

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ФАХОВИЙ КОЛЕДЖ РАКЕТНО-КОСМІЧНОГО МАШИНОБУДУВАННЯ
ДНІПРОВСЬКОГО НАЦІОНАЛЬНОГО УНІВЕРСИТЕТУ ім. О. ГОНЧАРА

КОНСПЕКТ ЛЕКЦІЙ

з дисципліни

«Хмарні технології»

ЗІ СПЕЦІАЛЬНОСТІ 121

«ІНЖЕНЕРІЯ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ»

Розробила: _____Ланська С.С.

Розглянуто та схвалено

на засіданні циклової комісії

програмної інженерії

Протокол № 2 від 21.09.2023 р.

Голова комісії _____Ланська С.С.

Дніпро

2023

Лекція № 1

з навчальної дисципліни «Хмарні технології»

Тема: Технологічні основи хмарних обчислень. Хмарні технології: основні поняття, завдання та тенденції розвитку

Мета заняття: надання необхідних теоретичних знань про призначення та стан розвитку хмарних технологій, поняття та завдання інформаційного забезпечення, а також використання новітніх інструментів обробки та збереження інформації.

Час – 2 години

План проведення лекції

- 1) Основні етапи розвитку апаратного і програмного забезпечення.
- 2) Сучасні технологічні рішення.
- 3) Хмарні технології: основні поняття, завдання та тенденції розвитку.

Література

- 1) Кононюк А.Е. Фундаментальная теория облачных технологий [Текст]: Книга 1. Общенаучные подходы формирования систем облачных технологий / А.Е. Кононюк. – К.: Освіта України, 2018 – 621 с.
 - 2) Зінченко О.В., Іщеряков С.М., Прокопов С.В., Сєрих С.О., Василенко В.В. Хмарні технології. – Навчальний посібник. – К: ФОП Гуляєва В.М., 2020. –74 с.
 - 3) Хмарні технології в освіті. Навчально-методичний посібник для студентів фізико-математичного факультету. – Житомир: ЖДУ, 2016. – 72 с.
-

Навчальні матеріали лекції

1.1 Основні етапи розвитку апаратного і програмного забезпечення.

Для того, щоб зрозуміти, як з'явилися «хмарні» обчислення, необхідно уявляти основні моменти процесу розвитку обчислень і обчислювальної техніки.

В наш час життя без комп'ютерів є неможливим. Впровадження обчислювальної техніки проникло майже в усі життєві аспекти, як особисті, так і професійні. Розвиток комп'ютерів був досить швидким. Початком еволюційного розвитку комп'ютерів став 1930 рік, коли двійкова арифметика була розроблена і стала основою комп'ютерних обчислень і мов програмування.

У 1939 році були винайдені електронно-обчислювальні машини, які виконують обчислення в цифровому вигляді.

В 1941 році в німецькій Лабораторії Авіації в Берліні з'явилася модель Z3 Конрада Цузе, що було однією з найбільш значних подій у розвитку комп'ютерів, тому що ця машина підтримувала обчислення як з плаваючою точкою, так і двійкову арифметику. Цей пристрій розглядають як найперший комп'ютер, який був повністю працездатним. Мовою програмування вважають «Turing-complete», якщо він потрапляє в той же самий обчислювальний клас, як машина Тьюринга.

Перше покоління сучасних комп'ютерів з'явилося в 1943, коли були розроблені Марк I і машина Колос. З фінансовою підтримкою від IBM (International Business Machines Corporation) Марк був сконструйований і розроблений в Гарвардському університеті. Це був електромеханічний програмований комп'ютер загального призначення. Перше покоління комп'ютерів було побудовано з використанням з'єднаних проводів і електронних ламп (термоелектронних ламп). Дані зберігалися на паперових перфокартах. Колос використовувався щоб допомогти розшифрувати зашифровані повідомлення.

Щоб виконати завдання розшифровки, Колос порівнював два потоки даних, прочитаних на високій швидкості з перфострічки. Колос оцінював потік даних, вважаючи кожний збіг, який було виявлено, ґрунтуючись на програмуєму Булеву функцію. Для порівняння з іншими даними був створений окремий потік.

Інший комп'ютер загального призначення цієї ери був ENIAC (Electronic Numerical Integrator and Computer — Електронний Числовий Інтегратор і Комп'ютер), який був побудований в 1946.

Він був першим комп'ютером, здатним до перепрограмування, щоб вирішувати повний спектр обчислювальних проблем. ENIAC містив 18 000

термоелектронних ламп, що важив більше ніж 27 тон, і споживав електроенергії 25 кіловат на годину. ENIAC виконував 100 000 обчислень в секунду. Винахід транзистора означав, що неефективні термоелектронні лампи могли бути замінені більш дрібними і надійними компонентами. Це було наступним головним кроком в історії обчислень.

Комп'ютери Transistorized відзначили появу другого покоління комп'ютерів, які домінували в кінці 1950-х і на початку 1960-х. Незважаючи на використання транзисторів і друкованих схем, ці комп'ютери були все ще великими і дорогими. В основному вони використовувалися університетами та урядом. Інтегральна схема або чіп були розвинені Джеком Кілбі. Завдяки цьому досягненню він отримав Нобелівську премію з фізики в 2000 році.

Винахід Кілбі викликав вибух у розвитку комп'ютерів третього покоління. Навіть при тому, що перша інтегральна схема була розроблена у вересні 1958, чіпи не використовувалися в комп'ютерах до 1963. Історію мейнфреймів прийнято відраховувати з появи в 1964 році універсальної комп'ютерної системи IBM System/360, на розробку якої корпорація IBM витратила 5 млрд. доларів.

Мейнфрейм — це головний комп'ютер обчислювального центру з великим об'ємом внутрішньої і зовнішньої пам'яті. Він призначений для завдань, що вимагають складних обчислювальних операцій. Сам термін «мейнфрейм» походить від назви типових процесорних стійок цієї системи. У 1960-х — початку 1980-х років System/360 була беззаперечним лідером на ринку. У той час такі мейнфрейми, як IBM 360 збільшили здатність зберігання і обробки, інтегральні схеми дозволяли розробляти мінікомп'ютери, що дозволило великій кількості маленьких компаній проводити обчислення.

Інтеграція високого рівня діодних схем призвела до розвитку дуже маленьких обчислювальних одиниць, що призвело до наступного кроку розвитку обчислень.

У листопаді 1971 Intel випустили перший в світі комерційний мікропроцесор, Intel 4004. Це був перший повний центральний процесор на одному чіпі і став першим комерційно доступним мікропроцесором. Це було

можливо через розвиток нової технології кремнієвого керуючого електрода. Це дозволило інженерам об'єднати набагато більше число транзисторів на чіпі, який виконував би обчислення на невеликій швидкості. Ця розробка сприяла появі комп'ютерних платформ четвертого покоління.

Комп'ютери четвертого покоління, які розвивалися в цей час, використовували мікропроцесор, який розміщує здатності комп'ютерної обробки на єдиному чіпі. Комбінуючи пам'ять довільного доступу (RAM), розроблену Intel, комп'ютери четвертого покоління були швидші, ніж будь-коли раніше і займали набагато меншу площу. Процесори Intel 4004 були здатні виконувати лише 60 000 інструкцій в секунду. Першим комерційно доступним персональним комп'ютером був MITS Altair 8800, випущений в кінці 1974 року.

У подальшому були випущені такі персональні комп'ютери, як Apple I і II, Commodore PET, VIC-20, Commodore 64, і, в нарешті, оригінальний IBM-PC в 1981. Ера PC почалася всерйоз до середини 1980-их. Протягом цього часу, IBM PC, Commodore Amiga і Atari ST були найпоширенішими платформами PC, доступними громадськості. Навіть при тому, що мікрообчислювальна потужність, пам'ять і зберігання даних потужності збільшилися набагато порядків, починаючи з винаходу з Intel 4004 процесорів, технології чіпів інтеграції високого рівня (LSI) або інтеграція надвисокого рівня (VLSI) сильно не змінилися. Тому більшість сьогоденішніх комп'ютерів все ще потрапляє в категорію комп'ютерів четвертого покоління.

Одночасно з різким зростанням виробництва персональних комп'ютерів на початку 1990-х почалася криза ринку мейнфреймів, пік якого припав на 1993 рік. Багато аналітиків заговорили про повне вимирання мейнфреймів, про перехід від централізованої обробки інформації до розподіленої (за допомогою персональних комп'ютерів, об'єднаних дворівневою архітектурою «клієнт-сервер»). Багато хто став сприймати мейнфрейми як вчорашній день обчислювальної техніки, вважаючи Unix- і PC-сервери більш сучасними і перспективними.

З 1994 року знову почалося зростання інтересу до мейнфреймів. Справа в тому, що, як показала практика, централізована обробка на основі мейнфреймів

вирішує багато завдань побудови інформаційних систем масштабу підприємства простіше і дешевше, ніж розподілена. Багато з ідей, закладених в концепції хмарних обчислень, також "повертають" нас до епохи мейнфреймів, зрозуміло з поправкою на час. Раніше у бесіді з Джоном Менлі, одним з провідних наукових співробітників центру досліджень і розробок HP в Брістолі, обговорювалася тема хмарних обчислень, і Джон звернув увагу на те, що основні ідеї cloud computing до болі нагадують мейнфрейми, тільки на іншому технічному рівні: «Все йде від мейнфреймів. Мейнфрейми навчили нас тому, як в одному середовищі можна ізолювати додатки, — вміння, критично важливе сьогодні ».

1.2 Сучасні технологічні рішення.

З кожним роком вимоги бізнесу до безперервності надання сервісів зростають, а на застарілому обладнанні забезпечити безперебійне функціонування практично неможливо. У зв'язку з цим найбільші ІТ-вендори виробляють і впроваджують більш функціональні і надійні апаратні і програмні рішення. Розглянемо основні тенденції розвитку інфраструктурних рішень, які, так чи інакше, сприяли появі концепції хмарних обчислень: зростання продуктивності комп'ютерів, поява багатопроцесорних і багатоядерних обчислювальних систем, розвиток блейд-систем, поява систем і мереж зберігання даних, консолідація інфраструктури.

Блейд-системи.

У процесі розвитку засобів обчислювальної техніки завжди існував великий клас завдань, що вимагають високої концентрації обчислювальних засобів. До них можна віднести, наприклад, складні ресурсномісткі обчислення (наукові завдання, математичне моделювання), а також завдання з обслуговування великого числа користувачів (розподілені бази даних, Інтернет-сервіси, хостинг).

Не так давно виробники процесорів досягли розумного обмеження нарощування потужності процесора, при якому його продуктивність дуже висока при відносно низькій вартості. При подальшому збільшенні потужності процесора, необхідно було вдаватися до нетрадиційних методів охолодження

процесорів, що досить незручно і дорого. Виявилося, що для збільшення потужності обчислювального центру більш ефективно збільшити кількість окремих обчислювальних модулів, а не їх продуктивність. Це призвело до появи багатопроцесорних, а пізніше і багатоядерних обчислювальних систем.

З'являються багатопроцесорні системи, які нараховують більше 4 процесорів. На поточний момент існують процесори з кількістю ядер 8 і більше, кожне з яких еквівалентно по продуктивності. Збільшується кількість слотів для підключення модулів оперативної пам'яті, а також їх ємність і швидкість.

Збільшення числа обчислювальних модулів в обчислювальному центрі вимагає нових підходів до розміщення серверів, а також призводить до зростання витрат на приміщення для центрів обробки даних, їх електроживлення, охолодження і обслуговування.

Для вирішення цих проблем було створено новий тип серверів XXI століття — модульні, частіше звані Blade-серверами, або серверами-лезами (blade - лезо).

За визначенням, даним аналітичною компанією IDC Blade-сервер або лезо — це модульна одноплатна комп'ютерна система, що включає процесор і пам'ять. Леза вставляються в спеціальне шасі з об'єднуючою панеллю (backplane), що забезпечує їм підключення до мережі і подачу електроживлення. Це шасі з лезами є Blade-системою. Воно виконане в конструктиві для установки в стандартну 19-дюймову стійку і в залежності від моделі та виробника займає в ній 3U, 6U або 10U (один U — unit, або монтажна одиниця, дорівнює 1,75 дюйма). За рахунок загального використання таких компонентів, як джерела живлення, мережеві карти і жорсткі диски, Blade-сервери забезпечують більш високу щільність розміщення обчислювальної потужності в стійці в порівнянні зі звичайними тонкими серверами заввишки 1U і 2U.

Технологія блейд-систем запозичує деякі риси мейнфреймів. В даний час лідером у виробництві блейд-систем є компанії Hewlett-Packard, IBM, Dell, Fujitsu Siemens Computers, Sun.

Розглянемо основні переваги блейд-систем:

1) Унікальна фізична конструкція. Архітектура блейд-систем заснована на детально відпрацьованій унікальній фізичній конструкції. Спільне використання таких ресурсів, як ресурси живлення, охолодження, комутації та управління, знижує складність і ліквідує проблеми, які характерні для більш традиційних стійкових серверних інфраструктур. Фізична конструкція блейд-систем передбачає розміщення блейд-серверів в спеціальному шасі і основним її конструктивним елементом є об'єднуюча панель. Об'єднуюча панель розроблена таким чином, що вона вирішує всі завдання комутації блейд-серверів із зовнішнім світом: з мережами Ethernet, мережами зберігання даних Fiber Channel, а також забезпечує взаємодію по протоколу SAS (SCSI) з дисковими підсистемами в тому ж шасі.

2) Кращі можливості управління і гнучкість. Блейд-сервери принципово відрізняються від стійкових серверів тим, що серверна полка має інтелект у вигляді модулів управління, який відсутній в стійках при розміщенні традиційних серверів. Для управління системою не потрібні клавіатура, відео і миша.

Управління блейд-системою здійснюється за допомогою централізованого модуля управління і спеціального процесора віддаленого управління на кожному блейд-сервері. Система управління шасі і серверами як правило мають досить зручне програмне забезпечення для управління.

З'являються можливості дистанційно керувати всією «Blade»-системою, в тому числі управління електроживленням і мережею окремих вузлів.

3) Масштабованість. При необхідності збільшення продуктивних потужностей, достатньо придбати додаткові леза і підключити до шасі. Сервери та інфраструктурні елементи в складі блейд-систем мають менший розмір і займають менше місця, ніж аналогічні стійкові рішення, що допомагає економити електроенергію і простір, виділені для ІТ. Крім того, завдяки модульній архітектурі, вони є більш зручними у впровадженні та модернізації.

4) Підвищена надійність. У традиційних стійкових середовищах для підвищення надійності встановлюється додаткове обладнання, засоби комутації та мережеві компоненти, що забезпечують резервування, що тягне за собою

додаткові витрати. Блейд-системи мають вбудовані засоби резервування, наприклад, передбачається наявність декількох блоків живлення, що дозволяє при виході з ладу одного блоку живлення забезпечувати безперебійну роботу всіх серверів, розташованих в шасі.

5) Зниження експлуатаційних витрат. Застосування блейд-архітектури призводить до зменшення енергоспоживання і тепла, що виділяється, а також до зменшення займаного обсягу. Крім зменшення займаної площі в ЦОД, економічний ефект від переходу на леза має ще кілька складових. Оскільки в них входить менше компонентів, ніж у звичайні стійкові сервери, і вони часто використовують низьковольтні моделі процесорів, що скорочують вимоги до енергозабезпечення та охолодження машин. Інфраструктура блейд-систем є більш простою в управлінні, ніж традиційні IT-інфраструктури на серверах. У деяких випадках блейд-системи дозволили компаніям збільшити кількість ресурсів під керуванням одного адміністратора (сервери, комутатори і системи зберігання) більш ніж в два рази. Керуюче програмне забезпечення допомагає IT-організаціям економити час завдяки можливості ефективного розгортання, моніторингу та контролю за інфраструктурою блейд-систем. Перехід до серверної інфраструктури, побудованої з лез, дозволяє реалізувати інтегроване управління системи і відійти від колишньої схеми роботи Intel-серверів, коли кожному додатку виділялася окрема машина. На практиці це означає значно раціональніше використання серверних ресурсів, зменшення числа рутинних процедур (таких, як підключення кабелів), які повинен виконувати системний адміністратор, і економію його робочого часу.

Системи зберігання даних.

Іншою особливістю сучасної історії розвитку обчислювальних систем поряд з появою блейд-серверів стали появи спеціалізованих систем і мереж зберігання даних. Внутрішні підсистеми зберігання серверів часто вже не могли надати необхідний рівень масштабованості і продуктивності в умовах лавиноподібного нарощування обсягів оброблюваної інформації. В результаті з'явилися зовнішні

системи зберігання даних, орієнтовані суто на вирішення завдань зберігання даних і надання інтерфейсу доступу до даних для їх використання.

Система Зберігання Даних (СЗД) — це програмно-апаратне рішення по організації надійного зберігання інформаційних ресурсів та надання до них гарантованого доступу.

Системи зберігання даних являють собою надійні пристрої зберігання, виділені в окремий вузол. Система зберігання даних може підключатися до серверів багатьма способами. Найбільш продуктивним є підключення по оптичним каналам (Fiber Channel), що дає можливість отримувати доступ до систем зберігання даних зі швидкостями 4-8 Гбіт/сек.

Відзначимо основні переваги використання СЗД:

1) Висока надійність і відмовостійкість – реалізується повним або частковим резервуванням всіх компонентів системи (блоків живлення, шляхів доступу, процесорних модулів, дисків, кеша і т.д.), а також потужною системою моніторингу та оповіщення про можливі і існуючі проблеми.

2) Висока доступність даних – забезпечується продуманими функціями збереження цілісності даних (використання технології RAID, створення повних і миттєвих копій даних усередині дискової стійки, реплікування даних на віддалену СЗД і т.д.) і можливістю додавання (поновлення) апаратури і програмного забезпечення в безперервно працюючу систему зберігання даних без зупинки комплексу.

3) Потужні засоби управління і контролю - управління системою через web-інтерфейс або командний рядок, вибір декількох варіантів сповіщення адміністратора про неполадки, повний моніторинг системи, працююча на рівні "заліза" технологія діагностики продуктивності.

4) Висока продуктивність - визначається числом жорстких дисків, об'ємом кеш-пам'яті, обчислювальною потужністю процесорної підсистеми, числом внутрішніх (для жорстких дисків) і зовнішніх (для підключення хостов) інтерфейсів, а також можливістю гнучкого налаштування і конфігурації системи для роботи з максимальною продуктивністю.

5) Безпроблемна масштабованість - зазвичай існує можливість нарощування числа жорстких дисків, об'єму кеш-пам'яті, апаратної модернізації існуючої системи зберігання даних, нарощування функціонала за допомогою спеціального ПЗ, працюючого на стойці, без значної переконфігурації або втрат якоїсь функціональності СЗД. Цей момент дозволяє значно економити і гнучкіше проектувати свою мережу зберігання даних.

Мережі зберігання даних.

SAN — це високошвидкісна комутована мережа передачі даних, що об'єднує сервери, робочі станції, дискові сховища і стрічкові бібліотеки. Обмін даними відбувається по протоколу Fibre Channel, оптимізованому для швидкої гарантованої передачі повідомлень і дозволяючому передавати інформацію на відстань від декількох метрів до сотень кілометрів.

Рушійною силою для розвитку мереж зберігання даних стало вибухове зростання об'єму ділової інформації (такий як електронна пошта, бази даних і високонавантажені файлові сервера), що вимагає високошвидкісного доступу до дискових пристроїв на блоковому рівні. Раніше на підприємстві виникали "острови" високопродуктивних дискових масивів SCSI. Кожен такий масив був виділений для конкретного застосування і видний йому як деяка кількість "віртуальних жорстких дисків". Мережа зберігання даних (Storage Area Network або SAN) дозволяє об'єднати ці "острови" засобами високошвидкісної мережі.

Основу SAN складає волоконно-оптичне з'єднання пристроїв по інтерфейсу Fibre Chanel, що забезпечує швидкість передачі інформації між об'єктами 1,2,4 або 8 Mbit/sec. Мережі зберігання допомагають підвищити ефективність використання ресурсів систем зберігання, оскільки дають можливість виділити будь-який ресурс будь-якому вузлу мережі. Розглянемо основні переваги SAN:

1) Продуктивність. Технології SAN дозволяють забезпечити високу продуктивність для завдань зберігання і передачі даних.

2) Масштабованість. Мережі зберігання даних забезпечують зручність розширення підсистеми зберігання, дозволяють легко використати придбані раніше пристрої спільно з новими облаштуваннями зберігання даних.

3) Гнучкість. Спільне використання систем зберігання даних, як правило, спрощує адміністрування і додає гнучкість, оскільки кабелі і дискові масиви не треба фізично транспортувати і перекомутувати від одного сервера до іншого. SAN дозволяє підключити нові сервери і дискові масиви до мережі без зупинки системи.

4) Централізоване завантаження. Іншою перевагою є можливість завантажувати сервера прямо з мережі зберігання. При такій конфігурації можна швидко і легко замінити збійний сервер, переконфігуруючи SAN таким чином, що сервер-заміна, завантажуватиметься з логічного диска збійного сервера.

5) Відмовостійкість. Мережі зберігання допомагають ефективніше відновлювати працездатність після збою. У SAN може входити віддалена ділянка з вторинним облаштуванням зберігання. У такому разі можна використати реплікацію — реалізовану на рівні контролерів масивів, або за допомогою спеціальних апаратних пристроїв. Попит на такі рішення значно зріс після подій 11 вересня 2001 року в США.

6) Управління. Технології SAN дозволяють забезпечити централізоване управління усією підсистемою зберігання даних.

1.3 Хмарні технології: основні поняття, завдання та тенденції розвитку.

Хмарні обчислення останнім часом привертають багато уваги. У засобах масової інформації, в інтернеті, навіть на телебачення можна зустріти захоплені матеріали, що описують прекрасні можливості, які може надати дана технологія.

Не дивлячись на те, що метафора «хмара» вже давно використовується фахівцями в галузі мережних технологій для зображення на мережних діаграмах складної обчислювальної інфраструктури (або ж Інтернету як такого), термін «Хмарні обчислення» з'явився на світ зовсім недавно. Згідно з результатами аналізу пошукової системи Google, термін «Хмарні обчислення» («Cloud Computing») почав набирати вагу в кінці 2007 — початку 2008 року, поступово витісняючи популярне в той час словосполучення «Грід-обчислення» («Grid Computing»). Судячи з заголовків новин того часу, однією з перших компаній, що

дали світові даний термін, стала компанія IBM, яка розгорнула на початок 2008 року проект «Blue Cloud» і спонсорувала Європейський проект «Joint Research Initiative for Cloud Computing».

На сьогоднішній день вже можна говорити про те, що хмарні обчислення міцно увійшли в повсякденне життя кожного користувача Інтернету (хоча багато хто про це і не підозрює). За деякими експертними оцінками, технологія хмарних обчислень може в три-п'ять разів скоротити вартість бізнес-додатків і більш ніж у п'ять разів вартість додатків для кінцевих споживачів. Але, незважаючи на загальне райдужне почуття по відношенню до хмарних технологій, досі немає єдиної думки про те, що таке «Хмарні Обчислення» і яким чином вони співвідносяться з парадигмою «Грід-обчислень». Для того, щоб розібратися в цьому, поглянемо спочатку на кілька існуючих визначень хмарних обчислень, з'ясуємо основні характеристики хмар і розглянемо, які загальні аспекти можна виявити в архітектурі хмарних рішень. Розглянемо переваги і недоліки, а також спробуємо класифікувати платформи хмарних обчислень. Наприкінці ми спробуємо зробити порівняння двох концепцій: грід-обчислень і хмарних обчислень.

Під хмарними обчисленнями (від англ. cloud computing, також використовується термін «хмарна (розсіяна) обробка даних») зазвичай розуміється надання користувачеві комп'ютерних ресурсів та потужностей у вигляді інтернет-сервісу. Таким чином, обчислювальні ресурси надаються користувачеві в «чистому» вигляді, і користувач може не знати, які комп'ютери обробляють його запити, під керуванням якої операційної системи це відбувається і т.д.

«Хмара — це великий пул легко використовуваних і легкодоступних віртуалізованих ресурсів (таких як апаратні комплекси, сервіси та ін.). Ці ресурси можуть бути динамічно перерозподілені (масштабовані) для підстроювання під змінну навантаження, забезпечуючи оптимальне використання ресурсів. Цей пул ресурсів зазвичай надається за принципом «оплата по мірі

використання». При цьому власник хмари гарантує якість обслуговування на основі певних угод з користувачем».

Всі визначення ілюструють одну просту думку: феномен хмарних обчислень об'єднує декілька різних концепцій інформаційних технологій і являє собою нову парадигму надання інформаційних ресурсів (апаратних і програмних комплексів). З боку власника обчислювальних ресурсів хмарні обчислення орієнтовані на надання інформаційних ресурсів зовнішнім користувачам. З боку користувача, хмарні обчислення — це отримання інформаційних ресурсів у вигляді послуги в зовнішнього постачальника, оплата за яку здійснюється в залежності від обсягу спожитих ресурсів відповідно до встановленого тарифу. Ключовими характеристиками хмарних обчислень є масштабованість і віртуалізація.

Масштабованість являє собою можливість динамічної настройки інформаційних ресурсів до мінливих навантажень, наприклад, до збільшення або зменшення кількості користувачів, зміні необхідної ємності сховищ даних або обчислювальної потужності. Віртуалізація, яка також розглядається як найважливіша технологія всіх хмарних систем, в основному використовується для забезпечення абстракції і інкапсуляції.

Абстракція дозволяє уніфікувати «сирі» обчислювальні, комунікаційні ресурси і сховища інформації у вигляді пулу ресурсів і вибудувати уніфікований шар ресурсів, який містить ті ж ресурси, але в абстрагованому вигляді. Вони представляються користувачам і верхнім слоям хмарних систем як віртуалізовані сервери, кластери серверів, файлові системи та СУБД.

