

БАЗОВІ ПІДХОДИ ТА ОСОБЛИВОСТІ ВИКОРИСТАННЯ ХМАРНИХ, ТУМАННИХ І РОСИСТИХ ОБЧИСЛЕНЬ

Розглянуто базові підходи, що використовуються при побудові сучасних хмарних сервісів. Показано, що хмарні сервіси характеризуються мобільністю, меншими сукупними фінансовими витратами організації інфраструктури ЦОД, широкими можливостями по автоматизації роботи мережевих ресурсів та гнучкістю у роботі з проектами за умов реорганізації системи. Проведено аналіз переваг, недоліків та актуальних сфер застосування приватних, публічних та гібридних хмарних сервісів. Також, у роботі вказано на особливості використання базових моделей надання послуг хмарним сервісом: моделі надання апаратних ресурсів (IaaS), моделі надання інфраструктури (PaaS), моделі надання платформи (SaaS) та моделі надання програмного комплексу (SaaS). Розглянуто ефективність заходів з підтримки надійності роботи хмарного сервісу, що включають у себе забезпечення роботи серверів інфраструктури сервісу у режимі реального часу та надійне збереження даних на серверах. З цією метою побудовано алгоритм навантаження інфраструктури хмарного сервісу під час активної роботи з абонентами та показано переваги застосування схеми балансування апаратних ресурсів мережі. З метою забезпечення надійного збереження даних у випадку аварійної ситуації запропоновано використовувати код, що корегує помилки; також обраховано переваги кодування над простим дублюванням, як функцію, що визначає співвідношення між кількістю додаткових серверів в обох випадках. Проведено аналіз параметрів, що характеризують ефективність роботи сервісів даного типу та тенденції еволюції хмарних обчислень. Визначено переваги та недоліки використання туманних і росистих обчислень, як етапу розвитку хмарних обчислень. Показано, що туманні обчислення дозволяють підвищити мобільність хмарних обчислень, але у рамках даного підходу залишається залежність користувача від надійності інтернет-зв'язку. Методологію росистих обчислень показано як застосування у архітектурі хмарного сервісу функції збереження «гарячих» даних на накопичувачах, що знаходяться у внутрішній мережі користувача і працюють у режимі FTP-серверів гібридної хмари з розділення ресурсів на публічні (ком'юніті-сервери) і приватні, а також з анонімізацією апаратних ресурсів. Проведений аналіз показав, що туманні та росисті сервіси подібні до хмарних сервісів, але у даному випадку аспекти безпеки інфраструктури мережі, конфіденційності даних, анонімності ресурсів та миттєвої еластичності сервісів мають більшу вагу.

Ключові слова: хмарні обчислення, туманні обчислення, росисті обчислення, масштабування системи, автоматизація сервісів, збереження даних, балансування інфраструктури.

VOLODYMYR LABZHYSKIY

National Technical University of Ukraine "Igor Sikorsky Kyiv Polytechnic Institute"

FUNDAMENTALS AND PECULIARITIES OF USING CLOUD, FOG AND DEW COMPUTING

The basic approaches of modern cloud services building are considered. It is shown that cloud services are characterized by mobility, lower financial costs of organizing data center infrastructure, wide opportunities for automation of network resources and flexibility in working with projects in the context of system reorganization. The analysis of advantages, drawbacks and hot areas of application of private, public and hybrid cloud services is carried out. Also, in the paper discussed features of the use of basic cloud service delivery models: IaaS (Hardware as a service), PaaS (Infrastructure as a service), SaaS (Software as a service). The efficiency of measures to support the reliability of the cloud service is considered, including the provision of the work of infrastructure servers in real time and reliable data storage on the servers. To achieve this goal the algorithm for loading the infrastructure of cloud service during active work with subscribers is constructed and the advantages of using the model of network hardware resources balancing are shown. In order to ensure reliable data storage in the event of an emergency, it is proposed to use a code that corrects errors; also, the advantages of encoding over simple duplication are calculated, as a function that determines the relationship between the number of additional servers in both cases. Efficiency of the services of this type and evolution trends of cloud computing was analyzed. The advantages and disadvantages of fog and dew computing application as a stage of development of cloud computing were determined. It is shown that fog computing can increase the mobility of cloud computing, but within the framework of this approach, the user's dependence on the reliability of Internet communication remains. The methodology of dew computing is shown as the application in the cloud service architecture of the function of storing "hot" data on the drives located in the user's internal network and operating in the mode of the FTP servers of the hybrid cloud from the division of resources into public (community servers) and private, and also with the anonymization of hardware resources. The analysis showed that fog and dew services are similar to cloud services, but in this case aspects of network infrastructure security, confidentiality of data, anonymity of resources and rapid elasticity are of greater importance.

Keywords: cloud computing, fog computing, dew computing, system scaling, service automation, data storage, infrastructure balancing

Вступ. Хмарні обчислення на сьогоднішній день є типовим засобом мережевих сервісів, що дає можливість надання кінцевим користувачам обчислювальні ресурси та ресурси інформаційних сховищ. Апаратні ресурси при цьому надаються користувачам без інформації про інфраструктуру хмари і виплати за них відповідають їх об'єму. Хмарні сервіси характеризуються мобільністю і, таким чином, ефективність їх застосування при достатній перепускності інтернет-каналу не залежить від локації користувача, меншими сукупними фінансовими витратами, а отже можливістю вивільнення більшої частини апаратних ресурсів організації, широкими можливостями по автоматизації роботи мережевих ресурсів інформаційних систем (МРІС) та потенціалом для побудови інфраструктури організації у відповідності до задач та розробленого програмного забезпечення (ПЗ), а не корегуванням задач і спрощенням ПЗ у відповідності до наявних

