

УДК 004.75.: 004.891.3

Д.М. МЕДЗАТИЙ, Д.О. ІВАНЧИШИН, О.В. КУВШИНОВ

Хмельницький національний університет

НАДІЙНІСТЬ CLOUD-CИСТЕМ

В роботі розглянуто особливості організації та функціонування cloud-систем і оцінено фактори, які впливають на їх надійність. Проведено аналіз відмов cloud-систем. Виділено складові cloud-систем, що потребують аналізу при комплексній оцінці надійності системи в цілому. Сформульовано основні принципи комплексного підходу щодо оцінки надійності cloud-систем.

In the article the main classes of cloud-systems have been considered. Analysis of cloud computing systems failure was conducted. In the article all failures were classified into two groups: request stage failures and execution stage failures. New reliability model for cloud computing systems was developed.

Ключові слова: надійність, розподілені системи, моделі надійності.

Вступ

Сучасний стан розвитку інформаційних технологій спрямований на розроблення та впровадження нових підходів до організації обчислювального процесу. Необхідність пошуку нових рішень зумовлена рядом факторів, серед яких найбільш значимими є наявність наукових-практичних задач, які вимагають високопродуктивних обчислювальних ресурсів; неможливість подальшого стрімкого нарощування продуктивності центральних процесорів через фізичні обмеження; висока вартість кінцевих високопродуктивних рішень та неефективне їх використання; нерівномірність обчислювальних навантажень. Перераховані фактори сприяють розробленню та впровадженню різноманітних технологій розподілених обчислень, які базуються на сучасних програмних та апаратних засобах, а також нових представленнях про обчислювальні ресурси та їх використання (розподілена обробка, кластеризація, віртуалізація та ін.).

Одним з таких підходів є cloud computing («хмарні» обчислення), що вважається наступним еволюційним кроком у розвитку розподілених обчислень. На сьогодні, не існує чіткого, стандартизованого визначення цієї технології. Національний інститут стандартів та технологій США (NIST) подає визначення cloud computing як технології (моделі обчислень), що дозволяє у прийнятній формі, за запитом отримати доступ до сумісно-використовуваного набору конфігурованих обчислювальних ресурсів (наприклад, мереж, серверів, систем збереження даних, програмного забезпечення, послуг тощо), які швидко можуть бути виділені та надані у користування при мінімальній взаємодії з провайдером послуг [1]. Метою такої моделі обчислень є підвищення ефективності використання розподілених ресурсів, об'єднання їх в єдину високопродуктивну систему здатну вирішувати масштабні обчислювальні задачі. В cloud-системах використовуються як відомі технології так і нові ідеї та рішення. Зокрема, cloud-системи тісно пов'язані з такими технологіями, як grid computing – один із варіантів розподілених обчислень, в якому різні обчислювальні ресурси поєднані в єдиний кластер для розв'язання складних задач [2], utility computing – підхід, коли доступ до комп'ютерних ресурсів (обчислювальних, систем збереження даних, програмного забезпечення) надається «за лічильником», тобто користувач сплачує лише за отримані ресурси та сервіси (за аналогією з комунальними послугами) [3], transparent computing – модель, в якій від користувача прихована складана внутрішня будова системи, натомість користувач отримує простий, зрозумілий та зручний у користуванні інтерфейс (front-end interface) [4].

Поєднання згаданих технологій спільно з особливостями, притаманними виключно cloud-computing, створюють передумови для побудови гнучких та високопродуктивних cloud-систем, які можуть використовуватись для розв'язання задач у різних галузях. Однак, на сьогодні, одним із стримуючих факторів щодо впровадження та подальшого розвитку cloud-систем є відсутність вичерпних відомостей про їх надійність. Отже, визначення та оцінка факторів, що впливають на надійність cloud-систем є актуальною задачею.

Постановка задачі

Розглянути особливості організації та функціонування cloud-систем з точки зору їх надійнісних характеристик. Оцінити ризики використання cloud-систем для розв'язання задач із заданим рівнем критичності. Запропонувати підхід до оцінки надійності cloud-систем.