Інкапсуляція додатків підвищує безпеку, керованість і ізольованість. Ще однією важливою особливістю хмарних платформ є інтеграція апаратних ресурсів і системного ПЗ з додатками, які надаються кінцевому користувачеві у вигляді сервісів.

Часто хмари порівнюють з мейнфреймами (mainframe), знаходячи між ними багато спільного. Принципова відмінність хмари від мейнфреймів у тому, що його обчислювальна потужність теоретично не обмежена. Друга принципова

відмінність у тому, що, простіше кажучи, термінали для мейнфреймів служили тільки для інтерактивної взаємодії користувача з запущеним на обробку завданням.

У хмарі ж термінал сам є потужним обчислювальним пристроєм, здатним не тільки накопичувати проміжну інформацію, але і безпосередньо керувати глобальною системою обчислювальних ресурсів.

Серед раніше виниклих (у 1990-х рр.) технологій обробки даних деяке поширення набули так звані *grid* обчислення. Цей напрямок спочатку розглядався як можливість використання вільних ресурсів процесорів та розвитку системи добровільної оренди обчислювальних потужностей. Ряд проектів (*GIMPS*, *distributed.net*, *SETI@home*) довели, що така модель обчислень є досить ефективною.

Ця технологія застосовується для вирішення наукових, математичних завдань, де потрібні значні обчислювальні ресурси. Відомо, що *grid*-обчислення також застосовуються для комерційних цілей. Наприклад, з їх допомогою виконуються деякі трудомісткі завдання, пов'язані з економічним прогнозуванням, аналізом сейсмічних даних, розробкою та вивченням властивостей вакцин та нових ліків. Дійсно, *grid*-обчислення та хмари мають багато подібних характеристик в архітектурі та застосовуваних принципах. Тим не менш, модель хмарних обчислень вважається сьогодні більш перспективною завдяки значно гнучкішій платформі до роботи з віддаленими обчислювальними ресурсами.

В даний час великі обчислювальні хмари складаються з тисяч серверів, розміщених у центрах обробки даних (ЦОД). Вони забезпечують ресурсами десятки тисяч додатків, які одночасно використовують мільйони користувачів.

Хмарні технології є зручним інструментом для підприємств, яким занадто дорого утримувати власні ERP, CRM або інші сервери, які потребують придбання та налаштування додаткового обладнання.

ERP (Enterprise Resource Planning — планування ресурсів підприємства) — організаційна стратегія інтеграції виробництва та операцій, управління трудовими

ресурсами, фінансового менеджменту та управління активами, орієнтована на безперервне балансування та оптимізацію ресурсів підприємства за допомогою спеціалізованого інтегрованого пакету прикладного програмного забезпечення, що забезпечує загальну модель даних та процесів для всіх сфер діяльності підприємства.

CRM (Customer Relationship Management) — система управління взаємовідносинами з клієнтами, тобто прикладне програмне забезпечення, призначене для автоматизації стратегій взаємодії із замовниками (клієнтами), зокрема, для підвищення рівня продажів, оптимізації маркетингу та покращення обслуговування клієнтів шляхом збереження інформації про клієнтів та історії взаємовідносин з ними, встановлення та покращення бізнеспроцедур та подальшого аналізу результатів.

Серед приватних користувачів широке поширення поступово отримують завдяки своїй зручності такі хмарні послуги, як, наприклад, що надаються компанією *Google* («Документи», «Календар» та ін.).

Причини зростання популярності хмарних технологій зрозумілі: можливості їх застосування дуже різноманітні та дозволяють економити як на обслуговуванні та персоналі, так і на інфраструктурі. Апаратне забезпечення може бути сильно спрощено при обробці даних та зберіганні інформації в віддалені центри даних. Всі ці проблеми майже повністю перекладаються на провайдера послуг.

До того ж такий підхід дозволяє стандартизувати програмне забезпечення, навіть якщо на комп'ютерах підприємства встановлені різні операційні системи (*Windows*, *Linux*, *MacOS* тощо). Хмарні технології полегшують забезпечення доступу до даних компанії як для клієнтів, так і для власних співробітників, що знаходяться поза офісу, але мають можливість підключитися через Інтернет.

Зрозуміло, що використання хмарних обчислень набагато зручніше. Найголовнішим недоліком, який можна відразу помітити, є повна залежність від постачальника цих послуг. Фактично підприємство (користувач) виявляється заручником провайдера сервісів та провайдера доступу до мережі Інтернет. Хоча надійність постачальників хмарних обчислень зростає, для забезпечення

надійності та безпеки даних необхідно докласти чимало зусиль, наприклад, мати дублюючі канали зв'язку, дублюючі потужності для можливості перемикання на них і, звичайно ж, подумати про доступність інформації та безпеки. Крім цього, хмарні обчислення зовсім не підходять для підприємств, що мають відношення до державної та військової таємниці. Жодна комісія не видасть сертифікат на таку систему під час роботи з інформацією, яка не підлягає розголошенню.

Як зазначено в, сучасні хмарні технології не тільки використовуються в готовому мережному та серверному обладнанні, а також поступово проникають на ринок вбудованих систем (embedded cloud) і стають причиною масштабної реструктуризації ринку. Використання вбудованих систем призводить до розміщення комп'ютерних процесорів у таких виробках, як лічильники обліку витрати ресурсів, інтелектуальні датчики, M2M-модулі, автомобілі, побутова техніка та ін. Це дозволяє керувати роботою пристроїв, збиранням даних та забезпеченням інтерактивних можливостей через підключення до комп'ютерної мережі.

Ідею підключення різних пристроїв до глобальної мережі називають Інтернетом речей (*Internet of Things — IoT*). На думку Кевіна Далласа, генерального менеджера Microsoft Windows Embedded, ідея Інтернету речей існує вже багато років, проте для неї реалізації не вистачало однієї ланки, щоб побудувати таку мережу, – хмари.

Так як кількість вбудованих комп'ютерів збільшується завдяки зниженню цін на процесори та повсюдному поширенню Інтернету, зростають також і обсяги переданих даних з подальшою їх обробкою (часто в режимі реального часу). Тому можна припустити, що найближчими роками роль Інтернету речей та хмарних обчислень буде збільшуватися.

Питання та завдання до контролю знань студентів

1) «Розподілені технології», тобто дані опрацьовуються з використанням не лише одного комп'ютера, а опрацювання розподіляється по декількох комп'ютерах, які підключені до мережі Internet — це...

- А Хмарні технології
- Б Хмарні обчислення
- В Хмарні сервіси

2) Модель зручного мережного доступу до загального фонду обчислювальних ресурсів, які можна швидко надати за умови мінімальних управлінських зусиль та взаємодії з постачальником — це

- А Хмарні технології
- Б Хмарні обчислення
- В Хмарні сервіси

3) Програмно-апаратне забезпечення, яке доступно користувачу через Інтернет у вигляді сервісу, який надає зручний інтерфейс для віддаленого доступу до обчислювальних ресурсів (програми даних)

- А Хмарні технології
- Б Хмарні обчислення
- В Хмарні сервіси

4) Сервіси, які призначені для того, щоб робити доступними користувачеві прикладне програмне забезпечення, простір для зберігання даних та обчислювальні потужності через Інтернет — це...

- А Хмарні технології
- Б Хмарні обчислення
- В Хмарні сервіси

5) Що не відноситься до загальновідомих переваг хмарних технологій?

- А Не потрібні потужні комп'ютери;
- Б Менше витрат на закупівлю програмного забезпечення і його систематичне оновлення;
- с) необмежений обсяг збереження даних;

В Недоступність з різних пристроїв і відсутня прив'язка до робочого місця.

Розробив: Ланська С.С.

Розглянуто та схвалено

на засіданні циклової комісії

програмної інженерії

Протокол № 2 від 21.09.2023 р.

Голова комісії _____ Ланська С.С.

Лекція № 2

з навчальної дисципліни «Хмарні технології»

Тема: Моделі розгортання хмарних технологій.

Класифікація систем хмарних обчислень. Компоненти хмарних додатків.

Мета заняття: надання необхідних теоретичних знань про властивості хмарних технологій, їх моделі обслуговування та послуги, що здійснюються за допомогою хмарних сервісів.

Час – 2 години

План проведення лекції

- 1) Моделі розгортання хмарних технологій.
- 2) Проблеми хмарних технологій.
- 3) Основні властивості хмарних технологій.
- 4) Моделі обслуговування хмарних технологій.
- 5) Послуги, що здійснюються за допомогою хмарних сервісів.

Література

- 1) Кононюк А.Е. Фундаментальная теория облачных технологий [Текст]: Книга 1. Общенаучные подходы формирования систем облачных технологий / А.Е. Кононюк.– К.: Освіта України, 2018 – 621 с.
- 2) Зінченко О.В., Іщеряков С.М., Прокопов С.В., Сєрих С.О., Василенко В.В. Хмарні технології. – Навчальний посібник. – К: ФОП Гуляєва В.М., 2020. –74 с.
- 3) Хмарні технології в освіті. Навчально-методичний посібник для студентів фізико-математичного факультету. – Житомир: ЖДУ, 2016. – 72 с.

Навчальні матеріали лекції

2.1 Моделі розгортання хмарних технологій

За моделлю розгортання хмари поділяють на приватні, загальнодоступні (публічні) та гібридні.

Приватні хмари — це внутрішні хмарні інфраструктури та служби підприємства. Ці хмари знаходяться в межах корпоративної мережі. Організація може керувати приватною хмарию самостійно або доручити це завдання зовнішньому підряднику. Інфраструктура може розміщуватися або у приміщеннях замовника, або у зовнішнього оператора, або частково у замовника та частково у оператора.

Ідеальний варіант приватної хмари — хмара, розгорнута на території організації, що обслуговується та контролюється її співробітниками.

Приватні хмари мають ті ж переваги, що і загальнодоступні, але з однією важливою особливістю: підприємство саме займається встановленням та підтримкою хмари. Складність та вартість створення внутрішньої хмари можуть бути дуже високі, а витрати на його експлуатацію можуть перевищувати вартість використання загальнодоступних хмар.

Слід зазначити, що приватні хмари мають переваги перед загальнодоступними: детальніший контроль над різними ресурсами хмари забезпечує компанії будь-які доступні варіанти конфігурації. Крім того, приватні хмари ідеальні, коли потрібно виконувати роботи, які не можна довірити загальнодоступній хмарі з міркувань безпеки.

Загальнодоступні (публічні) хмари — це хмарні послуги, надані постачальником. Вони знаходяться за межами корпоративної мережі. Користувачі цих хмар не мають можливості керувати цією хмарию або обслуговувати її, вся відповідальність покладено на власника цієї хмари. Постачальник хмарних послуг приймає він обов'язки по установці, управління, надання та обслуговування програмного забезпечення, інфраструктури додатків або фізичної інфраструктури. Клієнти платять лише за ресурси, які вони використовують.

Абонентом запропонованих сервісів може стати будь-яка компанія та індивідуальний юзер. Вони пропонують легкий і доступний за ціною спосіб розгортання веб-сайтів або бізнес-систем з великими можливостями масштабування, які в інших рішеннях були б недоступні.

Приклади: онлайн-сервіси Amazon EC2 та Amazon Simple Storage Service (S3), Google Apps/Docs, Salesforce.com, Microsoft Office Web.

Водночас послуги публічних хмар в основному надаються в вигляді стандартних конфігурацій, тобто виходячи з умов найбільш найпоширеніших випадків використання. Це означає, що у користувача залишається менше можливостей щодо вибору конфігурації проти системами, у яких ресурсами управляє сам споживач. Слід також мати на увазі, що оскільки споживачі слабо контролюють інфраструктуру, процеси, що вимагають суворих заходів безпеки та відповідності нормативним вимогам, не завжди підходять для реалізації у загальнодоступній хмарі.

Гібридні хмари є поєднанням загальнодоступних і приватних хмар. Зазвичай вони створюються підприємством, а обов'язки з управління ними розподіляються між підприємством та постачальником загальнодоступної хмари. Гібридна хмара надає послуги, частина яких належить до загальнодоступних, а частина — до приватних. Зазвичай такий тип хмар використовується, коли організація має сезонні періоди активності. Іншими словами, як тільки внутрішня ІТ-інфраструктура не справляється з поточними завданнями, частина потужностей перекидається на публічну хмару (наприклад, великі обсяги статистичної інформації, які в необробленому вигляді не становлять цінності для підприємства), а також для надання доступу користувачам до ресурсів підприємства (до приватної хмари) через публічну хмару. Добре продумана гібридна хмара може обслуговувати як такі, що вимагають безпеки критично важливі процеси, такі як отримання платежів від клієнтів, і більш другорядні.

Основним недоліком цього типу хмари є складність ефективного створення подібних рішень та управління ними.

Необхідно отримувати послуги з різних джерел та організувати їх так, ніби це було єдине джерело. Взаємодія між приватним та загальнодоступним компонентами може ще більше ускладнити рішення. Оскільки це відносно нова архітектурна концепція у сфері хмарних обчислень, для цієї моделі з'являються

все нові та нові практичні рекомендації та інструменти, та її широке поширення може затягнутися доти, доки вона не буде краще вивчена.

На думку Тома Біттмана, віце-президента та провідного аналітика американської дослідницької та консалтингової компанії "Gartner", серед перерахованих вище трьох моделей розгортання хмар найактуальнішою для бізнесу в даний момент є приватні хмари. Біттман виділив п'ять основних моментів, які допомагають отримати більш точне уявлення про влаштування приватної хмари.

Хмара — це лише віртуалізація. Хоча віртуалізація серверів та інфраструктури складає важливий фундамент приватних хмарних обчислень, самі по собі віртуалізація та управління віртуалізованим середовищем ще є приватною хмарию.

Віртуалізація дозволяє краще структурувати, об'єднувати в пул та динамічно надавати ресурси інфраструктури: сервери, десктопи, ємності для зберігання, мережне обладнання, сполучне ПЗ і т.д. Але щоб середовище технічно могло вважатися хмарним, потрібні ще й інші складові, такі як віртуальні машини, операційні системи чи контейнери сполучного ПЗ, високостійкі операційні системи, ПЗ grid-обчислень, ПЗ для абстрагування ресурсів зберігання, засоби масштабування та кластеризації.

Термін «приватна хмара» на відміну від загальнодоступної або гібридного відноситься до ресурсів, які використовуються єдиною організацією, або означає, що хмарні ресурси організації повністю ізольовані у хмарі від інших.

Хмара — необов'язкове джерело економії. Одне з головних помилок полягає в тому, що хмара заощаджуватиме гроші. Економія можлива, але не обов'язковий атрибут. Приватна хмара дозволяє більш ефективно перерозподіляти ресурси, щоб задовольнити корпоративні вимоги, та здатне зменшити капітальні витрати на обладнання. Але приватна хмара вимагає інвестицій в автоматизацію, і лише економія може не окупати всієї вартості. Отже, зниження витрат не є головним перевагою цієї моделі. З цього погляду головним стимулом до впровадження хмарної моделі має бути не економія, а швидкість виходу на ринок,

можливість швидкої адаптації та динамічного масштабування відповідно до попиту, які дозволяють підвищити швидкість запровадження нових сервісів.

Приватна хмара не завжди впроваджена у замовника. Приватна хмара означає конфіденційність, а не конкретне розташування, володіння ресурсами чи самостійне управління. Багато постачальники пропонують нелокальні приватні хмари, тобто виділяють ресурси єдиному замовнику, виключаючи спільне використання одного пулу кількома клієнтами. «Хмара називається приватним за його приватністю, а не за тим, де воно розгорнуто, хто ним володіє та несе відповідальність за управління», – наголошує Біттман. Деякі, наприклад, можуть свої ЦОД розміщувати у хостинг-провайдерів або об'єднувати в пул ресурси різних замовників, але ізолювати їх один від одного за допомогою віртуальної приватної мережі (Virtual Private Network – VPN) та інших подібних технологій.

Приватна хмара (як і публічна хмара) — це не лише інфраструктурні послуги. Серверна віртуалізація – велика тенденція і тому потужний двигун приватних хмарних обчислень. Але приватна хмара не зводиться лише до *інфраструктури як послугу (IaaS)*. Наприклад, для розробки та тестування нового ПЗ високорівнева *платформа як послуга (PaaS)* має більше сенсу, ніж просто надання віртуальних машин.

Сьогодні найшвидше зростаючий сегмент хмарних обчислень — це *IaaS*. Вона надає найнижчорівневі ресурси ЦОД в простий для використання формі, але не змінює фундаментально принципи роботи. Щоб створити нові програми, спочатку призначені для хмари та надають абсолютно нові послуги, які можуть дуже відрізнитись від того, що давали колишні програми, розробникам зручніше використовувати *PaaS*.

Приватна хмара може перестати бути приватною. З однією сторони, приватна хмара надає переваги хмарі: швидкість перебудови, масштабованість та ефективність, позбавляє від деяких загроз безпеці, потенційних і реальних, які характерні для загальнодоступних хмар. З іншого боку, із часом рівень обслуговування, безпека та контроль дотримання вимог у загальнодоступних хмарних сервісах безумовно будуть підвищуватись. Тому деякі приватні хмари,

можливо, цілком перейдуть до категорії загальнодоступних. Більшість же сервісів приватної хмари, швидше за все, еволюціонуватимуть у гібридні хмарні послуги, розширюючи доступні можливості за рахунок використання загальнодоступних хмарних послуг та інших сторонніх ресурсів.

2.2 Проблеми хмарних технологій.

За останні кілька років хмарні технології переросли з перспективної бізнес-концепції в один із найбільш швидкозростаючих сегментів ІТ-галузі. Зараз компанії все частіше усвідомлюють, що використовуючи хмару, вони можуть отримати швидкий доступ до найкращих у своїй галузі бізнес додатків або різко збільшити свої ресурси, і все це має незначну вартість. Але в міру того, як дедалі більше інформації про приватних осіб та компаній розміщується в хмарі, починає зростати занепокоєння з приводу того, наскільки безпечним є це середовище.

Безпека. Де ваші дані надійніше захищені, на вашому локальному жорсткому диску або на віддалених серверах? Одні стверджують, що дані клієнтів є більш безпечними при внутрішньому управлінні, тоді як інші стверджують, що постачальники хмарних послуг зацікавлені підтримувати довіру і використовують більш високий рівень безпеки. Однак не слід забувати, що у хмарі дані користувачів будуть розподілені між окремими комп'ютерами, незалежно від того, де в кінцевому підсумку зберігається базовий файл. Загрози, такі як злом веб-сайтів та вірусні атаки, є найбільшими проблемами безпеки даних хмарних технологій, тому що хакери можуть вторгнутися практично на будь-який сервер. Перш ніж використовувати технологію хмарних обчислень, потрібно подумати про ці речі. Після передачі важливих даних третій стороні переконайтеся, що у вас є хмарна система безпеки та управління. Згідно з опитуванням Crowd Research Partners, 9 з 10 експертів з питань кібербезпеки стурбовані питаннями безпеки у хмарах. Крім того, їх турбує порушення конфіденційності даних.

Недостатність ресурсів та досвіду. Це одна з ключових проблем процесу хмарної міграції. Відповідно до звіту RightScale, майже 75% респондентів відзначили це як проблему, тоді як 23% сказали, що це серйозна проблема.

Незважаючи на те, що багато працівників ІТ вдосконалюють свої знання, роботодавцям складно знайти співробітників з необхідним досвідом. Деякі організації також розраховують на рішення проблеми переходу до хмарних обчислень шляхом працевлаштування більшої кількості працівників, які мають сертифікати або навички в галузі хмарних технологій. Фахівці також пропонують забезпечити навчання нинішніх працівників, щоб зробити їх більш продуктивними та здатними використовувати найсучасніші технології.

Повне управління ІТ-послугами. ІТ працівники не мають повного контролю над обладнанням, доставкою інфраструктури та її функціонуванням у хмарному світі. Щоб викоринити різні невизначеності та труднощі при переході до хмари, ІТ спеціалісти повинні впровадити звичайні процедури контролю та управління. Основні ІТ компанії відіграють все більшу роль у посередництві, постачанні та контролі над хмарними послугами.

Зайві витрати у хмарі. Відповідно до звіту RightScale, організації втрачають майже 30% грошей, які вони інвестують у хмару. Компанії допускають кілька помилок, які можуть збільшити їх витрати. Іноді ІТспеціалісти, такі як розробники, вмикають хмарний сервіс, який передбачається використовувати деякий час, і забувають його знову вимкнути. А деяким компаніям заважають приховані пакети з витратами на хмару, які не використовуються. Допомогти організаціям можуть кілька технічних рішень: автоматизація, концепції для управління хмарними витратами, безсерверні послуги, функції автоматичного масштабування та численні інструменти управління, що надаються постачальниками хмарних послуг.

Робота з багатохмарним середовищем. В наш час більшість компаній працюють не лише з однією хмарию. Згідно зі звітом RightScale, майже 84% компаній дотримуються мультихмарної стратегії, а 58% вже мають свою гібридну тактику хмар, поєднану із використанням державних та приватних хмар. Щоб перемогти цей виклик, професіонали використовують навчання персоналу, активне управління відносинами з продавцями та проведення досліджень.

Хмарна міграція. Хоча випуск нового додатка в хмарі здійснити досить легко, перенесення існуючого додатка в середовище хмарних обчислень є більш складною процедурою. Згідно з повідомленням, 62% ІТ-спеціалістів заявили, що їхні проекти хмарної міграції були складнішими, ніж вони очікували. Поряд з цим 64% міграційних проектів зайняли більше часу, ніж передбачалося, а 55% вийшли за рамки свого бюджету. Зокрема, деякі організації, які переносять свої програми в хмару, повідомляли про простої під час міграції, проблеми із синхронізацією даних перед відключенням, проблеми з якісними інструментами міграції, повільну міграцію даних, складності з налаштуваннями питань безпеки та трудомістке усунення несправностей. Для вирішення цих питань необхідно збільшити виплати ІТ-працівникам, найняти власних спеціалістів, провести більше тестів перед міграцією.

Проблеми монополій. В даний час декілька провідних постачальників хмарних послуг, наприклад, Google Cloud Platform, Microsoft Azure, Amazon Web Services та IBM Cloud, панують над публічним хмарним ринком. Майже 90% опитаних ІТ-працівників висловили високий та помірний рівень занепокоєння щодо цієї проблеми. Зростаюча потужність високомасштабних постачальників послуг IaaS створює як шанси, так і проблеми для деяких учасників ринку та кінцевих користувачів. Великі постачальники послуг IaaS можуть набути небажаного впливу на клієнтів та ринок. Що стосується роботи в багатохмарному середовищі, необхідно забезпечити простіший спосіб передачі програм та даних через хмарні провайдери IaaS. Перед тим, як компанії приймуть певну хмарну послугу, вони повинні подумати, наскільки прямою буде передача цих даних в іншу хмару в майбутньому.

Конфіденційність. На відміну від традиційної обчислювальної моделі, хмарні обчислення використовують технологію віртуальних обчислень, особисті дані користувачів можуть розкидатися по різних віртуальних центрах обробки даних, а не залишатися в одному і тому ж фізичному місці, навіть передаватися через національні кордони. В цей час захист конфіденційності даних зіткнеться з

суперечками різних правових систем. З іншого боку, користувачі можуть отримувати приховану інформацію під час доступу до послуг хмарних обчислень.

Надійність. Сервери в хмарі мають такі ж проблеми, як і власні сервери-резиденти користувачів. Хмарні сервери також чутливі до простоїв та уповільнень, різниця полягає в тому, що користувачі мають більшу залежність від постачальника хмарних послуг. Існує велика різниця в моделях обслуговування, і коли клієнт обирає конкретного постачальника, то можете потрапити в залежність і тим самим створити потенційний ризик для бізнесу.

Відповідність. Ця проблема існує для всіх, хто користується хмарними сховищами чи службами резервного копіювання. Кожного разу, коли організація передає дані зі свого внутрішнього сховища в хмару, вона повинна притримуватись законів та правил галузі, які регулюються Загальним регламентом захисту даних (GDPR). Багато організацій потребують спеціаліста із захисту даних, який може забезпечити конфіденційність інформації відповідно до вимог закону.

Обчислення. Користувачі не можуть фізично володіти сховищем даних, залишаючи зберігання та контроль даних у руках хмарних постачальників.

Довготривала життєздатність. Слід бути впевненим, що дані, які ви вводите в хмару, ніколи не будуть втрачені, навіть якщо ваш постачальник хмарних обчислень буде ліквідований або його поглине більша компанія.

Хмарна інтеграція. Нарешті, кілька компаній, особливо ті, що мають гібридне хмарне середовище, повідомляють про проблеми, пов'язані з наявністю їхніх локальних додатків та інструментів та загальнодоступної хмари для спільної роботи. Згідно з опитуванням, 62% респондентів назвали інтеграцію застарілих систем найбільшим викликом у мультихмарі. Подібним чином, у звіті Software One щодо вартості хмар, 39% опитаних зазначили, що інтеграція застарілих систем є однією із найбільших проблем, яку вони відчули при використанні хмари. Поєднання нових хмарних програм та застарілих систем потребує ресурсів, досвіду та часу.

2.3 Основні властивості хмарних технологій

Національний Інститут стандартів та технологій NIST (National Institute of Standards and Technology, USA) у своєму документі “The NIST Definition of Cloud Computing” визначає наступні характеристики хмар:

- можливість високого ступеня автоматизованого самообслуговування системи із боку провайдера;
- наявність системи Broad Network Access;
- зосередженість ресурсів на окремих майданчиках для їх ефективного розподілу;
- швидка масштабованість (ресурси можуть необмежено виділятися та вивільнятися з великою швидкістю в залежності від потреб);
- керований сервіс (система керування хмарою автоматично контролює та оптимізує виділення ресурсів).

Самообслуговування на вимогу (On-demand self-service). У споживача є можливість отримати доступ до наданих обчислювальним ресурсам в односторонньому порядку принаймні потреби, автоматично, без необхідності взаємодії зі співробітниками кожного постачальника послуг.