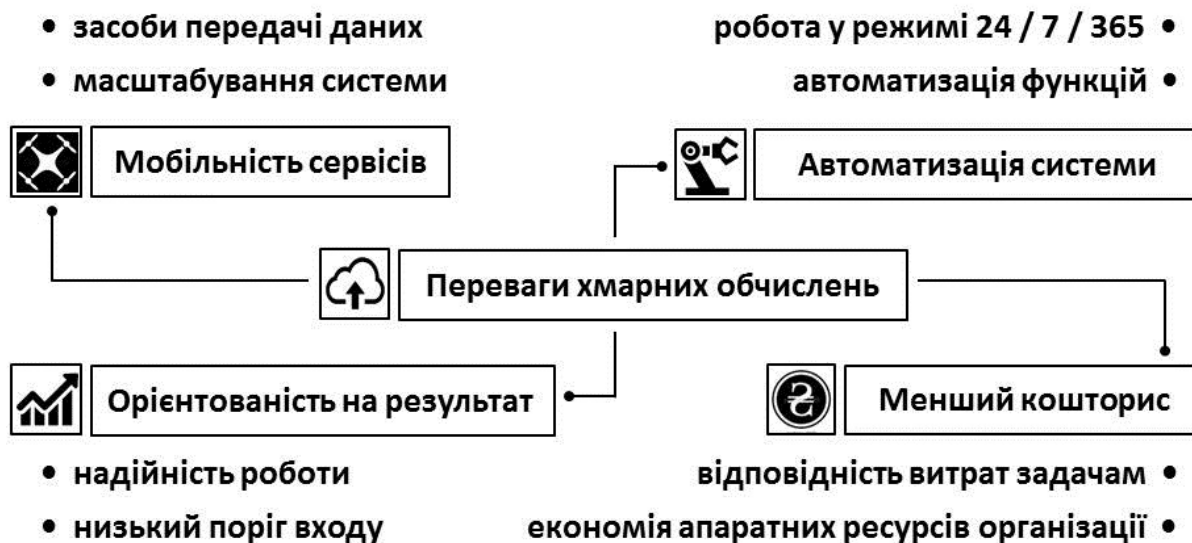


Рис. 1. Переваги застосування хмарних сервісів при побудові мережі організації

Таким чином, базова класифікація особливостей хмарних сервісів, що розглядаються, як переваги при проектуванні мережевої структури організації, може бути представлена у вигляді наступних груп[1-3]:

- наявність простих і мобільних засобів передачі даних;
- широкі можливості масштабування системи;
- низький вхідний поріг налаштування інформаційної мережі;
- надійність збереження даних;
- функціонування системи у режимі 24/7/365;
- високий рівень автоматизації сервісів;
- зменшення вартості використання апаратних ресурсів;
- відповідність рівня витрат складності поставлених задач.

Гнучкість хмарних сервісів, широкі можливості, що вони надають та низький вхідний поріг підключення до них визначає їх ключові переваги та тенденцію до розвитку і розповсюдження даного роду послуг.

1. Організація структури хмарного сервісу

Хмарні обчислення організуються шляхом об'єднання у мережу серверів, що створюють структури центрів обробки даних (ЦОД) та надають послуги інформаційних сховищ, віртуальних десктопів, платформ для використання програмних засобів, тощо (рис. 2). Сервери, робочі станції та портативні і планшетні персональні комп'ютери (ПК) кінцевих користувачів підключаються до хмарних сервісів через мережеві комутатори і використовують апаратні і програмні ресурси хмар у відповідності до рівня виплат організації[2, 3, 4-8].

Великі підприємства часто створюють власні сервіси хмарних обчислень, так звані приватні хмари, що знаходяться в межах корпоративної мережі. При цьому керування приватною хмарою і розташування її інфраструктури може як знаходитись в межах організації, так і бути доручено зовнішньому підряднику. Слід зазначити, що найбільш безпечна організація такого сервісу можлива у тому випадку, коли хмара розгорнута на території організації обслуговується та контролюється її співробітниками [4. 6].

Натомість, модель публічної хмари передбачає використання послуг, що надаються постачальником і знаходяться за межами корпоративної мережі [3, 7]. У даному випадку постачальник приймає на себе всі обов'язки по налаштуванню інфраструктури, управлінню сервісом, обслуговуванню програмного забезпечення, а клієнти платять за ті ресурси, які використовують. Абонентами публічних хмар часто стають невеликі компанії і індивідуальні користувачі, що пов'язано з їх фінансовою доступністю, простотою розгортання МРІС з можливістю їх подальшого масштабування. Недоліком даної моделі є стандартні конфігурації, пов'язані з статистикою найбільш розповсюджених випадків використання даного сервісу. Таким чином, у користувачів публічних хмар залишається менше можливостей по вибору конфігурації, що певною мірою нівелює розглянуті вище переваги хмарних обчислень. Слід також підкреслити, що, споживачі у даному випадку мало контролюють інфраструктуру хмарного сервісу, щодо заходів безпеки і відповідності сервісу нормативним вимогам.



Рис. 2. Базова схема взаємодії користувачів з хмарними сервісами

Останнім часом набувають популярності гібридні хмарні сервіси [8]. Гібридні хмари зазвичай створюються підприємством, а обов'язки по їх управлінню розподіляються між підприємством і постачальником публічної хмари. Гібридна хмара надає послуги, одна частина яких відноситься до приватних, а інша частина — до загальнодоступних послуг. Дана модель є найбільш актуальною для організацій з сезонними періодами активності. Основним недоліком гібридного хмарного сервісу є складність його проектування та внесення змін у його структуру. Необхідність регулярних взаємодій між приватним і загальнодоступним компонентами суттєво ускладнює дану архітектурну концепцію.

Таким чином, ефективність використання моделі хмарного сервісу [4-8] цілком залежить від галузі її застосування (табл. 1).

Табл. 1.

Особливості застосування базових моделей організації хмарних сервісів

	Модель організації хмарного сервісу		
	Приватна хмара	Публічна хмара	Гібридна хмара
Переваги	Контроль та безпека Вибір конфігурації	Низький вхідний поріг налагодження сервісу Відповідність витрат задачам	Контроль та безпека Вибір конфігурації Продаж апаратних ресурсів
Недоліки	Високий кошторис Значні витрати на запуск системи	Стандартні конфігурації Відсутність контролю	Складність організації системи
Рекомендації	Великі корпорації ІТ-компанії	Малі компанії Індивідуальні користувачі	Організації з сезонними періодами активності

*узагальнено автором за [4-8]

Незалежно від моделі при побудові хмарного сервісу необхідно пам'ятати про наступні вимоги до умов його функціонування:

- широкий мережевий доступ;
- самообслуговування за вимогою (on-demand self-service);
- миттєва еластичність (rapidelasticity);
- об'єднання ресурсів у пули (resourcepooling);
- керований сервіс (measuredservice).

