

УДК. 681.3.016



ИСПОЛЬЗОВАНИЕ GRID-КОНЦЕПЦИИ ДЛЯ ПРОЕКТИРОВАНИЯ РАСПРЕДЕЛЕННЫХ БАЗ ДАННЫХ В ИНФОРМАЦИОННО-ОБУЧАЮЩЕЙ СРЕДЕ

О.С. Богуш

ХНУРЭ, г. Харьков, Украина, olessya.bogush@gmail.com

В данной статье рассмотрены принципы построения распределенных баз данных в информационно-обучающей среде на основе совмещения модели клиент-сервер и технологии Grid. Предложены концепция распределенной базы данных в информационно-обучающей среде, выбрана архитектура программного обеспечения и метод распределения информации по вторичным серверам распределенной базы данных.

GRID-ТЕХНОЛОГИИ, ИНФОРМАЦИОННО-ОБУЧАЮЩАЯ СРЕДА, МОДЕЛЬ «КЛИЕНТ-СЕРВЕР», РАСПРЕДЕЛЕННАЯ БАЗА ДАННЫХ

Введение

При синтезе информационно-обучающей среды (ИОС) рано или поздно встает вопрос о способе хранения большого количества оцифрованных видеоданных, используемых в качестве учебных фильмов, видеолекций и т. д. Кроме того, возникает еще одна задача, которую невозможно решить одновременным наращиванием объема дискового пространства сервера. Речь идет об ограниченности выходного канала, через который пользователи ИОС получают видеоданные. Стандартной в настоящий момент является пропускная способность 100 Мбит/с. Если исходить из необходимости транслировать качественные видеофильмы в стандарте DVD (скорость передачи 22 Мбит/с) или MPEG (1 Мбит/с), то в первом случае уже пять пользователей, одновременно просматривающих видеофильмы, исчерпают лимит пропускной способности выходного канала сервера [1]. В этом случае исключается возможность доступа даже к текстовым материалам меньшего объема, но хранящимся на сервере. Во втором случае количество параллельных выходных видеопотоков на порядок больше, но резко падает показатель доступности сервера.

Обе вышеописанные проблемы — недостаточность дискового пространства и низкая пропускная способность выходного канала сервера — можно решить, если создать распределенную базу данных. При наличии специального программного обеспечения можно организовать множество параллельных синхронизированных по времени видеопотоков от различных источников к одному получателю. При этом проблема ограниченной пропускной способности выходного канала сервера не существенна, поскольку параллельные видеопотоки будут передаваться по разным линиям связи. При таком способе организации хранения данных необходимость наличия центрального сервера по-прежнему остается, но нагрузка на него значительно сократится, следовательно, вероятность доступности для новых запросов останется на высоком уровне.

Сегодня в самых разных организациях сосредоточено много вычислительных средств, которые реализуют различные задачи, однако, не используют всей своей мощности. Эти терминалы размещены в различных местах и выполняют разные приложения, используют средства хранения информации и различные системы доступа. С другой стороны существует ряд задач глобального масштаба, для решения которых применяются большие вычислительные мощности, которые необходимо распараллеливать. Эти предпосылки привели к появлению Grid-технологий. Grid-технологии рассматриваются в контексте консолидации не только вычислительных средств, но и всей совокупности вычислительных ресурсов. В распределенных приложениях необходимо решать проблему минимизации объема передаваемых данных, поэтому применение Grid-технологий охватывает не только компьютерные мощности, но и ресурсы средств хранения информации [2, 3].

1. Постановка задачи

Основная задача, решаемая в Grid — обеспечение доступа к ресурсам, а поскольку ресурсы распределенные, то функционирование обеспечивается специальной формой программного обеспечения — службами. В отличие от модели «клиент-сервер» набор служб устанавливается на каждом информационном ресурсе. Множество служб должно удовлетворять следующим ограничениям:

— каждый тип служб должен иметь стандартный протокол доступа, в соответствии с которым реализуется прикладной интерфейс клиентов. В рамках стандартных протоколов допустимы различные способы реализации служб;

— множества служб на различных ресурсах должны быть согласованными. Это предполагает унификацию наборов служб на основе тождественности их семантики, а также наличие общих правил, регламентов и организационных соглашений, на которые опирается конфигурирование служб.

Успех Grid-проекта должны обеспечивать следующие службы и соответствующие протоколы:

- протокол доступа к управлению ресурсами, который обеспечивает безопасное создание удаленных процессов и управление ими;
- служба метакаталогов, отвечающая за распределенный сбор данных и информационное обслуживание;
- служба инфраструктуры безопасности, поддерживающая однократную регистрацию, делегирование полномочий и отображение прав доступа на различные локальные системы.

Работы в области Grid направлены сегодня на существенное расширение состава служб, однако, для этой задачи необходима поддержка архитектуры служб.

2. Архитектура программного обеспечения

Сегодня ведущие мировые компании, такие как IBM, Sun Microsystems, HP, Oracle активно развивают решения для Grid. Они предлагают близкие по смыслу стратегии развития, направленные на значительную реструктуризацию корпоративной информационной инфраструктуры.