Огляд відомих cloud-систем

На сьогодні, ринок послуг у сфері cloud computing рішень є достатньо розвинутим. Найбільш відомими є рішення від Microsoft (Microsoft Azure), Google (Google Apps Engine), Amazon (Elastic Cloud Computing, Simple Storage Service), IBM (Blue Cloud), Nimbus, Oracle та інші. Крім великих корпорацій послуги cloud computing також надають менш потужні фірми. На ринку присутні безкоштовні рішення (iCloud, Cloudo, FreeZoho, SalesForce та ін.). Всі рішення різняться сервісами які надають, це може бути SaaS (Software-as-a-Service – програмне забезпечення як сервіс), PaaS (Platform-as-a-Service – платформа як

сервіс), IaaS (Infrastructure-as-a-Service – інфраструктура як сервіс), HaaS (Hardware-as-a-Service – апаратура як сервіс), крім того, в різних джерелах описані і інші варіації технології cloud computing, такі як дані як сервіс, бізнес як сервіс, середовище розробки як сервіс та інші [5-7]. Незважаючи на різновиди послуг, які надаються як сервіс, в основі всіх cloud-систем знаходяться достатньо типові апаратні та програмні засоби, які забезпечують функціонування системи та забезпечують відповідний рівень гнучкості та еластичності. Відповідно, апаратні та програмні засоби cloud-систем, з певною імовірністю можуть відмовити. Для мінімізації імовірності виходу з ладу системи та зменшення часу відновлення, в cloud-системах використовуються ті ж самі принципи, що і в більшості систем розподіленої обробки даних (резервування, перерозподіл обчислювальних ресурсів та інші). Ці підходи дозволяють частково приховати від користувача реальний стан речей та створити ілюзію безвідмовної роботи системи. Разом з тим, статистика відмов cloud-систем свідчить про те, що відомі підходи не завжди виявляються достатньо ефективними (таблиця 1).

Таблиця 1

Статистика відмов cloud-систем

№	Провайдер послуг	Сервіс	Дата	Тривалість простою
1	Google	Gmail, Google Apps Engine	24 лютого 2009 р.	2,5 години
2	Google	Google Search	31 січня 2009 р.	40 хвилин
3	Google	Google Gmail	9 березня 2009 р.	22 години
4	Google	Google Network	14 травня 2009 р.	2 години
5	Amazon	Amazon Elastic Cloud Computing	11 червня 2009 р.	7 годин
6	Amazon	Amazon Elastic Cloud Computing	9 грудня 2009 р.	5 годин
7	Amazon	Amazon Simple Storage Service	15 лютого 2008 р.	2 години
8	Amazon	Amazon Elastic Cloud Computing	21 квітня 2011 р.	27 годин
9	Microsoft	Microsoft Azure	13-14 березня 2008 р.	22 години
10	Microsoft	Microsoft Hotmail	12 березня 2009 р.	5 годин
11	Microsoft	Microsoft Sidekick	4 жовтня 2009 р.	6 днів
12	Flexiant	FlexiScale	31 жовтня 2008 р.	18 годин

Аналізуючи статистику, наведену у таблиці 1, слід відзначити, що не у всіх випадках наслідком відмов були лише простої. Під час відмови сервісу Microsoft Sidekick були повністю втрачені особисті дані користувачів [8], які згодом вдалося відновити, але лише частково. Отже, незважаючи на достатньо високий рівень впровадження відомих рішень щодо підвищення надійності cloud-систем, питання аналізу вузьких місць, з метою пошуку підходів до їх усунення та підвищення надійності і ефективності експлуатації останніх, залишається актуальним та потребує подальших досліджень.

Аналіз причин відмов cloud-систем

Для проведення аналізу причин відмов притаманних cloud-системам, слід розглянути загальну структуру системи, що надає cloud-сервіси (рис. 1).



Рис. 1. Структура cloud-системи

Запропонована структура є найбільш репрезентативною серед відомих рішень. Головним компонентом є система керування ресурсами cloud-системи. Система керування розв'язує чотири основні задачі:

- керування чергою, яка формується робочими запитом від множини користувачів, які отримують доступ до cloud-сервісів через Інтернет;
- керування обчислювальними ресурсами (персональними комп'ютерами, кластерами, суперкомп'ютерами та ін.), які можуть бути територіально розподіленими зі з'єднанням через мережу Інтернет;
- керування ресурсами збереження даних (бази даних, апаратура для зберігання інформації та ін.), як

локальними так і розподіленими;

планування запитів та поділ їх на під задачі з розподілом за наявними обчислювальними ресурсами з можливістю отримання доступу до необхідних даних.

