

ХМАРНІ ОБЧИСЛЕННЯ OPENGPSS CLOUD

Д.Г. Діденко

*Навчально-науковий комплекс «Інститут прикладного системного аналізу»**Національний технічний університет України «Київський політехнічний інститут», Україна***Вступ**

Імітаційне моделювання, як метод дослідження систем, вимагає все більше і більше обчислень. Для боротьби з цим можна використовувати Грід-системи у розрізі хмарних обчислень [1]. Грід розглядається як узгоджене, відкрите і стандартизоване середовище, що забезпечує гнучкий, безпечний, скоординований розподіл ресурсів у рамках віртуальної організації. А хмарні обчислення – це модель отримання повсюдного та зручного доступу на вимогу через мережу до спільного пулу обчислювальних ресурсів. Багато років ведуться дослідження відносно паралельного та розподіленого імітаційного моделювання в онлайн системі моделювання OpenGPSS (<http://www.simulation.kiev.ua>) [2-6], для підвищення швидкості роботи якої пропонується застосування Грід, а для зручності роботи з системою пропонується використовувати «хмару».

Концепція хмарного моделювання

В докладі пропонується OpenGPSS Cloud – це концепція хмарного обчислення для системи моделювання OpenGPSS. Тобто це реалізація OpenGPSS не на одному комп'ютері, а на обчислювальному кластері або хмарі:

$$\text{OpenGPSS Cloud} = \text{OpenGPSS} + \text{Cloud} =$$

$$\text{OpenGPSS} + \text{Грід} + \text{Сервіси}_{\text{ доступу}}$$

Побудоване таким чином хмарне середовище OpenGPSS Cloud буде мати наступні можливості [7]: моніторинг проведення імітаційних експериментів; керування процесом проведення імітаційних експериментів; зміна параметрів моделювання на «льоту»; багатокористувацький доступ; реалізація концепції SAAS («ПЗ як послуга»); збереження результатів експериментів; кросплатформеність клієнта; реалізація розподіленого моделювання з синхронізацією імітаційних моделей; незалежність від місця розташування клієнтів.

Іноді також до можливостей відносять «кросплатформеність хмари», але при цьому продуктивність системи може сильно зменшуватися. Під «кросплатформеністю клієнта» розуміється можливість запуску клієнтом веб-браузера на будь-якій апаратній та програмній платформі.

В якості Грїду пропонується використовувати Грід-систему Національного технічного університету України «Київський політехнічний інститут», до якого входить суперкомп'ютер, що

розміщений у «Центрі суперкомп'ютерних обчислень» НТУУ «КПІ» (<http://www.hpcc.org.ua>). У кластері центру використовується 44 вузла з двома чотирьох ядерними процесорами Intel Xeon E5440 @ 2.83ГГц та 8 Гб оперативної пам'яті у кожному та 68 вузла з двома двох ядерними процесорами Intel Xeon 5160 @ 3.00ГГц та 4 Гб оперативної пам'яті у кожному. Мережа обміну даними між вузлами: InfiniBand. Використовується операційна система CentOS 5.2 64 біт. Також встановлений локальний менеджер ресурсів Torque версії 2.3.6 з планувальником задач Maui 3.2.6. В якості компіляторів C++ використовуються intel 10.1 та gcc 4.1.2. З прикладного програмного забезпечення найбільш цікаві бібліотека паралельного програмування OpenMPI та пакети лінійної алгебри ATLAS, LAPACK й ScaLAPACK. Доступ відбувається за протоколом SSH. Ще є друга частина кластеру, але вона працює з операційною системою MS Windows Server 2008 HPC Edition.

Структура OpenGPSS Cloud складається з веб-серверу, «серверу застосувань OpenGPSS», «серверу моделювання OpenGPSS», Грід та відображена на рис.1.