Широкий мережний доступ (Broad network access). Надані обчислювальні ресурси доступні через мережу через стандартні механізми для різних платформ, тонких та товстих клієнтів (мобільних телефонів, планшетів, ноутбуків, робочих станцій та т.п.).

Об'єднання ресурсів у пули (Resource pooling). Обчислювальні ресурси провайдера об'єднуються в пули для обслуговування багатьох споживачів за багатоорендною (multi-tenant) моделлю. Пули включають різні фізичні і віртуальні ресурси, які можуть бути динамічно призначені та перепризначені в відповідно до споживчих запитів. Немає необхідності в тому, щоб споживач знав точне розташування ресурсів, проте можна вказати їхнє місцезнаходження на вищому рівні абстракції (наприклад, країна, регіон чи ЦОД). Прикладами такого роду ресурсів можуть бути системи зберігання, обчислювальні потужності, пам'ять, пропускну спроможність мережі.

Миттєва еластичність (Rapid elasticity). Ресурси можуть бути легко виділені та звільнені, в деяких випадках автоматично, для швидкого масштабування пропорційно попиту. Для споживача можливості надання ресурсів бачаться як необмежені, тобто вони можуть бути присвоєні в будь-якій кількості та у будь-який час.

Вимірюваний сервіс (Measured service). Хмарні системи автоматично керують та оптимізують ресурси за допомогою засобів вимірювання, реалізованих на рівні абстракції стосовно різноманітних сервісів (наприклад, управління зовнішньою пам'яттю, обробкою, смугою пропускання чи активними користувальницькими сесіями). Використані ресурси можна відстежувати і контролювати, що забезпечує прозорість як для постачальника, так і для споживача, що використовує сервіс.

2.4 Моделі обслуговування хмарних технологій

Існує три основні сервісні моделі хмарних обчислень - Інфраструктура як послуга (IaaS), Платформа як послуга (PaaS) та Програмне забезпечення як послуга (SaaS). Є чіткі відмінності між цими сервісами і тим, що вони можуть запропонувати користувачам з точки зору зберігання та об'єднання ресурсів. Ці сервіси ще називають шарами хмари і вони також можуть взаємодіяти між собою, формуючи одну всеосяжну модель хмарних обчислень, яка представлена на рисунку 2.1.



Рисунок 2.1 – Піраміда хмарних сервісів

Інфраструктура як послуга (IaaS). Як основа піраміди хмарних обчислень, IaaS є найбільш повним та гнучким видом хмарних сервісів. Це найпоширеніша модель обслуговування хмарних технологій, оскільки вона пропонує фундаментальну інфраструктуру віртуальних серверів, мереж, операційних систем та накопичувачів даних. По суті IaaS забезпечує повністю віртуалізовану обчислювальну інфраструктуру, яка забезпечується та управляється через Інтернет. Постачальник послуг IaaS управляє фізичною ланкою інфраструктури (сервери, простір для зберігання даних тощо), яка розташована в центрі обробки даних, але дозволяє клієнтам повністю налаштувати ці віртуалізовані ресурси відповідно до своїх конкретних потреб. За допомогою IaaS клієнт може купувати, встановлювати, налаштовувати та керувати будь-яким програмним забезпеченням, включаючи такі речі, як операційні системи, проміжне програмне забезпечення, програми, бізнес-аналітика та засоби розробки. Компанії платять лише за ту інфраструктуру, яку вони використовують, що дозволяє їм масштабувати свої обчислювальні потреби без необхідності нарощувати додаткові потужності. IaaS виключає капітальні витрати на створення власної інфраструктури. Це чудовий варіант для невеликих компаній та стартапів, які не мають ресурсів для придбання обладнання та програмного забезпечення, необхідного для створення власної мережі. Повсякденний тягар управління обчислювальною інфраструктурою також беруть на себе ІТ-відділи. Оскільки постачальник послуг IaaS постійно оновлює свою систему за допомогою найновішого програмного забезпечення, стає простіше запускати нові програми. IaaS пропонує найновіші засоби захисту та, як правило, пропонує такі послуги, як аварійне відновлення.

Приклади IaaS: Microsoft Azure, Amazon Web Services (AWS), Cisco Metacloud, Google Compute Engine (GCE).

РaaS (Платформа як послуга). РaaS розташований трохи вище у піраміди хмарних обчислень. У той час як IaaS постачає всі інструменти, доступні через хмару, і залишає замовникам можливість створювати те, що відповідає їхнім

потребам, PaaS трохи більш спеціалізований. Замість чистої інфраструктури, PaaS забезпечує структуру, необхідну для побудови, тестування, розгортання, управління та оновлення програмних продуктів. Сервіс використовує ту саму базову інфраструктуру, що і IaaS. PaaS надзвичайно корисний для будь-якої компанії, яка розробляє програмне забезпечення та веб-додатки. Постачальники хмарних обчислень розгортають інфраструктуру та програмне забезпечення, але підприємства можуть розробляти та запускати і власні програми. Структура сервісу PaaS наведена на рисунку 2.2.

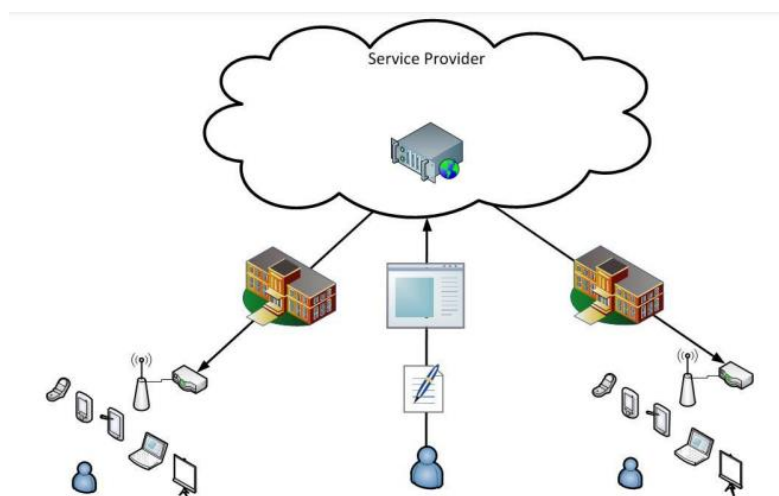


Рисунок 2.2 – Структура хмарного сервісу PaaS

Багато інструментів, необхідних для розробки для платформ (комп'ютери, мобільні пристрої, браузері тощо), є досить дорогими. Використовуючи PaaS, клієнти можуть отримати доступ до інструментів розробки, коли вони потребують, без необхідності їх купувати. Оскільки платформа доступна через Інтернет, усі команди віддалених ІТ-працівників можуть отримати доступ до тих самих ресурсів, щоб пришвидшити розробку продукту. Рішення PaaS є масштабованими та ідеально підходить для бізнессередовищ, де кілька розробників працюють над одним проектом. Це також зручно для ситуацій, коли потрібно використовувати існуюче джерело даних (наприклад, інструмент CRM).

Приклади PaaS: AWS Elastic Beanstalk, Apache Stratos, Google App Engine, Microsoft Azure.

Програмне забезпечення як послуга (SaaS). Для більшості людей SaaS є найбільш звичною формою хмарних обчислень. SaaS — це повністю розроблене програмне рішення, готове до придбання та використання через Інтернет і розташоване на вершині піраміди хмарних технологій. Постачальник SaaS надає програму «в аренду», управляє інфраструктурою, операційними системами, проміжним програмним забезпеченням та необхідними даними, гарантуючи доступність програмного забезпечення, коли і де це потрібно клієнтам. Багато додатків SaaS працюють безпосередньо через веб-браузери, усуваючи необхідність завантажувати чи встановлювати додаткові програми. Це значно зменшує проблеми з управлінням програмним забезпеченням і дозволяє компаніям оптимізувати свою діяльність за допомогою гібридних та багатохмарних рішень.

Додатки SaaS дозволяють компаніям швидко розпочинати роботу, а також швидко масштабувати операції. Не потрібно купувати або розгортати апаратне та програмне забезпечення, яке використовується для надання бізнес-послуг. Навіть складні додатки, такі як програми управління взаємовідносинами з клієнтами (CRM) або програми планування корпоративних ресурсів (ERP), можуть бути легко доступні найменшим організаціям.

Приклади SaaS: Microsoft Office 365, Salesforce, Cisco WebEx, Google Apps. Для наочності, узагальнимо сервіси архітектури «хмара» в одну схему, представлену на рисунку 2.3, на якій наведено класифікацію сервісів за типом послуг.

В таблиці 2.1 представлені провідні компанії, що надають послуги у сфері хмарних обчислень. Хмарні служби мають кілька загальних атрибутів:

1) Віртуалізація — хмарні обчислення широко використовують віртуалізацію серверів та сховищ для швидкого розподілу/перерозподілу ресурсів.

2) Багаторічна оренда — ресурси об'єднуються та розподіляються між багатьма користувачами, щоб отримати економію від масштабу.

3) Мережевий доступ — доступ до ресурсів здійснюється через веббраузер за допомогою різноманітних мережевих пристроїв (комп'ютер, планшет, смартфон).

4) За запитом — ресурси надаються з онлайн-каталогу заздалегідь визначених конфігурацій

5) Масштабування — ресурси можуть автоматично збільшуватися або зменшуватися в залежності від потреб замовника.

6) Визначення/повернення платежів — використання ресурсів відстежується та виставляється рахунок на основі домовленостей про обслуговування.

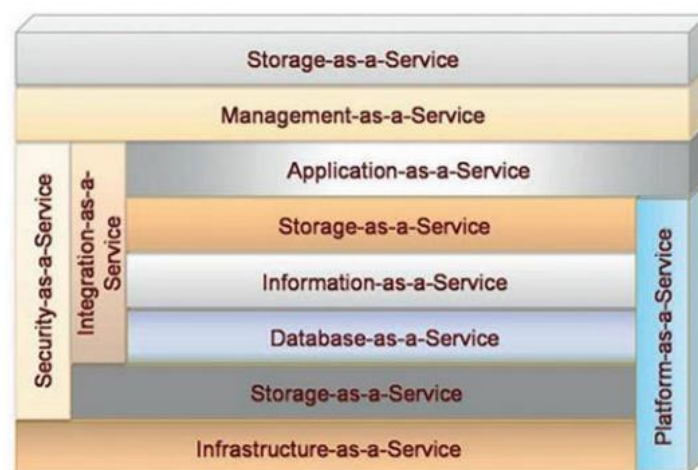


Рисунок 2.3 – Архітектура хмарних сервісів

Хмарні послуги пропонують численні переваги, які включають:

- швидше впровадження;
- доступ у будь-якому місці до програм та послуг;
- швидка масштабованість для задоволення попиту;
- зниження вартості інфраструктури, економія на енергії та обладнанні;
- більша продуктивність ІТ-персоналу та організації в цілому;
- покращена безпека та захист інформаційних активів.

Таблиця 2.1 – Послуги, що надаються хмарними постачальниками

Name of Company	IaaS	PaaS	SaaS
AWS	Amazon AC2	Amazon Web Services	Amazon Web Services
Microsoft	Microsoft Private Cloud	Amazon Azure	Amazon Office 365
Google	-	Google App Engine (Python, Java and many)	Google Applications
IBM	Smart Cloud Enterprise	Smart Cloud Applications Services	SaaS Products
Adobe	-	Adobe Creative Cloud	Acrobat, Flashplayer, etc.

2.5 Послуги, що здійснюються за допомогою хмарних сервісів.

Функція як послуга (FaaS). Часто відома як безсерверні обчислення, FaaS дозволяє клієнтам швидко реагувати на зміни без необхідності розподіляти ресурси заздалегідь. Хмарний провайдер обробляє інфраструктуру, дозволяючи клієнту зосередитися на розгортанні коду програми. Функції масштабуються автоматично, що робить їх ідеально пристосованими для динамічних навантажень. Клієнти платять лише за ресурси, якими вони користуються, що робить FaaS найоптимальнішою формою хмарних обчислень. Більшість програм FaaS досить прості і можуть бути розгорнуті дуже швидко. Клієнту хмари просто потрібно завантажити відповідний код функції та повідомити платформі, як ресурси необхідні під час її виконання. Нові види функцій можна масштабувати за запитом, і коли функція не використовується, вона не споживає жодних ресурсів. Основним недоліком FaaS є час виконання. Оскільки функції повинні надавати ресурси при кожному їх запуску, можуть бути незначні відставання в продуктивності, якщо програма вимагає великої обчислювальної потужності або виконується під час пікового навантаження. Більшість служб FaaS доступні через великих хмарних провайдерів, таких як AWS та Azure, що може призвести до блокування постачальника.

Приклади FaaS: AWS, функції Azure.

Storage as a Service (зберігання як сервіс) — це бізнес-модель, при якій велика компанія здає в оренду меншій компанії або окремій особі простір в своїй інфраструктурі сховища. На підприємстві постачальники SaaS орієнтуються на додаткові програми зберігання даних, просуваючи SaaS як зручний спосіб управління резервними копіями. Ключова перевага SaaS полягає в економії витрат — на персоналі, апаратному забезпеченні та фізичному просторі зберігання. Наприклад, замість того, щоб підтримувати велику кількість копій файлів і організовувати їх зберігання за межами сайту, мережевий адміністратор, який використовує SaaS для резервного копіювання, може вказати, для яких даних в мережі слід створювати резервні копії та як часто їх слід створювати. Його компанія підпише угоду про рівень обслуговування (SLA), згідно з якою постачальник SaaS погоджується надати простір для зберігання на основі ціни за гігабайт, що зберігається, і ціни за передачу даних, і дані компанії будуть автоматично передаватися в зазначений час через власну глобальну мережу (WAN) провайдера сховища або Інтернет. Якщо дані компанії коли-небудь пошкодяться або загубляться, адміністратор мережі може зв'язатися з постачальником SaaS та запросити копію даних.

DBaaS (також відомий як сервіс керованих баз даних) — це одна з найбільш затребуваних технологій в галузі управління інформаційними ресурсами, хмарний підхід до зберігання та управління структурованими даними. DBaaS дозволяє користувачам отримувати доступ до хмарної системи баз даних і користуватися нею без придбання та налаштування власного обладнання, встановлення власного програмного забезпечення для баз даних чи управління ними. Суть концепції DBaaS в тому, що користувачеві не потрібно встановлювати і підтримувати базу даних, йому досить зробити запит і отримати по ньому базу даних. Хмарний провайдер піклується про все — від періодичного оновлення і резервного копіювання до забезпечення того, щоб система баз даних залишалася доступною та безпечною цілодобово.

Ринок DBaaS та хмарних баз даних є одним із найбільш швидкозростаючих ринків SaaS, який, як очікується, зросте до 320 млрд. доларів США до 2025 року.

Постачальники баз даних та сховищ даних приєдналися до відомих хмарних постачальників, пропонуючи своє програмне забезпечення, що дозволяє клієнтам використовувати багато переваг хмарних обчислень для зберігання даних, пошуку та доступу до своїх додатків. Користувач отримує з інформаційних ресурсів необхідну інформацію, зібрану згідно з параметрами свого запиту, і може працювати з нею як зі звичайною базою даних: зберігати, редагувати, відправляти іншим користувачам, об'єднувати з іншими базами і інше. Як приклад можна назвати Amazon Web Services.

Інформація як послуга (IAS) дозволяє будь-яким програмам отримувати доступ до всіх типів інформації через інтерфейс програми. Послуга, що базується на всьому або «що-небудь як послуга», або «все як послуга», є фундаментальною конфігурацією IAS. Таким чином, IAS можна академічно підключити до моделі хмарної служби, яка реалізована в технології хмарних обчислень.

Модель *BPaaS (business process as a service)* — це надання послуг з вирішення бізнес-завдань, коли в основі рішення лежать хмарні технології. Gartner визначає бізнес-процес як послугу (BPaaS) як надання послуг аутсорсингу бізнес-процесів (BPO), які постачаються із хмари та створюються для багатонаціональних послуг. Як хмарний сервіс модель BPaaS доступна через Інтернет-технології. Можливості для роботи за цією моделлю виникають, якщо у компанії-замовника є потреба в автоматизації повторюваних типових робіт, але немає закріплених менеджерів. Один і той же сервіс може використовуватися декількома клієнтами BPaaS-провайдера. Оплата такого аутсорсингу проводиться за фіксованою на період ціною або за фактом споживання послуги.

Інтеграція як послуга (іноді скорочується як IaaS) — це модель надання хмарних послуг для інтеграції. Integration-as-a-Service забезпечує інтеграційне рішення, яке забезпечує зв'язок із серверними системами, джерелами, файлами та операційними програмами завдяки реалізації чітко визначених інтерфейсів, веб-сервісів та дзвінків між додатками та джерелами даних. Це забезпечує користувачам більш вільно пов'язане середовище, захищене від складної взаємозалежності. Модель інтеграції як послуги надає можливість інтеграції в

хмарі, що дозволяє обмінюватися даними між системами, а також сторонніми постачальниками в режимі реального часу.

Спектр послуги *безпека як сервіс (SECaaS)* великий, наприклад, це: постійний моніторинг безпеки інформаційних систем, запобігання втрати даних, безперервність діяльності і післяаварійне відновлення, безпека електронної пошти, антивірусне управління, фільтрація спаму, управління ідентифікацією і доступом, аудит безпеки, безпека веб-додатків, сканування інфраструктури на уразливості, тестування на проникнення, контроль правил мережевого доступу, шифрування і управління ключами шифрування, контроль своєчасних оновлень ПО і ОС.

Питання та завдання до контролю знань студентів

1) «Цей тип хмарних сервісів використовує мультиабонентську архітектуру: надає через браузер доступ до окремого додатку тисячам клієнтів». До якого терміну стосується дане визначення?

- A SaaS (Software As a Service);
- Б IaaS (Infrastructure as a Service);
- В DaaS (Desktop as a Service);
- Г Hybrid cloud.

2) На якій платформі розміщена «хмара» Adobe Creative Cloud?

- A Microsoft Azure;
- Б Chromium;
- В Sun Cloud;
- Г Amazon Web Services.

3) Що означає аббревіатура AWS (в області хмарних технологій)?

- A Amazon Web Services;
- Б Application Windows Service;
- В Action Wide Soft;
- Г Android Webview Support.

4) Хмарні сервіси — це:

А Сайти, на яких розміщується інформація щодо хмарних технологій;

Б Новітній вид мережеских послуг, які дозволяють інформаційними засобами віртуального середовища розширити програмно-технічні ресурси комп'ютерного пристрою користувача;

В Хмарне середовище, яке слугує виключно для бізнес-проектів;

Г Простий тип «хмарини» для передачі інформації.

Розробив: Ланська С.С.

Розглянуто та схвалено

на засіданні циклової комісії

програмної інженерії

Протокол № 2 від 21.09.2023 р.

Голова комісії _____ Ланська С.С.

Лекція № 3

з навчальної дисципліни «Хмарні технології»

Тема: огляд рішень провідних вендорів. Популярні хмарні сервіси для офісу та навчання

Мета заняття: розглянути рішення та приклади хмарних сервісів провідних вендорів — Microsoft, Amazon, Google та інших.

Час – 2 години

План проведення лекції

- 1) Огляд рішень провідних вендорів.
- 2) Інші постачальники хмарних технологій.
- 3) Популярні хмарні сервіси для офісу та навчання.

Література

- 1) Кононюк А.Е. Фундаментальная теория облачных технологий [Текст]: Книга 1. Общенаучные подходы формирования систем облачных технологий / А.Е. Кононюк. – К.: Освіта України, 2018 – 621 с.
- 2) Зінченко О.В., Іщеряков С.М., Прокопов С.В., Серих С.О., Василенко В.В. Хмарні технології. – Навчальний посібник. – К: ФОП Гуляєва В.М., 2020. – 74 с.
- 3) Войтович Н.В., Найдьонова А.В. Використання хмарних технологій Google та сервісів web 2.0 в освітньому процесі. Методичні рекомендації. – Дніпро: ДПТНЗ «Дніпровський центр ПТОТС», 2017 – 113 с.

Навчальні матеріали лекції

3.1 Огляд рішень провідних вендорів.

На сьогоднішній день існує безліч постачальників хмарних платформ, сховищ і ПЗ. У зв'язку з цим робляться спроби якимось чином порівняти їх між собою. В Інтернеті можна знайти багато оглядів на цю тему. Але через різноманітність послуг, що надаються, розібратися в тому, хто з них кращий,

досить важко. Тому необхідно вибрати кілька основних показників, які можуть допомогти при порівнянні, наприклад, ємність, захищеність, простота, функціональність, доступність та ін.

При цьому, безперечно, кожен користувач вибирає те чи інше рішення в залежності від необхідності.

Порівняння платформ AMAZON, GOOGLE та MICROSOFT.

Зараз основними постачальниками хмарної інфраструктури є Amazon, Google і Microsoft. Кожна з компаній має цілу лінійку наданих послуг. У цих матеріалах описані лише деякі з них, найпопулярніші. Також не обговорюється питання, до якої саме моделі належить та чи інша послуга та які вендори надають лише публічні хмари, а які можуть брати участь у створенні приватних хмар.

Google.

<https://developers.google.com>

Google Drive — хмарне сховище даних, що належить компанії Google, що дозволяє користувачам зберігати свої дані на серверах у хмарі та ділитися ними з іншими користувачами в Інтернеті. Google Drive відрізняється лаконічним інтерфейсом та пропонує встановити зручні програмні клієнти для смартфонів та планшетів на базі операційної системи Android, ПК та ноутбуків під керуванням операційної системи Windows або MacOS, мобільних пристроїв iPhone та iPad. У майбутньому очікується тісніша інтеграція сховища з операційною системою Chrome OS та підтримка Linux. Кожен користувач Google Drive отримує до 15 Гб вільного простору на всі сервіси Google (у тому числі Gmail і Photos). При цьому він сам може вирішити скільки місця виділити під пошту і який обсяг залишити під важливі файли. Працювати з файлами в Google Drive можна прямо у браузері. Google Drive можна перетворити на окрему папку в документах смартфона, планшета або ПК, і її вміст буде синхронізуватися автоматично.

Google Docs — безкоштовний онлайн-офіс, що включає текстовий, табличний процесори та сервіс для створення презентацій, а також інтернет-сервіс хмарного зберігання файлів з функціями файлообміну. Дозволяє створювати та

редагувати стандартні документи, таблиці та презентації, а також підтримує функції спільної роботи над ними.

Google App Engine — сервіс хостингу сайтів та web-додатків на серверах Google. Безкоштовно надається до 1 Гб дискового простору, 10 Гб вхідного трафіку на день, 10 Гб вихідного трафіку на день, 200 мільйонів гігациклів CPU на день та 2 000 операцій відправлення електронної пошти на день. Програми, що розгортаються на основі App Engine, повинні бути написані на Python, Java або Go. Пропонується набір API для сервісів сховища datastore API (BigTable) облікових записів Google, набір API для завантаження даних по URL, електронної пошти тощо.

Платформа Google конкурує з аналогічними сервісами від Amazon, які надають можливість розміщувати файли та веб-програми, використовуючи свою інфраструктуру. На відміну від багатьох звичайних розміщень додатків на віртуальних машинах, таких як Amazon EC2, платформа App Engine тісно інтегрована з додатками та накладає на розробників деякі обмеження.

Google Cloud Storage — сервіс хостингу файлів, що базується на IaaS. Усі файли, які записуються або перезаписуються на сервери, автоматично шифруються за алгоритмом AES-128. Є конкурентом продукту Amazon S3.

Amazon.

<http://aws.amazon.com>

Amazon Simple Storage Service (Amazon S3) — онлайнова веб-служба, пропонована Amazon Web Services, що надає можливість для зберігання та отримання будь-якого обсягу даних, у будь-який час з будь-якої точки мережі, так званий файловий хостинг. У березні 2012 року компанія Nasuni провела досвід, протягом якого по черзі передавала масивний обсяг даних (12 Тб) з одного хмарного сервісу до іншого. В експерименті брали участь найбільш рейтингові хмари: Amazon S3, Windows Azure та Rackspace. На подив дослідників, швидкість передачі даних сильно відрізнялася в залежності від того, яка хмара приймала дані. Найкращий показник швидкості запису даних виявився у Amazon S3,

передача даних із двох інших сервісів займала всього 4–5 годин, тоді як передача даних у Rackspace зайняла трохи менше тижня, а Windows Azure – 40 годин.

Amazon Elastic Compute Cloud (Amazon EC2) — веб-сервіс, що надає обчислювальні потужності у хмарі. Він дає користувачам повний контроль над обчислювальними ресурсами, а також доступне середовище для роботи. Amazon EC2 дозволяє користувачам створити Amazon Machine Image (AMI), який міститиме їх програми, бібліотеки, дані та пов'язані з ними конфігураційні параметри, або використовувати заздалегідь налаштовані шаблони образів для роботи Amazon S3. Amazon EC2 надає інструменти для зберігання AMI. Amazon S3 надає безпечне, надійне та швидке сховище для зберігання образів.

Microsoft

<http://www.windowsazure.com>

Microsoft SkyDrive — інтернет-сервіс зберігання файлів з функціями файлообміну, створений та керований компанією Microsoft. Сервіс SkyDrive дозволяє зберігати до 7 ГБ інформації (або 25 ГБ для користувачів, які мають право на безкоштовне оновлення) як стандартні папки. Користувачі можуть переглядати, завантажувати, створювати, редагувати та обмінюватися документами Microsoft Office (Word, Excel, PowerPoint та OneNote) безпосередньо у веб-браузері. Віддалений доступ до комп'ютера, який працює під керуванням Windows.

Windows Azure — це платформа хмарних сервісів, розроблена Microsoft. Реалізує моделі PaaS та IaaS. Платформа надає можливість розробки та виконання програм та зберігання даних на серверах, розташованих у розподілених центрах даних.

- *Windows Azure Compute* — компонент, що реалізує обчислення на платформі Windows Azure, надає середовище виконання на основі рольової моделі.

- *Windows Azure Storage* — компонент сховища, що надає сховище, що масштабується. Не має можливості використовувати реляційну модель і є

альтернативою (або доповнюючим рішенням) SQL Databases (SQL Azure) – «хмарною» версією SQL Server, що масштабується.

