
УДК 004.896(06)

Е.С.САКАЛО

ИССЛЕДОВАНИЕ И ПОСТРОЕНИЕ ОБЛАЧНОЙ МОДЕЛИ УСЛУГ В CLOUD-СРЕДЕ

Предлагается публичная модель облака для поисковых систем с использованием распределённых вычислений для больших наборов данных.

1. Введение

Облачные вычисления представляют собой динамически масштабируемый способ доступа к внешним вычислительным ресурсам в виде сервиса, предоставляемого посредством Интернета, при этом пользователю не требуется никаких особых знаний об инфраструктуре “облака” или навыков управления этой “облачной” технологией.

Под “облаком”, подразумевается не сам Интернет, а весь тот набор аппаратного и программного обеспечения, который обеспечивает обработку и исполнение клиентских заявок “с той стороны”. Если не мудрить с определениями и вглядеться в суть, даже самое простое действие посредством Интернета – запрос страницы сайта в какой-то степени представляет собой пример облачного вычисления. Интернет, по сути, и возник именно для того, чтобы предоставлять удалённый доступ к различным вычислительным ресурсам.

Таким образом, распространившееся в последнее время мнение об облачных вычислениях как о чём-то специфическом, имеющем отношение к научным исследованиям или мощным корпоративным системам как минимум необъективно, а с точки зрения клиента – конечного пользователя, и вовсе не соответствует действительности.

Как результат, концепция Cloud Computing – облачных вычислений – позволяет организациям повысить эффективность предоставления услуг, упростить управление и улучшить согласование ИТ-сервисов с меняющимися потребностями бизнеса.

В настоящее время применяются как публичные (public), так и частные (private) модели Cloud-среды.

Публичными моделями Cloud-решений может пользоваться любой человек с доступом в Интернет. Аспект аутсорсинга затрудняет поддержание целостности и конфиденциальности данных, обеспечение доступности данных и готовности сервиса, а также демонстрацию соответствия нормативным требованиям.

Фактически концепция Cloud Computing передает значительную часть контроля над данными и операциями от клиентской организации к ее поставщику Cloud-сервисов подобно тому, как организация доверяет часть своих ИТ-операций аутсорсинговой компании. Даже такие базовые задачи, как установка пакетов с исправлениями и конфигурирование межсетевых экранов, могут стать обязанностью поставщика Cloud-сервисов, а не конечного пользователя.

При применении модели доступа к ИТ-ресурсам Cloud Computing (облачные вычисления) информационные сервисы предоставляются таким образом, что обеспечивающие технологии становятся практически невидимыми за пределами пользователя устройства. А поскольку это позволяет отделить приложения и сервисы от обеспечивающей инфраструктуры и тем самым позволяет бизнесу быстрее адаптироваться к изменениям, облачные вычисления могут быть частью стратегии по повышению динамичности предприятия.

При разработке модели поисковой системы в Cloud-среде за основу была взята поисковая система Amazon CloudSearch. Amazon CloudSearch представляет собой полностью управляемый поисковый сервис в облаке, позволяет заказчикам легко и быстро интегрировать масштабируемые функции поиска в своих приложениях.

Amazon CloudSearch легко масштабируется, когда объем поиска данных увеличивается, а разработчики могут изменить параметры поиска и применить новые настройки в любое время без необходимости загружать данные еще раз. Эта технология направлена на упрощение процесса для веб-разработчиков, желающих интегрировать поиск в своих размещенных сайтах.

Поиск домена для каждого набора данных, которые необходимо сделать доступными, имеет один или несколько случаев, каждый из которых обладает конечным объемом оперативной памяти и ресурсов процессора для индексирования данных и обработки запросов. Количество поисковых случаев домена зависит от количества документов, а также объема и сложности запросов. В управляемых услугах поисковой системы в облачной среде система определяет размер и количество поисковых случаев, необходимых для достижения низкой задержки, высокой производительности поиска пропускной способности. Когда пользователь загружает данные и настройки индексов, система поиска строит индекс и выбирает соответствующий начальный экземпляр типа поиска, чтобы индекс клиента мог храниться в оперативной памяти.

Поисковая система будет масштабироваться до поиска области (или раздела индекса по нескольким экземплярам, если клиент уже в самом большом). И наоборот, когда объем данных сокращается, система масштабируется до поиска меньших случаев или меньшего типа поиска, например, если индекс помещается на одном разделе.

