НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ

«КПІ імені Ігоря Сікорського»

Теплоенергетичний факультет

Кафедра автоматизації проектування енергетичних процесів і систем

**Розрахунково-графічна робота**

з дисципліни «Високопродуктивні розподілені обчислювальні системи»

Виконав:

студент групи ТМ-01мп

Каліка Б.М.

Перевірив:

ст. в. Свістунов С.Я.

Київ 2020

**Варіант 9.**

**Чисельний метод - Метод Ейлера.**

**Опис програмного рішення для вирішення задачі**

Для розв’язання поставленої задачі був реалізований метод Ефлера за допомогою мови програмування python.

У ході виконання лабораторних робіт отримані результати були отримані за допомогою виконання програми у кластерній, грід, хмарній інфраструктурах. Відповідно до кожної з інфраструктур були створені відповідні скрипти для запуску програм і виконані відповідні команди.

Спробуємо сформулювати визначення кожної з інфраструктур на основі матеріалу з лекцій, літератури і власного бачення, приведемо приклади використання, а також їхні переваги та недоліки.

**Кластер**

Кластер - це зв'язаний набір повноцінних комп'ютерів, який використовується в якості єдиного обчислювального ресурсу та забезпечують користувачам більш високу доступність. В якості вузлів кластери можуть використовуватися як однакові (гомогенні кластери), так і різні (гетерогенні кластери) обчислювальні машини. За своєю архітектурою кластерна обчислювальна система є слабо зв'язаної. Для створення кластерів зазвичай використовуються або прості однопроцесорні персональні комп'ютери, або двох- або чотирипроцесорні SMP-сервери. При цьому не накладається ніяких обмежень на склад і архітектуру вузлів. Ці кластери використовуються для скорочення часу простою, дозволяючи іншому серверу продовжувати роботу в разі збою. Група серверів підключена до однієї системи. У той момент, коли один з цих серверів стає недоступним, робоче навантаження перерозподіляється на інший сервер до того, як клієнт відчуває будь-який час простою.

**Виконання поставлених задач:** Для виконання обчислення на кластері через інтерфейсний вузол потрібно створити bash скрипт. В скрипті описати інструкцію запуску і виконання програми і всі конфігурації необхідні для роботи програми. Так як інтерпретатор python уже встановлений, то просто пишемо команду на запуск програми python lab1.py. Після цього відправляємо завдання на виконання із зазначенням черги завдань **qsub -q bitpedu test.pbs**. Отже для запуску виконання задачі на навчальному кластері потрібно підключення через ssh і запустити задачу на виконання. В результаті отримуємо файл з вихідними даними або файлом з помилками і інші файли, які були зазначені в описі задачі.

**Переваги:** лекгокерованість, проста у впровадженні, основна архітектура є зрозумілою для користувача, тобто для користувача вся система, діє як один процесор, хоча насправді містить процесори, програми, мережеві пристрої та драйвери.

**Недоліки:** користувачу потрібно дочекатися вільних ресурсів для подання роботи в чергу і очікувати отримання результатів.

**Грід**

Грід - це одна з форм розподілених обчислень, в якій суперкомп'ютер складається з одиночних комп'ютерів і кластерів, географічно розподілених в різних місцях і працюють разом для виконання певних операцій. За своєю суттю грід - це технологія, яка використовується для вирішення наукових завдань, що вимагають значних обчислювальних ресурсів. Грід - використовуються в комерційній інфраструктурі для вирішення таких завдань, як економічне прогнозування, сейсмоаналіз, розробка та вивчення властивостей нових ліків. Всі беруть участь обчислювальні ресурси об'єднуються в одну єдину грід-мережу. Одним з найяскравіших представників такої мережі є LCG (LHC Computing Grid) - грід, призначений для обробки великих обсягів даних, що надходять з БАК (Великого адронного коллайдера). У його склад входить 170 обчислювальних центрів з 36 країн. Також в нашій країні існує наша власна національна грід-мережу на базі НАН Білорусі - BASNET-GRID.

Для побудови повністю функціональної грід-системи необхідне програмне забезпечення проміжного рівня (middleware), побудоване на базі існуючих інструментальних засобів яке дає високорівневі сервіси для задач і користувачам.