Описана структура cloud-систем визначає різні типи відмов, що можуть порушити гарантоздатність та надійність надання відповідних сервісів. Серед загальної множини виділимо наступні типи відмов: переповнення черги запитів, перевищення часу очікування, відсутність даних (доступу до засобів збереження інформації), відсутність обчислювальних ресурсів, відмови програмного забезпечення, відмови апаратного забезпечення, відмови системи комунікацій (відмови комп'ютерних мереж). Кожен тип відмов слід розглядати окремо та враховувати вплив відмов заданого типу на надійність cloud-систем в цілому.

Переповнення черги. Черга запитів має обмеження щодо максимальної кількості елементів, що можуть очікувати виділення ресурсів. Якщо запит, який надійшов, буде знаходитись у черзі задовго, то виникне відмова перевищення часу очікування. Якщо ж черга переповнена, то нові запити будуть ігноруватись і користувачі не зможуть отримати доступ до сервісу.

Перевищення часу очікування. Cloud-сервіс, як правило, має фіксований час відклику. Якщо час знаходження запиту у черзі перевищить час відклику системи, виникає відмова за часом очікування. Такий запит «викидається» з черги, не впливаючи на решту запитів.

Відсутність даних. В системі керування ресурсами, менеджер ресурсів збереження даних реєструє всі наявні ресурси. Існує імовірність, що раніше зареєстровані засоби збереження чи, зрештою, дані були переміщені, а менеджером актуалізована інформація про поточний стан ресурсів. Як результат, у випадку посилення на ці дані поточного запиту, виникає відмова за відсутністю даних.

Відсутність обчислювальних ресурсів. Відмови за відсутності обчислювальних ресурсів схожі з відмовами через відсутність даних. Будь-який обчислювальний ресурс у мережі може вийти з ладу або може бути вимкненим без нотифікації менеджера обчислювальних ресурсів. Відповідно, робочий запит направлений на такий обчислювач буде залишений без обслуговування, що призведе до відмови через відсутність обчислювального ресурсу.

Відмови програмного забезпечення. Як і будь-яка обчислювальна система, cloud-система складається з набору апаратних та програмних засобів з певним рівнем надійності. Отже є імовірність того, що програмне забезпечення відмовить через наявність прихованих помилок, невідповідність вихідних даних або внаслідок інших причин. Імовірність виникнення відмов програмного забезпечення у cloud-системах пов'язана з можливістю використання доволі різнотипного прикладного та системного програмного забезпечення на різних рівнях системи. В наслідок такої неоднорідності, не завжди вдається забезпечити необхідний рівень узгодженості програмних засобів. Крім того, широке використання віртуалізації у cloud-системах, з використанням складного системного програмного забезпечення, значно ускладнює задачу забезпечення необхідного рівня надійності програмних засобів.

Відмови апаратного забезпечення. Обчислювальні ресурси та ресурси збереження даних містять апаратне забезпечення, яке складається з компонентів, що можуть відмовляти. Без сумніву, апаратне забезпечення на якому працює «ядро» cloud-системи будують з використанням засобів підвищення надійності (структурне резервування). Це дозволяє значно підвищити надійність та гарантоздатність апаратного забезпечення «ядра», однак, cloud-система є розподіленою системою, яка містить достатньо великий набір апаратних засобів без підвищених вимог до надійності (персональні комп'ютери, неоднорідні кластери та ін.).

Відмови системи комунікацій. Використання в якості обчислювальних ресурсів розподілених систем висуває особливі вимоги до надійності засобів комунікацій. Засобами комунікацій у cloud-системах є комп'ютерні мережі (локальні, глобальні, мережа Інтернет). Втрата зв'язку обчислювача з засобами збереження даних, або системою керування ресурсами призводить до виникнення відмови. Особливо значна загроза відмов мережі при виконанні задач, що потребують значних витрат часу або опрацьовують значний об'єм даних.

