

МУЛЬТИАГЕНТНАЯ РАСПРЕДЕЛЕННАЯ СИСТЕМА МОНИТОРИНГА КОМПЛЕКСНОЙ СЕТЕВОЙ ИНФРАСТРУКТУРЫ

Костюков В. В - студент., Крючкова Е. Н. - к.ф-м.н., профессор,
Алтайский государственный технический университет (г. Барнаул)

На текущем этапе информатизации общества все более актуальной становится проблема обеспечения должного уровня качества протекания информационных процессов. В результате чего возникает потребность в определенном классе программных систем — систем мониторинга, предназначением которых, в общем случае, является систематическое наблюдение и анализ протекающих в информационной среде процессов. Полученные в результате мониторинга данные могут быть использованы как для улучшения процесса принятия решений, так и для выявления узких мест исследуемой системы.

Полезность и практическая ценность системы мониторинга определяется ее способностью адаптироваться к условиям динамически изменяющихся требований, среди которых, декларируются требования к функционалу системы, отказоустойчивости и масштабируемости по отношению к ее размеру [1]. Очевидно, что для комплексного выполнения этих требований, а также для возможности адаптации к их изменениям, необходимо появление нового класса инструментов мониторинга, лишенных недостатков классических клиент-серверных систем.

Авторами предлагается архитектура распределенной системы мониторинга, которая позволяет обеспечить выполнение перечисленных требований. Основопологающая идея предлагаемого подхода заключается в использовании механизма разработки и исполнения дополнительных модулей в процессе решения задач мониторинга, а также свойств распределенных систем в процессе эксплуатации.

Базовая теоретическая модель описывается с помощью понятий сервера и агента мониторинга. **Агент**, запущенный на определенном узле, представляется активной сущностью, непрерывно наблюдающей за его состоянием и передающей серверу сообщения об изменении этого состояния. **Сервер** — пассивная сущность, предоставляющая агентам ресурсы для приема сообщений их последующей обработки и хранения.

Задача мониторинга представляет собой шаблонную проблему получения и анализа некоторой информации о состоянии удаленного узла.

Под **модулем мониторинга** далее будем понимать абстракцию, характеризующуюся (рисунок 1): возможностью исполнения в операционной среде; входными данными, передаваемыми исполняющей системой; выходными данными, возвращаемыми исполняющей системе; интерфейсом, задающим правила исполнения модуля; реализацией, представляющей собой программный код, воплощающий функционал модуля. Понятие модуля мониторинга является следствием отображения задачи мониторинга из предметной области в программную среду.

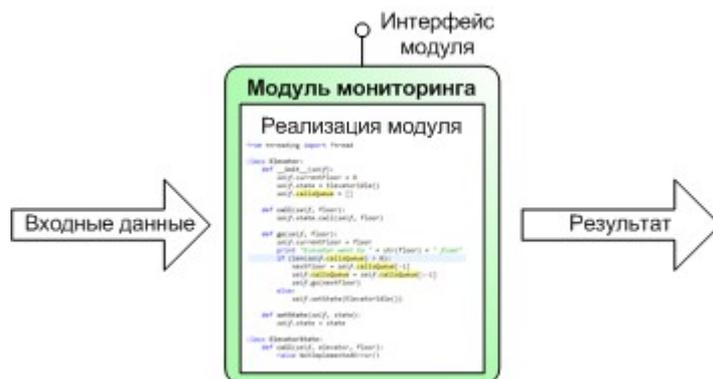


Рисунок 1 - Понятие модуля

В качестве инструмента, позволяющего решать задачи мониторинга, нами был спроектирован механизм разработки и исполнения модулей мониторинга, состоящий из системы исполнения и прикладного интерфейса программирования модулей. Данный механизм позволяет динамически расширять функционал системы с учетом изменяющихся требований путем разработки дополнительных модулей мониторинга.

Система исполнения модулей мониторинга (рисунок 2) реализует генерацию кода каркаса и исполнение модулей мониторинга с использованием ресурсов операционной среды, а также является промежуточным слоем между модулем мониторинга и агентом, в рамках которого он запускается. Данный слой позволяет разрабатывать модули без учета специфики физического расположения агентов (адресации, топологии сети).

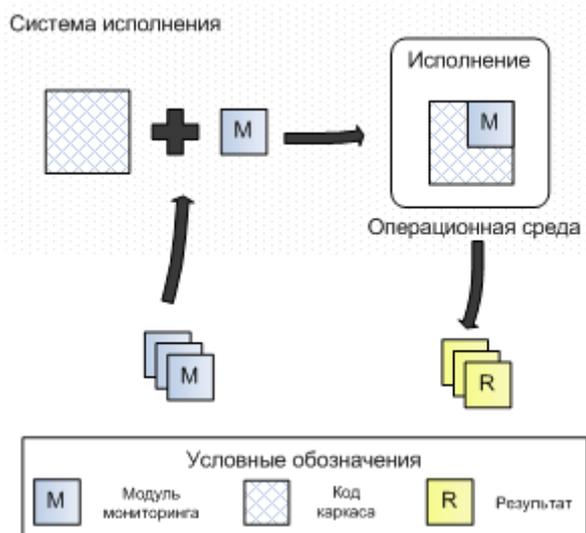


Рисунок 2 - Система исполнения

Код каркаса генерируется системой исполнения на основании текущего глобального состояния распределенной системы и содержит конструкции инициализации окружения, создания экземпляра модуля мониторинга, исполнения экземпляра с передачей параметров и ожиданием возвращаемого результата.

