

ГИБРИДНАЯ МОДЕЛЬ ВЗАИМОДЕЙСТВИЯ ТРИАДЫ АКТОРОВ «ПОТРЕБИТЕЛЬ-БРОКЕР-ПРОВАЙДЕР» В ОБЛАЧНЫХ СРЕДАХ¹

А.В. Скатков, Д.Ю. Воронин, В.И. Шевченко, Р.Н. Литвинова (Севастополь)

Известно [1, 2], что функционирование облачных вычислительных систем (ОВС) ориентировано на достижение одной или множества целей. Одним из характерных примеров целевого назначения ОВС является вычислительное обслуживание крупномасштабных научных задач [3 – 5]. Референтная модель [6] подразумевает, что процесс функционирования ОВС может быть описан взаимодействием Акторов, имеющих в соответствии с их типом (Потребитель, Брокер, Провайдер) индивидуальные целевые функции и сформированный соответствующим образом сценарий поведения. Ввиду сложности и масштабности подобных систем возникает целый ряд проблем, связанных с организацией взаимодействия базовых Акторов в рамках референтной модели [6]. В качестве опорного инструментального средства для организации взаимодействия триады Акторов «Потребитель-Брокер-Провайдер» предлагается гибридная имитационная модель, реализованная в среде AnyLogic.

Постановка задачи и ее решение при использовании гибридной модели

Заданы: отношения предшествования между задачами Потребителя и параметры требуемого вычислительного обслуживания; доступные тарифные планы и существующие директивные ограничения Брокера; параметры Провайдера; критерии эффективности функционирования облачной среды. Необходимо отразить в модели полный цикл обработки задач Провайдера, учитывая особенности взаимодействия базовых Акторов в облачной среде.

Разработанная гибридная модель представляет собой комплекс взаимосвязанных элементов, описывающих поведение базовых Акторов (Потребителя, Брокера, Провайдера) при использовании дискретно-событийного и агентного подхода к реализации моделей в среде AnyLogic [7, 8] (рис. 1).

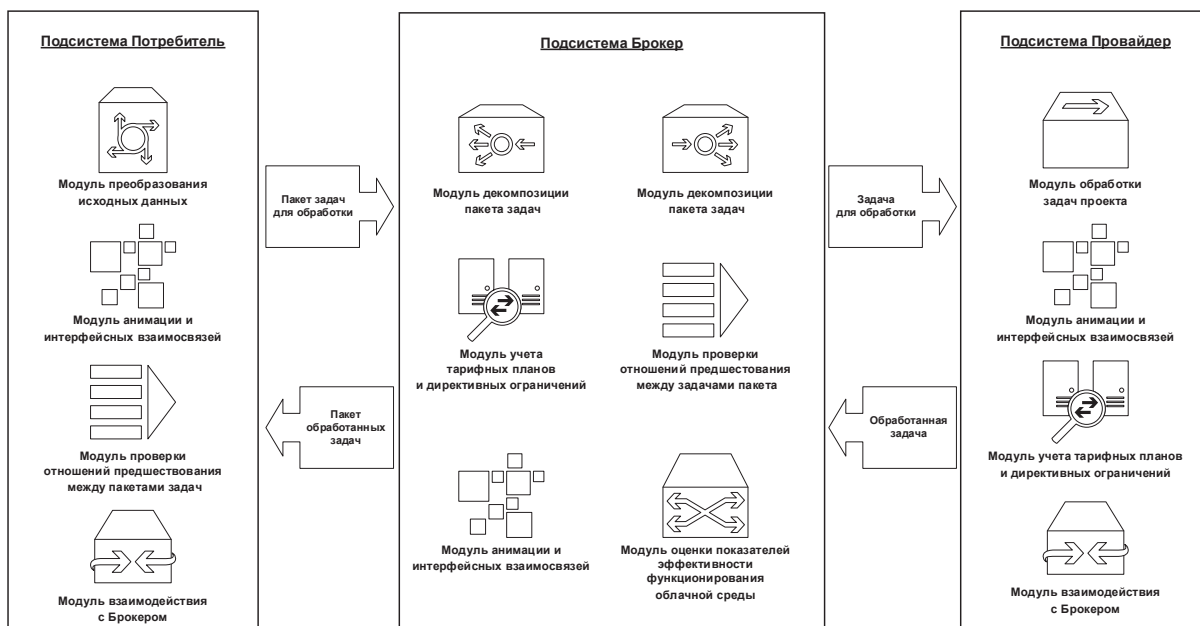


Рис.1. Структурно-функциональная схема гибридной модели

¹ Работа выполнена при финансовой поддержке Российского фонда фундаментальных исследований (грант № 15-29-07936).