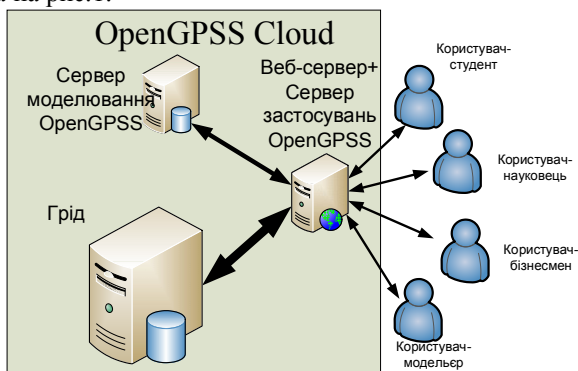


Рисунок 1 - Взаємодія користувачів з OpenGPSS Cloud

В OpenGPSS Cloud будуть доступні два режими моделювання:

- 1) «звичайне послідовне виконання» - виконання послідовного моделювання відразу на «сервері моделювання OpenGPSS». Цей режим використовується в разі неможливості під'єднання до Грїду, або наприклад при проведенні відладки моделі (debug-режим);
- 2) «пакетне виконання» – проведення експериментів для різних моделей або незалежних експериментів однієї моделі на обчислювальних ресурсах Грїду.

Реалізація хмарності вже сьогодні

Частина зазначених вище пунктів вже реалізована у системі моделювання OpenGPSS: наприклад, моніторинг та проведення імітаційних експериментів, реалізація концепції SAAS (веб-застосування, багатокори-

стувацький доступ, збереження результатів експериментів, кросплатформеність клієнта).

Також існують альтернативні розробки з використанням сервісно-орієнтованої архітектури SOA [7-9]. Але нажаль, вони не можуть вважатися «відкритими», тому що містять нестандартні «пропрієтарні» протоколи і розширення.

А що робити, якщо нема можливості побудувати хмару? Наприклад, не розроблене веб-застосування або нема відповідних прикладних інтерфейсів. Деякі з дослідників [8] розглядають доцільність використання систем автоматизації імітаційних досліджень (САІД) як програмних комплексів, що включають в собі систему імітаційного моделювання як ядро, і призначені для автоматизованої розробки та експлуатації імітаційних моделей. Ці САІД не що інше як аналог серверів застосувань для баз даних і також перекликаються з хмарними сервісами, тому що створюють чергу виконання завдань, виконують на моделюючому ядрі і повертають результат. Але система OpenGPSS також вміє це робити. Також у [8] пропонується ввести розподіл в залежності від цільової направленості користувачів, причому при роботі з імітаційними моделями можна виділити наступні групи користувачів: студент, науковець, бізнесмен та модельєр (розробник моделі). Особливості цих ролей наведені у таблиці 1.

Таблиця 1. Режими роботи САІД

№	Режим роботи	Графічний генератор коду	Редактор моделі	Допомога			Функціональність	Програвання (проведення експерименту)
				Робота з інтер-	Теоретична	Практичні		
1	2	3	4	5	6	7	8	9
1	Для студента	Так	Простий	Так	Так	Так	Мала	Так
2	Для науковця	Так	Складний	Так	Так	Так	Велика	Так
3	Для бізнесмена	Ні	Ні	Так	Ні	Ні	Мала	Так
4	Для модельєра	Так	Складний	Так	Ні	Так	Велика	Так

Використання розділень на ролі спростить розуміння програми та зробіть більш чіткими вимоги до графічного інтерфейсу користувача та функціональним обмеженням системи.

Висновки

1. Використання хмарних технологій є суттєвим кроком вперед для підвищення зручності та ефективності роботи систем імітаційного моделювання.

2. Підключення до Грідів підвищить швидкість проведення незалежних імітаційних експериментів.

3. Запропоновані САІД зможуть підвищити продуктивність розробки імітаційних моделей та спростити експлуатацію та проведення імітаційних експериментів користувачами.