Модули мониторинга разрабатываются в терминах предметной области с использованием **прикладного интерфейса программирования (API)** — высокоуровневого объектно-ориентированного набора инструментов. Прикладной интерфейс программирования является промежуточным слоем между модулем мониторинга и операционной средой, в которой он запущен. API призван сосредоточить программиста на решаемой задаче мониторинга, скрыв от него подробности реализации сложных моментов, таких как распределенная коммуникация с сервером, маршализация/демаршализация параметров и возвращаемого результата модуля, системные вызовы операционной системы.

Известно понятие глобального состояния [1], в соответствии с которым распределенная система функционирует в данное время (рисунок 3). В классической трактовке, это состояние определяется графом связности узлов, расположением запущенных экземпляров модулей и нагрузкой на узлы. В предлагаемой архитектуре сущность распределенного модуля представляет **вторичный сервер**. Это придает ему некоторые особенности элемента распределенной системы [1], например: масштабируемость — возможность запуска дополнительного экземпляра сервера; сериализуемость — возможность сохранения текущего состояния вторичного сервера; переносимость — возможность переноса сервера в распределенной среде с сохранением его внутреннего состояния. **Первичный сервер** является клиентом по отношению к распределенной системе. Удобно представлять такую архитектуру как структуру шаблона «Компоновщик» [2], в которой древовидная структура представляется графом связности узлов, агенты - листовыми объектами, вторичные сервера — составными объектами, наконец, первичный сервер — клиентом, которому требуется

единообразно трактовать листовые и составные объекты структуры.

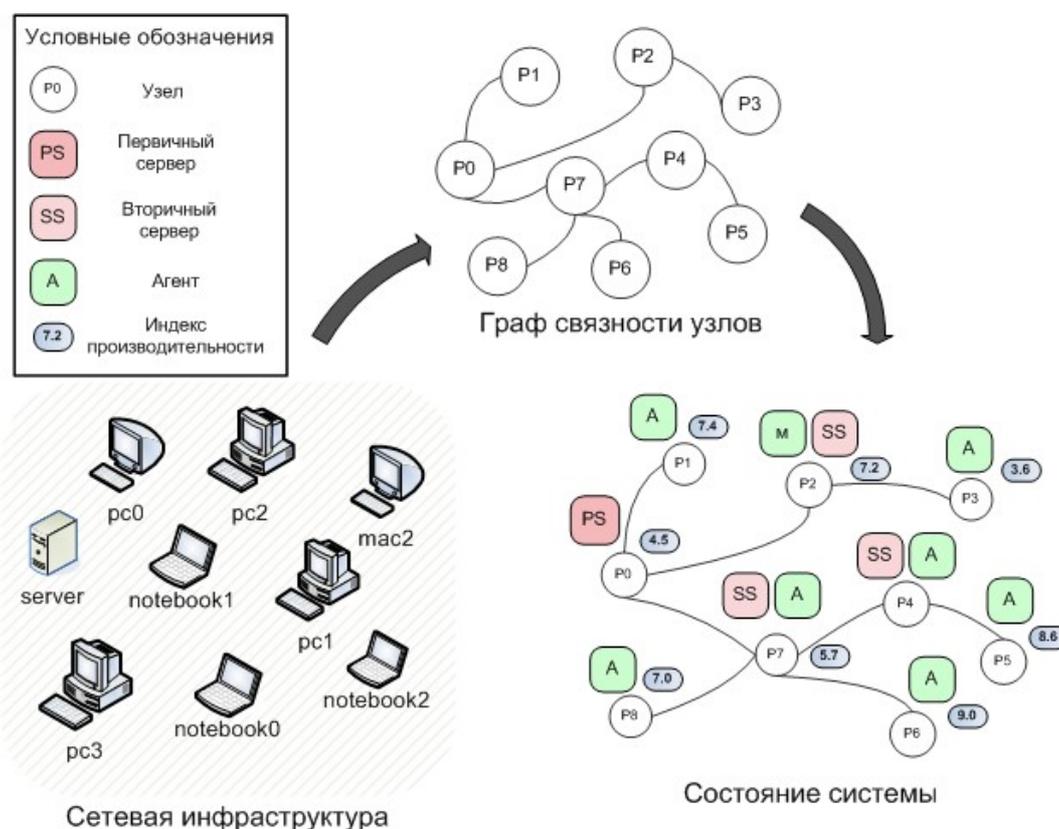


Рисунок 3 - Состояние системы

Роль нагрузки на узел играет **индекс производительности** — вещественное положительное число, определяющее количество свободных ресурсов узла по некоторой шкале. Индекс производительности узла совместно с установленным **пороговым значением** являются рычагами воздействия на глобальное состояние распределенной системы мониторинга. Сервера запущенные на узлах с индексом производительности ниже порогового значения, подвергаются масштабированию (запуску дополнительных экземпляров, сопровождаемому балансировкой нагрузки), и распределенная система переходит в более эффективное состояние.

С целью проверки гипотезы о практической возможности создания распределенной системы мониторинга, нами был разработан программный комплекс, реализующий прототип рассмотренной архитектуры. Опытные результаты эксплуатации прототипа показали его работоспособность и эффективность, что эмпирически доказывает ранее выдвинутую гипотезу.

Можно выделить несколько путей развития проекта: полнофункциональная реализация предложной архитектуры; реализация агентов и прикладного интерфейса программирования модулей для популярных операционных систем; разработка шаблонных модулей мониторинга для решения круга повседневных задач.

Список литературы

1. Э. Таненбаум. Распределенные системы. Принципы и парадигмы / Э. Таненбаум, М. Ван Стеен. — СПб.: Питер, 2003. — 877 с: ил. — (Серия «Классика computer science»).
2. Приемы объектно-ориентированного проектирования. Паттерны проектирования / Э. Гамма, Р. Хелм, Р. Джонсон, Д. Влиссидес. — СПб.: Питер, 2009. — 366 с.