В Hewlett Packard была разработана специальная концепция UDC (Utility Data Center), обеспечивающая минимальную нагрузку на средства передачи данных, решающая проблему защиты и распараллеливания БД и т.д. Базовым программным обеспечением Grid и международным стандартом де-факто является Globus.

3. Концепция распределенной базы данных в информационно-обучающей среде

В предлагаемом подходе к реализации распределенной базы данных совместно используется модель «клиент-сервер» и концепция Grid.

Реализация концепции Grid предполагает, что каждый из компьютеров, входящих в локальную сеть, одновременно является и клиентом — компьютером, запрашивающим ресурсы, и сервером, предоставляющим ресурсы. Такие серверы можно назвать вторичными, чтобы не путать с центральным сервером. Клиентская часть доступна владельцу персонального компьютера (ПК), а серверная часть доступна только для РБД.

Таким образом, реализация концепции Grid предполагает наличие трех сущностей:

- 1) центральный сервер;
- 2) вторичный сервер;
- 3) клиент.

В функции центрального сервера входят:

- хранение информации о месторасположении видеофрагментов на вторичных серверах;
- предоставление по требованию клиента сведений о месторасположении фрагментов одного видеофайла.

Функциями вторичного сервера являются:

- хранение видеофрагментов;
- организация выходного видеопотока по запросу клиента и передача требуемого видеофрагмента;
- организация входного видеопотока по запросу клиента и сохранение видеофрагмента.

Клиент здесь взаимодействует с серверами с целью сохранения или получения видеофайлов.

Реализованную в РБД модель «клиент-сервер» можно описать следующим образом: клиентом является компьютер, запрашивающий видеофайл, а серверами — центральный сервер, содержащий информацию о местоположении фрагментов требуемого видеофайла, и вторичные серверы, содержащие сами фрагменты.

Протокол взаимодействия клиента и серверов в простейшем варианте можно описать следующим образом: клиент запрашивает у центрального сервера данные о месторасположении фрагментов того или иного видео. После получения необходимой информации от центрального сервера клиент организует соединения с теми вторичными серверами, где хранятся фрагменты видео, после чего параллельно с получением видеоданных проигрывает видео.

4. Распределение видеофрагментов по вторичным серверам

Использование РБД может позволить эффективно создавать и впоследствии управлять распределенной базой данных видеофрагментов. Вся информация о видеофрагментах, а также сами видеофрагменты будут храниться в логически единой, но физически распределенной базе данных. Основная информация, необходимая для отклика на запрос видеофайла, должна находиться на центральном сервере, а сами видеофрагменты будут находиться на вторичных серверах.

Центральный сервер должен хранить следующие данные для распределенной записи/чтения фрагментов:

- нумерацию сетевых устройств (коммутаторов, маршрутизаторов);
- нумерацию вторичных серверов;
- нумерацию магистральных сегментов в каждой зоне, разделяемых сетевыми устройствами (коммутаторами, маршрутизаторами);
- соответствие между номером сетевого устройства и номером вторичного сервера;
- нумерацию зон сети (ветвей сети от центра — считая центром центральный сервер);
- количество магистральных сегментов на одну копию фрагмента видеозаписи (зависит от коэффициента доступности (дублирования));
- идентификатор фрагмента;

- идентификатор файла;
- номер фрагмента в наборе фрагментов одной видеозаписи;
- описание фрагмента;
- размер фрагмента в байтах;
- продолжительность фрагмента в секундах;
- таймаут обращения к вторичному серверу (зависит от размера фрагмента в секундах), по истечении которого клиент обращается за чтением фрагмента к центральному серверу.

Прежде чем видеофайлы смогут быть доступны для просмотра, они должны быть разбиты на фрагменты и распределены по вторичным серверам в соответствии с установленными требованиями.

Требования к разбиению и распределению видеофайлов на фрагменты:

1. Размер фрагмента видеофайла должен быть определен путем создания экспериментальных видеосетей, однако, в качестве исходного значения может быть выбрана цифра, пропорциональная значениям, используемым в отработанной технологии, реализованной в системах IP-телефонии (3 КБайт).

2. Фрагменты видеофайла должны быть распределены по вторичным серверам таким образом, чтобы фрагменты с последовательными номерами не находились на одном и том же вторичном сервере. Удовлетворение данного требования позволит временно освобождать вторичный сервер для отклика на другие запросы.

3. Фрагменты видеофайла со стартовыми номерами должны находиться на наиболее доступных для большинства пользователей вторичных серверах, для того чтобы пользователь, запросивший тот или иной файл, смог максимально быстро начать его просмотр. Остальные фрагменты клиент будет получать параллельно с просмотром файла.

