РЕАЛИЗАЦИЯ СИСТЕМЫ МОНИТОРИНГА И ДИНАМИЧЕСКОЙ БАЛАНСИРОВКИ НАГРУЗКИ ДЛЯ ВЫЧИСЛИТЕЛЬНЫХ ВЕБ-СЕРВИСОВ

Афанасьев А.П., Волков С.Ю., Колпаков В.Г., Корх А.В., Курочкин И.И., Смирнов С.А., Рубцов А.О., Сухорослов О.В.
Институт проблем передачи информации РАН

Реализация вычислительных веб-сервисов (ВВС) в рамках облачной модели Software as a Service (SaaS) обладает рядом неоспоримых преимуществ, как для потребителей, так и для поставщиков данных сервисов. Однако вместе с тем существует ряд проблем, связанных с реализацией данных сервисов в облачной среде. Реализация масштабируемых ВВС требует разработки специальных технических решений по мониторингу и динамической балансировке нагрузки между узлами облачной вычислительной системы (ОВС).

Разработан экспериментальный образец системы мониторинга и динамической балансировки нагрузки (СМДБН), реализующий комплексный подход к решению проблем мониторинга и балансировки нагрузки с учетом характерных особенностей ВВС. Архитектура разработанной СМДБН включает в себя следующие программные модули:
- агент, выполняющий сбор информации о потреблении ресурсов узла ОВС;
- сервис мониторинга, обеспечивающий накопление полученной от агентов информации о потреблении ресурсов узлами ОВС и реализующий программный интерфейс для доступа к накопленным данным;
- модуль распределения заданий ВВС, реализующий динамическую балансировку нагрузки между узлами ОВС.

Агент предназначен для сбора и передачи информации о текущей загрузке узла ОВС в сервис мониторинга. Агент выполняется в фоновом режиме на каждом узле ОВС. С определенной периодичностью агент осуществляет сбор значений требуемых метрик и отправку полученных данных в систему обмена сообщениями. Агент поддерживает сбор следующих метрик: коэфициент загрузки (load average); загрузка центрального процессора; потребление оперативной памяти; потребление дискового пространства. Модуль позволяет при необходимости расширять набор собираемых метрик, в том числе для нужд других систем, например сторонней системы мониторинга общего назначения. Агенты могут использоваться для сбора информации не только для целей СМДБН, но и для других систем, в первую очередь – для сторонней системы мониторинга общего назначения, поддерживающей администрирование ОВС. Из этих соображений, вместо прямой передачи данных в сервис мониторинга, агент направляет их в систему обмена сообщениями RabbitMQ, которая выступает в качестве посредника между агентами и системами-потребителями собранной информации.

Сервис мониторинга (СМНИ) предназначен для агрегации данных о текущей загрузке со всех узлов ОВС, контроля наступления определенных типов событий и предоставления доступа к этим данным и событиям для других модулей. СМНИ получает данные от агентов мониторинга путем подписки на канал системы обмена сообщениями. Для каждой пары узел-метрика модуль хранит в памяти последнее полученное от агента значение метрики, которое принимается за “текущее” значение. Долговременное хранение полученных данных в рамках СМНИ не предусмотрено. СМНИ определяет возникновение следующих типов событий на основе анализа послупающих данных и своего внутреннего состояния: подключение узла, отключение узла, перегрузка узла. СМНИ реализует программный интерфейс (API) для получения текущей информации о загрузке узлов ОВС, а также подписки на уведомления о событиях, генерируемых СМНИ. При наступлении события СМНИ рассылает уведомления о нем всем подписанным внешним модулям. В рамках разработанной СМДБН главным потребителем данной информации является модуль распределения заданий.

Модуль распределения заданий (РЗВВС) предназначен для распределения заданий ВВС между узлами ОВС в соответствии с определенным алгоритмом динамической балансировки нагрузки. Реализация РЗВВС основана на обработке следующих типов событий, поступающих от сервиса мониторинга и внешних модулей: поступление нового задания (от фронтенда ВВС), завершение выполнения задания (от модуля управления запуском заданий), подключение нового узла (от модуля СМНИ), отключение узла (от модуля СМНИ), перегрузка узла (от модуля СМНИ). Модуль поддерживает использование различных реализаций алгоритма балансировки нагрузки. Для этого логика обработки перечисленных выше событий выделена в динамически подключаемый компонент со стандартным интерфейсом. Различные реализации данного компонента могут соответствовать различным алгоритмам балансировки нагрузки.

Работа выполнена в рамках ФЦП «Исследования и разработки по приоритетным направлениям развития научно-технологического комплекса России на 2007-2013 годы» при финансовой поддержке Минобрнауки, государственный контракт № 14.514.11.4041.