На рис. 2 представлено диалоговое окно реализации подсистемы Потребитель. С точки зрения программной реализации каждый Потребитель является агентом, который может находиться в следующих состояниях: «создаю пакет данных для крупномасштабной вычислительной задачи», «отправляю запрос Брокеру для получения вычислительного обслуживания», «ожидаю окончательной обработки всего пакета задач», «свободен» (пакет задач обработан в полном объеме). Для анимации хода вычислительной обработки пакета задач предусмотрена диаграмма, представляющая собой набор пиктограмм, поставленных в соответствие задачам пакета. Для каждой обработанной задачи пакета нотифицируется номер обрабатывающего ее Провайдера. Это позволяет отобразить результаты диспетчерских процессов. Структура пакета задач является иерархической и представляет собой набор агентов, хранящих отношения предшествования между задачами и прочие их индивидуальные параметры (трудоемкость, тип требуемых IT-сервисов и т.д.). При выполнении отношений предшествования задачи пакета отправляются через Брокера на соответствующего Провайдера.

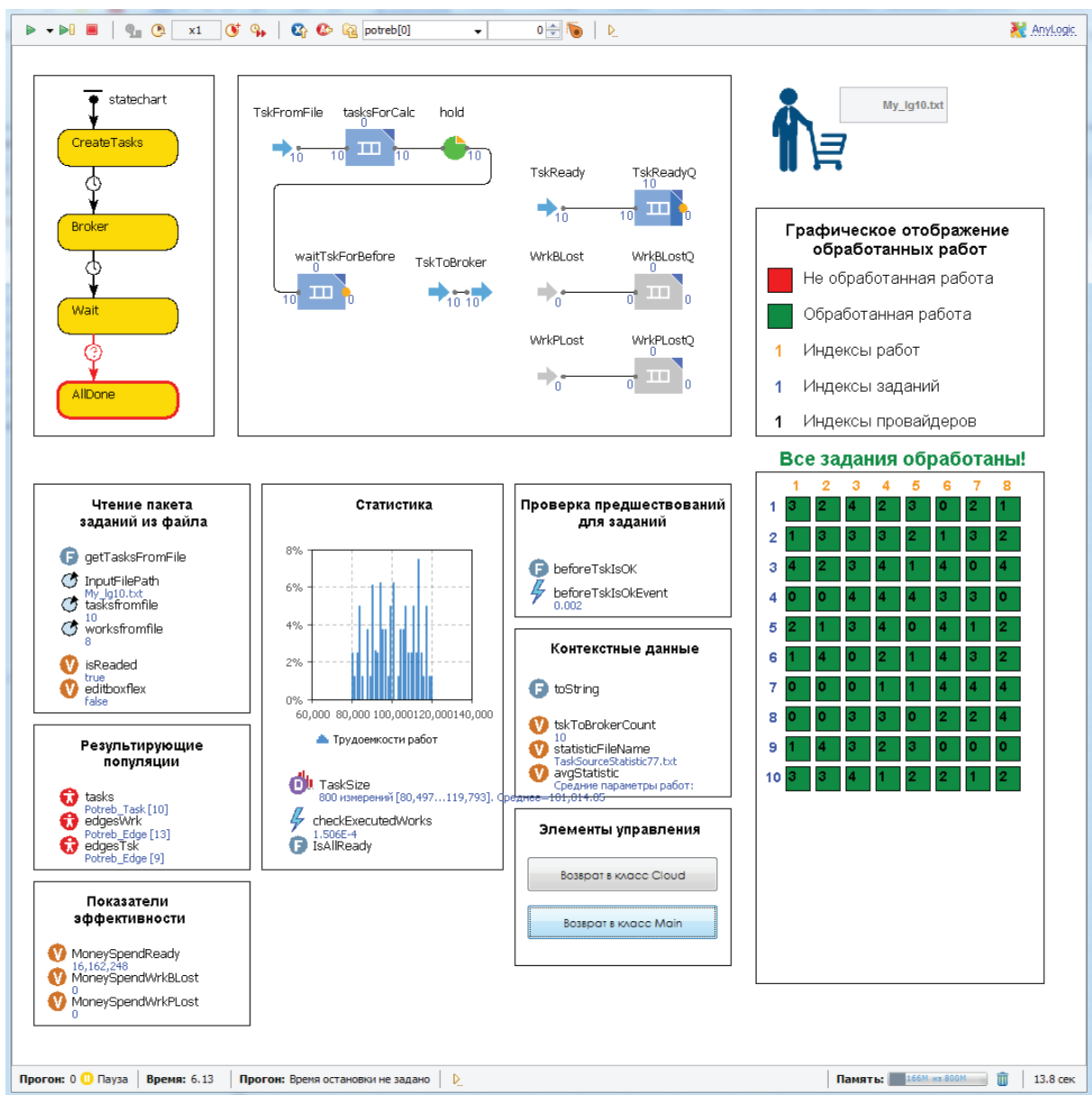


Рис. 2. Диалоговое окно реализации подсистемы Потребитель

Диалоговое окно реализации подсистемы Брокер представлено на рисунке 3. Брокер проверяет соблюдение директивных ограничений по обработке пакета задач и если нарушений не зафиксировано, то в соответствии с выбранной стратегией поиска подходящего Провайдера и предоставленными тарифными планами происходит передача задачи Провайдеру для дальнейшей ее обработки [9]. Диалоговое окно реализации подсистемы Провайдер представлено на рис. 4.

