

Реплікація в розподіленій дискретно-подійній системі імітаційного моделювання OpenGPSS

Д.Г. Діденко

Аспірант кафедри АСОІУ (Автоматизовані Системи Обробки інформації та Управління)

*Національного Технічного Університету України
"Київський Політехнічний Інститут"*

Системи розподіленого імітаційного моделювання [1, 2] із кожною хвилиною все більше і більше застосовуються для моделювання складних задач у таких сферах економіки як літакобудування, автомобілебудування, логістика та ін. Однією з таких систем є розробка Київського Центра Імітаційного Моделювання розподілена дискретно-подійна система імітаційного моделювання OpenGPSS[3], що працює з мовою імітаційного моделювання GPSS [4, 5]. Ця система являє собою кластер імітаційного моделювання – об'єднання серверів моделювання, та підтримує динамічну переконфігурацію кластера у випадку відмови або приєднання нового сервера.

Забезпечення відмовостійкості системи досягається за допомогою реплікації та персистентного розміщення об'єктів системи у СУБД (зберігання даних у таблицях СУБД).

Кожний з серверів моделювання зберігає інформацію про GPSS-сценарії, GPSS-сущності (одноканальні пристрої (Facility), багатоканальні пристрої (Storage), черги (Queue), функції (Function), логічні перемикачі (Logic Switch), матриці (Matrix), зберігаємі величини (Save Value), таблиці (Table), списки користувача (User Chain), генератори (Generate)) та проводить імітаційні експерименти.

1. Реплікація у системі OpenGPSS

Застосування реплікації у системі OpenGPSS випливає із особливостей архітектури для забезпечення надійності роботи кластера і виконання вимог забезпечення надійного проведення імітаційного експерименту в умовах постійних відмов, включення нових серверів та змін топології з'єднань у кластері (динамічна переконфігурація кластера).

З точки зору даних, кластер моделювання – це розподілена асинхронна БД, що розгорнута на множині взаємодіючих серверів. Усі сервери рівнозначні. Під “сущностями” будемо розуміти GPSS-сценарії, GPSS-сущності та записи про користувачів. Для досягнення незаперечуваності даних у якості відповідних унікальних ідентифікаторів сущностей застосовуємо GUID, що згенеровані різними серверами.

Використання будь-якого головного сервера у системі для генерації ідентифікаторів та синхронізації годинників зменшує можливості масштабованості системи [6], тому що головний сервер буде вузьким місцем для всієї системи відносно пропускної спроможності каналу зв’язку цього сервера. Крім того застосування системи “єдиного годинника” неможливо через

те, що не в усі моменти часу граф зв'язків серверів є зв'язним – при розділі кластеру на частини, кожний “підкластер” продовжує функціонувати незалежно.

Одним з рішень вище згаданих вимог асинхронності створення сущностей та неможливості використання головного сервера є відкладене вилучення сущностей. Для цього створюється список ідентифікаторів “застарілих” сущностей, і періодично система автоматично їх вилучає з сервера. Необхідно лише об'єднувати “застарілі” списки з усіх серверів та зберігати їх на кожному сервері. Треба зауважити, що цей список лише зростає і ніколи не зменшується, тому що ніколи невідома кількість серверів на яких використовувалися сущності, і, виходячи з цього, ніколи достеменно не відомо чи всі такі сущності вилучені з усіх серверів усього кластера.

2. Реалізація реплікації у системі

Програмна реалізація сервера моделювання основана на СУБД Oracle [7, 8] у вигляді таблиць і пакетів на вбудованій процедурній мові PL\SQL [9, 10, 11]; потік керування передається за допомогою завдань (job [10]). Клієнтська же частина – це динамічні HTML-сторінки, що створені на PHP [12].

Сервер моделювання має агентну архітектуру та функціонально складається з наступних частин: агент імітаційного моделювання AgSim, агент реплікації AgRep; агент розділу навантаження AgSpl; агент синхронізації AgSnc; агент прийняття-передачі повідомлень AgTrf; агент продуктивності сервера AgPwr; агент взаємодії з користувачем AgUsr; агент вилучення “застарілих” даних AgGbr, що можуть взаємодіяти один з одним через системні таблиці та інші інтерфейси (наприклад, канали (pipe) або розклад завдань (job)).

Реплікація у системі складається з реплікації користувача і реплікації даних та реалізована у вигляді двох агентів: агент реплікації AgRep і агент вилучення “застарілих” даних AgGbr.

Модуль реплікації користувача.

Якщо користувач поточного завдання не створений на іншому сервері, тоді першим кроком буде створення користувача та копіювання всіх додаткових даних про нього (наприклад, електрона пошта користувача).

Механізм запуску агента.

Агентам системи OpenGPSS притаманні всі властивості агентів [13]:

- *автономність* – властивість діяти без зовнішнього керуючого впливу, тому, по-перше, агент працює в ізольованій сесії, по-друге, керування агенту передається асинхронно за допомогою запуску запланованих завдань (job) у СУБД Oracle;

- *активність* (pro-activeness) – властивість виконувати поставлені перед агентом задачі;

- *реактивність* – агент відповідає на зміни навколоїшнього середовища;

- *комунікативність* – властивість взаємодіяти з іншими агентами – в реалізації агентів використовуються системні інтерфейси поточного сервера для планування наступного запуску агентів.