– *Windows Azure Fabric* — за своїм призначенням є контролером і ядром платформи, виконуючи функції моніторингу в реальному часі, забезпечення стійкості до відмови, виділення потужностей, розгортання серверів, віртуальних машин і додатків, балансування навантаження та управління обладнанням.

Платформа Windows Azure має API, побудоване на REST, HTTP та XML, що дозволяє розробникам використовувати хмарні сервіси з будь-якою операційною системою, пристроями та платформами.

Рішення на платформі Amazon зазвичай базуються на тому, що локальний комп'ютер розглядається як сервер, що входить до складу загального ЦОД, і вносяться необхідні зміни в конфігурацію, або необхідно створити кілька віртуальних машин залежно від необхідного рівня масштабування.

На інших вищезгаданих платформах використовують спеціальні засоби динамічного масштабування, скриптові програми, безліч різних API. Частина з них дуже зручна, в інших випадках необхідно докладати багато зусиль для суттєвої чи навіть майже повної переробки вихідної програми.

3.2 Інші постачальники хмарних технологій.

IBM SmartCloud

<http://www.ibm.com/cloud-computing/us/en/products.html>

Хмарне рішення, пропонуване компанією IBM, а саме IBM SmartCloud, реалізує всі три моделі (IaaS, SaaS, PaaS) в рамках не тільки публічної, але приватної та гібридної хмар. До його складу входить хмарний сервіс, раніше званий IBM Lotus Live, що надає бізнес-програми за моделлю SaaS. Містить повний набір інтерактивних сервісів, які надають рішення для організації захищеної системи електронної пошти, проведення web-конференцій і колективної роботи. Сервіси вільні від реклами та не збирають інформацію про клієнта, а також не є споживчими програмами, націленими на бізнес-діяльність. З користувачів стягується щомісячна плата. Елементи управління системою

захисту, розгорнуті для Lotus Live, забезпечують приватність та керований доступ до важливої інформації при виконанні бізнес-операцій. Усі клієнтські взаємодії кодуються стійкими алгоритмами шифрування та здійснюються за протоколом SSL для HTTP та через RC2 у протоколі системи миттєвого обміну повідомленнями Lotus Sametime. Резервні копії системи шифруються.

Rackspace Cloud

<http://www.rackspace.com/cloud/>

Платформа пропонує набір продуктів для автоматизації хостингу та хмарних обчислень, реалізується модель PaaS. Об'єднує у собі Cloud Files, Cloud Servers, Cloud Sites. Завдяки серверній віртуалізації користувачі отримують можливість розгорнути сотні хмарних серверів одночасно та створювати архітектуру, Платформа пропонує набір продуктів для автоматизації хостингу та хмарних обчислень, реалізується модель PaaS. Об'єднує в собі Cloud Files, Cloud Servers, Cloud Sites. Завдяки серверній віртуалізації користувачі отримують можливість розгорнути сотні хмарних серверів одночасно і створювати архітектуру, яка забезпечує високу доступність. Є конкурентом Amazon Web Services.

Oracle Exalogic Elastic Cloud

<http://www.oracle.com/us/solutions/cloud/overview/index.html>

Компанія ORACLE працює над концепцією програмного забезпечення як послуги протягом багатьох років. Компанія визнана одним з провідних постачальників програмного забезпечення, побудованого за технологією хмари, і працює з більш ніж 5,5 мільйонів користувачів. Компанія ORACLE пропонує вибір між моделями розгортання ПЗ як з використанням її центрів даних та заснованими на передплаті, так і з моделями розгортання ПЗ на території компанії замовника.

Для більшості центрів даних, що переходять на технологію приватних хмарних обчислень, ORACLE як перший крок пропонує консолідацію обчислювальних ресурсів і перехід на платформи та інфраструктуру, що розділяються і масштабуються.

Серверне обладнання, виділене під індивідуальні завдання middleware, БД та інші додатки, розраховане на пікове навантаження і має зарезервовану потужність, яка не використовується постійно. Кожен сервер може містити практично несумісні програмні компоненти від різних постачальників ПЗ, що у ряді випадків збільшує витрати на підтримку та підвищує управлінські витрати.

Перехід на об'єднану архітектуру хмарних обчислень ORACLE зі стандартизованими програмами, що розділяються, на вимогу істотно знижує витрати. Консолідація обчислювальних ресурсів може бути виконана як на рівні IaaS, з використанням технологій віртуалізації, так і на рівні PaaS, за допомогою стандартизації та об'єднання на основі БД та/або middleware архітектури.

Консолідація на рівні PaaS є великою цінністю, тому що зменшує різноманітність ПЗ, об'єднуючи його на основі стандартизованих програмних інтерфейсів. Консолідація на рівні IaaS може також забезпечити більш високу ефективність поділу апаратних засобів, але нічого не дає для зменшення складності підтримки ПЗ. Найбільш популярною є консолідація ПЗ на рівні БД ORACLE, що частково поєднує переваги PaaS та IaaS.

Для хмарних обчислень компанія ORACLE пропонує дві ключові технології: віртуалізація та кластеризація серверів. Віртуалізація дозволяє легко розгортати нові програми на вимогу і є добрим способом поділу апаратних засобів між завданнями. Об'єднання в кластери важливе для підвищення диверсифікації ресурсів між додатками, тим самим підвищуючи їх доступність та стійкість до відмов.

Платформа ORACLE PaaS

ORACLE PaaS є масштабованою платформою, спільною для всіх хмарних програм як приватних, так і громадських центрів даних. Платформа ORACLE PaaS заснована на БД ORACLE та додатках Oracle Middleware. Вона дає можливість різним організаціям об'єднувати існуюче ПЗ з використанням загальної архітектури, що дозволяє створювати нові додатки, які використовують існуючі ПЗ для розширення спектра послуг, що надаються на вимогу.

Платформа ORACLE PaaS надає послуги БД на вимогу, засновані на БД ORACLE та апаратних комплексах Oracle Exadata, а також послуги ПЗ Middleware на вимогу на основі Oracle WebLogic та Oracle Exalogic. Oracle Exadata – це спеціалізована машина БД, а Oracle Exalogic є машиною, оптимізованою для виконання програм Middleware, написаних мовою JAVA. Обидві машини масштабуються та стійкі до відмов. Вони спроектовані та налаштовані для спільної роботи.

Для розробки нових програм програмісти можуть використовувати знайомі середовища проектування, такі як JDeveloper, NetBeans і Eclipse, а також мережеві інструменти WebCenter Page Composer, BI Composer і BPM Composer. Для взаємної інтеграції нових та розроблених раніше додатків у приватних та громадських хмарах компанія ORACLE пропонує Oracle SOA Suite та Oracle BPM Suite, а також Oracle Data Integration та Oracle GoldenGate. Ідентифікація користувачів та розподіл прав здійснюються за допомогою Oracle Identity and Access Management. За взаємодію користувачів із хмарою відповідає програмний комплекс Oracle WebCenter, що дозволяє користувачам здійснювати спільну роботу.

Інфраструктура ORACLE IaaS

Компанія ORACLE пропонує набір елементів технології «Інфраструктура на вимогу» (IaaS), у тому числі обчислювальні сервери, послуги зберігання та передачі інформації, віртуалізацію ПЗ, операційні системи та системи управління ПЗ.

Інфраструктура ORACLE IaaS включає сервери, засновані на технології SPARC та x86, встановлені у стійках та лезах (blades); технології зберігання на FLASH, дискових масивах та стрічках; варіанти віртуалізації Oracle VM для x86, Oracle VM для SPARC та Oracle Solaris Containers; операційні системи Oracle Solaris, Oracle Linux та Oracle Enterprise Manager.

Гнучка інфраструктура ORACLE IaaS підтримує об'єднання гетерогенних ресурсів, масштабованість, швидке розгортання та високу доступність

прикладного ПЗ, дозволяючи ефективно управляти громадськими та приватними IaaS.

У 2010 році Oracle відкрила власну публічну хмару Oracle Cloud, що надає як технологічне програмне забезпечення за моделлю PaaS, так і бізнес-додатки за моделлю SaaS.

Salesforce.com

<http://www.salesforce.com>

Система управління взаємовідносинами із клієнтами (CRM-система – Customer Relationship Management) надається замовникам виключно під моделі SaaS. Під назвою Force.com компанія надає PaaS-платформу для самостійної розробки додатків, а під брендом Database.com – систему хмарної управління БД. В залежності від рівня передплати доступні різні технічні можливості. Так, у безкоштовній версії Force.com передплатники можуть створити не більше десяти сутностей, а в необмеженій версії з ціною \$75 на користувача на місяць – до 2 000. Передплатники можуть розміщувати розроблені програми на платформі Force.com у спеціальному каталозі (AppExchange) та пропонувати свої розробки іншим замовникам, зокрема на комерційній основі.

Як система управління БД платформа Force.com використовує три кластера Oracle RAC, що реплікуються, з восьми вузлів кожен. Кластери розташовані у трьох віддалених один від одного ЦОД. В одній схемі Oracle Database обробляються дані відразу кількох компаній-передплатників. Використовується стабільна схема даних, попередньо підготовлена до розширення додатковими об'єктами таким чином, що в тих самих таблицях зберігаються дані різних передплатників незалежно від відмінності в специфічних атрибутах об'єктів для різних передплатників. Широко використовується секціонування таблиць БД.

Parallels

<http://www.parallels.com>

Пропонує цілу низку продуктів для автоматизації хостингу та хмарних обчислень, заснованих на IaaS. Parallels Cloud Server поєднує в собі Parallels Cloud Storage, Parallels Virtuozzo Containers та Parallels Hypervisor, дозволяючи значно

підвищити надійність, продуктивність та рентабельність серверів. Parallels Cloud Storage – це гнучке, масштабоване рішення, що допомагає збільшити доступність та продуктивність хмарних серверів. Parallels Virtuozzo Containers – рішення для серверної віртуалізації, що забезпечує високий рівень щільності та продуктивності, а також досить швидке виділення ресурсів та динамічну зміну сервіс-планів. Технологія Parallels Hypervisor дозволяє самостійно створювати віртуальні машини.

Можливість розміщувати безліч клієнтів на ізольованих один від одного віртуальних серверах на одному фізичному сервері дозволяє сервіс-провайдерам оптимізувати ресурси та підвищувати прибуток. PACI (Parallels Automation for Cloud Infrastructure) – рішення, що включає все необхідне для надання хмарних послуг на будь-якій апаратній платформі. Клієнти можуть розширювати, звужувати та видаляти хмарні послуги, розподіляти навантаження між кількома віртуальними машинами чи контейнерами та захищати свої дані за допомогою брандмауерів. PACI забезпечує автоматичне балансування навантаження та безпеку (включаючи брандмауери) для всіх віртуальних серверів.

3.3 Популярні хмарні сервіси для офісу та навчання.

Світовий досвід впровадження та використання хмарних технологій свідчить про перспективність його використання й у вітчизняній системі освіти. На сьогоднішній день в україномовному сегменті мережі Інтернет найбільшою популярністю серед освітян користуються сервіси хмарних обчислень корпорацій Microsoft та Google. Саме ці корпорації, постійно удосконалюючи свої службові сервіси, дозволяють організувати швидке впровадження технологій хмарних обчислень у навчально-виховні процеси освітніх закладів. Потужний інструментарій та інноваційні функціональні можливості освітніх «хмар» дозволяють сучасним педагогам використовувати ці технології у своїй професійній діяльності максимально ефективно.

Microsoft для впровадження хмарних технологій у систему навчання пропонує ряд інструментів, що базуються на взаємодії викладача та студента

(пакет Microsoft Office 365), а саме система електронної пошти Outlook, інтерактивні календарі, контакти Outlook Line, веб-додатки та архіви SkyDrive, система обміну миттєвими повідомленнями LyncOnline, телефонна мережа Skype, соціальна мережа Yumner міні-сайти тощо. Служба Microsoft Office 365 може підтримувати як персональне використання онлайнових інтерактивних додатків, так і корпоративне їх використання багатьма користувачами.

Google, в свою чергу, пропонує сучасний інструментарій для побудови навчальних порталів (служба G Suite for Education). G Suite for Education – це набір хмарних служб, які допоможуть педагогам і учням продуктивно працювати і спілкуватися в будь-який час в будь-якому місці на будь-яких пристроях.

Окрім G Suite for Education, корпорація Google розробляє та надає велику кількість додатків та сервісів для освітнього середовища, доступ до яких можливий у вікні будь-якого браузера при наявності підключення до Інтернету. Найуживаніші з них є наступні сервіси: Google ArtProject (віртуальні інтерактивні музеї світу), Google Earth (безкоштовна, вільно завантажувана програма, що відображає віртуальний глобус), Google Maps (набір онлайнових карт), Google Translate (онлайнний перекладач), Youtube (відеохостинг).

Найбільш поширеною системою сервісів на основі хмарних технологій, що застосовується в освітньому процесі, є служба G Suite for Education (донедавна відома як Google Apps).

G Suite for Education – це набір стандартних хмарних (тобто розміщених на серверах компанії Google) додатків для планування спільної діяльності, колективної роботи і спілкування, публікації матеріалів, хостингу відеоматеріалів та багатьох інших інструментів, доступний в домені .edu.

Використання сервісів Google в освітній галузі надає ряд переваг:

- безкоштовність (доступні всі базові можливості, відсутність обмежень на період використання);
- один акаунт – всі сервіси (реєстрація необхідна на самому початку, нові сервіси можна підключати за необхідністю, наявність та налаштування персонального організатора всіх сервісів);

- знайомий інтуїтивно зрозумілий інтерфейс (відповідає стандартним офісним програмам, має україномовну версію);
- хмарне зберігання інформації (можливість втрати даних виключена, спрощений доступ за прямими посиланнями, можливість редагування та доступу з будь-якого комп'ютера, що має підключення до мережі Інтернет);
- мінімальні вимоги для доступу (відсутність необхідності встановлення додаткових програм, підтримка різними браузерами, доступ з портативних пристроїв (смартфонів, планшетів));
- сумісне створення документів (об'єднання людей, що працюють над спільним документом, оперативна узгодженість документів, інтерактивність, відображення змін в реальному часі);
- історія всіх змін (ведення статистики змін, можливість відновлення документу попередньої редакції);
- розмежування прав на доступ (різні права на доступ (редагування, перегляд, коментування), можливість вбудовування у вигляді посилань);
- підтримка та розвиток (оновлення інтерфейсу та можливостей, інтеграція з сучасними технологіями Веб 2.0 тощо);
- спільнота користувачів (обмін думками та можливостями, ефективні приклади та досвід, широка аудиторія для тестування інструментів).

Служби G Suite for Education допомогли в корені змінити систему навчання в багатьох навчальних закладах по всьому світу. У наш час продукти Google стали для закладів освіти тією технологічною базою, що допомогла підняти ефективність спільної роботи педагогів та учнів на новий рівень.

G Suite for Education містить дві категорії сервісів:

1) Основні сервіси:

- Gmail – повнофункціональний поштовий клієнт, що дозволяє обмінюватись миттєвими повідомленнями, користуватись голосовим та відеочатом, має мобільний доступ, а також захист від вірусів та спаму. Основною особливістю даного поштового сервісу є потужний алгоритм пошуку по поштової кореспонденції. Продумана ієрархія повідомлень в Gmail дозволяє бачити

повідомлення в контексті, і, якщо існують відповіді на відправлене або отримане повідомлення, система Gmail автоматично відображає їх у хронологічній послідовності разом з початковим повідомленням. Цей ланцюжок повідомлень дозволяє відслідковувати усі повідомлення та продовжувати обговорення в одному місці.

- Google Calendar (Календар) – це, перш за все, веб-інструмент управління та планування. Створення календаря учнівських та педагогічних заходів, календарне планування роботи над будь-яким проектом, спільне використання календарів для створення та перегляду розкладу занять і консультацій – ось декілька прикладів використання можливостей сервісу Google Calendar в освітньому процесі і не тільки.

- Google Cloud Search (Хмарний пошук) – веб-сервіс, що дозволяє користувачам шукати та опрацьовувати матеріали у Основних сервісах G Suite, та пропонує корисну інформацію та інструкції.

- Google Drive (Диск) – хмарне середовище з набором веб-інструментів для зберігання та передачі файлів різних форматів, а також для перегляду відео.

- Google Docs (Документи) – веб-сервіс, що дозволяє створювати, редагувати, експортувати текстові документи, а також надавати колективного доступу до роботи з ними.

- Google Sheets (Таблиці) – веб-сервіс, що дозволяє створювати, редагувати та експортувати електронні таблиці, а також надавати колективного доступу до роботи з ними.

- Google Slides (Презентації) – веб-сервіс, за допомогою якого користувач може створювати, редагувати, експортувати презентації в режимі онлайн, а також надавати колективного доступу до роботи з ними.

- Google Forms (Форми) – інструмент, за допомогою якого можна легко і швидко планувати заходи, складати опитування та анкети, а також збирати іншу інформацію.

- Google Sites (Сайти) – веб-сервіс, що дозволяє користувачам створювати сайти в домені G Suite Basic або на зовнішніх ресурсах. Користувач може

створити сайт за допомогою веб-інструменту, а потім надати доступ до нього групі інших користувачів або ж зробити його вільнодоступним. Власник сайту сам вирішує, хто має право відвідувати сайт та редагувати його контент.

– Google Hangouts, Google Talk, Hangouts Meet – веб-сервіси, за допомогою яких користувачі можуть спілкуватися один з одним в режимі реального часу. Google Hangouts та Google Talk дозволяють обмінюватися текстовими повідомленнями, здійснювати дзвінки та проводити полегшені відеозустрічі з одним чи декількома користувачами. За допомогою Hangouts Meet можна проводити відеозустрічі з великою кількістю учасників. Адміністратори доменів G Suite можуть вибрати, які з сервісів будуть доступні в домені, в тому числі дозволити проводити розширені відеозустрічі Hangouts Meet в доповнення до класичних полегшених відеозустрічей Google Hangouts або замість них.

– Google Keep – веб-сервіс, що дозволяє користувачам створювати та редагувати замітки, списки та малюнки, а також надавати до них доступ та можливість колективної роботи над ними.

– Google Сейф – веб-сервіс для пошуку та експорту вмісту Google Диска та Gmail. За допомогою Google Сейфа користувачі можуть архівувати дані, створювати правила зберігання та видалення повідомлень з певним вмістом в Gmail, а також шукати контент у всьому домені. Крім того, функції надання електронних документів дозволяють створювати папки для зберігання даних в юридичних цілях. Щоб зберігати архівовані дані на серверах Google, користувачам необхідно продовжити або придбати підписку на Google Сейф. Якщо в акаунті користувача активовано сервіс Gmail, він може зберігати (в тому числі на певний строк) та експортувати історію чатів в Google Talk та Google Hangouts, а також шукати в цих сервісах потрібну інформацію.

2) Додаткові сервіси:

– YouTube – сервіс, що надає послуги з відеохостингу та дозволяє користувачам завантажувати, переглядати та коментувати відеозаписи. Активні користувачі даного сервісу створюють власні канали.

– Google Maps (Карты) – набір додатків, побудованих на основі безкоштовного картографічного сервісу та технологій, які надає компанія Google. Сервіс являє собою карту та супутникові знімки планети Земля. Для багатьох регіонів доступні високодеталізовані аерофотознімки, для деяких – з можливістю перегляду під кутом 45° з чотирьох сторін світу. Додатково пропонуються знімки Місяця та Марсу.

– Blogger – зручний сервіс для ведення блогів: дозволяє користувачу створювати власну сторінку в Інтернеті для спілкування між людьми, об'єднаними спільними інтересами.

– Google Analytics – безкоштовний зручний та корисний сервіс для створення детальної статистики відвідувачів веб-сайтів а також збору даних про їхні дії на сайті. Інструмент формує велику кількість звітів, на підставі яких можна будувати стратегію просування сайту. Тобто, встановлення лічильника Google Analytics дає можливість зрозуміти – хто, коли і навіщо відвідував сайт. Всебічний аналіз цільової аудиторії дозволяє оперативно реагувати на запити користувачів.

– Google Earth (Планета Земля) – безкоштовна, вільно-завантажувана програма компанії Google що відображає віртуальний глобус. Користувачі можуть створювати свої власні мітки та накладати свої зображення поверх супутникових (це можуть бути карти чи більш детальні знімки, отримані з інших джерел).

– Google Groups (Групи) – сервіс, що надає можливість користувачам брати участь у обговореннях та створювати власні групи.

– Google Scholar (Академія) – безкоштовна пошукова система за повними текстами наукових публікацій усіх форматів та дисциплін. Індекс Google Scholar містить дані з більшості рецензованих онлайн журналів найбільших наукових видавництв Європи та Америки.

– Google Translator (Перекладач) – безкоштовний сервіс, що дозволяє миттєво перекладати слова, фрази та веб-сторінки з англійської на більш ніж 100 мов та назад.

Безпосередньо в навчальному процесі сервіси, які надає компанія Google, можна використовувати для інтерактивного спілкування з учнями (електронна пошта, онлайн-спільноти, колективні сховища знань), спільного використання онлайн-додатків (календар, веб-конференції, спільна робота з документами). Представлені сервісами Google онлайн- та офлайн-матеріали дають змогу користувачу перевіряти правильність виконаних вправ, не покидаючи простору; застосовувати найрізноманітніші наукові, енциклопедичні та довідникові видання; опрацьовувати навчальний матеріал у відповідному темпі та режимі.

Питання та завдання до контролю знань студентів

1) Групи Google – це:

А інструмент управління та групової роботи на основі модерованих форумів та списків розсилок;

Б хмарні сервіси на базі Google;

В це платний хмарний інтернет-сервіс і програмне забезпечення компанії Microsoft, що розповсюджується за схемою «програмне забезпечення послуги»;

Г набір безкоштовних веб-додатків – текстового і табличного процесора, менеджера презентацій і редактора нотаток.

2) Яка «хмара» включає в себе 25 Гб вільного місця для зберігання файлів і синхронізації фотографій?

А Windows Live;

Б Dropbox;

В iCloud;

Г Amazon CloudDrive.

3) У 2014 році платформа Windows Azure була перейменована в:

А Apple Azure;

Б Microsoft Azure;

В SLA;

Г CobiT Azure.

4) В якому році Google офіційно запустила хмарне сховище Google Drive?

А 2012;

Б 2001;

В 2014;

Г 2005.

5) Це є безкоштовне онлайнове сховище в хмарі, яке ви отримуєте разом із обліковим записом Microsoft:

А OneDrive;

Б Google Docs;

В Dropbox;

Г iCloud.

6) Продукт MS Office Web Apps - це :

А набір безкоштовних веб-додатків – текстового і табличного процесора, менеджера презентацій і редактора нотаток;

Б система цифрової дистрибуції відеоігор, сервіс, що використовує концепцію хмарних обчислень (англ. Cloud Computing), сьоме покоління гральних консолей;

В захищене хмарне сховище з можливостями резервного копіювання, синхронізації, спільної роботи, розподілу прав доступу та іншими стандартними особливостями;

Г сайти, на яких розміщується інформація щодо хмарних технологій.

Розробив: Ланська С.С.

Розглянуто та схвалено

на засіданні циклової комісії

програмної інженерії

Протокол № 2 від 21.09.2023 р.

Голова комісії _____ Ланська С.С.

Лекція № 4

з навчальної дисципліни «Хмарні технології»

Тема: Інтернет речей (IoT) та Смарт-технології. Застосування технології інтернет речей у сучасному світі.

Мета заняття: розкрити принципи застосування технології Інтернет речей у сучасному світі техніки.

Час – 2 години

План проведення лекції

- 1) Інтернет речей IoT.
- 2) Застосування технології інтернет речей у сучасному світі.

Література

- 1) Кононюк А.Е. Фундаментальная теория облачных технологий [Текст]: Книга 1. Общенаучные подходы формирования систем облачных технологий / А.Е. Кононюк. – К.: Освіта України, 2018 – 621 с.
- 2) Косинський В. І. Сучасні інформаційні технології: навч. посіб. для студ. ВНЗ / В. І. Косинський, О. Ф. Швець. – [2-е вид., випр.]. – К. : Знання, 2012. – 318с.
- 3) Хмарні технології в освіті. Навчально-методичний посібник для студентів фізико-математичного факультету. – Житомир: ЖДУ, 2016. – 72 с.

Навчальні матеріали лекції

4.1 Інтернет речей IoT.

Поняття «Інтернет речей» (Internet of Things, IoT) з'явилося 1999-го року. Існує кілька визначень, що розкривають суть цього концепції.

За версією компанії IDC «Інтернет речей — це мережа мереж з унікально ідентифікованими кінцевими точками, які спілкуються між собою у двох напрямках за протоколами Internet Protocol (IP) та зазвичай без людського втручання».

Фахівці компанії Gartner дають таке визначення: «Інтернет речей — це мережа фізичних об'єктів, які мають вбудовані технології, що дозволяють здійснювати взаємодію із зовнішнім середовищем, передавати відомості про свій стан і приймати дані з поза».

Існує ще більш технологічне визначення компанії McKinsey — «Інтернет речей — це датчики та приводи (виконавчі пристрої), вбудовані у фізичні об'єкти та пов'язані через дротові або бездротові мережі з використанням протоколу IP, який зв'язує Інтернет».

Іншими словами, «Інтернет речей — це неосяжне скупчення підключених датчиків, камер, смартфонів, комп'ютерів та пристроїв, які взаємодіють із додатками, веб-сайтами, соціальними медіа та іншими пристроями. Щоб витягти з цього максимальну користь, необхідно обробляти та аналізувати дані, які генерують усі ці об'єкти, в реальному часі».

Можна вивести умовну формулу, що характеризує Інтернет речей (IoT):

$$\text{IoT} = \text{Сенсори (датчики)} + \text{Дані} + \text{Мережі} + \text{Послуги}$$

Під конкретною «Інтернет річчю (thing)» прийнято розуміти «будь-який натуральний або штучний предмет, якому може бути присвоєна IP-адреса і який має можливість передачі даних по мережі».