Труднощами, що виникають при побудові систем подібного класу, зазвичай називають управління ресурсами, забезпечення захисту інформації та самого процесу обчислень, а також стандартизацію обчислювальної платформи і "незалежність" виконуваного програмного забезпечення (ПО). Під "незалежністю" мається на увазі міжплатформова переносимість ПЗ, що не прив'язаного до типу операційної системи і апаратного пристрою комп'ютера, обмін даними через загальновідомі і стандартизовані протоколи. Для досягнення цих якостей середовище повинне будуватися на основі принципів відкритих систем. Принцип відкритих систем полягає у використанні стандартних інтерфейсів. Але, від декларації необхідності реалізації принципів відкритих систем середовище ще не набуває властивостей відкритості. Для цього необхідно застосувати ряд задекларованих етапів, щоб система могла вважатися відкритою. Даний процес називається технологією відкритих систем.

**Виконання поставлених задач:** для виконання обчислення в грід інфраструктурі за допомогою технології віддаленого доступу до ресурсів (**ARC**) потрібно створити (замовити) проксі-сертифікат, який діє обмежений час і запустити задачу на виконання. Оформити опис задачі для виконання і скрипт для виконання необхідних команд. Задача поміщається в чергу, час від часу перевіряємо її статус. Після успішного її виконання – копіюємо результати у вигляді файлів в свою домашню директорію.

**Переваги:** можливість запускати програму на одній з машин, що не працюють у грід інфраструктурі, а не запускати його на тій самій призначеній машині, яка може бути зайнятою на час запиту. Програми можна записати як підзадачі, кожна з яких можна призначити процесору, просто мінімізуючи час, необхідний для запуску всієї роботи.

**Недоліки:** вимагає високошвидкісного доступу до Інтернету. Якщо операція, що виконується в даний час, залежить від попередніх завдань, в такому випадку обробка не може виконуватися паралельно на розподілених ресурсах. Тоді програма не може отримати переваги від грід з точки зору високопродуктивних обчислень.

**Хмара**

Хмара - це модель надання можливості повсюдного і зручного мережевого доступу на вимогу до пулу поділюваних конфігуруються обчислювальних ресурсів (наприклад, мереж, серверів, засобів зберігання, додатків і сервісів), які можуть оперативно надаватися і звільнятися при мінімальному зусиллі управління або взаємодії з провайдером (постачальником). Ця модель хмари представлена (описана) п'ятьма основними характеристиками, трьома сервісними моделями і чотирма моделями розгортання. Хмарна модель володіє п'ятьма основними властивостями, і складається з трьох моделей служб і чотирьох моделей розгортання:

* Самообслуговування по вимогам (On-demand self-service).
* Широкий мережевий доступ (Broad network access).
* Обєднання ресурсів в пули (Resorce pooling).
* Миттєва еластичність (Rapid elasticity).
* Вимірюваність (Measured service).

У хмарній інфраструктурі під керуванням OpenStack для нас був створений проєкт СloudKPI, в якому можна запустити свій віртуальний сервер. Доступи до цього серверу отримали індивідуально у викладача. Для виконання створювали піддиректорію зі своїм іменем (можна було вказати іншу назву) в якій зберігалися файли для роботи. В скрипті палаштовували змінні оточення для доступу до хмарної інфраструктури BITP OpenStack. Також створювали та описували файл з конфігурацією для віртуальної машини (необхідні модулі та бібліотеки). Також була можливість перевірки статусу виконання завдання на віртуальній машині, а також отримати посилання на віртуальну машину через Web. За допомогою посилання заходили на віртуальну машину, перевіряли файли і копіювати їх у свою директорію (для полегшення використовували програму WinSCP).

**Переваги:** адаптованість до змін робочого навантаження, безпека, надійність, доступність, балансування навантаження, масштабованість.

**Недоліки:** вимагає стабільного з’єднання з Інтернетом, користувачі обмежені утилітами, операційними системами, прикладним ПЗ, що надаються хостинговою компанією.

**Висновки**

У цій роботі було розглянуті кластерна, грід та хмарна обчислювальні інфраструктури. Було описано процес роботи з кожною з них (конфігурація, використання, розрахунки та отримання результатів), наведені переваги і недоліки кожної з них.

Таке порівняння полегшить розуміння інфраструктур, оскільки їх особливості схожі в концептуальному плані.