

## **АЛГОРИТМ ДИСПЕТЧИРОВАНИЯ НАГРУЗКИ НА ОБЛАЧНУЮ ИНФРАСТРУКТУРУ**

**Балашов Н.А., Баранов А.В., Белов С.Д., Кадочников И.С., Кореньков В.В.,  
Кутовский Н.А., Нечаевский А.В., Пелеванюк И.С.**

Объединенный институт ядерных исследований, ул. Жолио-Кюри 6, 141980 Дубна,  
Московская область, Российская Федерация

Для предоставления вычислительных ресурсов локальным пользователям, а также участия в российских и международных проектах с применением облачных вычислений и грид-технологий в ОИЯИ создано и развивается облако класса “инфраструктура как сервис” на базе платформы OpenNebula.

Наличие выделенных но простаивающих виртуальных машин (ВМ) очевидно нежелательно, но большинство решаемых с помощью облачной инфраструктуры задач не генерируют предсказуемую равномерную нагрузку. Проблема оптимального использования вычислительных ресурсов неизбежно возникает и в ситуации с физическими компьютерами, и с виртуальными машинами. Дополнительная гибкость облачной инфраструктуры по сравнению с традиционной позволяет повысить эффективность за счёт т.н. механизма “overcommitment” (перерасхода), то есть выделения большего количества виртуальных ресурсов, чем обладает физический сервер.

Когда простаивающие ВМ начинают активно работать, может возникнуть перегрузка физической машины, то есть суммарная требуемая мощность может превысить её возможности. Для минимизации риска перегрузки необходимо динамическое распределение нагрузки и выбор безопасного уровня перерасхода. Идея системы диспетчирования облачных инфраструктур с перерасходом не уникальна, но полноценных, активно развивающихся продуктов для платформы OpenNebula на данный момент нет.

Разрабатываемая в ОИЯИ система интеллектуального диспетчирования облачных ресурсов представляет собой гибкую платформу из взаимодействующих компонентов: базы данных, веб-интерфейса, библиотеки API, службы планирования и набора алгоритмов управления облаком. Ключевым для работы планировщика является алгоритм диспетчирования. Выбор оптимальной стратегии зависит от решаемых облаком задач и создаваемых ими нагрузок, а также желаемых показателей оптимизации и приемлемых рисков перегрузки по памяти и процессору.

В качестве возможного подхода мы предлагаем разделить виртуальные машины на классы по ожидаемой средней нагрузке. Каждому классу соответствует набор физических узлов с настроенным уровнем перерасхода ресурсов, куда диспетчируются машины данного класса.

Построенная платформа позволяет реализовать более сложные стратегии, но для отладки планировщика и получения первых результатов простой и понятный алгоритм имеет большие преимущества.