также упрощение администрирования системы. Для разработки и сравнения алгоритмов, необходимых для организации кластера, был проведен анализ статистики работы существующего облачного приложения с мультиклиентской архитектурой. Основываясь на данной статистике, был выявлен ряд зависимостей между характеристиками клиента и порождаемым им потоком запросов. Результатом исследований стала разработка имитационной модели кластера.

Имитационная модель является графическим приложением под операционную систему Linux и разработана с использованием языка программирования C++ и фреймворка Qt. Модель работает в условном дискретном времени, от одного события до другого, и состоит из четырех основных типов объектов: очереди событий, диспетчера, обработчиков и источников событий, а также наблюдателей. Источники событий генерируют новые события и помещают их в очередь на обработку. После этого диспетчер извлекает первое событие из очереди и направляет его соответствующему обработчику. Обо всех этапах обработки каждого события ядро модели оповещает всех наблюдателей, которые ответственны за сбор статистики о работе моделей, построение отчетов и выполнение ряда служебных задач. Разработанная модель способна генерировать большое количество графических отчетов о своей работе.

В ходе первых экспериментов с использованием модели было выявлено, что даже в тех случаях, когда интенсивность входного потока меньше пропускной способности кластера, использование примитивных стратегий может приводить к падению производительности приложения. В рамках исследования планируется дальнейшее совершенствование модели и выявление ключевых факторов, влияющих на эффективность работы предлагаемой системы управления кластером.