Задачи для достижения цели: 1) Детально проанализировать работу нескольких поисковых сред [1, 5]. 2) Построить архитектуру программы, основанную на облачных вычислениях [1, 6, 7].

В качестве объекта разработки выступает поисковая система в Cloud-среде. С учетом всех недостатков существующих поисковых систем возникает острая задача поиска алгоритма для построения поисковой системы на основе облачных вычислений.

2. Внедрение облачных вычислений при проектировании настройки облака

Для решения задачи внедрения облачных вычислений при проектировании настройки облака в работе использовалась технология Map Reduce. В рамках этой технологии Google была предложена абстрактная модель, которая позволила выражать простые вычисления, в то же время пряча сложные детали параллелизации: обработку ошибок, распределение данных, балансировку нагрузки.

Первая реализация этой модели была выполнена на основе распределенной файловой системы той же компании GFS (Google File System). Эта реализация активно используется в программных продуктах самой Google, но является сугубо проприетарной и недоступна для использования вне Google.

Альтернативная, свободно доступная реализация Hadoop MapReduce (с открытыми исходными текстами) была выполнена в проекте Hadoop сообщества Apache. Архитектура облачной модели основана на доминировании веб-приложений над обычными функциями ОС. Ключевая роль при этом отводится браузеру. Стратегия создания нового продукта подразумевает архитектуру, нетребовательную к аппаратным ресурсам персонального компьютера, используемого для выхода в сеть Интернет. Эта тенденция переноса центра тяжести с пользователя ПК на Интернет-ресурсы и соответствует идеологии облачных вычислений. В этой модели вычисления производятся над множествами входных пар “ключ-значение”, и в результате каждого вычисления также производится некоторое множество результирующих пар “ключ-значение”. Для представления вычислений в среде MapReduce используются две основные функции: Map и Reduce. Обе функции явно кодируются разработчиками приложений в среде MapReduce.

Функция Map в цикле обрабатывает каждую пару из множества входных пар и производит множество промежуточных пар “ключ-значение”. Среда MapReduce группирует все промежуточные значения с одним и тем же ключом I и передает их функции Reduce.

Функция Reduce получает значение ключа I и множество значений, связанных с этим ключом. В типичных ситуациях каждая группа обрабатывается (в цикле) таким образом, что в результате одного вызова функции образуется не более одного результирующего значения.

Реализации MapReduce от Google и Hadoop ориентированы на использование в кластерной распределенной среде со следующими основными характеристиками:

- узлы среды выполнения MR-приложений обычно представляют собой компьютеры общего назначения с операционной системой Linux;
- используется стандартное сетевое оборудование с адаптерами, рассчитанными на скорости передачи в 100 мегабит в секунду или 1 гигабит в секунду, но средняя пропускная способность существенно ниже;
- кластер состоит из сотен или тысяч машин, так что вполне вероятны отказы отдельных узлов;
- для хранения данных используются недорогие дисковые устройства, подключенные напрямую к отдельным машинам;
- для управления данными, хранящимися на этих дисках, применяется распределенная файловая система;
- пользователи представляют свои задания в систему планирования; каждое задание состоит из некоторого набора задач, которые отображаются планировщиком на набор узлов кластера. Рассмотрим реализацию модели MapReduce в распределенной среде.

Вызовы Map распределяются по нескольким узлам кластера путем разделения входных данных на M непересекающихся групп (split). Входные группы могут параллельно обрабатываться на разных машинах. Вызовы Reduce распределяются путем разделения пространства промежуточных ключей на R частей с использованием некоторой функции разделения (например, функции хэширования). Число разделов R и функция разделения задаются пользователем.

Выполнение MR-программы происходит следующим образом. Сначала среда MapReduce расщепляет входной файл на M частей, размер которых может задаваться пользователем. Затем сразу в нескольких узлах кластера запускается основная программа MapReduce. Один из ее экземпляров играет специальную роль и называется распорядителем (master). Остальные экземпляры являются исполнителями (worker), которым распорядитель назначает работу. Распорядитель должен назначить исполнителям для выполнения M задач Map и R задач Reduce. В библиотеке MapReduce поддерживается возможность чтения входных данных в нескольких разных форматах. Например, в режиме “text” каждая строка трактуется как пара “ключ-значение”, где ключ – это смещение до данной строки от начала файла, а значение – содержимое строки. В другом распространенном формате входные данные представляются в виде пар “ключ-значение”, отсортированных по значениям ключа. В каждой реализации формата входных данных известно, каким образом следует расщеплять данные на осмысленные части, которые обрабатываются отдельными задачами Map (например, данные формата “text” расщепляются только по границам строк).