У 2009 році кількість підключених до мережі предметів перевищила кількість людей, відбувся перехід від Інтернету до Інтернету речей.

Прийнято виділяти рішення Інтернету речей як для корпоративного сектора (B2B), так і для сектора сполучення (B2C).

Рішення Інтернету речей для корпоративного та споживчого сектору представлено в таблиці 4.1.

Таблиця 4.1 – Рішення Інтернету речей для корпоративного та споживчого сектору

для корпоративного сектору (B2B)	IoT для споживчого сектора (B2C)
розумний транспорт та безпілотники;	пристрої, що носяться (електроніка);
розумне місто (smart city);	розумний дім (smart home);
страхова телематика;	розумний одяг;
розумні робочі місця;	smart TV;
розумні електромережі (smart grid);	розумні пристрої для тварин;
розумні заводи (smart factory, industrial Internet, IIoT);	інше.
точне землеробство;	
розумні свердловини (smart well);	
геолокаційний маркетинг;	
розумні склади;	
інше.	

Датчики та живлення

Інтернет починається або закінчується однією подією: простий рух, зміна температури або, може бути, важіль замикає замок. На відміну від багатьох існуючих ІТ-пристроїв, Інтернет речей здебільшого пов'язаний з фізичною дією або подією. Він формує реакцію на якийсь фактор реального світу. Іноді при цьому один-єдиний датчик може згенерувати величезний обсяг даних, наприклад, акустичний датчик для профілактичного огляду обладнання. В інших випадках всього одного біта даних достатньо, щоб передати життєво важливі відомості про стан здоров'я пацієнта. Якою б не була ситуація, системи датчиків еволюціонували і, відповідно до закону Мура, зменшилися до субнанометрових розмірів і стали істотно дешевше. Саме до цього апелюють ті, хто прогнозує, що до Інтернету речей будуть підключені мільярди пристроїв, і саме тому ці прогнози виправдаються. 20

Тому, розглядаючи Інтернет Речей, необхідно розглядати мікроелектромеханічні системи, датчики і інші типи недорогих граничних пристроїв і їх електрофізичних властивостей. Також це стосується силових і енергетичних систем, необхідних для живлення цих граничних пристроїв. Не

можна вважати, що граничні пристрої забезпечуються енергією за замовчуванням. Мільярди маленьких датчиків все одно потребують великої кількості енергії. З питанням живлення також пов'язані питання організації хмарних сервісів IoT.

Передача даних

Велика увага при розробці IoT приділяється встановленню з'єднання і роботі мереж. Інтернету речей не існувало б без надійних технологій передачі даних з найвіддаленіших і несприятливих областей в найбільші центри збору даних компаній Google, Amazon, Microsoft і IBM. Словосполучення «Інтернет речей» містить слово «Інтернет», тому необхідно вивчати питання, що стосуються мережних технологій, обміну даними та навіть теорії сигналів. Базова опора Інтернету речей - це не датчики і не програми, а можливість встановити з'єднання.

Передача даних і встановлення мережевого з'єднання базуються на базі систем зв'язку ближньої дії - персональних мереж (PAN), зазвичай побудованих без дотримання правил IP-протоколу. Це може бути як проводові так і бездротові мережі. До бездротових IoT-мереж/протоколів як правило відносяться протоколи Bluetooth, mesh-мережі, Zigbee, ZWave. Для IIoT це також Wireless Hart та ISA100. Це яскравий приклад різноманіття бездротових систем зв'язку IoT. Перелік дротових мереж ще більший, так як сюди входять усі можливі промислові мережі та протоколи.

Крім PAN використовуються бездротові локальні мережі та системи зв'язку на основі IP-протоколу, включаючи широкий діапазон Wi-Fi-мереж на основі стандартів IEEE 802.11, 6LoWPAN і технології Thread. Нерідко використовуються телекомунікації на основі стільникових стандартів (3G, 4G LTE) і нові стандарти, що забезпечують роботу Інтернету речей і міжмашинної взаємодії, такими як Cat-1 і Cat-NB, а також пропрієтарні протоколи LoRaWAN і Sigfox, що використовуються саме для IoT.

Маршрутизація Для передачі даних від датчиків в Інтернет-простір необхідні дві технології: маршрутизатор-шлюз і опорні інтернет-протоколи, що забезпечують ефективність обміну даними. Маршрутизатор особливо важливий в таких аспектах, як безпека, управління і напрям даних. Граничні маршрутизатори

(Edge routers) керують і стежать за станом відповідних mesh-мереж, а також вирівнюють і підтримують якість даних. Також велике значення належить конфіденційності та безпеки даних. Маршрутизатор відіграє важливу роль в створенні віртуальних приватних мереж, віртуальних локальних мереж і програмовизначених глобальних мереж. Вони в буквальному сенсі можуть містити тисячі вузлів, що обслуговуються єдиним граничним маршрутизатором, і в якійсь мірі маршрутизатор служить розширенням для хмари (edge device).

На цьому рівні використовується ряд протоколів, необхідних для обміну даними між вузлами, маршрутизаторами і хмарними сервісами в межах IoT-системи. Інтернет речей відкрив дорогу новим IoT-протоколам, які виходять на один рівень з традиційними протоколами HTTP і SNMP, які застосовуються вже кілька десятиків років. Для передачі IoT-даних потрібні ефективні, енергозберігаючі протоколи з малою затримкою, здатні легко і безпечно відправляти дані в хмару і з нього. Зокрема тут використовуються такі протоколи, як всюдисущий MQTT, AMQP і CoAP.

Туманні і граничні обчислення, аналітика і машинне навчання

На цьому етапі необхідно вирішити, що робити з потоком даних, що надходять в хмарний сервіс з граничного вузла (Edge Device). Щоб навчитися правильно оцінювати, як система буде розвиватися і рости, необхідно розібратися у всіх тонкощах і складнощах архітектури хмарних систем, який вплив на IoT-систему робить запізнювання. Крім того, не все треба відправляти в хмару. Пересилання всіх IoT-даних обходиться значно дорожче, ніж їх обробка на кордоні мережі (граничні обчислення, Edge Computing) або включення граничного маршрутизатора в зону, яку обслуговує хмарний сервіс (туманні обчислення, Fog computing). Туманні обчислення також стандартизуються, зокрема є стандарт туманних обчислень, наприклад архітектура OpenFog. Дані, які були отримані шляхом перетворення аналогового фізичного впливу в цифровий сигнал, можуть мати велику вагу. Саме тут в гру вступають засоби аналітики і процесори правил IoT-системи. Ступінь складності введення в дію IoT-системи залежить від того, яке рішення проектується. У деяких ситуаціях все

досить просто: наприклад, коли на граничний маршрутизатор, який контролює кілька датчиків, потрібно встановити простий процесор правил, що відслідковує аномальні скачки температури. Інша ситуація - величезна кількість структурованих і неструктурованих даних в режимі реального часу передається в 23 хмарне озеро даних, що вимагає високої швидкості обробки (для прогнозової аналітики) і довгострокового прогнозування на базі високотехнологічних моделей машинного навчання, таких як рекурентна нейронна мережа в пакеті аналізу сигналів з кореляцією по часу. Тут є певні проблеми і складнощі аналітики, які вирішуються різними підходами та методами, наприклад складними обробниками подій, байесовськими мережами і формування нейронних мереж.

В даний час IoT найбільш активно використовується в технологіях «розумного будинку»: дистанційне керування через Інтернет домашніми пристроями, віддалений моніторинг та керування системами опалення, освітлення, медіа-пристроями, електронними системами безпеки, оповіщеннями про вторгнення, протипожежними системами та ін. Широке застосування технологія IoT знаходить в енергетиці (смарт-лічильники, системи виявлення втрат або крадіжок в електричній мережі). У нафтогазовому секторі для віддаленого моніторингу трубопроводів. В автостраховання починає практикуватися розрахунок страховки, що базується на віддаленому моніторингу водіння користувачів. У логістиці широко використовуються системи відстеження маршруту автомобіля, моніторинг вантажоперевезень, контроль відвантаження та складування. У торгівлі розвивається віддалений моніторинг та облік товарів, забезпечених RFID-мітками, інвентаризація в реальному часі, бездротові платіжні рішення, наприклад, використанням технології NFC.

Створення власного пристрою для IoT можливе за допомогою сервісу IBM IoT Foundation і платформи IBM Bluemix.

Набирає популярності пакет Azure IoT Suite – хмарне рішення компанії Microsoft для однойменної хмарної інфраструктури.

Очікується, що у далекій перспективі IoT призведе до трансформації у «Всеосяжний Інтернет».

4.2 Застосування технології інтернет речей у сучасному світі

Архітектура IoT складається з чотирьох функціональних рівнів, які наведені на рисунку 4.1. Найнижчий рівень (рівень сенсорів і сенсорних пристроїв) складається з об'єктів, інтегрованих із сенсорами (датчиками), які забезпечують збір і обробку інформації в реальному масштабі часу. Наступний рівень (шлюзів і мереж) складається з конвергентної мережевої інфраструктури, яка створюється шляхом інтеграції різнорідних мереж у єдину мережеву платформу. Сервісний рівень містить певний набір послуг, які автоматизують низку технологічних і господарських операцій. Четвертий рівень архітектури IoT включає різні типи додатків для відповідних промислових секторів і сфер діяльності.

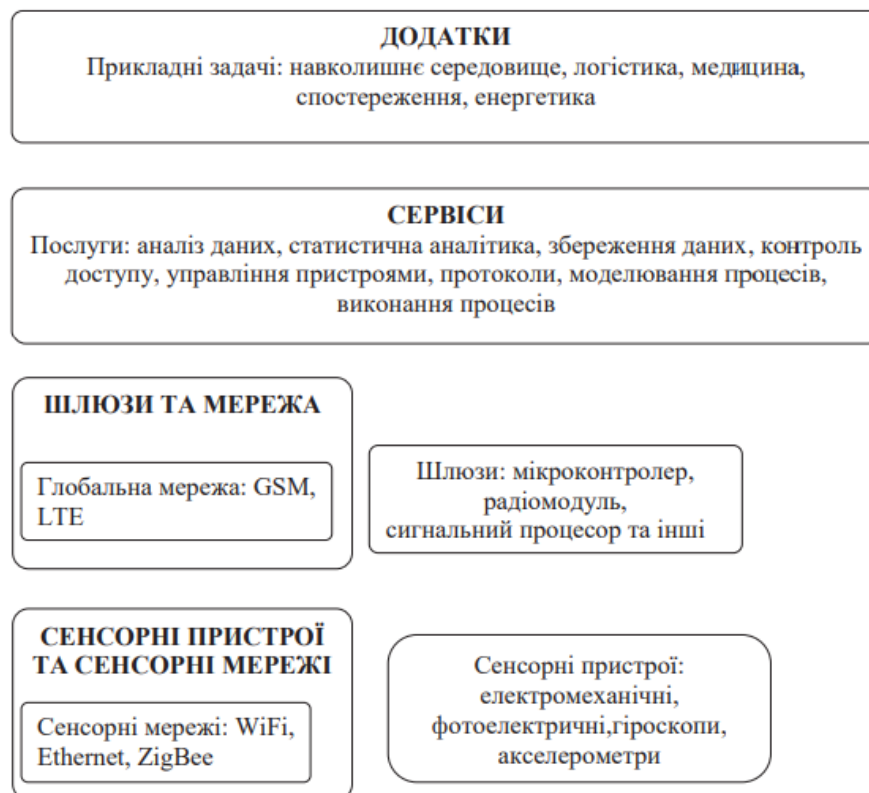


Рисунок 4.1 – Функціональні рівні архітектури IoT

Варто зауважити, що M2M (міжмашинна взаємодія) давно використовується в різних галузях *економіки*. Ця технологія є надійним способом збору даних, однак на стадії прийняття рішень вимагає обов'язкової участі людини. Відмінність Інтернету речей полягає саме в автоматизації рутинних дій, заснованих на аналізі даних.

Інтернет речей можна визначити як інформаційно-технологічну концепцію побудови інформаційних і комунікаційних інфраструктур на основі обчислювальної мережі, яка з'єднує речі (фізичні об'єкти), оснащені інформаційними технологіями для здійснення комунікаційного обміну один з одним і глобальною інформаційно-комунікаційною інфраструктурою або безпосередньо, або через інтегровані з ними інші пристрої, які мають адресу протоколу Інтернет (IP) без участі людини з метою збору, передачі, накопичення та обробки інформації.

Застосування IoT в енергетиці, дивись рисунок 4.2, дозволить в якомусь сенсі перетворити електроенергетику. На зміну ієрархічній системі «виробництво – передача – збут», у якій всі процедури жорстко визначені регламентами, а узгодженість досягається за рахунок державного регулювання, учасники дізнаються про дії один одного з новин, прийде гнучка система продуктивної взаємодії в режимі реального часу. Кожен елемент системи буде «бачити» інші елементи, розуміти їх можливості і потреби і використовувати свій потенціал найкращим чином. Така зміна дозволить вийти на принципово новий рівень надійності і ефективності в роботі енергетичної системи. Ключовими сферами у всіх елементах електроенергетики, на які вплине впровадження IoT, є:

- технології, в тому числі підвищиться їх надійність;
- економічність, в тому числі скоротяться витрати;
- поява нових ринків, створення нових властивостей і бізнесів.

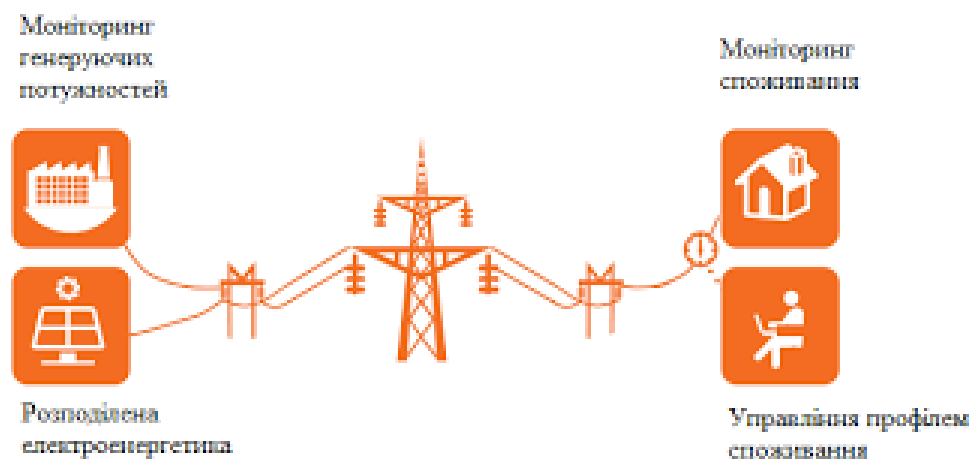


Рисунок 4.2 – Области застосування IoT в електроенергетиці

У сфері охорони здоров'я технології IoT допомагають підвищити ефективність роботи медичних установ, скоротити час перебування в стаціонарі, надати пацієнтам нові сервіси для контролю за станом здоров'я, збирати і аналізувати додаткову інформацію про хід лікування. Так, дистанційний моніторинг здоров'я допомагає знизити витрати за рахунок оперативного контролю медичних показників і спростити взаємодію між лікарями і пацієнтами.

Області застосування IoT у сфері охорони здоров'я наведено на рисунку 4.3.

Пристрої IoT також беруть участь у вирішенні низки управлінських, адміністративних і логістичних завдань: моніторинг і управління людськими ресурсами, віддалена діагностика медичного обладнання, локальне позиціонування персоналу, пацієнтів і переносних пристроїв, управління запасами медикаментів і витратних матеріалів.



Рисунок 4.3 – Області застосування IoT у сфері охорони здоров'я

«Розумне сільське господарство», наведено на рисунку 4.4, ставить перед собою мету максимально автоматизувати сільськогосподарську діяльність, підвищити урожайність і якість продукції. Точне землеробство (GPSдатчики, дрони) – це широкий спектр технологій від планування посіву і підготовки

грунту, моніторингу стану та управління посівом, контролю рівня вологості, мінералізації ґрунту і температурного режиму до збору врожаю. При використанні «розумних теплиць» (датчики, пристрої та комплектуючі, ПЗ для віддаленого управління теплицями) операційна економія досягається шляхом більш ефективної витрати добрив, хімікатів і води.

Ця технологія дозволяє оптимізувати кількість персоналу, який потрібен для догляду за культурами, і знизити втрати, що виникають через людський фактор. «Розумні ферми» (датчики, пристрої і ПЗ для моніторингу) дозволяють підвищити продуктивність тварин і якість продукції. За оцінкою експертів ринку, автоматизовані системи відгодівлі, доїння і моніторингу здоров'я поголів'я худоби можуть підвищити надої на 30-40%.

Послуги, які викликають потенційний інтерес в контексті «розумного міста»: моніторинг стану будівель; управління відходами; контроль якості повітря; моніторинг шуму; перевантаження автотранспорту; міське енергоспоживання; розумне паркування; розумне освітлення; моніторинг параметрів навколишнього середовища.

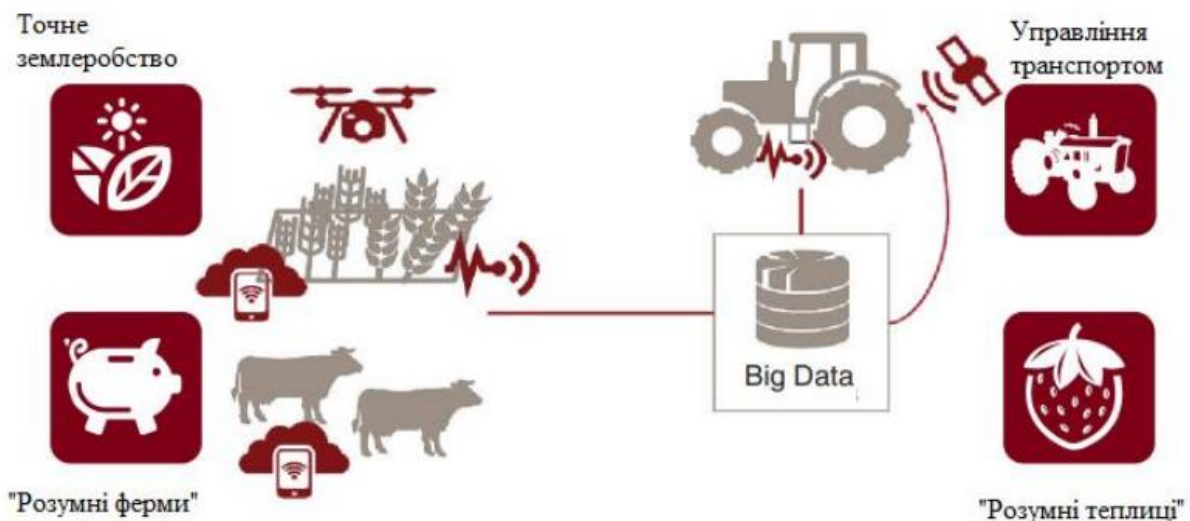


Рисунок 4.4 – Области застосування IoT у сфері сільського господарства

Моніторинг стану будівель. Належне обслуговування будівель міста вимагає постійного моніторингу їх фактичного стану і виявлення районів, найбільш схильних до впливу зовнішніх факторів. Міський IoT може надати розподілену

базу даних вимірювань цілісності конструкції будівлі, зібрану відповідними датчиками, розташованими в будівлях, такими як датчик вібрації і деформації, датчики контролю рівня забруднення, а також датчики температури і вологості повітря, для отримання повної характеристики умов навколишнього середовища. Ця база даних дозволить проводити цілеспрямовані і випереджальні дії по обслуговуванню і ремонту будівель. Базу даних можна зробити загальнодоступною, щоб городяни знали про роботи, які проводяться по збереженню історичної спадщини міста.

Управління відходами. Управління відходами є основною проблемою у багатьох сучасних містах, що зумовлено як вартістю послуг, так і проблемою зберігання сміття на звалищах. Застосування IoT технологій у цій галузі може призвести до економічних і екологічних переваг. Для реалізації служби управління відходами на базі IoT потрібно підключення «розумних сміттєвих контейнерів» до центру управління колекторно-вантажного автопарку.

Контроль якості повітря. Міський IoT може забезпечити контроль якості повітря в місцях роботи і відпочинку людей, а також у парках або на трасах. Реалізація такої послуги вимагає, щоб по всьому місту були встановлені датчики вимірювання якості повітря з доступністю результатів вимірювань для громадськості, наприклад на карті міста через відповідну програму.

Моніторинг шуму. Шум можна розглядати як форму акустичного забруднення. IoT може запропонувати послугу моніторингу шуму для вимірювання кількості шуму. Крім побудови просторово-часової карти шумового забруднення в цьому районі, така служба може також використовуватися для забезпечення громадської безпеки за допомогою алгоритмів виявлення звуку, які можуть розпізнавати, наприклад, шуми, які виникають при дорожньо-транспортних пригодах або конфліктних зіткненнях.

Перевантаження автотранспорту. Незважаючи на те, що системи моніторингу трафіку на основі камер вже доступні і розгорнуті в багатьох містах, рішення на базі IoT дозволять забезпечити більш щільний потік джерел

інформації. Моніторинг дорожнього руху може бути реалізований за допомогою сенсорів і GPS, встановлених на сучасних автомобілях.



Рисунок 4.5 – Области застосування IoT у транспортуванні і зберіганні вантажів

Міське енергоспоживання. Міський IoT може надати послугу для моніторингу енергоспоживання всього міста, що дозволяє міській владі отримати повне уявлення про кількість енергії, необхідної для різних послуг (освітлення вулиць, транспорту, камер управління, обігріву/охолодження громадських будівель). Така послуга дозволить ідентифікувати основні джерела енергоспоживання, встановити пріоритети і оптимізувати їхню роботу. Для забезпечення цієї послуги пристрої моніторингу енергоспоживання повинні бути інтегровані з енергосистемою міста.

Розумне паркування. Послуга інтелектуального паркування заснована на дорожніх датчиках і інтелектуальних дисплеях, які направляють автомобілістів по кращій дорозі для паркування в місті. Переваги, одержані від цієї послуги, різні: більш швидке визначення місця для паркування означає зменшення обсягу викидів вуглекислого газу від автомобіля, меншу завантаженість доріг і більше щасливих громадян. Послуга інтелектуального паркування може бути безпосередньо інтегрована в міську інфраструктуру IoT. Крім того, використовуючи технології зв'язку малого радіусу дії, такі як радіочастотні ідентифікатори або зв'язок ближнього поля, можна реалізувати електронну систему верифікації паркувальних місць, зарезервовану для машин швидкої допомоги, поліції, які можуть на законних підставах використовувати паркувальні місця.

Розумне освітлення. Оптимізація ефективності вуличного освітлення є важливим завданням. Зокрема, цей сервіс може оптимізувати інтенсивність освітлення вуличного ліхтаря залежно від часу доби, погодних умов і присутності людей. Для реалізації такого сервісу необхідно включити вуличні ліхтарі в інфраструктуру «розумного міста».

Моніторинг параметрів навколишнього середовища. Іншим важливим застосуванням технологій IoT є моніторинг параметрів навколишнього середовища у громадських будівлях (школах, музеях, офісах та інших установах) за допомогою різних типів датчиків і виконавчих механізмів, які керують освітленням, температурою і вологістю повітря. Контролюючи ці параметри, можна підвищити рівень комфорту людей.

Застосування IoT у сфері логістики є одним із пріоритетних напрямів у вирішенні питання скорочення витрат і затримки вантажів у дорозі, підвищення прозорості операцій і мінімізації участі людини у цих процесах.

Рішення та технології для підключення транспорту до мережі вже широко доступні по всій території України. Відстеження геопозицій і стану перевезених вантажів, моніторинг транспортної інфраструктури – перспективні напрями розвитку, які мають значний потенціал до зростання. Окремі технології IoT, такі

як автономний транспорт і роботизовані склади, поки не знайшли застосування в Україні, однак у середньостроковій перспективі їх впровадження неминуче.

Промисловий Інтернет Речей —це система об'єднаних комп'ютерних мереж і підключених до них промислових (виробничих) об'єктів з вбудованими датчиками і програмним забезпеченням для збору та обміну даними, з можливістю віддаленого контролю і управління в автоматизованому режимі, без участі людини.

На першому етапі впровадження ІоТ на промислове обладнання встановлюють датчики, виконавчі механізми, контролери та людино-машинні інтерфейси. В результаті стає можливим збір інформації, яка дозволяє керівництву отримувати об'єктивні і точні дані про стан виробництва. Оброблені дані надаються всім підрозділам підприємства. Це допомагає налагодити взаємодію між співробітниками різних підрозділів і приймати обґрунтовані рішення.

Отримана інформація може бути використана для запобігання позаплановим простоям, поламам устаткування, скороченню позапланового техобслуговування та збоям в управлінні ланцюжками поставок, тим самим дозволяючи підприємству функціонувати більш ефективно.

При обробці величезного масиву неструктурованих даних, що надходять з датчиків, їх фільтрація і адекватна інтерпретація стає пріоритетним завданням. Тому особливого значення набуває представлення інформації в зрозумілому користувачеві вигляді. Для цього використовуються передові аналітичні платформи, призначені для збору, зберігання і аналізу даних про технологічні процеси і події, що відбуваються в реальному масштабі часу.

Промисловий Інтернет Речей дозволяє створювати виробництва, які виявляються більш ощадливими, гнучкими і ефективними, ніж існуючі. Бездротові пристрої з підтримкою протоколу ІР, включаючи смартфони, планшети і датчики, вже активно використовуються на виробництві. Наявні дротові мережі датчиків в найближчі роки будуть розширені і доповнені бездротовими мережами, завдяки чому на підприємствах суттєво розширяться

зони застосування систем моніторингу та управління. Наступний етап оптимізації виробничих процесів буде характеризуватися все більш щільною конвергенцією кращих інформаційних і операційних технологій.

Питання та завдання до контролю знань студентів

1) Що таке "Інтернет речей"?