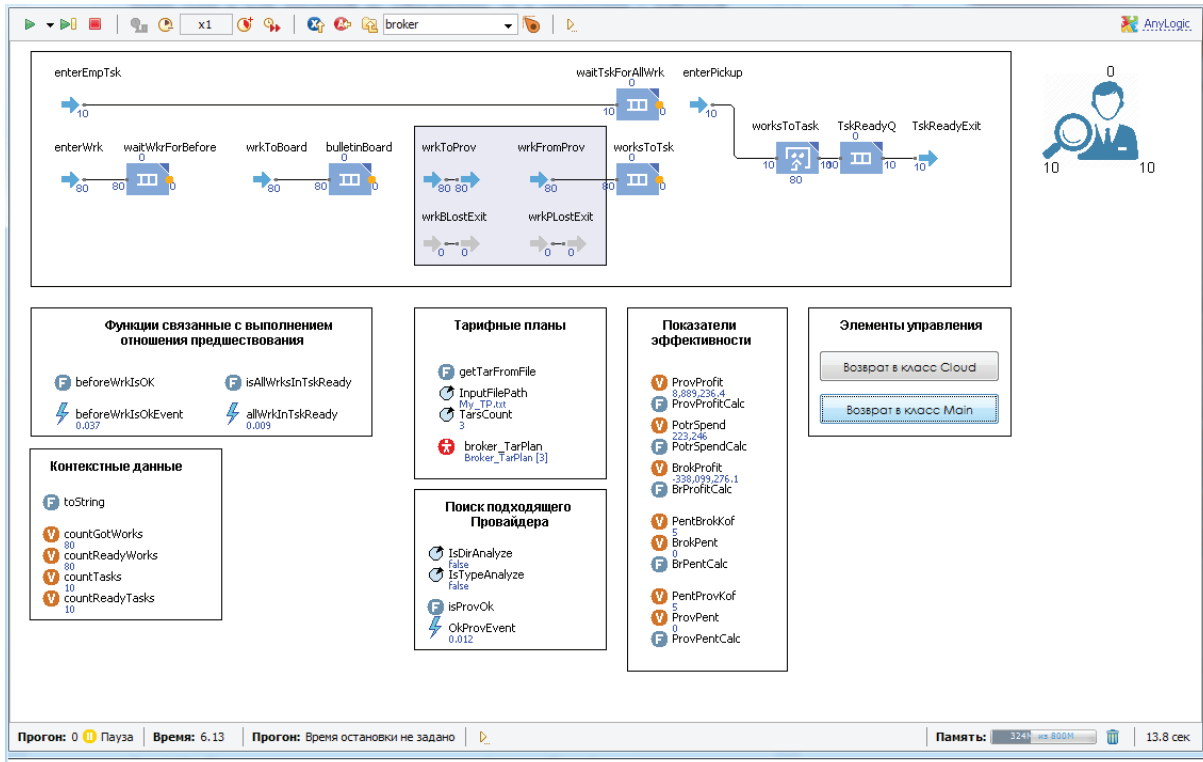


Рис. 3. Диалоговое окно реализации подсистемы Брокер

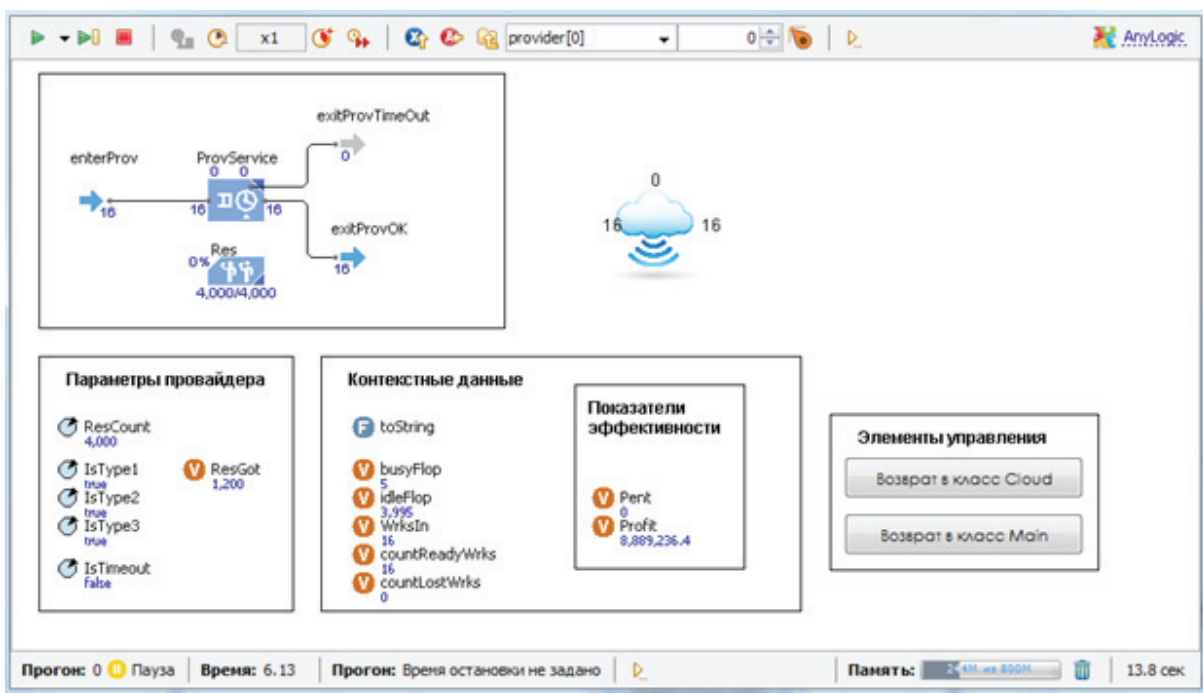


Рис. 4. Диалоговое окно реализации подсистемы Провайдер

В соответствии с текущим тарифным планом Провайдер выделяет на задачу квант своих ресурсов. Далее моделируется процесс обработки задачи пакета: она задерживается в системе на определенное время, длительность которого определяется вычислительной трудоемкостью задачи и объемом выделенных ей ресурсов. При соблюдении директивных ограничений задача считается выполненной, иначе – потерянной. Выполненные задачи пересылаются обратно Потребителю, предварительно зафиксировав всю необходимую информацию о ходе ее обработки в Брокере.