Пользователи могут добавить к реализации собственные форматы входных данных, обеспечив новую реализацию интерфейса reader (в реализации Hadoop – RecordReader). Reader не обязательно должен читать данные из файла, можно легко определить reader, читающий данные из базы данных или из некоторой структуры в виртуальной памяти.

Аналогичным образом поддерживаются возможности генерации данных в разных форматах, и имеется простая возможность определения новых форматов результирующих данных.

Основными особенностями системы есть: скорость загрузки, выхода в Интернет, получения электронной почты, тесная интеграция с интернет-сервисами, надежность работы, обеспечение безопасности в автоматическом режиме, автоматическое обновление, простота. Как и объем данных, система автоматически масштабирует поиск домена в соответствии с требованиями.

Ключевым элементом платформы есть средства автоматического обнаружения, способные выделить компоненты сервиса и их изменения, включая виртуальные и физические серверы, сетевые соединения и приложения.

Каталог сервисов предоставляет меню облачных сервисов и должен сопровождаться реализацией специального пользовательского портала самообслуживания.

Вычислительные облака состоят из тысяч серверов, размещенных в датацентрах, обеспечивающих работу десятков тысяч приложений, которые одновременно применяют миллионы пользователей. Непременным условием эффективного управления такой крупномасштабной инфраструктурой является максимально полная автоматизация. Кроме того, для обеспечения различных видов пользователей - облачных операторов, сервис-провайдеров, посредников, ИТ-администраторов, пользователей приложений - защищенного доступа к вычислительным ресурсам облачная инфраструктура должна предусматривать возможность самоуправления и делегирования полномочий.

Концепция облачных вычислений значительно изменила традиционный подход к доставке, управлению и интеграции приложений. По сравнению с традиционным подходом, облачные вычисления позволяют управлять большими инфраструктурами, обслуживать различные группы пользователей в пределах одного облака.

3. Заключение

Предложена облачная модель услуг. Это гибкая, экономически эффективная и проверенная технология для доставки ИТ-сервисов организациям или индивидуальным потребителям с использованием сети Интернет. Ресурсы Cloud-среды быстро разворачиваются и легко масштабируются, при этом инициализация всех процессов, приложений и сервисов осуществляется по требованию, независимо от местоположения потребителя или используемого компьютерного устройства. Модель с применением облачных вычислений является динамично масштабируемой для доступа к внешним вычислительным ресурсам в виде сервиса, предоставляемого посредством Интернета, при этом пользователю не требуется никаких особых знаний об инфраструктуре «облака» или навыков управления этой «облачной» технологией.

Практическая значимость. Данная модель – это единая архитектура, которая объединяет возможности по использованию «частных», «гибридных» и «публичных» облаков, а также позволяет совмещать их с традиционной ИТ-средой. Благодаря применению стандартных модульных компонентов, а также использованию разработанного метода позволяет оперативно находить необходимую информацию.

Список литературы: 1. *A distributed storage system for structured data* / Chang, J. Dean. Seattle, USA, 2006. С. 205-218. 2. Dean, J. and Ghemawat, S. MapReduce: Simplified data processing on large clusters. San Francisco, USA, 2004. С. 137-150. 3. Barroso, L. A., Dean, J., and Urs Hylzle, U. Web search for a planet: The Google cluster architecture. Seattle, USA, 2003. С. 22-28. 4. *Hadoop Map/Reduce Tutorial* [Электронный ресурс]. Электрон. дан. Режим доступа: http://hadoop.apache.org/common/docs/current/mapred_tutorial.html, свободный. 5. CloudEra [Электронный ресурс] Электрон. дан. Режим доступа: <http://www.cloudera.com/>, свободный. 6. How to Create and Debug an Amazon Elastic MapReduce Job Flow, [www/ URL: http://aws.amazon.com/articles/3938](http://aws.amazon.com/articles/3938) 7. Екатерина Баранова, «Концепция Cloud computing» www.itcontent.ru/archives/blog/cloud_computing.

Поступила в редколлегию 18.11.2013

Сакало Евгений Сергеевич, канд. техн. наук, доцент кафедры информатики ХНУРЭ. Научные интересы: облачные технологии, проектирование облаков различной сложности. Адрес: Украина, 61166, Харьков, пр. Ленина, 14, тел.0577021419. Email: yevgen.sakalo@gmail.com.