А це мережа підключених до Інтернету побутових речей з чи без датчиків і сенсорів;

Б це глобальна мережа підключених до Інтернету фізичних пристроїв - "речей", оснащених датчиками, сенсорами і пристроями для передавання інформації;

В це всі речі, які можна продати або купити через мережу Інтернет;

Г це куплені на розпродажах через мережу Інтернет речі, які мають сенсори і датчики.

2) Назвіть "мінуси" Інтернету речей.

А менше побутових проблем, більше часу на хобі;

Б прилади не мають жодних антивірусів чи навіть перевірки користувача;

В відслідковування стану здоров'я пацієнтів та надання швидкої допомоги за необхідності;

Г необхідна абсолютна надійність мережі, адже найменший її збій несе загрозу людині.

3) Інтернет речей...

А Internet of Things – IoT;

Б це мережа речей, які підключені до мережі Інтернет;

В це система, що об'єднує реальні речі у віртуальну мережу;

Г перетворює звичні для нас речі у нові пристрої, створюючи як розумні годинники, так і розумні міста

4) Середовище Інтернет речей не обмежується розумним будинком, воно може охоплювати...

А всю країну;

Б певну організацію;

В все місто;

Г всі країни.

5) Переваги Інтернету речей:

А безпека особистих даних;

Б стеження в режимі реального часу;

В збільшення продуктивності і безпеки виробничих процесів;

Г миттєвий контроль та реагування в складних автономних системах.

Розробив: Ланська С.С.

Розглянуто та схвалено
на засіданні циклової комісії
програмної інженерії

Протокол № 2 від 21.09.2023 р.

Голова комісії _____ Ланська С.С.

Лекція № 5

з навчальної дисципліни «Хмарні технології»

Тема: Поняття віртуалізації комп'ютерних систем та мереж. Огляд систем віртуалізації мереж, комп'ютерних ресурсів, додатків та сховищ даних.

Визначення віртуалізації рівня додатків та операційних систем.

Мета заняття: формування знань про віртуалізацію та розподілені віртуальні машини

Час – 2 години

План проведення лекції

- 1) Віртуалізація та її застосування.
- 2) Розподілені віртуальні машини

Література

- 1) Кононюк А.Е. Фундаментальная теория облачных технологий [Текст]: Книга 1. Общенаучные подходы формирования систем облачных технологий / А.Е. Кононюк. – К.: Освіта України, 2018 – 621 с.
- 2) Косинський В. І. Сучасні інформаційні технології: навч. посіб. для студ. ВНЗ / В. І. Косинський, О. Ф. Швець. – [2-ге вид., випр.]. – К.: Знання, 2012. – 318с.
- 3) Хмарні технології в освіті. Навчально-методичний посібник для студентів фізико-математичного факультету. – Житомир: ЖДУ, 2016. – 72 с.

Навчальні матеріали лекції

5.1 Віртуалізація та її застосування.

Віртуалізація — це створення віртуальної, а не фактичної, версії чогось, наприклад операційної системи (ОС), сервера, пристрою зберігання даних або мережевих ресурсів.

У віртуалізації використовується програмне забезпечення, яке імітує функціональність обладнання для створення віртуальної системи. Ця практика

дозволяє ІТ-організаціям керувати кількома операційними системами, більш ніж однією віртуальною системою та різними програмами на одному сервері. Переваги віртуалізації включають підвищення ефективності та економії масштабу.

Віртуалізація ОС — це використання програмного забезпечення, яке дозволяє частину обладнання одночасно запускати кілька зображень операційної системи. Ця технологія почала діяти на мейнфреймі десятиліття тому, що дозволяє адміністраторам уникати витрачання дорогих процесорних потужностей.

Як працює віртуалізація. Віртуалізація описує технологію, при якій додаток, гостьова операційна система (гостьова ОС) або сховище даних відлучаються від справжнього базового обладнання або програмного забезпечення. Ключове використання технології віртуалізації — віртуалізація сервера, яка використовує програмний рівень — гіпервізор — для емуляції базового обладнання. Це часто включає пам'ять процесора, введення/виведення (введення/виведення) та мережевий трафік. Гіпервізори беруть фізичні ресурси та відокремлюють їх, щоб їх можна було використовувати у віртуальному середовищі. Вони можуть сидіти поверх ОС або їх можна встановити безпосередньо на апаратне забезпечення. Останнє - це те, як більшість підприємств віртуалізують свої системи.

Гіпервізор Xen — це програма з відкритим кодом, яка відповідає за управління взаємодіями низького рівня, що виникають між віртуальними машинами (ВМ) та фізичним обладнанням. Іншими словами, гіпервізор Xen дозволяє одночасно створювати, виконувати та керувати різними віртуальними машинами в одному фізичному середовищі.

За допомогою гіпервізора гостьова ОС, яка зазвичай взаємодіє з справжнім обладнанням, зараз робить це за допомогою програмної емуляції цього обладнання; часто гостьова ОС не має уявлення, що це на віртуалізованому обладнанні. Хоча продуктивність цієї віртуальної системи не дорівнює продуктивності операційної системи, що працює на справжньому апаратному забезпеченні, концепція віртуалізації працює, оскільки більшість гостьових

операційних систем і додатків не потребують повного використання базового обладнання. Це дозволяє досягти більшої гнучкості, контролю та ізоляції, усуваючи залежність від певної апаратної платформи. Хоча спочатку призначена для віртуалізації сервера, концепція віртуалізації поширилася на додатки, мережі, дані та настільні комп'ютери.

Процес віртуалізації виконує наступні кроки:

- 1) Гіпервізори відокремлюють фізичні ресурси від їх фізичного середовища.
- 2) Ресурси беруть і поділяють за потребою від фізичного середовища до різних віртуальних середовищ.

- 3) Користувачі системи працюють і виконують обчислення у віртуальному середовищі.

- 4) Після запуску віртуального середовища користувач або програма може надіслати інструкцію, яка потребує додаткових ресурсів з фізичного середовища. У відповідь гіпервізор передає повідомлення фізичній системі і зберігає зміни. Цей процес відбуватиметься майже з рідною швидкістю.

Віртуальне середовище часто називають гостьовою машиною або віртуальною машиною. VM діє як один файл даних, який можна перенести з одного комп'ютера на інший і відкрити в обох; очікується, що це буде виконувати однаково на кожному комп'ютері.

Види віртуалізації. Напевно ви мало знаєте про віртуалізацію, якщо ви коли-небудь ділили свій жорсткий диск на різні розділи. Розділ — це логічний поділ жорсткого диска для створення, фактично, двох окремих жорстких дисків.

Є шість областей ІТ, де віртуалізація просувається:

- 1) *Віртуалізація мережі* — це метод об'єднання наявних ресурсів у мережі шляхом розподілу наявної пропускної здатності на канали, кожен з яких не залежить від інших і може бути призначений — або переназначений — на певний сервер чи пристрій у режимі реального часу. Ідея полягає в тому, що віртуалізація маскує справжню складність мережі, розділяючи її на керовані частини, подібно до того, що ваш розділений жорсткий диск полегшує управління вашими файлами.

2) *Віртуалізація пам'яті* — це об'єднання фізичного сховища з декількох мережних пристроїв зберігання даних у єдиний пристрій зберігання, яким керує центральна консоль. Віртуалізація сховища зазвичай використовується в мережах сховищ.

3) *Віртуалізація сервера* — це маскування ресурсів сервера - включаючи кількість та особу окремих фізичних серверів, процесорів та операційних систем - від користувачів сервера. Наміром є позбавити користувача від необхідності розуміння та управління складними деталями серверних ресурсів, одночасно збільшуючи обмін ресурсами та їх використання, а також підтримуючи можливість розширення.

Шар програмного забезпечення, що дозволяє цю абстракцію, часто називають гіпервізором. Найпоширеніший гіпервізор — Тип 1 — призначений для того, щоб сидіти прямо на голому металі та забезпечувати можливість віртуалізації апаратної платформи для використання віртуальними машинами. KVM віртуалізація — це гіпервізор віртуалізації на основі ядра Linux, який надає переваги віртуалізації типу 1, подібні до інших гіпервізорів. KVM ліцензується під відкритим кодом. Гіпервізору типу 2 потрібна операційна система хоста і частіше використовується для тестування та лабораторій.

4) *Віртуалізація даних* — це абстрагування традиційних технічних деталей управління даними та даних, таких як розташування, продуктивність чи формат, на користь ширшого доступу та більшої стійкості, пов'язаної з потребами бізнесу.

5) *Віртуалізація на робочому столі* — це віртуалізація завантаження робочої станції, а не сервера. Це дозволяє користувачеві віддалено отримувати доступ до робочого столу, як правило, використовуючи тонкого клієнта за столом. Оскільки робоча станція по суті працює на сервері центрів обробки даних, доступ до неї може бути і більш безпечним, і портативним. Ліцензію на операційну систему все ще потрібно враховувати, а також інфраструктуру.

6) *Віртуалізація додатків* — це відведення шару програми від операційної системи. Таким чином, додаток може працювати в капсульованому вигляді, не

залежаючи від операційної системи під ним. Це може дозволити додатку Windows запускатись на Linux і навпаки, крім додавання рівня ізоляції.

Віртуалізацію можна розглядати як частину загальної тенденції в ІТ підприємства, яка включає автономні обчислення, сценарій, в якому ІТ-середовище зможе керувати собою на основі сприйнятої діяльності та корисних обчислень, в яких потужність обчислювальної машини розглядається як утиліта, що клієнти можуть платити лише за потреби. Звичайною метою віртуалізації є централізація адміністративних завдань при одночасному покращенні масштабованості та навантаження.

Переваги. До переваг використання віртуалізованого середовища належать:

1) Зниження витрат. Віртуалізація зменшує кількість апаратних серверів, необхідних у компанії та центрі обробки даних. Це знижує загальну вартість придбання та обслуговування великої кількості обладнання.

2) Легше відновлення після катастроф. Відновлення стихійних лих у віртуальному середовищі дуже просте. Регулярні знімки дають сучасні дані, що дозволяє віртуальним машинам зробити резервне копіювання та відновлення. Навіть у надзвичайній ситуації віртуальну машину можна перемістити на нове місце за лічені хвилини.

3) Легше тестування. Тестування менш складне у віртуальному середовищі. Навіть якщо допущена велика помилка, тест не потрібно зупиняти і повертатися до початку. Він може просто повернутися до попереднього знімка і продовжити тест.

4) Швидше резервне копіювання. Резервні копії можуть бути зроблені як віртуального сервера, так і віртуальної машини. Автоматичні знімки робляться протягом дня, щоб гарантувати актуальність усіх даних. Крім того, віртуальні машини можна легко мігрувати між собою та ефективно перерозподіляти.

5) Підвищення продуктивності праці. Менше фізичних ресурсів призводить до меншого часу, витраченого на управління та обслуговування серверів. Завдання, які можуть тривати дні або тижні у фізичному середовищі, можна виконати за лічені хвилини. Це дозволяє співробітникам витрачати більшу

частину свого часу на більш продуктивні завдання, наприклад, на збільшення доходів та стимулювання ділових ініціатив.

Віртуалізація надає компаніям вигоду від максимізації їх виробництва. Додатковими перевагами для бізнесу та центрів обробки даних є:

1) Однодумні сервери. Віртуалізація забезпечує економічно ефективний спосіб розділення електронної пошти, баз даних та веб-серверів, створюючи більш всеосяжну та надійну систему.

2) Швидке розгортання та перерозподіл. Коли фізичний сервер виходить з ладу, резервний сервер може не завжди бути готовим або оновленим. Також може бути недоступним зображення або клон сервера. Якщо це так, то процес передислокації може бути трудомістким і виснажливим. Однак якщо центр обробки даних віртуалізований, то процес швидкий і досить простий. Віртуальні засоби резервного копіювання, такі як Veeam, доступні для прискорення процесу на кілька хвилин.

3) Зменшення тепла та покращення енергозбереження. Компанії, які використовують багато апаратних серверів, ризикують перегріти свої фізичні ресурси. Найкращий спосіб не допустити цього — зменшити кількість серверів, що використовуються для управління даними, і найкращий спосіб зробити це шляхом віртуалізації.

4) Краще для навколишнього середовища. Компанії та центри обробки даних, які використовують багато обладнання, залишають великий слід вуглецю; вони повинні брати на себе відповідальність за забруднення, яке вони створюють. Віртуалізація може допомогти зменшити ці ефекти, значно зменшивши необхідну кількість охолодження та потужності, тим самим допомагаючи очистити повітря та атмосферу. Як результат, компанії та центри обробки даних, які здійснюють віртуалізацію, покращать свою репутацію, а також підвищуючи якість їх відносин із клієнтами та планетою.

5) Легша міграція до хмари. Віртуалізація наближає компанії до переживання повністю хмарного середовища. Віртуальні машини можуть бути навіть розгорнуті з центру обробки даних для створення хмарної інфраструктури.

Можливість охопити хмаровий спосіб мислення за допомогою віртуалізації робить міграцію до хмари ще простішою.

6) Відсутність залежності від постачальника. Віртуальні машини є агностичними в апаратній конфігурації. Як результат, віртуалізація апаратного та програмного забезпечення означає, що компанії не потрібно залежати від постачальника цих фізичних ресурсів.

Обмеження. Перш ніж перейти до віртуалізованого середовища, важливо врахувати різні авансові витрати. Необхідні інвестиції у програмне забезпечення для віртуалізації, а також у апаратне забезпечення, яке може знадобитися для забезпечення віртуалізації, може бути дорогим. Якщо існуючій інфраструктурі більше п'яти років, потрібно буде врахувати бюджет первинного оновлення. На щастя, багато підприємств мають можливість пристосувати віртуалізацію, не витрачаючи великих грошей. Крім того, витрати можна компенсувати, співпрацюючи з керованим постачальником послуг, що надає щомісячні опції лізингу чи придбання.

Існують також питання ліцензування програмного забезпечення, які необхідно враховувати при створенні віртуалізованого середовища. Компанії повинні забезпечити чітке розуміння того, як їхні постачальники бачать використання програмного забезпечення у віртуалізованому середовищі. Це стає менш обмеженням, оскільки все більше постачальників програмного забезпечення адаптується до посиленого використання віртуалізації.

Перехід до віртуалізації потребує часу та може спричинити криву навчання. Впровадження та контроль віртуалізованого середовища вимагає від кожного ІТ-персоналу пройти навчання та володіти досвідом з віртуалізації. Крім того, деякі програми не дуже добре адаптуються, коли вони потрапляють у віртуальне середовище. ІТ-персонал повинен бути готовий до вирішення цих проблем і повинен вирішити їх перед перетворенням.

Існують також ризики безпеки, пов'язані з віртуалізацією. Дані мають вирішальне значення для успіху бізнесу і, отже, є загальною ціллю для атак. Шанси на порушення даних значно збільшуються під час використання

віртуалізації. Нарешті, у віртуальному середовищі користувачі втрачають контроль над тим, що вони можуть зробити, оскільки для виконання одного і того ж завдання існує декілька посилань, які повинні співпрацювати. Якщо якась частина не працює, то вся операція вийде з ладу.

5.2 Розподілені віртуальні машини

Віртуальна машина (VM) — це віртуальне середовище, яке функціонує як віртуальна комп'ютерна система з власним процесором, пам'яттю, мережевим інтерфейсом та сховищем, створеним у фізичній апаратній системі. Програмне забезпечення під назвою гіпервізор відокремлює ресурси машини від апаратного забезпечення та розподіляє їх відповідним чином, щоб вони могли використовуватись в машині управління.

Фізичне обладнання, оснащене гіпервізором, таким як віртуальна машина на базі ядра (KVM), називається хостом, тоді як багато віртуальних машин, які використовують його ресурси, є гостями. Гіпервізор розглядає обчислювальні ресурси (наприклад, процесор, пам'ять та сховище) як пул ресурсів, який легко переміщати між наявними гостями або на нові віртуальні машини.

VM ізольовані від решти системи, і декілька VM можуть існувати на одному апаратному забезпеченні, наприклад, на сервері. Їх можна переміщувати між хост-серверами залежно від попиту або використовувати ефективніше ресурси.

Операційна система в VM працює так само, як операційна система або додаток, як правило, на апаратному забезпеченні хоста, тому досвід користувача з VM буде подібним.

Після того, як ми визначилися, що таке віртуальна машина, ми можемо прокоментувати характеристики, які мають більшість віртуальних машин, які існують на даний момент.

Переважає більшість віртуальних машин, таких як Virtualbox або VMWare, дозволяють встановлювати практично будь-яку операційну систему, таку як Linux, Android, Mac OS X, Windows, Chrome OS тощо. Однак є й інші віртуальні

машини, такі як Virtual PC, Hiper -V або Parallels, які в основному призначені для віртуалізації Windows.

Кожна з операційних систем, яку ми віртуалізуємо, абсолютно не залежить від інших операційних систем. Таким чином, у випадку, якщо одна з віртуальних машин перестане працювати, решта продовжить працювати без будь-яких проблем.

Після того, як операційна система встановлена у віртуальній машині, ми повинні використовувати віртуалізовану операційну систему так само, як і ми, якби ми її встановили на комп'ютері.

Віртуальна машина має всі елементи, доступні для реального комп'ютера. Він має жорсткий диск, оперативну пам'ять, привід CD-ROM, мережеву карту, відеокарту тощо, але на відміну від реального комп'ютера, ці елементи, замість фізичного, є віртуальними.

Усі елементи віртуальної машини інкапсульовані у набір файлів. Це дозволяє нам копіювати віртуальну операційну систему з одного комп'ютера на інший, або ми можемо робити резервні копії без проблем і дуже легко і дуже швидко.

Пояснити детальну роботу віртуальної машини дуже складно, і мало хто має необхідні знання для цього. Однак приблизно можна сказати, що віртуальна машина — це програмне забезпечення, яке через шар віртуалізації спілкується з обладнанням, яке ми маємо в наявності в нашому комп'ютері, отримуючи таким чином емуляцію всіх компонентів реального комп'ютера. Таким чином віртуальна машина зможе емулювати жорсткий диск, оперативну пам'ять, мережеву карту, процесор тощо.

Коли ми знаємо це, коли ми відкриваємо віртуальну машину, таку як Virtualbox, у нас є графічне середовище, яке дозволить нам налаштувати та призначити ресурси для кожного з фізичних компонентів, які віртуальна машина емулює. Так, наприклад, практично на всіх віртуальних машинах ми повинні визначати деталі наступного типу:

- 1) Місце, яке ми хочемо призначити нашому жорсткому диску.

- 2) Оперативна пам'ять, яку ми хочемо призначити віртуальній машині.
- 3) Пам'ять нашої відеокарти.
- 4) Налаштування мережі, яка нам потрібна.

Після налаштування цих параметрів ми створимо віртуальну машину для встановлення операційної системи. Таким чином нам залишиться лише відкрити щойно створену віртуальну машину та встановити операційну систему так, ніби це справжній звичайний комп'ютер.

Надалі детально розглядаються пункти, які слід виконати для встановлення та використання операційної системи у віртуальній машині.

Утиліти та переваги, які ми можемо отримати від віртуальної машини, численні. Деякі з видів використання, які ми можемо надати віртуальним машинам, є наступними:

- для тестування операційних систем. Якщо ви все життя використовували Windows і хочете спробувати іншу операційну систему, наприклад, Linux Mint, ви можете це зробити через віртуальну машину. Також процес установки у віртуальну машину надзвичайно простий, оскільки нам не доведеться турбуватися про створення додаткових розділів на нашому жорсткому диску тощо;

- використовувати програмне забезпечення, яке недоступне в нашій операційній системі. Наприклад, якщо ми користувачі Linux і хочемо використовувати Photoshop, ми можемо це зробити через віртуальну машину;

- іноді доводиться використовувати програмне забезпечення, яке може працювати лише в застарілих операційних системах. Отже, якщо у нас є програма, яка може використовуватися лише в Windows 98, ми можемо створити віртуальну машину з Windows 98 та запустити та використовувати програмне забезпечення без проблем;

- ми можемо експериментувати в операційній системі, яка працює всередині віртуальної машини, роблячи речі, які ми б не наважилися зробити з нашою операційною системою, наприклад, застосувати оновлення програмного забезпечення, безпечно переглядати вебсторінку, яку ми вважаємо підозрілою тощо;

- ми можемо використовувати віртуальні машини як пісочницю для того, щоб, наприклад, виконувати шкідливі програми або відкривати підозрілі електронні листи в контрольованому та безпечному середовищі;

- ми можемо створити/моделювати комп'ютерну мережу лише одним комп'ютером. Ми можемо використовувати цю мережу віртуалізованих комп'ютерів для навчальних цілей і таким чином отримувати знання про мережеве адміністрування;

- якщо ви розробник програмного забезпечення, ви можете перевірити, чи правильно розроблена програма працює в декількох операційних системах;

- для перевірки альфа-версії, бета-версії та випуску кандидатських версій певних програм та операційних систем;

- щоб встановити веб-сервер, VPN-сервер, поштовий сервер або будьякий інший тип сервера;

- для тестування безлічі програм у Windows та запобігання забрудненню реєстру через установки та видалення програм.

Деякі з переваг, які надають віртуальні машини та віртуалізація, полягають у наступному:

- якщо сервер чи віртуалізована операційна система не налаштовані, відновити її надзвичайно просто, порівняно з реальною машиною. Якщо ми вживемо необхідних запобіжних заходів, ми зможемо відновити стан, у якому була віртуалізована операційна система або сервер, дуже легко і дуже швидко;

- якщо говорити про ділове середовище, віртуалізація операційних систем та серверів передбачає економію економії та значне місце. За допомогою віртуалізації ми уникаємо інвестицій у безліч фізичних засобів, економлячи гроші та простір;

- як ми щойно бачили, використання віртуальних машин передбачає наявність меншої фізичної оснащеності. Тому факт віртуалізації серверів або операційних систем може означати значну економію на обслуговуванні та споживанні енергії;

– завдяки віртуалізації та динамічному балансуванню ми можемо таким чином підвищувати тарифи на сервер. Якщо у нас є веб-сервер, ми можемо призначити додаткові ресурси для сервера, такі як оперативна пам'ять та процесор, у максимальні навантаження, щоб запобігти падінню сервера і, таким чином, підвищити швидкість обслуговування. Після завершення пікового навантаження ми можемо перенаправити ресурси, застосовані до веб-сервера, на іншу потребу. Тому, окрім підвищення тарифів на обслуговування, ресурси можна оптимізувати краще;

– якщо ми використовуємо віртуальну машину у виробничому середовищі, ми можемо розширити ресурси операційної системи чи сервера дуже простим способом. Нам просто потрібно отримати доступ до програмного забезпечення для віртуалізації та призначити більше ресурсів дуже простим способом;

– створити середовище для тестування всіх видів надзвичайно просто. Таким чином ми легко отримаємо тестове середовище, повністю ізольоване від решти систем;

– віртуальні машини та віртуалізація дозволяють легко та просто користуватися однією послугою на віртуалізованому сервері. Таким чином, навіть якщо один з віртуалізованих серверів буде скинутий, інший продовжить працювати.

Питання та завдання до контролю знань студентів

- 1) В чому суть віртуалізації?
- 2) Що таке віртуальна машина, хостова віртуальна машина, віртуальна машина рівня додатків, паралельна віртуальна машина?
- 3) Які програмні продукти відносяться до віртуальних машин?
- 4) Опишіть призначення гіпервізора.
- 5) Охарактеризуйте типи гіпервізорів (схематично або таблично).
- 6) Для чого використовують віртуальні машини?
- 7) Виділіть і поясніть переваги і недоліки віртуалізації

Розробив: Ланська С.С.

Розглянуто та схвалено
на засіданні циклової комісії
програмної інженерії

Протокол № 2 від 21.09.2023 р.

Голова комісії _____ Ланська С.С.

Лекція № 6

з навчальної дисципліни «Хмарні технології»

Тема: відмінності між віртуальними машинами та контейнерами.

Мета заняття: розглянути технології контейнерної та апаратної віртуалізації.

Час – 2 години

План проведення лекції

- 1) Відмінності між віртуальними машинами та контейнерами.
- 2) Області використання віртуальних машин та контейнерів.

Література

- 1) Кононюк А.Е. Фундаментальная теория облачных технологий [Текст]:

Книга 1 Общенаучные подходы формирования систем облачных технологий /
А.Е. Кононюк. – К.: Освіта України, 2018 – 621 с.

- 2) Косинський В. І. Сучасні інформаційні технології: навч. посіб. для студ.

ВНЗ /В. І. Косинський, О. Ф. Швець. – [2-ге вид., випр.]. – К.: Знання, 2012. – 318с.

- 3) Хмарні технології в освіті. Навчально-методичний посібник для студентів

фізико-математичного факультету. – Житомир: ЖДУ, 2016. – 72 с.

Навчальні матеріали лекції

6.1 Відмінності між віртуальними машинами та контейнерами.

Як віртуальні машини, так і контейнери можуть допомогти отримати максимальну віддачу від наявних ресурсів комп'ютерного обладнання та програмного забезпечення. Контейнери є новими дітьми в блоці, але віртуальні машини були і залишаються надзвичайно популярними в центрах обробки даних усіх розмірів.

Якщо ви шукаєте найкраще рішення для запуску власних служб у хмарі, потрібно розуміти ці технології віртуалізації, як вони порівнюються один з одним, і які найкращі способи використання для кожного з них. Ось наш короткий вступ.

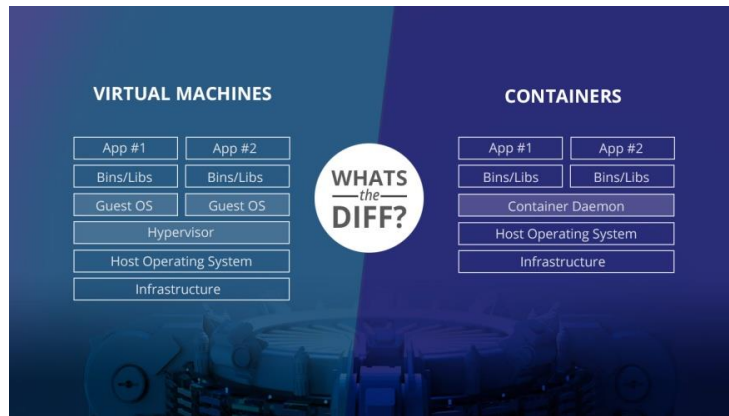


Рисунок 6.1 –Різниця між віртуальними машинами та контейнерами

Основні означення – віртуальні машини та контейнери.