Високий рівень перепускності інтернет-каналу, що надається мережевими комутаторами, є найбільш очевидною вимогою і «вузким місцем», що призвело до еволюції хмарних сервісів у туманні та росисті.

При цьому самообслуговування за вимогою означає, що при наявності достатньо широкого мережевого доступу абонент використовує апаратні ресурси хмари без звернення до провайдера. Зазначені ресурси мають об'єднуватися у пули, що включають у себе фізичні та віртуальні ресурси, з метою обслуговування великою кількістю споживачів у рамках багатоарендної (multi-tenant) схеми. Наявність миттєвої еластичності сервісу вказує на те, що ресурси можуть бути автоматично вивільнені і надані для швидкого масштабування системи у відповідності до запиту користувача та згідно з системою контролю керованого сервісу [4-6].

Моделі послуг хмарного сервісу поділяються на наступні:

- модель надання апаратних ресурсів (HaaS: Hardware as a service) та
- модель надання інфраструктури (IaaS: Infrastructure as a service);
- модель надання платформи (PaaS: Platform as a service);
- модель надання програмного комплексу (SaaS: Software as a service).

Переваги та недоліки даних моделей, а також рекомендації і приклади їх застосування представлені у табл. 2.

Табл. 2.

Особливості застосування базових моделей надання послуг хмарним сервісом

	Модель надання послуг хмарним сервісом		
	IaaS та HaaS	PaaS	SaaS
Переваги	Ефективне використання ресурсів Автоматичне масштабування	Плавна розгортка нових версій ПЗ	Низький рівень витрат Ітеративне оновлення програмного комплексу
Недоліки	Довгострокові витрати Організація безпеки	Організація безпеки Зниження контролю над апаратними ресурсами	Відсутність контролю над апаратними і програмними ресурсами
Рекомендації	Великі корпорації	ІТ-компанії	Малі компанії Індивідуальні користувачі
Приклади	IBM SmartCloudEnterprise, Amazon EC2, Windows Azure, GoogleCloudStorage, VMWare	IBM SmartCloudApplicationServices, AmazonWebServices, GoogleAppEngine	Gmail, GoogleDocs, Netflix, IBM LotusLive, Unyte, Salesforce.com, WebEx

*узагальнено автором за [4-6]

У рамках моделі HaaS користувач забезпечується обладнанням (сервери, мережеве обладнання, накопичувачі та ін.), на основі якого розгортає свою власну інфраструктуру, а у подібній до HaaS схемі IaaS апаратні ресурси надаються у відповідності з масштабуванням системи. Споживач при цьому не здатний керувати інфраструктурою хмари, але має контроль над операційною системою, протоколами зберігання та захисту даних, а також йому надається вибір мережевих компонентів. Таким чином, споживач організує захист платформ і додатків, а провайдер хмарного сервісу — захист інфраструктури. З іншого боку PaaS будується шляхом додання до моделі IaaS операційної системи (ОС) та інтерфейсу програмування API (Application Programming Interface). Споживач при цьому не може керувати базовою інфраструктурою хмари (мережами, серверами, ОС та інформаційними сховищами), але натомість отримує контроль над додатками і параметрами конфігурації середовища хостингу. Найбільш від IaaS відрізняється модель SaaS у рамках якої користувач отримує доступ до ПЗ через мережевий доступ. У даній схемі питання оновлення та отримання ліцензій на ПЗ регулюються провайдером, а виплати здійснюються за фактичне використання ПЗ.

2. Забезпечення надійності роботи хмарного сервісу

Базові заходи з підтримання надійності роботи хмарного сервісу включають у себе забезпечення роботи серверів інфраструктури сервісу у режимі реального часу та надійне збереження даних, у тому разі у випадку аварійної ситуації, що виникла внаслідок технічних проблем, або зловмисних дій [1-5].

Навантаження інфраструктури хмарного сервісу під час активної роботи з абонентами визначається через показник навантаження окремих серверів:

$$L = K_z + \frac{C \cdot N}{f \cdot M} \quad (1)$$

де K_2 — коефіцієнт завантаження серверу операційною системою і додатковими процесами, що не пов'язані з запитами абонентів, C — кількість циклів однієї сесії роботи сервісу, M — коефіцієнт, пов'язаний з кількістю центральних процесорів (ЦП) у кожному сервері та кількістю ядер у кожного процесору, а f — тактова частота ЦП.

Рівняння (1) дозволяє розрахувати завантаження кожного серверу як функцію від кількості користувачів хмарного сервісу та визначити умови перезавантаження всієї інфраструктури. Очевидним рішенням, що надає можливість уникнути випадку перезавантаження інфраструктури є вкладання коштів у апаратні ресурси сервісу: збільшення тактової частоти ЦП, використання багатоядерних ЦП і багатопроцесорних серверів, а також збільшення кількості серверів. Але практика показує, що більш ефективним є застосування методу балансування, що використовується у рамках стратегії миттєвої еластичності сервісу.