Слід відмітити, що різні типи відмов корелюються між собою. Наприклад, відмови планувальника можуть збільшити час очікування та призвести до виникнення відмов через переповнення та перевищення часу очікування. Це одна з причин того, що для розрахунку надійності cloud-систем не можна використати жодну з відомих моделей, які окремо розглядають надійність програмного забезпечення, надійність апаратури, мережі і таке інше. Тому, для коректної оцінки надійності cloud-систем слід розробити модель їх надійності.

Модель надійності cloud-систем

Одним з ключових моментів при побудові моделі надійності cloud-систем є необхідність означення цього поняття, саме з точки зору кінцевого користувача. Отже, визначимо надійність cloud-системи, як імовірність того, що cloud сервіс чи послуга будуть надані користувачу у повному обсязі за визначений час. Враховуючи типи відмов, описані раніше, це означає, що робочий запит буде оброблений планувальником у допустимих часових межах; набір підзадач, обумовлений сервісом, буде сформовано; будуть наявні необхідні обчислювальні ресурси та дані; робота системи комунікації не перерветься під час виконання задачі.

За таких умов, імовірні відмови доцільно поділити на дві групи: відмови запитів (переповнення черги та перевищення часу очікування) та відмови виконання (відсутність обчислювальних ресурсів чи

даних, відмови програмного чи апаратного забезпечення, відмови мережі). Тоді, ці групи відмов доцільно розглядати як незалежні. З врахуванням цього надійність cloud-системи буде визначатись співвідношенням:

$$P_C = P_{зан} \cdot P_{вик}, \quad (1)$$

де P_C – загальна надійність cloud-системи; $P_{зан}$ – надійність підсистем опрацювання запитів; $P_{вик}$ – надійність підсистем етапу розв’язання задачі.

Надійність підсистеми опрацювання запиту описується класичною моделлю на основі Марківських процесів, вважаючи що потік нових запитів є Пуасонівським.

Більш цікавим є питання визначення надійності підсистем етапу виконання задач. Як зазначалося раніше, відмова на цьому етапі може бути спричинена декількома типами відмов. Аналізуючи перераховані відмови, відзначимо, що відсутність обчислювальних ресурсів або ресурсів збереження даних виникає на початковому етапі виконання запиту i , у певному наближенні, може бути віднесена до відмов апаратного забезпечення. Тоді, загальна відмова, cloud-системи може бути спричинена одним з трьох факторів: відмова програмного забезпечення, відмова апаратного забезпечення, відмова мережі. Якщо абстрагуватись від конкретних взаємозв’язків елементів кожної складової надійності, та представити їх звичайною множиною, то надійність елементу буде визначатись наступним виразом [9]:

$$P_i = e^{-I_i \cdot T_i}, \quad (2)$$

де P_i – надійність i -го елементу; I_i – інтенсивність відмов i -го елементу; T_i – час, протягом якого i -й елемент cloud-системи зайнятий виконанням задачі (обслуговуванням запиту).

Інтенсивність відмов (I_i) буде різною для окремих елементів, однак, як для програмних засобів так і для компонентів апаратури та мережі інтенсивність відмов буде константою [9]. Цей факт значно спрощує модель надійності, дозволяючи уникнути громіздких обчислень.

Час, протягом якого ресурс зайнятий виконанням певної задачі, доцільно визначити окремо для програмного забезпечення, мережі та апаратного забезпечення. Наприклад, час виконання програмним засобом задач щодо обслуговування cloud-системи, може визначатись співвідношенням:

$$T_{i,k}^{ПЗ} = \frac{\text{Кількість інструкцій}}{\text{Швидкість опрацювання}}, \quad (3)$$

де $T_{i,k}^{ПЗ}$ – час виконання k -ї програми на i -му обчислювальному ресурсі. Швидкість опрацювання, в даному випадку, може вимірюватись у MIPS (мільйонів інструкцій за секунду). Відповідно, кількість інструкцій слід подавати у еквівалентному значенні (за типом інструкцій).