Що таке віртуальні машини?

Віртуальна машина (VM, ВМ) — це емуляція комп'ютерної системи. Простіше кажучи, це дає змогу запускати, як здається, багато окремих комп'ютерів на апаратних засобах, які є фактично одним комп'ютером.

Операційні системи (ОС) та їхні програми спільно використовують апаратні ресурси з одного хост-сервера або з пулу хост-серверів. Кожна ВМ вимагає власної базової ОС, а її апаратне забезпечення є віртуалізованим. Гіпервізор або монітор віртуальної машини - це програмне забезпечення, прошивка або апаратне забезпечення, яке створює і запускає віртуальні машини. Він розташований між апаратним забезпеченням і віртуальною машиною і необхідний для віртуалізації серверу.

З часу появи доступних технологій віртуалізації та послуг хмарних обчислень, ІТ-відділи великих і малих підприємств використовуються віртуальні машини для зниження витрат і підвищення ефективності.

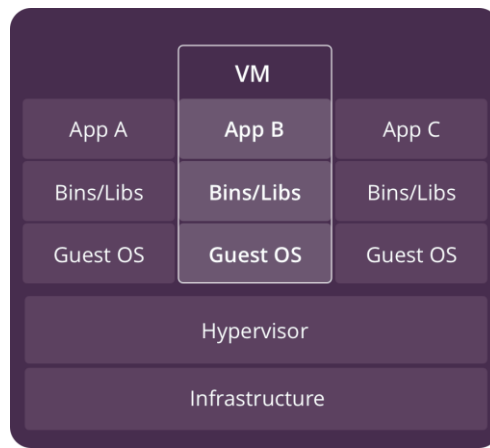


Рисунок 6.2 – Організація віртуальної машини

Однак віртуальні машини можуть зайняти багато системних ресурсів. Кожна віртуальна машина працює не тільки з повною копією операційної системи, але і з віртуальною копією всього обладнання, яке операційна система потребує для запуску. Це значно збільшує об'єм необхідної оперативної пам'яті і необхідний процесорний час. Це все ще економічно вигідно, порівняно з запуском окремих реальних комп'ютерів, але для деяких застосувань це може бути надмірним, що призвело до розробки контейнерів.

Переваги віртуальних машин.

Для застосунків (applications) доступні усі ресурси ОС:

- існують перевірені часом інструменти керування
- існують перевірені часом засоби безпеки
- Краще відомі засоби керування безпекою

Популярні постачальники віртуальних машин:

- VMware vSphere;
- VirtualBox;
- Xen;
- Hyper-V;
- KVM.

Що таке контейнери?

За допомогою контейнерів, замість віртуалізації базового комп'ютера, як віртуальної машини (VM), віртуалізована тільки сама ОС.

Контейнери працюють поверх фізичного сервера, в якого його хостова ОС типово є Linux або Windows. Контейнери розділяють між собою єдине ядро хостової ОС і, як правило, також бінарні файли і бібліотеки. Ці спільні компоненти доступні лише для читання. Спільне використання ресурсів ОС, таких як бібліотеки, значно зменшує необхідність відтворення коду операційної системи і означає, що сервер може запускати кілька робочих навантажень за допомогою однієї установленної операційної системи. Таким чином, контейнери є надзвичайно легкими — вони мають розмір порядку кількох десятків мегабайт і запускаються протягом кількох секунд. У порівнянні з контейнерами запуск віртуальних машин проходить протягом кількох хвилин, і вони займають на порядок більше пам'яті, ніж еквівалентний до них за функціями контейнер.

На відміну від віртуальних машин, все, чого вимагає контейнер — це наявна операційна система, підтримуючі застосунки і бібліотеки, а також системні ресурси для запуску певної програми. На практиці це значить, що на одному сервері з контейнерами ви можете помістити в два-три рази більше застосунків, ніж з використанням ВМ. Крім того, за допомогою контейнерів ви можете створити портативне, послідовне робоче середовище для розробки, тестування та розгортання.

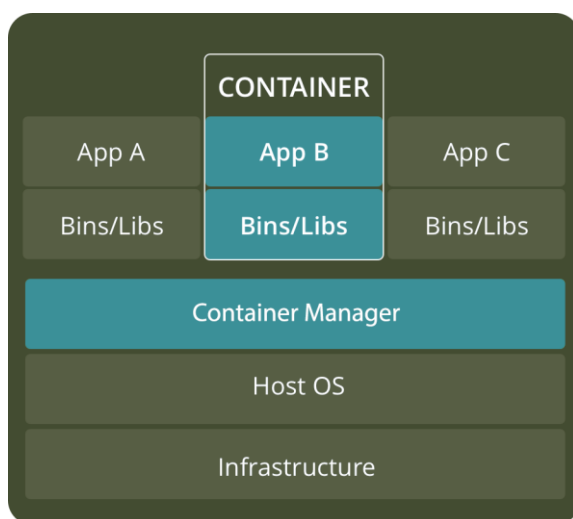


Рисунок 6.3 – Організація віртуальної контейнеру

Типи контейнерів.

Linux Containers (LXC). Оригінальна технологія контейнера Linux, відома як LXC. LXC — це метод віртуалізації на рівні операційної системи Linux для запуску декількох ізольованих систем Linux на одному хості.

Docker. Docker був запущений як проєкт з побудови контейнерів LXC з одним застосунком, вводячи кілька змін до LXC, які роблять контейнери більш портативними та гнучкими у використанні. Пізніше він перетворився на своє власне середовище виконання. На вищому рівні Docker - це утиліта Linux, яка може ефективно створювати, відправляти і запускати контейнери.

Переваги контейнерів:

- 1) Зниження ресурсів керування ІТ.
- 2) Зменшений розмір знімків.
- 3) Швидший запуск за стосунків.
- 4) Зменшені та спрощені оновлення безпеки.
- 5) Менше коду для передачі, міграції, завантаження робочих навантажень.

Популярні постачальники контейнерів:

- Linux Containers;
- LXC;
- LXD;
- CGManager;
- Docker;
- Windows Server Containers.

6.2 Області використання віртуальних машин та контейнерів

І контейнери, і віртуальні машини мають переваги та недоліки, і остаточне рішення буде залежати від ваших конкретних потреб, але є деякі загальні правила:

- 1) Віртуальні машини є кращим вибором для запуску додатків, які потребують усіх ресурсів та функціональності операційної системи, коли потрібно запускати декілька додатків на серверах або мати широкий спектр операційних систем для керування.

2) Контейнери є кращим вибором, коли найбільшим пріоритетом є максимізація кількості додатків, що працюють на мінімальній кількості серверів.

Різниця між віртуальними машинами та контейнерами наведено в таблиці 6.1.

Таблиця 6.1 – Відмінності віртуальних машин та контейнерів

Віртуальні машини	Контейнери
Важковаговий	Легковаговий
Обмежена продуктивність	Вихідна продуктивність
Кожна віртуальна машина працює у власній ОС	Усі контейнери поділяють хостову ОС
Віртуалізація на рівні обладнання	Віртуалізація на рівні ОС
Запускається порядку кількох хвилин	Запускається порядку кількох секунд або сотень мілісекунд
Виділяє необхідну пам'ять	Потребує менше пам'яті
Повністю ізольована і, отже, більш безпечна	Ізоляція на рівні процесу, можливо, менш безпечна

У більшості випадків, ідеальна установка, ймовірно, буде включати обидва варіанти. З поточним станом технологій віртуалізації гнучкість віртуальних машин і мінімальні вимоги до ресурсів контейнерів працюють разом, щоб забезпечити середовища з максимальною функціональністю.

Якщо у вашій організації працює велика кількість екземплярів тієї самої операційної системи, то слід перевірити, чи добре підходять контейнери. Вони просто можуть заощадити вам значний час і гроші ніж віртуальні машини.

Питання та завдання до контролю знань студентів

1) Які технології відносяться до технологій проектування глобальних мереж?

- A) Fast Ethernet;
- B) Frame Relay;
- B) FDDI;

Г) X.25;

Д) АТМ;

Е) ISDN.

2) Які бувають типи віртуальних каналів?

А) мережеві;

Б) комутовані;

В) постійні;

Г) високошвидкісні;

Д) розподілені.

3) Який з описів глобальних мереж є найкращим?

А) використовуються для об'єднання локальних мереж, розділених значними географічними відстанями;

Б) об'єднують робочі станції, термінали та інші пристрої, розташовані в межах міста;

В) об'єднують локальні мережі, розташовані в межах великого будинку;

Г) об'єднують автоматизовані робочі місця, термінали та інші пристрої, розташовані в межах будівлі.

4) Який з описів ISDN є найкращим?

А) це цифровий сервіс для передачі голосу і даних по існуючих телефонних лініях;

Б) забезпечує з'єднання маршрутизатор-маршрутизатор і хост-мережа як по синхронним, так і по асинхронним лініям зв'язку;

В) використовує високоякісне цифрове обладнання і є найшвидшим протоколом глобальних мереж;

Г) підтримує багатоточечні та двоточечні з'єднання, а також використовує символи кадру і контрольні суми.

Розробив: Ланська С.С.

Розглянуто та схвалено

на засіданні циклової комісії

програмної інженерії

Протокол № 2 від 21.09.2023 р.

Голова комісії _____ Ланська С.С.

Лекція № 7

з навчальної дисципліни «Хмарні технології»

Тема: розподілені комп'ютерні системи. Розподілені алгоритми.

Мета заняття: розглянути технології розподілених комп'ютерних систем та алгоритмів.

Час – 2 години

План проведення лекції

- 1) Розподілені комп'ютерні системи.
- 2) Розподілені алгоритми.
- 3) Організація мережі обчислювального кластеру.
- 4) Паралельна віртуальна машина (PVM).

Література

- 1) Кононюк А.Е. Фундаментальная теория облачных технологий [Текст]:

Книга 1 Общенаучные подходы формирования систем облачных технологий / А.Е. Кононюк. – К.: Освіта України, 2018 – 621 с.

- 2) Косинський В. І. Сучасні інформаційні технології: навч. посіб. для студ. ВНЗ / В. І. Косинський, О. Ф. Швець. – [2-ге вид., випр.]. – К. : Знання, 2012. – 318с.

- 3) Хмарні технології в освіті. Навчально-методичний посібник для студентів фізико-математичного факультету. – Житомир: ЖДУ, 2016. – 72 с.

Навчальні матеріали лекції

7.1 Розподілені комп'ютерні системи

Єдиного визначення розподіленої комп'ютерної системи (РКС) в даний час не тіснує. Із множини різних визначень можна привести іронічне висловлення Лесли Лампорта: «Розподіленою комп'ютерною системою можна назвати таку систему, в якій відмова комп'ютера, про існування якого ви навіть не підозрювали, може зробити ваш власний комп'ютер непридатним до

використання». Це визначення він дав в 1987р в листі колегам по поводу чергового відключення електроенергії в машинному залі.

Ендрю Таненбаум в своїй фундаментальній праці «Распределённые системы. Принципы и парадигмы» запропонував більш строге визначення: «Розподілена обчислювальна система (РОС) — це набір з'єднаних каналами зв'язку незалежних комп'ютерів, які з точки зору користувача деякого програмного забезпечення виглядають єдиним цілим». В цьому визначенні фіксуються два суттєвих моменти: автономність вузлів РКС і представлення системи користувачем, як єдиної структури. При цьому, основним зв'язуючим елементом РОС являється програмне забезпечення. В кінці минулого століття Ян Форстер визначив задачу розподілених обчислювальних систем як гнучке, безпечне, координоване розподілення ресурсів серед динамічних наборів користувачів, організацій і ресурсів. Він запропонував назвати такі розподілені обчислювальні системи терміном Грід. Розвиваючи ідею Грід, комерційні розробники в 2007-2008 р запропонували концепцію хмарних обчислень, в основі якої було надання високомасштабованих віртуальних обчислювальних ресурсів кінцевому користувачу через інтернет в вигляді послуг.

РКС представляє собою програмно-апаратний комплекс, орієнтований на вирішення певних задач. З однієї сторони кожний обчислювальний вузол являється автономним елементом, а з іншої програмна складова РОС має забезпечити користувачам видимість роботи з єдиною обчислювальною системою. В зв'язку з цим виділяють наступні важливі характеристики РОС, а саме можливість роботи з:

- різними типами пристроїв;
- різними постачальниками пристроїв;
- різними апаратними платформами;
- різними операційними системами.

Має бути забезпечена можливість простого розширення і масштабування, постійна доступність ресурсів і прихованість особливостей комунікацій від користувачів. Для забезпечення роботи гетерогенного обладнання РОС як

єдиного цілого, стек програмного забезпечення зазвичай розбивають на два шари. На верхньому шарі розміщуються розподілені додатки, відповідаючи за вирішення певних прикладних задач засобами РОС. Їх функціональні можливості базуються на нижньому шарі — проміжному програмному забезпеченні (ППЗ). ППЗ взаємодіє з системним ПЗ і мережним рівнем для забезпечення прозорості роботи додатків в РОС, як наведено на рисунку 7.1. Щоб РОС могла бути представлена користувачу як єдина система, використовують наступні типи прозорості в РОС:

- прозорий доступ до ресурсів — від користувачів має бути прихована різниця в представленні даних і в способах доступу до ресурсів РОС;
- прозоре місцезнаходження ресурсів — місце фізичного розташування потрібного ресурсу має бути несуттєвим для користувача;
- реплікація — приховання від користувача того, що в реальності існує більше однієї копії використовуємих ресурсів;
- паралельний доступ — можливість одночасного використання одного і того ж ресурсу різними користувачами незалежно один від одного. При цьому факт сумісного використання ресурсу має залишатися прихованим від користувача;
- прозорість відмов — відмова (відключення) яких-небудь ресурсів РОС не повинна впливати на роботу користувача і його додаток.

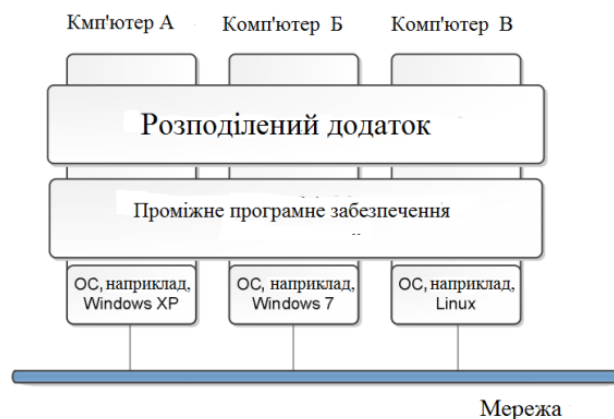


Рисунок 7.1 – Шари програмного забезпечення в РОС

Отже, організація обчислень в розподілених комп'ютерних системах ґрунтується на мережевих технологіях і має певні особливості. Структура розподіленої комп'ютерної системи складається з вузлів, об'єднаних за допомогою комп'ютерної мережі. В якості вузлів можуть використовуватися персональні комп'ютери (ПК), робочі станції, або багатопроцесорні комп'ютерні системи (КС). Кожний вузол РКС працює під керуванням власної операційної системи (КС з локальною пам'яттю використовують лише одну операційну систему, яка керує всіма процесорами). Структура РКС зображена на рисунку 7.2.



Рисунок 7.2 – Структура РКС

Взаємодія вузлів в РКС ґрунтується на посилянні повідомлень з використанням мережевих технологій, що потребує використання спеціальних програмних засобів — сокетів, віддалених методів, в яких задіяні протоколи, IP-адреси та інше.

Можливість об'єднати десятки тисяч і навіть сотні тисяч комп'ютерів дозволяє створювати дуже потужні розподілені системи, які легко розширюються, перебудовуються і значно дешевші.

Кластерні, мультікластерні та Грід — це типові розподілені обчислювальні системи. В розподілених системах спільне скоординоване використання ресурсів, різних за своєю структурою, підтримується відповідними технологіями та інфраструктурами.

У кластерах передбачена плавна зміна розмірів і подолання відмов: коли один підвузол у кластері відмовляє, то інший готовий зайняти його місце так, щоб додатки (включаючи і його користувачів) практично цього не помітили. Ранні кластери були більш-менш закритими системами; це були процесори із сильним зв'язком, підключені до загальної шини оперативної і зовнішньої пам'яті і колективно використовуючи дані на високих швидкостях. Розширення можливостей інформаційного обміну дозволили кластерам покинути середовище колективної шини й почати існування в значно більшому масштабі. Такий кластер залежить від якості й швидкості виділеного з'єднання для передачі даних. Глобальний кластер використовує відкритий Інтернет для передачі даних. Багатоядерні кластерні системи забезпечують два рівня обробки — на множині ядер і на множині вузлів. Це ускладнює розробку програмного забезпечення і потребує використання великої кількості взаємодіючих процесів у розподіленій програмі.

Сучасні РКС є мультиархітектурними і для них характерні ієрархічна організація і різні пропускні спроможності каналів зв'язку між їхніми ресурсами (вузлами, процесорами і їх ядрами). В архітектурному плані РКС це сукупність взаємодіючих елементарних комп'ютерів (ЕК) з засобами комунікації та зовнішніми пристроями. Кількість ЕК в РКС допускає варіювання від одиниць до сотень тисяч (наприклад, в MVS-15000BM їх 1148, в IBM Roadrunner – 122400, в IBM BlueGene – 884736). Ефективність виконання завдання суттєво залежить від досягнення мінімуму накладних витрат на міжмашинний обмін інформацією і дисбаланс завантаження ЕК. Програмне забезпечення для програмування в РКС ґрунтується на засобах, пов'язаних з особливостями передавання даних в комп'ютерних мережах. Такі засоби представлені в вигляді спеціальних бібліотек (WinAPI, PVM, MPI). Мови програмування Java, Ада, С# мають вбудовані засоби програмування для РКС, а поширення потреби в розробленні розподілених додатків привело до необхідності створення проміжного програмного забезпечення (middleware), яке виконує функції інтерфейсу між розподіленою програмою і розподіленою системою і яке дозволяє автоматизувати процес його

розроблення і виконання (CORBA, COM/DCOM). Гама засобів розроблення розподілених програм зараз досить значна і безперервно розширюється, включаючи сокети, віддалені процедури, віддалені об'єкти та ін.

Розподілена програма — це набір розділів, з яких складається програма і які розміщуються та виконуються на різних вузлах РКС. Модель клієнт-сервер є класичною для розподілених обчислень. Сервер — об'єкт, який надає послуги (або спільні ресурси) іншим об'єктам (обчислювальні сервери, сервери друку, файл-сервери, Web-сервери). Клієнт — це об'єкт, який звертається до серверу за послугами. Розроблення розподіленого додатку пов'язано з побудовою паралельного алгоритму розв'язання завдання, визначенням в ньому серверної і клієнтської частини, розроблення алгоритмів серверу і клієнта та організації взаємодії сервера і клієнта з використанням необхідних бібліотек, або мовних засобів.

7.2 Розподілені алгоритми

Алгоритмами є покрокові визначення потоку обчислень і інформаційного обміну. Сьогодні алгоритмічно обумовленими в мережних розподілених обчисленнях є три основні архітектурні моделі, які розрізняються по ступені синхронізації і роздільності:

- 1) Модель синхронної передачі повідомлень.
- 2) Модель асинхронної передачі повідомлень.
- 3) Модель із асинхронним спільним використанням пам'яті.

З алгоритмічної позиції системи можуть вважатися асинхронними, якщо немає фіксованої верхньої межі часу для доставки повідомлення, або часу на обробку між кроками. Наприклад, електронний лист може бути доставлено адресатові всього за кілька секунд, але це також може зайняти кілька днів, залежно від характеру мережі. Як такі асинхронні моделі не можуть, по визначенню, забезпечити надійні часові гарантії. З іншого боку, надійність асинхронних моделей може бути підвищена іншими їх властивостями, наприклад, стійкістю повідомлень.

Крім питань синхронності, предметом розподілених алгоритмів також є мережна топологія; визначення форм і границь динамічних мереж з метою ефективної передачі даних. Найпростіша модель для розуміння та організації інформації і потоку інформації в середовищі мережних розподілених обчислень показана на рисунку 7.3.



Рисунок 7.3 – Проста модель мережних розподілених обчислень

У цій моделі вихідні дані одного вузла стають входом іншого вузла. Взагалі, інформація може передаватися в обох напрямках. Проміжним ПЗ позначаються протоколи, фільтри, перетворювачі, брандмауери тощо. Проміжне ПЗ є центральним поняттям мережних розподілених обчислень і його можна розуміти як ПЗ, що з'єднує два і більше розділених застосувань. Як видно, проміжне ПЗ ортогональне (незалежне) стосовно мережних розподілених обчислень, що часто дає можливість використовувати проміжне ПЗ на одиночному вузлі і забезпечувати зв'язок між локальними додатками.

Передача повідомлень є основою всіх видів передачі даних, до яких відносяться як віддалений виклик процедури RPC (Remote Procedure Call), так і об'єкти, що виконуються. Коли вузол вирішує здійснити доступ, або спільно використовувати дані, відсутні в його пам'яті, але які перебувають у межах фізичної досяжності (тобто за межами високошвидкісної шини, чи внутрішнього механізму передачі даних), то необхідно передавати повідомлення. Хоча існує багато різних варіантів, ця проста концепція є фундаментальною складовою частиною мережних РО.

Передача повідомлень в мережних РО може приймати багато форм. Певний тип переданих даних повинен підходити для певної операції і це ж справедливо відносно форми їхньої передачі. Реалізація широкомовного розсилання повідомлення, наприклад, багато в чому відрізняється від передачі «один до одного». На природу повідомлень, безперечно, впливають часові співвідношення. Із простої передавальної моделі виникає складний набір реалізацій, що базується на трьох загальних поглядах на повідомлення:

- перетворення даних;
- синхронність;
- маршрутизація.

Це спричиняють різні особливості схем передачі повідомлень, які широко використовуються сьогодні.

Сокети.

Сокети — низькорівневий механізм, який застосовується для організації розподілених обчислень (запропоновані при створенні ОС UNIX). Сокет — своєрідний мережевий роз'єм на логічному рівні, який дозволяє передавати і приймати дані між комп'ютерами, з'єднаних мережею. Сокет дозволяє серверу обслуговувати клієнта, чекаючи на його підключення, а клієнту звертатися до сервера. Для створення сокету застосовують порт. Порт — абстрактне поняття, яке є пронумерованим сокетом на конкретному вузлі системи. Сервер «прослуховує» порт доти, поки клієнт не з'єднається з ним. Сервер може обслуговувати кількох клієнтів. При цьому оптимальний серверний додаток має бути реалізованим в вигляді кількох процесів (багатопотоковий сервер), кожен з яких відповідає за зв'язок зі своїм клієнтом. Механізм сокетів реалізовано в Java, Win32.

Віддалені процедури.

Механізм виклику віддаленої процедури (механізм RPC — Remote Procedure Call) реалізує високорівневу концепцію взаємодії розділів розподіленої програми. Зовнішньо віддалена процедура нічим не відрізняється від звичайної процедури. Особливість її полягає в тому, що розділ, в якому знаходиться

віддалена процедура (сервер) і розділ, де здійснюється виклик віддаленої процедури (клієнт), знаходяться в різних вузлах розподіленої системи, тому виклик та виконання віддаленої процедури пов'язано з трьома діями:

- передавання вхідних параметрів віддаленої процедури з вузла клієнта на вузол серверу;
- виконання віддаленої процедури на вузлі серверу (де вона знаходиться);
- повернення результату виконання віддаленої процедури з вузла серверу на вузол клієнта за допомогою вихідних параметрів.

Таким чином аргументи віддаленої процедури передаються в мережі в вигляді повідомлень між вузлами розподіленої системи. Клієнт ініціює виконання віддаленої процедури створенням і запуском нового процесу на вузлі серверу і чекає його завершення, після чого клієнт продовжує своє виконання (відмінність від механізму рандеву в тому, що останній потребує реалізацію програми серверу в вигляді активного модуля – процесу, а RPC дозволяє використовувати для розміщення віддаленої процедури пасивні модулі, наприклад, пакети).

Обчислювальний Кластер

РКС — розподілена інфраструктура (як правило, географічно), об'єднуюча багато ресурсів різних типів, доступ до яких користувач може отримати з будь-якого місця, незалежно від місця їх розташування. Основою РКС являються обчислювальні ресурси, представлені в вигляді кластерів ЕОМ, суперЕОМ і різних варіантів їх об'єднання. Можна сказати, що кластер — це група обчислювальних вузлів, об'єднаних швидкісною мережею, яка являє собою для користувача єдиний апаратний ресурс, а кластерні обчислення — це особлива технологія високопродуктивних обчислень, яка зародилася разом з розвитком комунікаційних засобів і стала прекрасною альтернативою використанню надпотужних комп'ютерів. Коли з'явилися доступні канали зв'язку з високою пропускною здатністю, з'явилась концепція віртуального суперкомп'ютера, де масштабна задача виконується сумісно в єдиній мережі кластером звичайних комп'ютерів. До розряду кластерних обчислень можна віднести всякі паралельні обчислення, де всі комп'ютери системи використовуються як один уніфікований

ресурс. Існує кілька типів кластерів: кластери високої доступності, MPP-кластери, високопродуктивні кластери, кластери розподіленої завантаження. Grid є різновидом кластерних систем. Вона має свої особливості, обумовлені масштабом. Обчислювальні вузли Грід розташовані на значній відстані один від одного і доступність того чи іншого із них не гарантована. Це накладає певні вимоги до керування ресурсами.