Методику балансування апаратних ресурсів мережі хмарного сервісу можна показати на прикладі, представленому на рис. 3. На представленій схемі розглядається механізм обробки 8000 запитів абонентів чотирма серверами S1, S2, S3 і S4, де S1 є найбільш потужним сервером, що здатний обробляти до 4000 запитів абонентів, межа перепускності серверу S2 складає 2000 запитів, а для серверів S3 і S4 — 1000 запитів. При організації інфраструктури хмарного серверу без застосування засобів балансування кожен сервер отримує рівномірне навантаження (рис. 3а) і, таким чином, перепускність всієї мережі визначається перепускністю найбільш слабких серверів S3 і S4 (рис. 3в). При застосуванні схеми, що включає у себе засоби балансування, розподіл навантаження на кожний окремий сервер автоматично визначається відповідно його потужності (рис. 3б), і, таким чином, перепускність інфраструктури хмарного серверу визначається сумарним значенням перепускності окремих серверів (рис. 3в). Отже засоби балансування виконують основне завдання стратегії миттєвої еластичності: визначення пропорцій розподілу запитів між серверами і їх динамічне корегування при зміні умов функціонування системи

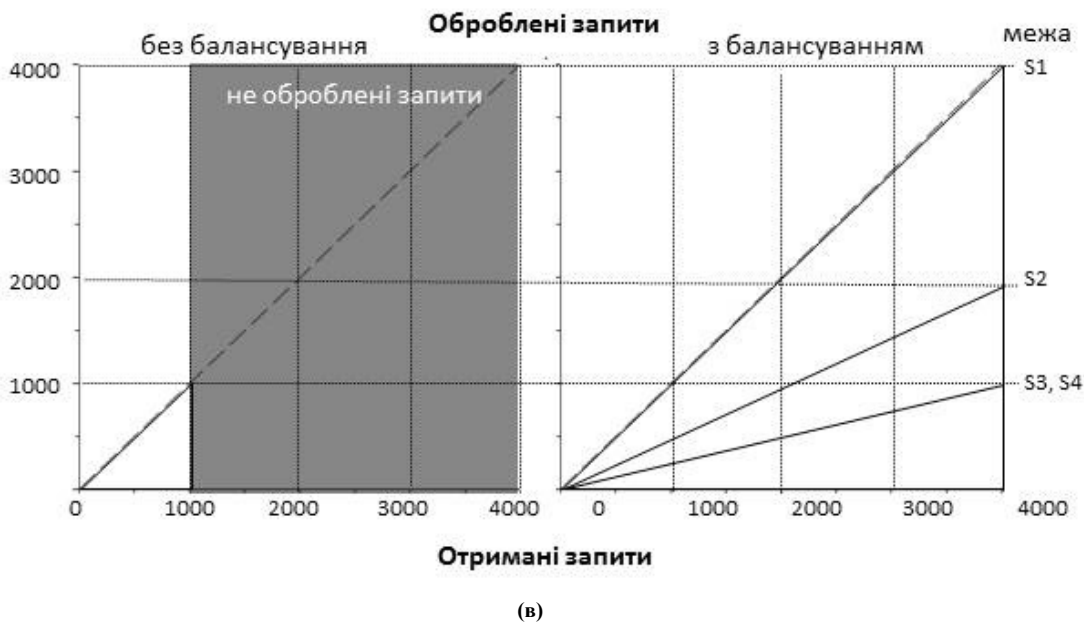
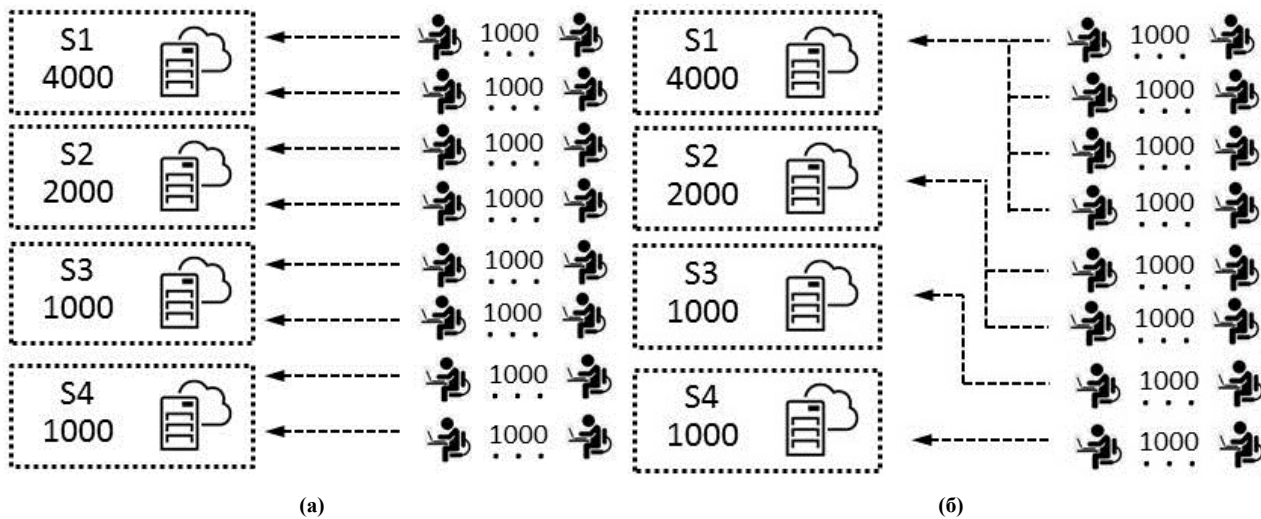


Рис. 3. Балансування навантаження серверів хмарного сервісу

Для вирішення задачі балансування у рамках інфраструктури окремого хмарного сервісу необхідно ввести поняття відносного рейтингу окремого i -го серверу R_i . Дана величина розраховується на основі даних про кількість запитів, що надходить на даний сервер (N_i), загальну кількість запитів, що надходять до мережі ($N_T = \sum_{i=1}^m N_i$), та рівня завантаження серверу ЦП серверу L_i :

$$R_i = \frac{N_i}{L_i \sum_{i=1}^m N_i} \quad (2)$$

Величина відносного рейтингу дозволяє автоматично у режимі реального часу обрахувати оптимальний рівень навантаження, для кожного серверу у інфраструктурі (рис. 4), попередньо, співвідносячи завантаження окремого серверу L_i і усереднений показник завантаження для хмарного серверу $L_i = L_i / \sum_{i=1}^m L_i$. Застосування даної моделі показано на рис. 4, результат відповідає рішенням задачі балансування, що було виконано вище аналітичним шляхом (рис. 3в)

