

УПРАВЛЕНИЕ ПОТОКАМИ СООБЩЕНИЙ В РАСПРЕДЕЛЁННЫХ ПРИЛОЖЕНИЯХ

Работа с сообщениями подразумевает взаимодействие между компонентами системы посредством передачи сообщений. Сервис передачи сообщений Java (JMS, Java Messaging Service) позволяет реализовать это взаимодействие в распределённом Java приложении, а хранение сообщений в очередях позволяет асинхронно обрабатывать получаемые сообщения на сервере приложений без дополнительных усилий.

Написание распределённых приложений связано с преодолением ряда сложностей не приходящих обычным клиент-серверным приложениям. Среди таковых: обеспечение стабильной работы при изменяющейся вычислительной мощности, балансировка нагрузок на вычислительные единицы, управление потоками данных и прочие. Для преодоления вышеозначенных проблем широкое распространение получило промежуточное ПО, ориентированное на пересылку сообщений между компонентами системы. Для платформы Java существует стандарт на прикладной интерфейс подобного ПО – JMS, Java Messaging Service.

Обычно, при написании приложений с использованием JMS используется топология типа "звезда с центральным сервером. Клиенты соединяются с брокером и взаимодействуют друг с другом в слабосвязанной манере через очереди и топки сообщений

Такой подход хорошо подходит для распределения нагрузки между небольшим числом заранее известных вычислительных единиц, но не подходит для динамических систем с большим количеством часто приходящих и уходящих устройств. Также, единственный JMS сервер является уязвимым местом данной системы.

При данных условиях лучше использовать кластеризованную систему брокеров сообщений, настроив её на авто добавление новых узлов. Данная конфигурация способна динамически менять свою топологию при добавлении или удалении новых вычислительных устройств. Когда ресурсы кластера становятся недостаточными, с лёгкостью добавляется новый брокер, который автоматически начинает принимать сообщения клиентов.

Для автоматического добавления новых узлов используется адаптивный алгоритм, учитывающий множество параметров: загрузку существующих узлов, количество ещё необработанных сообщений, длительность обработки сообщений каждым узлом.

Средство мониторинга состояния сети представляет собой программный супервизор, напи-

санный также на языке Java, и набора клиентских утилит на специфичных для целевой операционной системы языках (командные файлы Windows или скрипты командной оболочки для Linux). Опрос узлов может производиться по нескольким протоколам: HTTP, JMX или непосредственно по JMS. В случае нехватки передаточной способности сети брокеров, или её излишней мощности производится посылка управляющих команд на запуск новых брокеров или выключение старых.

Построение описанного выше распределённого приложения с использованием JMS и супервизора брокеров включает следующие этапы:

Этап 1. Строится оценочная топология сети вычислительных устройств.

Этап 2. Настраивается кластер брокеров сообщений, учитывающий специфику построенной сети. Устанавливается конфигурация балансировки поступающих сообщений между узлами кластера.

Этап 3. В процессе работы сети, клиенты производят подписку на сообщения брокеров.

Этап 3.1. Супервизор отслеживает нагрузку узлов и при необходимости посылает команды управления на клиентские узлы.

Этап 3.2. На клиентских узлах, потенциально участвующих в обмене сообщениями запущены сервисы, отслеживающие команды супервизора. При необходимости производится запуск или остановка брокеров, а также их включение или отсоединение от общей сети.

Этап 4. При значительном изменении топологии сети может производиться дополнительная калибровка кластера брокеров с перезапуском всей сети.

Список литературы

1. Снайдер Б. ActiveMQ in Action. / Manning, 2011.- 408 с.
2. Кэноби О. Advanced IBM MQ Techniques. / IBM Publishing, 2008. - 361 с.
3. Хопп Г. Enterprise Integration Patterns. / Addison-Wesley, 2011. - 700 с.

Крюков Станислав Юрьевич, магистрант факультета информационных технологий и управления Белорусского государственного университета информатики и радиоэлектроники, kafitas@bsuir.by

Научный руководитель: Ревотюк Михаил Павлович, доцент кафедры информационных технологий автоматизированных систем БГУИР, к.т.н., доцент, rmp@bsuir.by