Структура Грід — це віртуальна організація, влаштована над простором реальних комп'ютерів, мереж, адміністративних зон. Задача віртуальної організації — створити середовище, де частини одного додатку, які вконуються на різних комп'ютерах, будуть взаємодіяти між собою і до системи можна динамічно підключати будь-який новий ресурс. При цьому їх комунікації не повинні залежати від середовища виконання. Незалежно від того, де відбувається обчислення: в рамках одного комп'ютера, в локальній чи глобальній мережі, об'єднуючій багато організацій — це повинно залишатись прозорим для додатку.

Обчислювальний кластер.

Це система, зазвичай, складається з одного серверного вузла і одного або більше клієнтських вузлів, з'єднаних за допомогою Ethernet, або деякої іншої мережі. Обчислювальний кластер, як правило, працює під управлінням однієї з версій ОС Linux — багатокористувацької операційної системи і обчислювальний кластер — це технологія кластеризації комп'ютерів, які працюють під управлінням ОС Linux на різновид паралельного віртуального суперкомп'ютеру. Ключем до поняття кластерних обчислень є ідея єдиного образу системи. У кластерах передбачена плавна зміна розмірів і подолання відмов: коли один підвузол у кластері відмовляє, то інший готовий зайняти його місце так, щоб додатки (включаючи і його користувачів) практично цього не помітили.

Кластер складається з окремих машин (вузлів) і об'єднуючої їх мережі (комутатора). Крім ОС, необхідно встановити та налаштувати мережні драйвери, компілятори, програмне забезпечення для підтримки паралельного програмування і розподілу обчислювального навантаження.

Вузли кластеру: підходящим вибором в даний момент є системи на базі процесорів Intel. Варто встановити на кожен вузол 2-4Gb оперативної пам'яті. Одну з машин виділити в якості центральної (консоль кластеру), куди встановити великий жорсткий диск, можливо, більш потужний процесор і більше пам'яті, ніж на робочі вузли. Робити консоль кластеру більш потужною машиною має сенс, якщо необхідно мати на цьому комп'ютері крім інтерфейсу командного рядка більш зручне операційне оточення, наприклад віконний менеджер (KDE, Gnome), офісні програми, програми візуалізації даних і т.п.

Будова кластеру

Розгортання кластеру — завдання не надто складне. Для цього підійде будь-який дистрибутив. Який саме з дистрибутивів Linux ставити в якості базової ОС — не має значення. Ubuntu, Mandriva, Alt Linux, Red Hat, SuSE. Вибір залежить тільки від уподобань користувача.

7.3 Організація мережі обчислювального кластеру

Мережа — це модульна і адаптуюча комутаційна система, яку можна налаштувати відповідно до самих різних вимог. Її модульність полегшує додавання нових компонентів, або переміщення існуючих, а адаптивність спрощує внесення змін і удосконалень. Мережа кластеру нічим принципово не відрізняється від мережі робочих станцій, тому в найпростішому випадку для побудови кластеру необхідні звичайні мережеві карти та хаби / комутатори, які використовувалися б при облаштуванні якогось комп'ютерного класу. Однак, у випадку кластеру є одна особливість. Мережа кластеру в першу чергу призначена не для зв'язку машин, а для зв'язку обчислювальних процесів. Тому чим вищою буде пропускна здатність мережі, тим швидше будуть виконуватись паралельні завдання, запущені на кластері, отже робочі характеристики мережі набувають першочергового значення.

Для побудови обчислювальних кластерів використовують найрізноманітніше мережеве обладнання. Але слід пам'ятати, що пропускна

здатність мережі, яка зв'язує вузли кластеру, у багатьох випадках виявляється вирішальною для продуктивності кластеру.

Використовуване мережне обладнання характеризують зазвичай двома параметрами:

- латентність;
- пропускна здатність.

Латентність — це середній час між викликом функції передачі даних і самою передачею. Час витрачається на адресацію інформації, спрацювання проміжних мережних пристроїв, інші накладні витрати, що виникають при передачі даних.

Фактична пропускна здатність і латентність не тільки характеризують кластер, але й обмежують клас задач, які можуть ефективно вирішуватися на ньому. Так, якщо завдання вимагає частої передачі даних, кластер, який використовує мережеве обладнання з великою латентністю (наприклад, Gigabit Ethernet), буде велику частину часу витрачати навіть не на передачу даних між процесами, а на встановлення зв'язку, в той час як вузли будуть простоювати, і ми не отримаємо значного збільшення продуктивності. Втім, якщо пересилаються великі обсяги даних, вплив періоду латентності на ефективність кластеру може знижуватися за рахунок того, що сама передача вимагатиме досить великого часу, може бути навіть у рази більшою за період латентності.

7.4 Паралельна віртуальна машина (PVM)

Основою обчислювального середовища кластеру є паралельна віртуальна машина (PVM). PVM — це пакет програм, який дозволяє використовувати пов'язаний в локальну мережу набір різномірних комп'ютерів, як один великий паралельний комп'ютер. Таким чином, проблема великих обчислень може бути досить ефективно вирішена за рахунок використання сукупної потужності та пам'яті великої кількості комп'ютерів.

У загальному випадку кількість завдань може перевершувати число процесорів, включених в PVM. Крім того, до складу PVM можна включати досить

різномірні обчислювальні машини, несумісні з систем команд і форматів даних. Отже, паралельною віртуальною машиною може стати як окремо взятий ПК, так і локальна мережа, що включає в себе суперкомп'ютери з паралельною архітектурою, універсальні ЕОМ, графічні робочі станції і все ті ж малопотужні ПК. Важливо лише, щоб процеси, які включаються до PVM-обчислювальних засобів були інформацією у використовуваному програмному забезпеченні PVM. Завдяки цьому користувач може вважати, що він спілкується з однією обчислювальною машиною, в якій можливе паралельне виконання багатьох завдань.

PVM дозволяє користувачам використовувати існуючі апаратні засоби, для вирішення більш складних завдань при мінімальній додатковій вартості. Головна мета використання PVM – це підвищення швидкості обчислень за рахунок їх паралельного виконання. Функціонування PVM засноване на механізмах обміну інформацією між завданнями, виконуваними в її середовищі. У цьому відношенні найбільш зручно реалізовувати PVM в рамках багатопроцесорних обчислювальних комплексів, виділивши віртуальній машині кілька процесорів і загальну або індивідуальну (залежно від умов) оперативну пам'ять. Використання PVM допускається як на багатопроцесорних комп'ютерах (SMP), так і на обчислювальних комплексах, побудованих за допомогою кластерної технології. При використанні PVM, як правило, значно спрощуються проблеми швидкого інформаційного обміну між завданнями, а також проблеми узгодження форматів представлення даних між завданнями, виконуваними на різних процесорах.

Ефективне програмування для PVM починається з того, що алгоритм обчислень слід адаптувати до складу PVM і до її характеристик. Це творче завдання, яке в багатьох випадках повинне вирішуватися програмістом. Крім завдання розпаралелювання обчислень з необхідністю виникає і завдання керування обчислювальним процесом, координації дій і завдань учасників цього процесу. Іноді для керування потрібно створювати спеціальну задачу, яка сама не беручи участь в обчисленнях, забезпечує узгоджену роботу решти завдань.

При паралельних обчисленнях необхідно програмувати спеціальні дії з координації роботи завдань, такі як процеси запуску задач на процесорах кластеру, управління обміном даних між завданнями та ін. Найбільш простий спосіб організації паралельного процесу виглядає наступним чином. Спочатку запускається одне завдання (master), яке в колективі завдань буде відображати функції координатора робіт. Це завдання виробляє деякі підготовчі дії, наприклад, ініціалізація початкових умов, після чого запускає інші завдання (slaves), яким може відповідати або той же виконуваний файл, або різні виконувані файли. Такий варіант організації паралельних обчислень переважно використовується при ускладненні логіки керування обчислювальним процесом, а також коли алгоритми, реалізовані в різних завданнях, істотно розрізняються, або є великий обсяг операцій (наприклад, введення - виведення), які обслуговують обчислювальний процес в цілому.

Отже резюмуючи розглянуті питання, кластери – це багатопроцесорні ЕОМ, побудовані на елементах серійного виробництва. Це відноситься не тільки до апаратури, але і до програмного забезпечення. Основою операційною системою для кластерів є ОС Linux. В свою чергу кластери можна розділити на два помітно різні по спроможності класи:

- кластери типу Beowulf, які будуються на базі звичайної локальної мережі ПЕОМ. Щоб отримати такий кластер, до мережі ПЕОМ потрібно додати тільки системне програмне забезпечення, яке безоплатно розповсюджується в інтернеті;

- монолітні кластери, все обладнання яких компактно розміщене в спеціалізованих шафах. Це дуже швидкі машини, кількість процесорів в яких може досягати сотень та тисяч. Процесори в монолітних кластерах не можуть використовуватися в персональному режимі.

Питання та завдання до контролю знань студентів

- 1) Розподілені обчислення. Компоненти програмного забезпечення РОС.
- 2) Зв'язок в РОС. Модель OSI.
- 3) Характеристика сучасних РОС.

4) Поняття паралельної віртуальної машини.

Розробив: Ланська С.С.

Розглянуто та схвалено

на засіданні циклової комісії

програмної інженерії

Протокол № 2 від 21.09.2023 р.

Голова комісії _____ Ланська С.С.

Лекція № 8

з навчальної дисципліни «Хмарні технології»

Тема: Грід-системи та технології. Архітектура Грід

Мета заняття: ознайомитись з концепціями Грід-систем та їх відмінностями від розподілених систем

Час – 2 години

План проведення лекції

- 1) Концепція Грід. Відмінності від традиційних розподілених систем.
- 2) Вимоги та характеристики грід-системи.
- 3) Стандарти побудови архітектури грід.

Література

- 1) Кононюк А.Е. Фундаментальная теория облачных технологий [Текст]: Книга 1. Общенаучные подходы формирования систем облачных технологий / А.Е. Кононюк. – К.: Освіта України, 2018 – 621 с.
- 2) Косинський В. І. Сучасні інформаційні технології: навч. посіб. для студ. ВНЗ / В. І. Косинський, О. Ф. Швець. – [2-ге вид., випр.]. – К. : Знання, 2012. – 318с.
- 3) Хмарні технології в освіті. Навчально-методичний посібник для студентів фізико-математичного факультету. – Житомир: ЖДУ, 2016. – 72 с.

Навчальні матеріали лекції

8.1 Концепція Грід. Відмінності від традиційних розподілених систем

Останні п'ятнадцять років є роками зародження та розвитку нового напрямку в інформаційних технологіях, назву якому (як традиційно вважається) дали у 1998 році Я. Фостер та К. Кессельман – «Грід» (англ. «Grid»). Grid як засіб сумісного використання обчислювальних потужностей та сховищ даних дозволяє вийти за межі простого обміну даними між комп'ютерами і, зрештою, перетворити їх глобальну мережу на свого роду гігантський віртуальний

комп'ютер, доступний у режимі віддаленого доступу з будь-якої точки, незалежно від місця розташування користувача.

Ідея використовувати мережу суперкомп'ютерів для вирішення задач, мабуть, зародилася набагато раніше (спроби робилися з 60-х років XX століття), однак зараз вона набула завершеної форми «концепції Grid». Традиційно, причетними до розвитку Grid-обчислень вважають фізиків ядерщиків — і дотепер їх потреба в обробці колосальних об'ємів дослідних даних є рушійною силою для реалізації програм по впровадженню Grid, згадати хоча б діяльність Європейського центру ядерних досліджень (CERN). Однак Grid має потенційно велику кількість й інших областей застосування, оскільки пропонує універсальний підхід до розв'язку проблеми нестачі обчислювальних ресурсів — адже очевидно, що в загальному випадку мережа суперкомп'ютерів є спроможною вирішити складніші задачі, ніж кожен з її складових вузлів окремо.

Найближче за змістом до «Грид» є поняття «power grid» — мережа електроживлення, розподілений ресурс загального користування, коли кожен може легко під'єднатися через розетку і використовувати стільки електроенергії, скільки йому потрібно. Аналогічно користувачі з допомогою Grid отримують можливість прямого підключення до віддаленої обчислювальної мережі, не цікавлячись, звідки беруться потрібні для роботи обчислювальні ресурси й дані, які для цього використовуються лінії передачі, паролі чи протоколи тощо. При цьому аналогом інфраструктури електричних мереж (ліній електропередачі, підстанцій, трансформаторів, диспетчерських пунктів та ін.) виступає Grid — інфраструктура з програмним Grid забезпеченням (ПГЗ), з допомогою якого виконується „віртуалізація” ресурсів. Розвиток ПГЗ починався від базових засобів, що підтримують дистанційний доступ до ресурсів, пройшов стадію окремих систем, їх пакетів і привів до створення платформ — взаємноузгоджених наборів засобів, здатних дати комплексне рішення завдання обслуговування Gridінфраструктури виробничого призначення.

Концепція Grid базується на наступних основоположних моментах:

- швидке та постійне збільшення продуктивності мікропроцесорів масового виробництва (сучасний персональний комп'ютер на базі процесора може зрівнятися за швидкістю обчислень із суперкомп'ютерами 10-літньої давнини);
- поява швидкісних оптоволоконних ліній зв'язку - сьогодні в розвинених країнах базові лінії зв'язку в мережі Інтернет мають пропускну здатність 10 Гб/с і вище, а підключення до мережі багатьох наукових організацій відбувається на швидкості в 1-2 Гб/с;
- феномен Інтернету, глобалізація процесу обміну інформацією й інтеграції світової економіки;
- безперервне вдосконалювання технологій і засобів інформаційної безпеки.

По суті, грід є «надбудовою» над Інтернетом. Її основне призначення – організація розподілених обчислень для рішення серйозних задач науки й технології, які вимагають більших обчислювальних ресурсів, а саме, потужності комп'ютерів, ресурсів зберігання даних, часу обчислень. На відміну від безструктурної мережі Інтернет, грід – чітко впорядкована система. Певною мірою грід можна умовно назвати обчислювальним Інтернетом. Користувач, підключаючись до гріда, одержує доступ до потужності тисяч машин, на яких він може здійснювати обчислення й зберігати як величезні масиви даних, так і інформацію, отриману в результаті їхньої обробки. У цій мережі приділяється першорядна увага проблемам безпеки – адже анонімність, зручна при спілкуванні в Інтернеті, може стати надзвичайно небезпечною при роботі з науковими або практичними даними. Запитуючи яку-небудь інформацію в глобальній базі даних гріда, користувач одержить вичерпну відповідь на питання про її достовірність, повноту й доступність.

Необхідно зазначити відмінність грід-технології від технології паралельних обчислень. Основними перешкодами для здійснення нетривіальних паралельних обчислень в грід-середовищі є нестабільність, погана передбачуваність часу відгуку на запит. Причому це пов'язано не лише з тим, що в комп'ютерних мережах інформаційні пакети проходять через безліч мережевих пристроїв, але і з

відмінностями в протоколах зв'язку, які використовуються у "зовнішніх" комп'ютерних мережах і для міжпроцесорного обміну всередині суперкомп'ютерів. Це не дозволяє ефективно організувати паралельні обчислення з інтенсивним обміном інформацією між процесорами, які виконують окремі підзадачі в gridсередовищі.

Грід-технологія не є технологією паралельних обчислень, вона призначена для віддаленого запуску окремих завдань на територіально розподілених ресурсах. Тому якщо громіздке завдання може бути розбите на ряд маленьких, незалежних (які не обмінюються жодними даними) частин, - grid-технологія виявляється особливо ефективним і відносно дешевим рішенням. Навпаки, суперкомп'ютери виявляються для таких обчислень невиправдано дорогими і неефективними. Існують і гібридні проекти, метою яких є досягнення максимальної швидкості обчислень за рахунок глобального розподілу цих обчислень між суперкомп'ютерами - при цьому grid координує використання різних суперкомп'ютерів, а власне нетривіальне розпаралелювання відбувається всередині суперкомп'ютера.

8.2 Вимоги та характеристики грід-системи

Є два основних критерії, що відрізняють Грід-системи від інших систем, що забезпечують роздільний доступ до ресурсів:

1) Грід-система координує розрізнені ресурси. Ресурси не мають загального центру керування, а грід-система займається координацією їхнього використання, наприклад, балансуванням навантаження. Тому проста система керування ресурсами кластера не є системою грід, тому що здійснює централізоване керування всіма вузлами даного кластера, маючи до них повний доступ. Грід-системи мають лише обмежений доступ до ресурсів, що залежить від політики того адміністративного домена, у якому цей ресурс перебуває.

2) Грід-система будується тільки на базі стандартних і відкритих протоколів, сервісів і інтерфейсів.

Додаткові властивості, які повинні мати грід-системи:

- гнучкість (можливість забезпечення поділюваного доступу потенційно до будь-яких видів ресурсів);
- масштабованість (працездатність при збільшенні чи зменшенні складу системи);
- гнучка і потужна підсистема безпеки (стійкість до атак зловмисників, забезпечення конфіденційності);
- можливість контролю над ресурсами (застосування локальних і глобальних політик і квот);
- гарантії якості обслуговування;
- можливість одночасної, скоординованої роботи з декількома ресурсами.

Найчастіше реалізації Грід-систем забезпечують роботу з такими типами ресурсів:

- обчислювальні ресурси – окремі комп'ютери, кластери;
- ресурси зберігання даних – диски і дискові масиви, стрічки, системи масового зберігання даних;
- мережні ресурси;
- програмне забезпечення – яке-небудь спеціалізоване ПЗ.

Є різниця між технологією грід і реалізаціями грід-систем. Технологія грід – це найбільш загальні і універсальні аспекти, однакові для будь-якої системи (архітектура, протоколи, інтерфейси, сервіси).

Нижче коротко перераховуються та пояснюються основні властивості грід, до яких відносять:

- спільне використання розподілених ресурсів. Воно відкриває можливості з співробітництва, яких були б важко досягти іншими засобами. Водночас виникають питання «справедливості» розподілу ресурсів, створення та управління віртуальними організаціями (ВО).
- об'єднання потужностей. Таким чином будується система з сумарною потенційною обчислювальною потужністю, що перевершує потужності її складових, і при цьому досягається більш ефективне використання апаратних

засобів (зменшується простоювання). Постають жорсткі вимоги до каналів зв'язку.

- віртуалізація. Включає в себе такі поняття, як: приховування (маскування) від користувача складності апаратної та програмної реалізації системи та її складових, географічних відстаней між вузлами, приналежності вузлів різним організаціям, створення ілюзії роботи з «віртуальним суперкомп'ютером».

- неоднорідність (гетерогенність). Типовий Grid складається з множини різноманітних апаратних засобів та різноманітного програмного забезпечення (і має успішно функціонувати в таких умовах). Децентралізоване управління. Немає одного єдиного «власника» всієї системи, що вимагає використання механізмів розподіленого управління.

- інтероперабельність. Функціональна сумісність роботи різних компонентів Grid та навіть різних Grid -інфраструктур базується на стандартизації інтерфейсів. Підходи, що не враховують стандартів, є мало перспективними.

- прозорість доступу. Grid має надавати доступ до ресурсів системи користувачам, не зважаючи на конкретну топологію мережі чи локальну реалізацію механізмів доступу до тих чи інших вузлів та їх ресурсів.

- масштабованість. Grid має забезпечувати механізми включення нових джерел ресурсів, користувачів, сховищ даних тощо без впливу на існуючих учасників: Grid повинен мати здатність динамічно реконфігуруватись. Безпека. Одна з головних вимог до Grid-системи – безпека доступу до ресурсів, що зумовлює обмежений набір дозволених операцій у авторизованих користувачів та програм.

8.3 Стандарти побудови архітектури грід

У останні декілька років технологія Grid еволюціонувала від ретельно конфігурованої інфраструктури, яка підтримувала виконання обмеженого числа додатків категорії Grant Challenge на високопродуктивній апаратурі, до динамічного середовища, розвиток якого направляє міжнародне співтовариство.

У міру становлення технології Grid реальністю до процесу її розвитку все більш залучається індустрія. Участь комерційних організацій прискорює розробку надійного програмного забезпечення, що підтримує середовища Grid за межами академічних лабораторій. У свою чергу, це впливає як на архітектуру Grid, так і на пов'язані з нею протоколи і стандарти. Пристосування до архітектури грід технології Web-сервісів принесло істотну користь та привело до появи дещо фрагментованого середовища розробки. Розробники програмного забезпечення і Grid-сервісів добиваються відповідності угодам і стандартам, широко поширеним в їх співтоваристві. Проте по різних політичних і технічних причинах є декілька точок зору, що змагаються, щодо того, як слід реалізовувати архітектуру, і на які стандарти потрібно спиратися. Це суперництво гальмує розробників програмного забезпечення Grid, оскільки вони не упевнені, що майбутні стандарти включатимуть ті, що використовуються сьогодні. Основною організацією по стандартизації Grid є Global Grid Forum (GGF www.ggf.org). Крім того, роботи по стандартизації ведуться в Organization for the Advancement of Structured Information Standards (OASIS www.oasis.org), World Wide Consortium (W3C www.w3c.org), Distributed Management Task Force (DMTF www.dmtf.prg), Web Services Interoperability Organization (WS-I www.ws-i.org), Internet2 (www.internet2.edu) і Liberty Alliance (www.projectliberty.org). Найбільш важливим стандартом, покликаним визначити загальну, стандартну і відкриту архітектуру Grid, є стандарт Open Grid Services Architecture (OGSA), що розвивається GGF. У березні 2004 р. була випущена перша версія стандарту (OGSA 1.0), а в червні 2005 р. вийшла друга версія стандарту. OGSA є сервіс-орієнтованою архітектурою, в якій специфікується набір розподілених обчислювальних патернів, що реалізуються з використанням Web-сервісів. Стандарт призначається для визначення всіх основних сервісів, які можуть використовуватися в додатках e-business або e-science, включаючи управління роботами і ресурсами, комунікації і безпеку. Робота по специфікації інтерфейсів сервісів, семантики, протоколів і інших технічних деталей надана різним робочим групам усередині GGF і іншим організаціям по стандартизації Grid.

Перша конкретизація OGSA була здійснена в документі Open Grid Services Infrastructure (OGSI), випущеному в липні 2003 р. Цей документ базувався на понятті Grid-сервіса, розширенні Web-сервіса, в якому забезпечувався стандартний набір механізмів для управління станом. У OGSI 1.0 визначається набір принципів і розширень для використання WSDL і XML Schema при організації Web-сервісів з підтримкою стану. Критики OGSI відзначали ряд проблем в цьому стандарті: дуже великий об'єм; потреба в розширенні стандартного WSDL; дуже сильна об'єктна орієнтованість. Це привело до виникнення руху за визначення альтернативної інфраструктури Grid, заснованої на чистих специфікаціях Web-сервісів.

20 січня 2004 р. HP, IBM, Fujitsu і Globus Alliance оголосили про випуск WS-Resource Framework (WSRF www.globus.org/wsrf). Цей документ складається з набору специфікацій для виразу зв'язку між ресурсами, що володіють станами, і Web-сервісами. У специфікаціях визначаються конкретні формати повідомлень і зв'язані визначення на XML. Остаточні результати були передані двом новим технічним комітетам OASIS: WSResource Framework (WSRF) TC і WS-Notification (WSN) TC. WSRF TC відповідає за стандартизацію специфікацій WS-Resource Lifetime (визначаються способи управління життєвим циклом ресурсу і специфікуються Web-сервіси для ліквідації ресурсу); WS-Resource Properties (визначаються способи запиту і модифікації ресурсів, що описуються XML-документами Resource Property); WS-ServiceGroup (визначаються способи представлення і управління колекціями Web-сервісів і/або WS-ресурсами); WS-BaseFaults (визначається базовий XML-тип, використовуваний при обміні повідомленнями в Web-сервісах для інформування про збої). WSN TC займається стандартизацією трьох специфікацій, що відносяться до інтерфейсів Web-сервісів: WS-BaseNotification (асинхронне сповіщення, що включає інтерфейси виробника і споживача); WS-BrokeredNotification (асинхронне сповіщення); WS-Topics (організація і категоризація тем для підписки).

Деякі організації, що ведуть або планують грід-проекти, користуються альтернативними специфікаціями, що включають Basic Profile (BP1.0) від WS-I,

Web Services Grid Application Framework (WS-GAF, North-East Regional e-Science Centre www.neresc.ac.uk/ws-gaf) і WS-I+ (Open Middleware Infrastructure Institute www.omii.ac.uk). Специфікація BP1.0 була опублікована в квітні 2004 р. і містила керівництво по використанню SOAP. WSDL і UDDI. У WS-GAF пропонується підхід, відмінний від OGSI, до розширення функціональності Web-сервісів для задоволення потреб Grid-додатків. У WSI+ указуються існуючі стандарти, які є потенційно сумісними із стандартами Grid, що розвиваються. Фактичним стандартом безпеки в Grid є Grid Security Infrastructure (GSI <http://forge.gridforum.org/projects/gsi-wg>). У двох нових проектах досліджуються альтернативні рішення, які можуть вплинути на стандарти GSI. У проектах GridShib (<http://grid.ncsa.uiuc.edu/GridShib>) і ESPGRID (<http://e-science.ox.ac.uk/oesc/projects>) створені нові механізми і стратегії розподіленої аутентифікації, що дозволяють віртуальним організаціям в Grid інтегруватися з традиційною інфраструктурою корпоративної безпеки.

Питання та завдання до контролю знань студентів

- 1) Що таке грід?
- 2) Яке призначення проміжного програмного грід забезпечення (middleware)?
- 3) Які базові грід-сервіси Ви знаєте?
- 4) Нащо вводиться сертифікат користувача?
- 5) Як відбувається взаємодія користувача з грід-системою?
- 6) Які європейські грід-інфраструктури і грід-проекти Ви знаєте?
- 7) Які параметри національної грід-інфраструктури в Україні Ви знаєте?

Розробив: Ланська С.С.

Розглянуто та схвалено

на засіданні циклової комісії

програмної інженерії

Протокол № 2 від 21.09.2023 р.

Голова комісії _____ Ланська С.С.

