

процесс есть, вызывающему процессу придется подождать, в противном случае запрос удовлетворяется.

Переменные состояния не являются счетчиками. В отличие от семафоров они не аккумулируют сигналы, чтобы впоследствии воспользоваться ими. Это означает, что в случае выполнения операции signal на переменной состояния, с которой не связано ни одного заблокированного процесса, сигнал будет утерян. Проще говоря, операция wait должна выполняться прежде, чем signal. Это правило существенно упрощает реализацию. На практике это правило не создает проблем, поскольку отслеживать состояния процессов при необходимости не очень трудно. Процесс, который собирается выполнить signal, может оценить необходимость этого действия по значениям переменных.

Существует несколько языков программирования, поддерживающих мониторы, хотя и не всегда в соответствии с моделью Хоара и Бринча Хансена. Один из таких языков – Java, объектно-ориентированный язык, поддерживающий потоки на уровне пользователя и позволяющий группировать методы (процедуры) в классы. Добавление в описание метода ключевого слова synchronized гарантирует, что если хотя бы один поток начал выполнение этого метода, ни один другой поток не сможет выполнять другой синхронизированный метод из этого класса.

Мониторы являются структурным компонентом языка программирования, и компилятор должен их распознавать и организовывать взаимное исключение. В Pascal, C и многих других языках нет мониторов. В этих языках также нет и семафоров, но их легко добавить: нужно всего лишь присоединить к библиотеке две короткие программы, написанные на ассемблере и реализующие системные вызовы up и down. Компиляторы при этом не обязаны знать об их существовании. Разумеется, операционная система должна знать о семафорах, но даже если у вас операционная система с семафорами, вы можете писать программы для нее на C или C++. Если же операционная система с мониторами, необходим язык со встроенными мониторами.

Другая проблема, связанная с мониторами и семафорами, состоит в том, что они были разработаны для решения задачи взаимного исключения в системе с одним или несколькими процессорами, имеющими доступ к общей памяти. Помещение семафоров в разделенную память с защитой в виде команд TSL может исключить состояния состязания. Эти примитивы будут неприменимы в распределенной системе, состоящей из нескольких процессоров с собственной памятью у каждого, связанных локальной сетью.

Вывод из всего вышесказанного следующий: семафоры являются примитивами слишком низкого уровня, а мониторы могут использоваться только в некоторых языках программирования.

Список литературы

1. Таненбаум Э., Бос Х. Современные операционные системы. – СПб.: Питер, 2015. – 1120 с.
2. Бершадская Е.Г. Анализ технологий поддержки научных исследований // XXI век: итоги прошлого и проблемы настоящего плюс. 2015. – № 3 (25). – С. 11–17.

ПРИМЕНЕНИЕ ОБЛАЧНЫХ ТЕХНОЛОГИЙ

Кашицин И.М., Сальников И.И.

Пензенский государственный технологический университет, Пенза, e-mail: alexey314@ya.ru

Суть концепции облачных технологий заключается в предоставлении конечным пользователям удаленного динамического доступа к услугам, вычислительным ресурсам и приложениям через Интернет. Большинство сервис-провайдеров предлагают облач-

ные вычисления в форме VPS-хостинга, виртуального хостинга, и ПО-как-услуга(SaaS). Облачные услуги долгое время предоставлялись в форме SaaS, например, Microsoft Hosted Exchange и SharePoint.

В 2011 году WINDOWS AZURE была объявлена коммерческой системой. Как и традиционная ОС, WINDOWS AZURE позволяет запускать приложения и хранить данные, но происходит это не на компьютере пользователя, а в вычислительных облаках.

Существуют два типа рабочих версий облачного приложения: веб-роль (Web role) и рабочая роль (Worker role). Первая умеет обрабатывать HTTP- или HTTPS-запросы, и на ее виртуальной машине (BM) запущен сервер Internet Information Services (IIS). Программист имеет возможность создать версию веб-роли с помощью ASP.NET либо Windows Communication Foundation (WCF), а также воспользоваться любой другой технологией .NET, работающей с IIS. Приложение может быть создано на любом языке программирования.

Приложения, созданные на основе WINDOWS AZURE, предоставляются как сервис физическим лицам, корпоративным пользователям или и тем, и другим одновременно. С помощью WINDOWS AZURE независимый разработчик программного обеспечения может создавать приложения для бизнес-пользователей, применяя принципы программного обеспечения как сервиса.