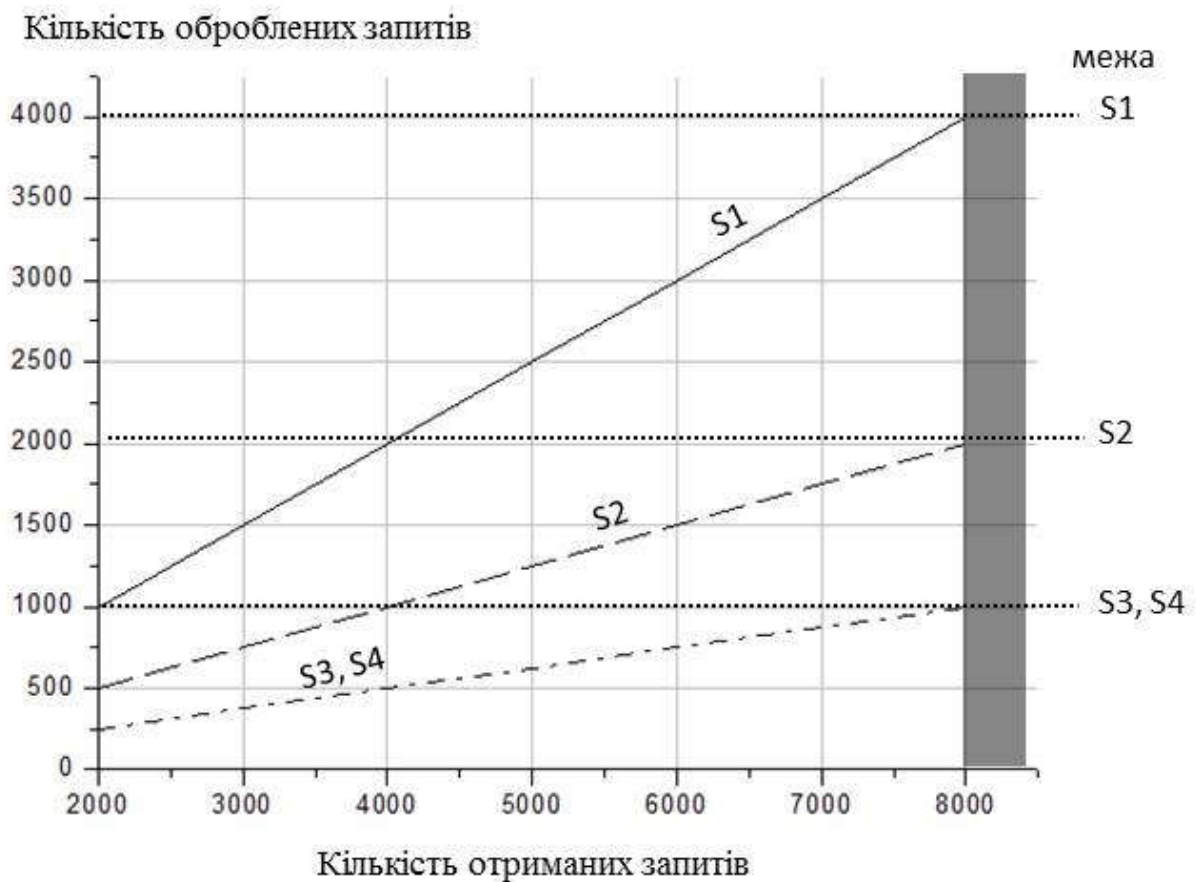


Рис. 4. Розподіл запитів між серверами хмарного сервісу при балансуванні навантаження

Розглянемо ефективність роботи даних засобів при виникненні аварійної ситуації, коли у представленій схемі виходить з ладу найбільш потужний сервер S1, що брав на себе половину навантаження системи (рис. 5).

Схема де не використовується балансування навантаження на серверу така ситуація призведе до втрати половини запитів. З іншого боку при балансуванні падає лише перепускність хмарного сервісу (рис. 5). Таким чином, розроблений алгоритм дозволяє перерозподілити пропорції обробки запитів між серверами так, щоб зберегти рівномірне завантаження інфраструктури, що змінюються під час функціонування розподіленої системи. Представлений випадок також дозволяє надати рекомендації по організації апаратних ресурсів, вказати на необхідність збільшення кількості серверів, щоб окремих сервер не відповідав за значну частину апаратних ресурсів системи.

Тим не менш, методика балансування не надає засобів для забезпечення надійності збереження даних на серверах у випадку їх виходу з ладу. Простим методом у даному випадку є засоби реплікування, що полягають у дублюванні даних шляхом кратного збільшення кількості серверів у інфраструктурі хмарного сервісу. Але така методика вимагає значних фінансових витрат, тому більш ефективним засобом є використання коду, що корегує помилки.