4. Желательно, чтобы каждый фрагмент видеофайла имел такое количество копий на вторичных серверах, чтобы в любой момент времени необходимая копия фрагмента была доступна для получения хотя бы с одного из вторичных серверов. Кроме того, копии всех фрагментов должны храниться на центральном сервере на тот случай, если при необходимости получения фрагмента он не будет доступен ни на одном из вторичных серверов.

Для разбиения фрагментов с последующим их распределением по вторичным серверам необходимо создать метод, учитывающий перечисленные выше требования. Кроме того, при построении метода распределения видеофрагментов по вторичным серверам необходимо учитывать еще множество вариантов. Например должна ли система при получении видеофайла на хранение сразу же разбить его на фрагменты или разбивать на фрагменты

нужно только при первом запросе данного файла от клиента? Нужно ли оставлять копии фрагментов на центральном сервере или хранить их нужно только на вторичных серверах? Нужно ли удалять редко используемые фрагменты с вторичных серверов для освобождения там места для более часто запрашиваемой информации?

Таким образом, метод распределения видеофрагментов должен определять следующее:

- количество фрагментов, на которые необходимо разбить файл (зависит от количества вторичных серверов в локальной сети, от размера видеофайла, от размера метаданных, от типа и конфигурации локальной сети);
- количество копий одного и того же фрагмента и их месторасположение;
- необходимость хранения дубликата каждого фрагмента на центральном сервере.

Все эти варианты необходимо рассмотреть с точки зрения их целесообразности в той или иной ситуации, при наличии тех или иных характеристик используемой распределенной базы данных.

Для нормальной работы коммутатора требуется отсутствие замкнутых маршрутов в сети. Эти маршруты могут создаваться администратором специально для образования резервных связей или же возникать случайным образом, что вполне возможно, если сеть имеет многочисленные связи, и кабельная система плохо структурирована или документирована [4].

Получившееся разбиение вторичных серверов по зонам учитывается затем при распределении стартовых видеофрагментов. Стартовые видеофрагменты размещаются в наиболее доступных зонах, для того чтобы пользователи могли максимально быстро начать просмотр запрошенного видеофайла.

В процессе функционирования системы вторичные серверы должны обмениваться служебными пакетами, для того чтобы удостовериться в доступности друг друга. Если возникает ситуация, когда ранее доступный сервер становится недоступным, необходимо вновь запускать процедуру определения зон. Данная процедура может быть инициирована со вторичных серверов, а также центральным сервером.

Выводы

В области дистанционного обучения уже сегодня Grid-технологии заслуживают пристального внимания. Разработанные на данный момент принципы в рамках Grid-технологий позволяют решать задачи, связанные с распределенным хранением и оперативной доставкой мультимедийных данных большого объема пользователям информационно-обучающей среды. Для пользователей инфор-

мационно-обучающей среды всегда важно, чтобы центральный сервер был максимально доступен для новых запросов, которые не всегда связаны с необходимостью просмотра видеороликов или прослушивания звукозаписей. Поскольку передача всего лишь нескольких видео- и аудио-поток может полностью занять выходной канал центрального сервера, его необходимо максимально разгрузить, а этого можно добиться, распределив мультимедийные данные на вторичных серверах. В этом случае выходной канал центрального сервера всегда будет открыт для новых запросов, а параллельная передача нескольких видеопотоков от многих источников к одному получателю значительно увеличит скорость получения запрошенных пользователем видеоданных.

При распределении видеофрагментов по серверам стартовые фрагменты каждого видеофайла имеет смысл размещать на наиболее доступных вторичных серверах, для того чтобы максимально сократить время ожидания начала проигрывания файла на клиенте.

Рассмотренные в статье принципы распределения данных позволяют значительно сократить нагрузку на центральный сервер, распределив не только данные, но и потоки передачи данных. Однако для снижения нагрузки на центральный сервер не

обязательно организовывать параллельную передачу видеопотоков от нескольких источников к одному получателю, что связано с необходимостью создания сложного программного обеспечения. Достаточно оформить видеофрагменты одного видеофайла, распределенные по вторичным серверам, в формате потокового видео и передать пользователю файл, содержащий ссылки на эти фрагменты. В этом случае центральный сервер также не будет занят передачей видеопотоков, а задержки при получении видеофайла пользователем будут незначительны, поскольку потоковое видео начинает проигрываться сразу же при получении первых кадров. Перспективными остаются задачи проектирования информационной структуры центрального и периферийных серверов, а также организации информационного обеспечения системы распределенных ресурсов.

Список литературы: 1. В.Л. Коваленко, Д. М. Корягин. Эволюция и проблемы Grid // Открытые системы. — 2003. — № 1. — С. 23-26. 2. Н.И. Дуброва. Учет и контроль для «коммуникационных вычислений» // Новые технологии. — 2004. — № 1. — С. 74-78. 3. П. Анни. Этот Grid — непрост...// Открытые системы. — 2005. — № 4. — С. 45-56. 4. Копьютерные сети. Принципы, технологии, протоколы / В. Г. Олифер, Н. А. Олифер. — СПб.: Питер, 2002. — 346 с.

Поступила в редколлегию 29.10.2007