Примером может послужить решение, разработанное американской компанией Alinean, Inc. Ее сфера деятельности – предоставление по запросу аналитических средств в области анализа продаж и маркетинга. Системы Alinean позволяют оценить нужды и возможности бизнеса в будущем, предложить решение для наращивания мощностей и подсчитать, когда начнут окупаться инвестиции. Пользователями Alinean являются корпоративные клиенты, находящиеся в разных уголках земного шара. Среди них IBM, HP, Microsoft, Intel, AT&T, VMware, Oracle, Siemens, Symantec и др. В дата-центре Alinean, находящемся в Орландо (Флорида, США), сервис по запросу предоставляли 20 серверов, работающих 24 часа в сутки семь дней в неделю. Объем бизнеса рос, и мощностей стало не хватать, да и содержание внутреннего ЦОД становилось все дороже.

Благодаря масштабируемости WINDOWS AZURE позволяет вести учет огромного количества пользователей. Создавая облачное решение, компания-разработчик может рассчитывать не только на корпорации, но и на физических лиц. Такое приложение было сделано новозеландской компанией TicketDirect International, которая, работая в онлайн-режиме, осуществляет 45% всех продаж билетов на культурные и спортивные мероприятия Новой Зеландии. Предыдущая, традиционная, система продажи билетов, функционировавшая на базе Microsoft SQL Server 7 и SQL Server 2000, была написана на Visual Basic 6. Приложение без проблем обслуживало несколько сотен продаж в течение часа. Но в дни распродаж, когда объявлялась скидка на посещение популярного мероприятия, до системы пытались одновременно «достучаться» тысячи людей [1]. Неудивительно, что компьютерный парк продавца билетов не выдерживал такого наплыва пользователей.

WINDOWS AZURE предоставила TicketDirect масштабируемую инфраструктуру как сервис с возможностью оплаты по факту. В результате в момент распродаж приложение начинает использовать дополнительные мощности. Теперь компании TicketDirect не потребуется закупать оборудование только для того, чтобы покрыть временные всплески

активности. Ограничений практически не существует. В облаках компания способна обслужить несколько популярных мероприятий, начинающих свои распродажи в одну и ту же минуту.

На данный момент идет активная разработка и совершенствование технологии облачных вычислений [2]. Но речь идет именно о разработке, а не об использовании. Сегодня многие боятся именно самого факта, что информация будет храниться сторонние люди. И хотя почти невозможность утери либо кражи данных уже доказана, немногие готовы довериться подобным сервисам. Так же сказывается недостаточное на данный период времени качество, стабильность и скорость интернет-соединений, что создает ощутимые трудности для разработчиков.

При использовании облачных вычислений, потребители информационных технологий могут существенно снизить капитальные расходы – на построение центров обработки данных, закупку серверного и сетевого оборудования, аппаратных и программных решений по обеспечению непрерывности и работоспособности – так как эти расходы поглощаются провайдером облачных услуг. Кроме того, длительное время построения и ввода в эксплуатацию крупных объектов инфраструктуры информационных технологий, и высокая их начальная стоимость ограничивают способность потребителей гибко реагировать на требования рынка, тогда как облачные технологии обеспечивают возможность практически мгновенно реагировать на увеличение спроса на вычислительные мощности.

Однако, несмотря на эти существенные недостатки, плюсы от внедрения данной технологии ясны всем. Ведь это экономия для потребителей, борьба с пиратством для разработчиков, минимизация затрат в IT сфере для бизнеса, унификация сетевых стандартов для всех пользователей.

Список литературы

1. Черняк Л. Интеграция – основа облака // Открытые системы. СУБД. 16 сентября 2011 г.
2. Жулев С.А., Ведюшкина А.Е., Артюшина Е.А. Проблемы использования облачных технологий при изучении СУБД MS SQL SERVER // Международный студенческий научный вестник. – 2015. – № 3-2. – С. 271.

ОПТИМИЗАЦИЯ ФОРМЫ КОНСТРУКЦИЙ ПРИ ЗАДАННОМ ТЕПЛОМ ВОЗДЕЙСТВИИ

Ладыгин Е.А., Курносое В.Е.

