

экспоненциальному и нормальному законам распределения $F(x)$, подтверждают справедливость приведенных выводов.

Полученные результаты могут быть использованы для статистической обработки данных различных технических и экономических исследований в случаях выборок конечного объема.

Литература.

Г. Дэйвид. Порядковые статистики. – М.: Наука, 1979. - 336с.

Караберов А.В., студент факультета Компьютерных наук; Научный руководитель - **Панфёрова И.Ю.** доцент кафедры Информационных управляющих систем, Харьковский национальный университет радиоэлектроники

ПРИМЕНЕНИЕ ГИБРИДНОЙ МОДЕЛИ ТРАНЗАКТОРОВ ДЛЯ ПОСТРОЕНИЯ ВСТРАИВАЕМОЙ МОБИЛЬНОЙ БАЗЫ ДАННЫХ

RESTful web-services is now a preferable way of physical realization of SaaS models. Such services need responsive mobile application clients. The key factor providing responsiveness is data models persistence. Because of growing complexity of mobile clients, where all background interactions are made in concurrent, multithreaded environment the main problem is concurrency control. This paper considers the way of creating a mobile database for such a mobile client, which is based on unifying actors model, transactional memory and shared-nothing architecture in one computational unit, called a transactor.

Основной бизнес-моделью продажи и использования программного обеспечения на текущий момент является SaaS-модель. В случае использования SaaS-модели поставщик разрабатывает веб-приложение и самостоятельно управляет им, предоставляя заказчику доступ к программному обеспечению через Интернет. Современной технологией реализации SaaS-модели на данный момент является концепция веб-сервисов. Клиентами веб-сервисов часто выступают мобильные приложения, которые должны обеспечивать высокую скорость работы и малое время отклика и реагирования. Ключевой особенностью для обеспечения выполнения данных требований является локальное кэширование данных предметной области, для которой проектируется веб-сервис. В случае использования проблемно-

ориентированного подхода к проектированию архитектуры мобильного приложения под этими данными подразумеваются объектные модели предметной области. В случае, когда необходима высокая скорость работы приложения для объектных моделей должно выполняться свойство персистентности, то есть они должны быть сохранены в локальной базе данных. Главным препятствием на пути создания локальной мобильной базы данных является обеспечение конкурентного доступа к данным и высокой скорости выполнения транзакций, не нарушая при этом ACID-принципов. К сожалению, все современные методы обеспечения конкурентного доступа к данным, сериализуемости и атомарности транзакций преимущественно основаны на модели потоков и блокировок (threads and locks). Данная модель имеет два существенных недостатка: низкая производительность и опасность возникновения взаимоблокировок транзакций.

В качестве альтернативы предлагается другой принцип взаимодействия и работы транзакций в конкурентной среде, основанный на модели акторов. Модель акторов [1] — это математическая модель, где примитивом параллельного численного расчета является вычислительная сущность под названием актор. Ключевыми особенностями акторов являются:

- возможность отправлять конечное, счетное число сообщений другим акторам;
- выбирать тип поведения для следующего сообщения;
- порождать новые акторы.

В терминологии распределённых вычислений систему, построенную на основе модели акторов, можно отнести к классу SN-систем (shared nothing), в которых каждый узел независим от других, система не имеет общих данных, а обмен информацией происходит на основе механизма передачи сообщений. Существует большое количество распределенных серверных баз данных, поддерживающих данный принцип. Их принято относить к классу NewSQL баз данных. К таким БД можно отнести NonStop SQL, Datomic, HStore [2]. На данный момент не существует ни одного мобильного NewSQL решения. Это можно объяснить тем, что модель акторов и SN-архитектура по большей части реализуются для больших распределенных систем с числом узлов больше ста. Но, так как все современные смартфоны планшетные компьютеры имеют многоядерные процессоры с числом ядер от 2 до 8, и уже анонсированы мобильные чипы от компании Adapteva с числом ядер до 64 [3], а также мобильные операционные системы, такие как iOS, Android, Tizen предоставляют возможность легко реализовывать модель многопоточных вычислений, становится осуществимой возможность реализации мобильной, встраиваемой базы данных на основе модели акторов и новой модели транзакционной памяти. Модель транзакционной памяти (transactional memory, TM) позволяет представлять транзакции как структуры данных в отличие от

обычных SQL хранилищ, где транзакция – это набор команд (утверждений, то есть, по сути строк). Для создания базы данных подобного типа предлагается ввести новую фундаментальную гибридную вычислительную сущность, объединяющую концепции актора и транзакционной памяти — транзактор [4]. Транзактор представляет собой поток, который может отправлять и принимать сообщения другим транзакторам, то есть реализует модель актора, но при этом он также может использовать возможности ТМ модели и читать и записывать данные в атомарный блок памяти. Подобно акторам каждый транзактор имеет «почтовый ящик» (mailbox), который сохраняет все посланные данному транзактору сообщения в виде очереди. Также транзактор имеет дескриптор транзакции – это структура данных, которая хранит всю информацию, связанную с данной транзакцией, подобно дескриптору процесса в операционных системах. Следует сказать, что дескриптор транзакции создаётся и сохраняется в локальной переменной потока, каждый раз, когда инициализируется новый атомарный блок. Транзакторы дают возможность программисту отправлять и принимать сообщения в рамках транзакции. Следует сказать, что если транзактор будет завершён аварийно, то будут уничтожены и все сообщения, ассоциированные с ним.

В случае применения модели транзакторов для проектирования мобильной базы будет сохранен высокий уровень изолируемости транзакций, характерный для ТМ-модели, и при этом будет сохранена высокая производительность, характерная для модели акторов. Также за счет отсутствия разделяемых данных и возможности асинхронно отправлять и принимать сообщения от других транзакторов исключена возможность взаимоблокировки. Модель транзакторов для мобильной базы данных легка в реализации, поскольку уже существуют готовые, высокопроизводительные реализации модели акторов (язык Erlang) и транзакционной памяти (база данных Datomic и язык Clojure). Внедрение базы данных, построенной по данному принципу, в мобильные приложения-клиенты SaaS веб-сервисов существенно повысит производительность этих приложения для конечного пользователя.

Литература

1. John Mitchell Concepts in programming languages // Cambridge University Press. — Cambridge, 2003 г., 529 стр.
2. Mattnew A. How will the database incumbents respond to NoSQL and NewSQL // The 451 Group. — NY., 2011 г.
3. Adapteva Epiphany chip [Электронный ресурс] — Режим доступа: <http://www.adapteva.com/epiphanyiv/>, свободный. — Загл. с экрана.
4. Lesani M. Transactors: Unifying transactions and actors // EPFL. — Lausanne, 2009 г.