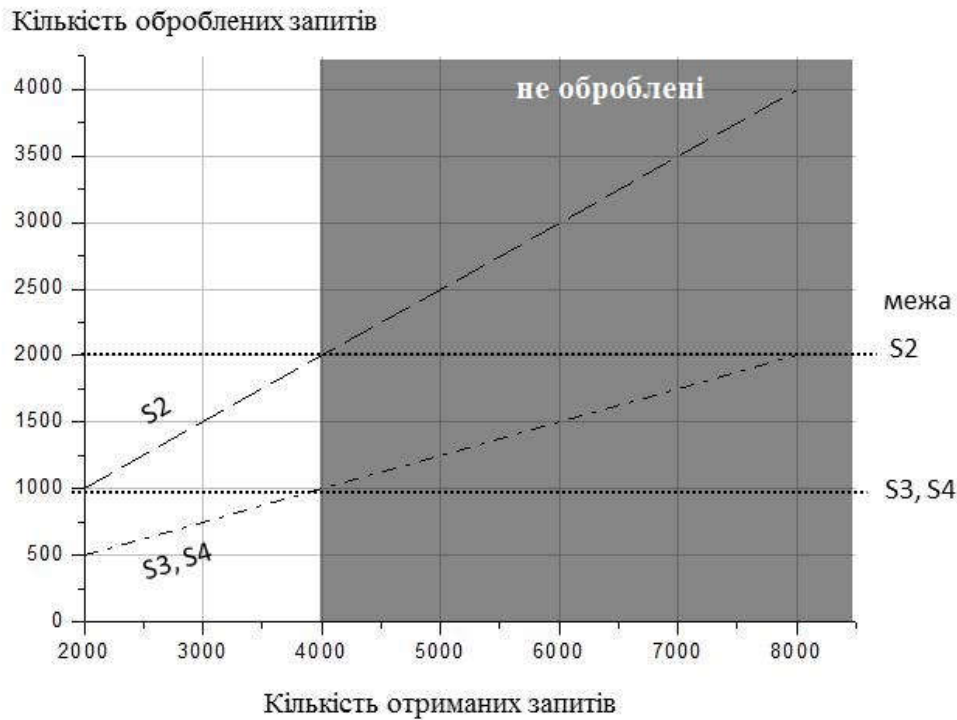


Рис. 5. Розподіл запитів між серверами хмарного сервісу при аварійній ситуації

Розглянемо розподілену систему хмарного сервісу [1-4], що включає N серверів $S_1, S_2 \dots S_N$. Нехай на кожному n -му сервері знаходиться інформаційний блок W_n , причому довжини цих блоків покладемо за $|W_n| = m$, а i -ий елемент блоку W_n позначимо як w_{ni} . До інформаційного блоку $W_i = w_{i1} \dots w_{im}$ застосовується засоби кодування, що описуються функцією K і визначаються елементами перевірки c :

$$K(w_i) = w_i c_i = w_i c_{i1} \dots c_{ir} \quad (3)$$

При цьому для збереження елементів перевірки застосовуються r додаткових серверів $Sc_1 \dots Sc_r$ (табл. 3).

Таблиця 3.

Застосування r додаткових серверів для захисту даних на n основних серверах

Сервер	S_1	S_2	...	S_N	Sc_1	Sc_2	...	Sc_r
Дані	w_{11}	w_{21}	...	w_{N1}	c_{11}	c_{21}	...	c_{r1}
	w_{12}	w_{22}	...	w_{N2}	c_{12}	c_{22}	...	c_{r2}

	w_{1m}	w_{2m}	...	w_{Nm}	c_{1m}	c_{2m}	...	c_{rm}

Переваги кодування над простим дублюванням визначаються співвідношенням між кількістю додаткових серверів, що мають бути застосовані в першому та другому випадку, тобто співвідношенням між величинами r та n . Для найбільш простого методу кодування, що базується на використанні коду Хеммінга параметр r як функція від n визначається наступним чином:

$$r \gg \log_2(n) \quad (4)$$

Побудуємо дану залежність і порівняємо кількість серверів обравши $r_{\min} = 3 \cdot \log_2(n)$, а множник дублювання — x2 (рис. 6).

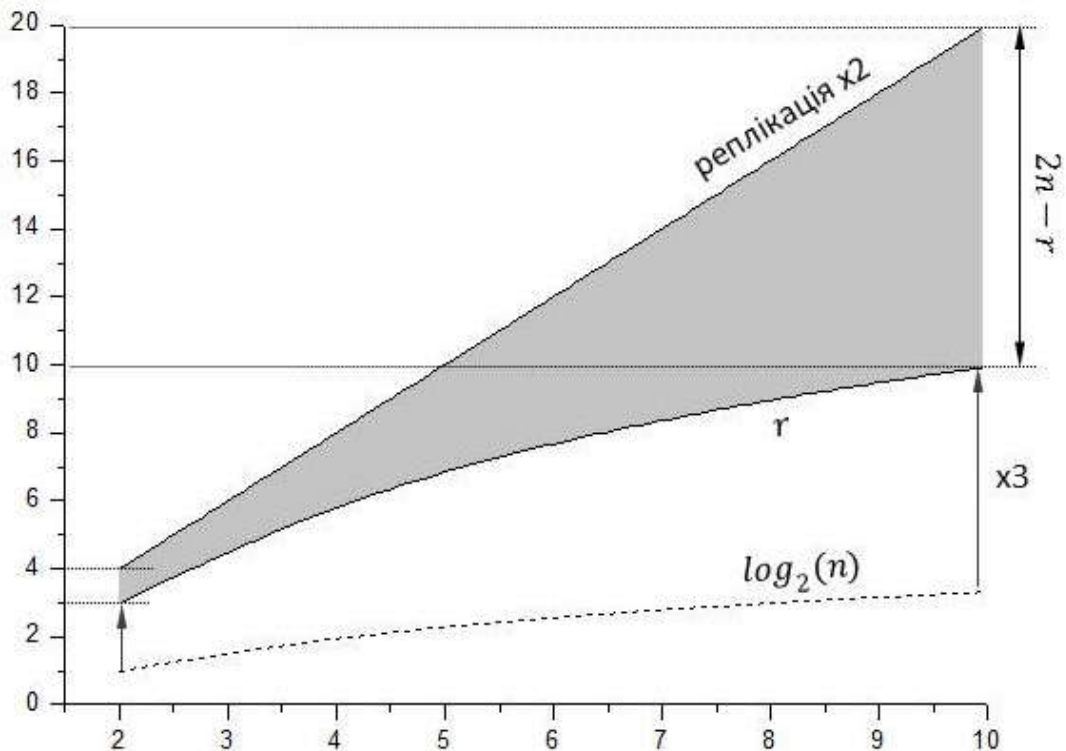


Рис. 6. Залежність кількості додаткових серверів при застосуванні коду Хеммінга і реплікуванні даних

Залежність представлена на рис. 6 наочно демонструє переваги коду Хеммінга над простим дублювання даних. Характерно, що ефективність даного підходу суттєво зростає при збільшенні кількості серверів.