Пензенский государственный технологический университет, Пенза, e-mail: alexey314@ya.ru

Задача выбора оптимальной конфигурации конструкции актуальна ввиду необходимости снижения материалоемкости изделий, сокращения затрат

на проектирование. Ее решение может быть основано на исследовании различных вариантов конструктивного исполнения изделия, их сопоставлении и выборе лучшего варианта исполнения в соответствии с критерием оптимизации.

Процессы деформирования, теплопередачи, диффузии и другие определяют конфигурацию многих изделий. Эти процессы описываются уравнениями в частных производных. В работах [1,2] предложено использовать эволюционные дискретные логико-математические модели, позволяющие автоматическим формировать системы разрешающих уравнений.

Область проектирования разбивается на элементы объема, каждый из которых может быть либо «пустым», либо «заполненным материалом».

Рассмотрим решение задачи автоматического определения конфигурации теплонагруженной конструкции при конечно-разностной аппроксимации уравнения теплопроводности Фурье:

$$\frac{\partial \theta}{\partial \tau} = a \left(\frac{\partial^2 \theta}{\partial x^2} + \frac{\partial^2 \theta}{\partial y^2} + \frac{\partial^2 \theta}{\partial z^2} \right). \quad (1)$$

Здесь: $\theta = \theta(x, y, z, \tau)$ температура, функция координат и времени; a – коэффициент температуропроводности материала. Построение уравнения баланса целесообразно выполнить на основе интегро-интерполяционного метода [3].

Для построения логико-математического уравнения баланса как внутреннего, так и граничного элемента объема и описания распределения материала по элементам целесообразно использовать импликативную алгебру выбора (ИАВ) Л.И. Волгина [4].

В частном случае логико-алгебраические модели на основе ИАВ могут включать бинарные операции конъюнкции ($\wedge$) или дизъюнкции ($\vee$), которые фактически являются операциями отождествления или выбора переменных:

$$z_1 = \wedge_1(y_1, y_2) = y_1 I(\alpha_2 - \alpha_1) + y_2 I(\alpha_1 - \alpha_2); \quad (2)$$

$$z_2 = \vee_1(y_1, y_2) = y_1 I(\alpha_1 - \alpha_2) + y_2 I(\alpha_2 - \alpha_1). \quad (3)$$

В выражениях (2,3) $I(x)$ есть единичная функция (оператор Хевисайда), y_1 и y_2 – предметные переменные, α_1 и α_2 – переменные импликативной алгебры выбора, действительные числа.

Логико-математическое уравнение теплового баланса для произвольного элемента объема области проектирования, «заполненного материалом», может быть записано в виде

$$\alpha^{000} \left(\frac{\alpha^{+00} J^{+X} - \alpha^{-00} J^{-X}}{h_x^0} + \frac{\alpha^{0+0} J^{+Y} - \alpha^{0-0} J^{-Y}}{h_y^0} + \frac{\alpha^{00+} J^{+Z} - \alpha^{00-} J^{-Z}}{h_z^0} + \frac{\alpha_T^{000} Q_3^{000}}{h_x^0 h_y^0 h_z^0 \tau} + \right. \\ \left. + \frac{\bar{\alpha}^{+00} J_s^{+X} - \bar{\alpha}^{-00} J_s^{-X}}{h_x^0} + \frac{\bar{\alpha}^{0+0} J_s^{+Y} - \bar{\alpha}^{0-0} J_s^{-Y}}{h_y^0} + \frac{\bar{\alpha}^{00+} J_s^{+Z} - \bar{\alpha}^{00-} J_s^{-Z}}{h_z^0} \right) = \\ = \frac{\alpha^{000} \alpha_d^{000} C_3^{000} (\theta_{t+\tau}^{000} - \theta_t^{000})}{h_x^0 h_y^0 h_z^0 \tau}. \quad (3)$$

Здесь предметные переменные: C_3^{000} – теплоемкость элемента; Q_3^{000} , $\Delta \theta_3^{000}$ – собственное тепловыделение и изменение температуры элемента за время наблюдения τ ; $+X, \dots, -Z$ – обозначение граней элемента. Переменные ИАВ: α^{+00} , α^{-00} , ..., α^{00-} – переменные выбора потоков по граням элемента объема; $\bar{\alpha}^{+00}$, $\bar{\alpha}^{-00}$, ..., $\bar{\alpha}^{00-}$ – отрицания их значений; α_T^{000} , α_d^{000} – переменные наличия собственного тепловыделения элемента и выбора стационарной или нестационарной задачи.