Модуль реплікації GPSS-сценарій.

Запуск клієнтом імітаційного експерименту – це створення нового завдання у системі, що складається з чотирьох етапів:

1) *етап блокування “застарілого” завдання* - поточне завдання при перезапуску стає “застарілим”;

2) *етап зупинки “застарілих” завдань* полягає в тому, щоб зупинити неактуальні (“застарілі”) завдання, які виконуються агентом імітаційного моделювання AgSim та зробити відмітку про неможливість подальшого виконання цих завдань у майбутньому;

3) *етап видалення “застарілих” завдань* – видаляємо всі неактуальні завдання з поточного сервера;

4) *етап вставки нових завдань* – дані для нових завдань, які є актуальними для двох серверів, копіюються з поточного сервера на інший.

Агент реплікації AgRep.

Агент реплікації AgRep копіює GPSS-модель у внутрішньому форматі (таблиця GPSS-моделей, таблиці GPSS-сущностей) з одного сервера імітаційного моделювання на інший. На кожному сервері зберігається своя копія GPSS-моделі, і далі при балансуванні навантаження передаються лише початкові та кінцеві маркери обробки. Саме тому підвищується швидкість обробки при наявності локальних даних.

Реактивність агента AgRep полягає в тому, що після зміни структури GPSS-моделі (перевантаження GPSS-блоків) або зміни в проведенні імітаційного експерименту, агент реплікації стає першочерговим у списку задач сервера для того, щоб якнайшвидше продублювати ці дані на інших серверах для підвищення надійності.

Активність цього агента визначена досягненням однаковості даних (крім даних імітаційних експериментів) на всіх серверах кластера, за допомогою реплікації користувачів, реплікації GPSS-сценаріїв, планування наступного запуску агента AgRep. У разі відмови працездатність системи зберігається доти доки функціонує останній сервер.

Видалення “застарілої” інформації з системи.

Кожного разу, після проведення реплікації на сервері залишається “застаріла” інформація. Саме тому, час від часу на кожному сервері починає працювати Агент вилучення “застарілих” даних AgGbr. *Активність* агента AgGbr полягає у вилученні “застарілих” даних з поточного сервера для зменшення пам’яті, що відведена під роботу сервера. Агент AgGbr також і *реактивен* – при збільшенні завантаження сервера частість запусків агента збільшується.

Виникнення конфліктів вставки одних і тих самих даних двома (або більше) агентами з різних серверів вирішується автоматично на рівні СУБД Oracle, як проблема блокування унікальних записів при вставці з різних сесій. У цьому випадку “перший за часом” (відносно поточного сервера) агент успішно записує дані і працює далі, а всі інші блокуються – переходят у режим очікування, доки не завершить свою сесію підтвердженням (commit) або скасуванням змін (rollback) перший агент.

3. Висновки

У докладі розглянуті питання реплікації у розподіленій дискретно-подійній системі імітаційного моделювання OpenGPSS. Запропонован підхід реплікації з асинхронним створенням та знищеннем сутностей, для чого реалізовані агент реплікації AgRep і агент вилучення “застарілих” даних AgGbr мовою PL\SQL в СУБД Oracle.

ЛІТЕРАТУРА

1. SPEEDES. <http://www.speedes.com>
2. Mascarenhas E., Knop F., Vernon R. ParaSol: A multithreaded system for parallel simulation based on mobile threads. Winter Simulation Conference 1995.
3. Киевский Центр Имитационного Моделирования. <http://www.simulation.kiev.ua>
4. Шрайбер Томас Дж. Моделирование с использованием GPSS. – М.: Машиностроение, 1980. – 593 с.
5. Томашевский В.Н., Жданова Е.Г. Имитационное моделирование в среде GPSS. – М.: Бестселлер, 2003. – 416 с.
6. Таненбаум Э. С., Ван Стейн М. Распределенные системы. Принципы и парадигмы. – СПб.: Питер, 2003 – 880 с.
7. Бобровски Стив. Oracle8: Архитектура. Пер. с англ. И. Драшников. -М.:Лори, 1998. -210 с.: ил. ISBN 5-85582-040-8 (рус.)
8. Кайт Том. Oracle для профессионалов. Книга 1. Архитектура и основные особенности: Пер. с англ. – М.: ДиасофтЮП, 2003. – 672 с.
9. Прибыл Б., Фейерштайн С. Oracle PL\SQL для профессионалов 3-е изд.: Пер. с англ. - СПб.: Питер, 2003 - 941 с.: ил. ISBN 5-318-00528-4 (рус.), ISBN 0-596-00381-1 (англ.)
10. Кайт Том. Oracle для профессионалов. Книга 2. Расширение возможностей и защита: Пер. с англ. – М.: ДиасофтЮП, 2003. – 848 с.
11. Скотт Урман. Oracle8: Программирование на языке PL/SQL. Пер. с англ. И. Драшников. -М.:Лори, 1999. -608 с.: ил. ISBN 5-85582-043-2 (рус.) 0-07-882305-6 (англ.)
12. Котеров Д., Костарев А. PHP 5. – СПб.: BHV. – 1120 с.
13. Wooldridge M., Jennings R. Intelligent Agents: Theory and Practice. Knowledge Engineering Review, Jan 1995.