3. Тенденції розвитку концепції хмарних обчислень

Для розуміння процесу еволюції хмарних обчислень у туманні (fog computing) та росисті обчислення (dew computing) необхідно розглянути етимологію зазначених словосполучень [9-12]. Назва «хмарні обчислення» виникла внаслідок організації архітектури даних сервісів у єдину мережу доступ до якої можна отримати через мережеві комутатори. Очевидно, що вузьким місцем сервісу стає пропускна здатність комутаторів, таким чином, у випадку реальної хмари, не можна говорити про абсолютну мобільність сервісу, його незалежність від точки виклику запиту. Вирішення даної проблеми призвело до появи концепції «граничних обчислень» (edge computing) у рамках котрої були розроблені «туманні обчислення». Термін «туман» був обраний для наочного представлення концепції: туман є хмарою, що опустилася до землі, таким чином, туманні обчислення розглядають абонента, як частину внутрішньої мережі хмари. Отже, якщо у хмарних обчисленнях абоненту пропонується обрати найбільший сервіс, то у туманних — найближчий, згідно з параметром перепускності інтернет-каналу [9-11]. Такий підхід дозволяє суттєво підвищити мобільність хмарних обчислень, але у його рамках залишається залежність користувача від надійності інтернет-зв'язку. Тому далі було запропоновано використати у архітектурі хмарного сервісу ідею випадання роси, тобто збереження «гарячих» даних на накопичувачах, що знаходяться у внутрішній мережі користувача [12]. Проведений аналіз надає можливість побачити, що методологія туманних і росистих обчислень не суперечить початковій концепції хмарного сервісу, а лише доповнює її.

Поява хмарних обчислень та їх еволюція у туманні та росисті обчислення пов'язана з трьома ключовими факторами, що вказують на розвиток ІТ-індустрії: інформаційна ємність окремих накопичувачів, частота процесорів і перепускність інтернет-каналу. Графік наведений на рис. 7 показує, що реалізація хмарних, туманних та росистих обчислень стала можливою лише завдяки експоненційному росту всіх трьох показників.

У рамках концепції росистих обчислень множини даних, якими обмінюються абоненти можна розглядати як водяну пару, що конденсується у краплі роси на холодних поверхнях, тобто копіюється у відповідні пули на спеціально виділені сервери. Це сервери провайдерів, що розміщуються у житлових комплексах, торговельних і розважальних центрах, спорткомплексах, тощо. На сьогоднішній день мультимедійні сервери вбудовують у поїзди, екскурсійні автобуси і літаки. При цьому переносні комп'ютери і смартфони абонентів працюють у режимі FTP-серверів гібридної хмари з розділення ресурсів на публічні (ком'юніті-сервери) і приватні, а також з анонімізацією апаратних ресурсів.

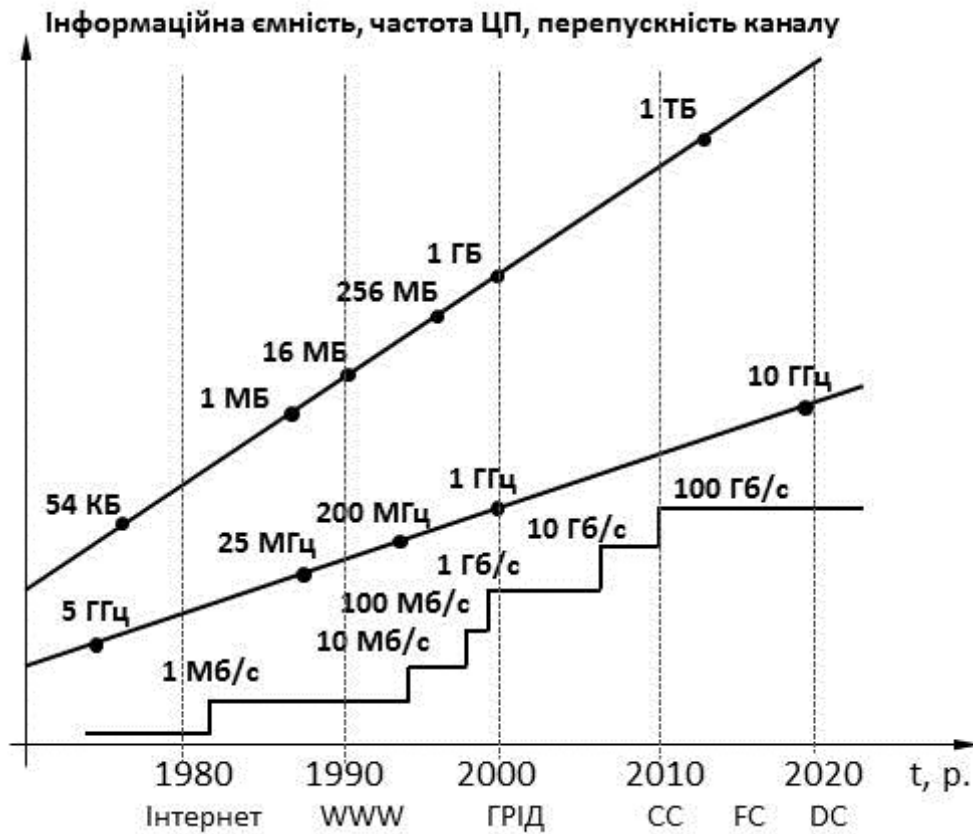


Рис. 7. Тенденції появи та розвитку сервісів хмарних обчислень у залежності від характеристик інфраструктури центрів обробки даних.

При використанні хмарних, туманних та росистих обчислень у роботі організації визначення актуальності конкретного сервісу залежить від швидкості з якою вам необхідно отримати дані та провести обчислення. Дані, що знаходяться у процесі використання мають «конденсуватися», тобто зберігатися на внутрішніх серверах компанії. Блок актуальних даних по які часто звертається персонал може знаходитися у межах сфери туманних обчислень, це дозволить зменшити час затримки при отриманні актуальних даних. Інформаційні сховища та блоки обчислень математичних моделей, що потребують значних апаратних ресурсів, слід зберегти рамках великого хмарного сервісу (рис. 8)