Література

1. Таненбаум Э., ван Стеен М. Распределённые системы. Принципы и парадигмы. - СПб.: Питер, 2003. - 877 с.: ил.
2. Діденко Д.Г. Особливості переходу від послідовного моделювання у системі GPSS\World до розподіленого моделювання у системі OpenGPSS. // IV науково-практична конференція з міжнародною участю «Математичне та імітаційне моделювання систем (МОДС-2009)». - Київ. - 2009. - С.197-201.
3. Діденко Д.Г.. Реалізація тиражування обчислювального експерименту в розподіленій системі моделювання OpenGPSS. Наукові вісті. м. Київ - 2007. - № 5. - С. 49-53.
4. Діденко Д.Г. Взаимодействие агентов в распределенной дискретно-событийной системе имитационного моделирования OpenGPSS. // Третья всероссийская научно-практическая конференция «Имитационное моделирование. Теория и практика (ИММОД-2007)», г.Санкт-Петербург. - 2007. - т.1 - С.272-276.
5. Діденко Д.Г. Агент реплікації в розподіленій дискретно-подійній системі імітаційного моделювання OpenGPSS. // Міжнародна наукова конференція «Інтелектуальні системи прийняття рішень та прикладні аспекти інформаційних технологій», м. Євпаторія. - 2006. - т.1 - С.264-266.
6. Томашевский В.Н., Діденко Д.Г. Агентная архитектура распределенной дискретно-событийной системы имитационного моделирования OpenGPSS. Системні дослідження та інформаційні технології. № 4, 2006. - К.: ВПК «Політехніка», 2006. С.123–133.
7. Александров В.В. Единая облачная имитационная среда GPSS Cloud. // Пятая всероссийская научно-практическая конференция по имитационному моделированию и его применению в науке и промышленности «Имитационное моделирование. Теория и практика (ИММОД-2011)». - г. Санкт-Петербург. - 2011. - т.1 - С.315-319.
8. Власов С.А., Девятков В.В., Кобелев Н.Б. Имитационные исследования: от классических технологий до облачных вычислений. // Пятая все-

российская научно-практическая конференция по имитационному моделированию и его применению в науке и промышленности «Имитационное моделирование. Теория и практика (ИММОД-2011)». - г. Санкт-Петербург. - 2011. - т.1 - С.42-50.

9. Александров В.В. Организация имитационного моделирования в среде GPSS на основе SOA. // Четвёртая всероссийская научно-практическая конференция по имитационному моделированию и его применению в науке и промышленности «Имитационное моделирование. Теория и практика (ИММОД-2009)». - г. Санкт-Петербург. - 2009. - т.1 - С.206-209.

УДК 681.086

ЯЗЫК АНАЛИТИК-2010. ОСОБЕННОСТИ И ОБЛАСТЬ ПРИМЕНЕНИЯ

А.Л. Ляхов, Д.Н. Гвоздик

Полтавский национальный технический университет имени Юрия Кондратюка

Данная работа посвящена описанию языка АНАЛИТИК-2010, который является входным языком в системе компьютерной алгебры (СКА) АНАЛИТИК-2010. Работы по созданию языков и систем компьютерной алгебры семейства АНАЛИТИК[1-7], были начаты в 60-х годах прошлого столетия академиком В.М. Глушковым. Первые разработки этого направления, аппаратно реализованные на ЭВМ серии МИР, были ориентированы на решение инженерных и исследовательских задач. Языки семейства АНАЛИТИК предназначены для решения разнообразных по тематике научных и прикладных задач численно-аналитическими методами.

Основным объектом систем компьютерной алгебры (Maple, Mathematica, MathCad) в настоящее время является выражение, подобное выражению математического анализа. Однако более глубокие исследования задач и алгоритмов, которые проводились в процессе разработки СКА АНАЛИТИК привело к тому, что сейчас объектом языка АНАЛИТИК является структура данных - "комплекс", которая имеет более высокий уровень связности, чем структура данных, используемая в других системах [8].

Операндами функций и операторов систем компьютерной алгебры обычно являются именуемые выражения, которые содержат переменные. Эти переменные в свою очередь могут именовать подобные выражения. Именуемые выражения других функций и операторов содержат такие же переменные. Таким образом, множество выражений, которые подаются на вход пользовательской программы или появляются на каком-либо этапе выполнения (промежуточные данные), образуют целостные структуры. Отличительной особенностью АНАЛИТИКА является то, что