Исходные данные определяют причинно-следственные связи между работами в пакете, а также отношения предшествования между пакетами заданий. Структура анализируемого пакета задач предполагает, что пока не будет обработан первый пакет заданий все остальные – не смогут быть отправлены на обработку. На рис. 5 представлен набор диаграмм, иллюстрирующих ход обработки пакетов заданий Потребителя. Каждая строка матрицы соответствует пакету работ (заданию). Номер столбца сопоставлен номеру работы в пакете.



Рис. 5. Результаты функционирования модели

Представленный пример иллюстрирует слабую загруженность облачной вычислительной среды на начальных этапах обработки пакета задач Потребителя из-за весьма жестких ограничений отношений предшествования. Пока не был выполнен первый пакет заданий, остальные ожидали его завершения и не отправлялись на обработку. Внутри ячеек представлены номера Провайдеров, выполнивших обработку соответствующих задач.

Заключение

При использовании среды многоподходного имитационного моделирования AnyLogic предложена и реализована гибридная имитации модель для анализа взаимодействия триады акторов «Потребитель-Брокер-Провайдер» в облачных средах. Она ориентирована на организацию вычислительного обслуживания крупномасштабных научных задач мониторинга многомерных объектов в конвергентных средах, а также интегрирует методы сетевого анализа проектов, агентную и дискретно-событийную парадигму синтеза имитационных моделей сложных систем, что позволяет ЛПП повысить эффективность решений, принимаемых в рамках расширенной референтной модели облачного компьютера.

Литература

1. **Nelson L.S., Raouf B.** Cloud Architectures, Networks, Services, and Management. – John Wiley & Sons, Inc, 2015. <http://onlinelibrary.wiley.com/book/10.1002/9781119042655> (дата обращения 15. 09.2017).
2. **Скатков А.В., Воронин Д.Ю., Харченко В.С.** Технологии Web, Grid, Cloud для гарантоспособных ИТ-инфраструктур. – Харьков: Изд-во ХАИ, 2013. 868 с.
3. **Vecchiola C., Pandey S. and Buuya R.** (2009) ‘High-performance cloud computing: a view of scientific applications’, ISPAN 2009: Proceeding of 10th International Symposium on Pervasive Sytems, Algorithms, and Network, Kaohsiung, Taiwan. P.4–16.
4. **Keahey K. and Freeman T.** (2008) ‘Science clouds: early experiences in cloud computing for scientific applications’, CCA 2008: Proceeding of Cloud Computing and Applications, Chicago, USA. P.1–6.
5. **Ostermann S., Iosup A., Yigitbasi N., Prodan R., Fahringer T. and Epema D.H.J.** (2009) ‘A performance analysis of ec2 cloud computing services for scientific computing’, CloudComp 2009: Proceeding of 1st International Conference on Cloud Computing, Munich, Germany. P.115–131.
6. **Liu F., Tong J., Mao J., Bohn R., Messina J., Badger L., Leaf D.** NIST Cloud Computing Reference Architecture. Recommendations of the National Institute of Standards and Technology. Cloud Computing Program Information Technology Laboratory National Institute of Standards and Technology Gaithersburg, MD 20899-8930 September 2011 (http://www.nist.gov/customcf/get_pdf.cfm?pub_id=909505).
7. **Боев В.Д.** Компьютерное моделирование: Пособие для практических занятий, курсового и дипломного проектирования в AnyLogic 7. СПб.: ВАС, 2014. 432 с.
8. **Карпов Ю. Г.** Имитационное моделирование систем. Введение в моделирование с AnyLogic 5. СПб.: БХВ-Петербург, 2005. 400 с.
9. **Skatkov A.V., Shevchenko V.I. and Voronin D.Y.** Game-theoretical Management Model for IT-services of ERP-systems Guaranteed Level Assurance in Cloud Environments // Proceedings of the 5th IEEE International Conference on Informatics, Electronics & Vision, 2016, Dhaka, Bangladesh. P. 1113–1116. doi: 10.1109/ICIEV.2016.7760172.