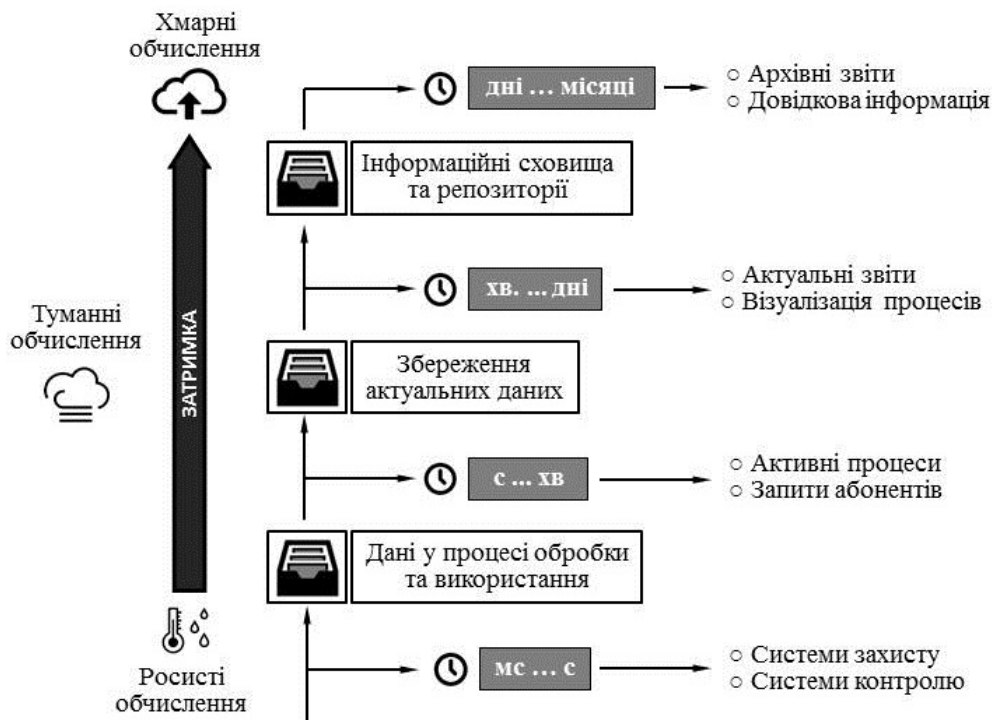


Рис. 8. Тенденції переходу від хмарних до туманних та росистих обчислень.

Туманні та росисті сервіси з точки зору вимог до організації архітектури подібні до звичайних хмарних сервісів, але певні аспекти та вимоги до їх роботи у даному випадку мають більшу вагу:

- безпека інфраструктури мережі;
- конфіденційність даних та анонімність ресурсів;
- миттєва еластичність сервісу.

Це пов'язано з відкритістю даних сервісів, що унеможливорює чітке проектування його архітектури, перевірку апаратних ресурсів і ПЗ на етапі розробки. Ком'юніті-сервер повинен бути найбільш захищеним і безпечним, що включає у себе шифровку даних, контроль у режимі реального часу на предмет виявлення потенційно небезпечного ПЗ, аналіз цілісності, що повинна забезпечуватися за допомогою хеш-кодів та перевірку контенту на ліцензійність.

Висновки. Розглянуто базові підходи організації структури хмарних серверів. Показано, що основними перевагами концепції хмарних обчислень при побудові мережевих ресурсів є мобільність передачі даних, низький вхідний поріг налаштування системи та простота подальшого масштабування системи, надійність збереження даних, функціонування системи у режимі реального часу та високий рівень автоматизації сервісів. Проведено аналіз режимів роботи публічної, приватної та гібридної хмари, вказано на можливі галузі застосування кожного з видів сервісів. Вказано, що базові заходи з підтримання надійності роботи хмарного сервісу включають у себе забезпечення роботи серверів у режимі реального часу шляхом балансування та надійне збереження даних у випадку аварійної ситуації. Запропоновано для вирішення проблеми необхідності широкого каналу доступу до хмарного сервісу використовувати туманні і росисті обчислення. Вказано, що визначення актуальності застосування концепції хмарних, туманних чи росистих обчислень залежить від максимального припустимого часу затримки у роботі сервісу.

Література

1. Bhowmik, S. (2017). Cloud computing. Cambridge: Cambridge University Press.
2. Dimitrov, V. (2016). Cloud Programming Models (MapReduce). Encyclopedia of Cloud Computing, 596-608. doi:10.1002/9781118821930.ch49
3. Gordon, D. G. (2016). Legal Aspects of Cloud Computing. Encyclopedia of Cloud Computing, 462-475. doi:10.1002/9781118821930.ch38
4. Appendix A: Example Of Microsoft Azure Cloud Service: Filemanager. (2016). Trustworthy Cloud Computing, 299-308. doi:10.1002/9781119114215.app1
5. Carlson, E. (2016). Enterprise Cloud Computing Strategy and Policy. Encyclopedia of Cloud Computing, 361-371. doi:10.1002/9781118821930.ch30
6. Barouti, S., Alhadidi, D., & Debbabi, M. (2013). Symmetrically-Private Database Search in Cloud Computing. 2013 IEEE 5th International Conference on Cloud Computing Technology and Science. doi:10.1109/cloudcom.2013.95
7. Metheny, M. (2013). Security and Privacy in Public Cloud Computing. Federal Cloud Computing, 71-102. doi:10.1016/b978-1-59-749737-4.00004-6
8. Cheng, P., & Qu, H. (2014). Design and Realization Based on Cloud Stack Hybrid Cloud Computing Platform. Advanced Materials Research, 989-994, 2297-2300. doi:10.4028/www.scientific.net/amr.989-994.2297
9. IEEE Cloud Computing Call for Papers Connecting Fog and Cloud Computing. (2016). IEEE Cloud Computing, 3(4), C2-C2. doi:10.1109/mcc.2016.83
10. Linthicum, D. S. (2017). Connecting Fog and Cloud Computing. IEEE Cloud Computing, 4(2), 18-20. doi:10.1109/mcc.2017.37
11. Cloud and Fog Computing. (2017). Secure Connected Objects, 238-247. doi:10.1002/9781119426639.ch17
12. Sojaat, Z., & Skalaa, K. (2017). The dawn of Dew: Dew Computing for advanced living environment. 2017 40th International Convention on Information and Communication Technology, Electronics and Microelectronics (MIPRO). doi:10.23919/mipro.2017.7973447

Рецензія/Peer review : 14.11.2017 р.

Надрукована/Printed :21.01.2018 р.

Стаття рецензована редакційною колегією