Час зайнятості мережі визначимо як:

$$T_m^M = \frac{\text{Кількість інформації}}{\text{Пропускна здатність}}, \quad (4)$$

де T_m^M – час, протягом якого m -а лінія зв’язку мережі передає інформацію (від одного вузла іншому). Кількість інформації та пропускна здатність повинні бути одного відповідати одна одній. Якщо кількість інформації подано у мегабайтах, то пропускна здатність подається у мегабайтах за секунду.

Отже, сумарний час «зайнятості» апаратного забезпечення, складається з часу роботи програмних засобів та часу витраченого на транспортування даних. Відповідно до (3) та (4) цей час визначається як:

$$T_i^{OP} = \sum_{k=1}^n T_{i,k}^{ПЗ} + \sum_{m=1}^r T_m^M, \quad (5)$$

де T_i^{OP} – час завантаженості i -го обчислювального ресурсу (апаратного забезпечення); n – кількість програм (модулів), що виконуються на даному обчислювальному засобі для розв’язання поставленої під задачі; r – кількість комунікаційних ліній мережі, задіяних у процедурах обміну даними між i -м обчислювальним ресурсом та певними ресурсами збереження інформації.

Визначивши загальний час, що витрачається на опрацювання задач відповідного робочого запиту, відповідно до (2) можна визначити загальну надійність виконання запиту з надання сервісу cloud-системою:

$$P_{вик} = \prod_{i=1}^S P_i = \prod_{i=1}^S e^{-I_i \cdot T_i}, \quad (6)$$

де S – загальна кількість елементів cloud-системи, які приймають участь в опрацюванні запиту.

Запропонований підхід розрахунку надійності cloud-системи дозволяє достатньо швидко визначити показники надійності з точки зору користувача. На нашу думку, саме надійність надання сервісів користувачу є визначальною. Відповідно до технології transparent computing користувач не бачить що саме і як відбувається у cloud-системі. Головним для нього залишається цілісність даних та гарантія отримання сервісу у повному обсязі.

Висновки

Cloud computing є однією з найбільш прогресивних технологій розподілених обчислень, яка надає

користувачу можливість у найкоротші терміни отримати всі необхідні ресурси. Ця можливість досягається завдяки таким властивостям cloud-систем як гнучкість, еластичність та віртуалізація. Разом з тим, велика кількість провайдерів cloud-послуг та відсутність чітких критеріїв порівняння останніх, ускладнює процес прийняття користувачем остаточного рішення щодо вибору cloud-системи. Також, одним із стримуючих факторів впровадження cloud-систем є відсутність методик оцінки їх надійності. Запропонована в роботі модель надійності cloud-систем, яка враховує особливості відомих типів відмов та взаємозв'язки між ними, дозволяє провести розрахунок та оцінити надійність надання сервісу відповідною cloud-системою. Саме рівень надійності надання послуг є визначальним для кінцевого користувача.

Подальші дослідження у даному напрямку доцільно зосередити на аналізі запропонованої моделі, доведенні її адекватності та апробації. Результатом цих досліджень повинна стати уточнена модель та метод визначення надійності cloud-систем.

Література

1. Peter Mell The NIST Definition of Cloud Computing. Recommendations of the National Institute of Standards and Technology / Peter Mell, Timothy Grance. – NIST Special Publication 800-145. Computer Security Division. Information Technology Laboratory, Gaithersburg, MD 20899-8930. – 2011. – 7 p.
2. Foster The Grid 2: Blueprint for a New Computing Infrastructure / I. Foster, C. Kesselman. – Los Alamos, Morgan-Kaufmann, 2003. – 748 p.
3. C.S. Yeo Utility Computing on Global Grids. Technical Report / C.S. Yeo, R. Buyya, M.D. de Assunção, et al. – GRIDS-TR-2006-7, Grid Computing and Distributed Systems Laboratory, The University of Melbourne, Australia. – 2006. – 26 p.
4. Y. Zhang Transparent computing: A new paradigm for pervasive computing / Y. Zhang, Y. Zhou // Proceedings of the 3rd International Conference on Ubiquitous Intelligence and Computing (UIC-06), LNCS 4145, 1– 11. – 2006. – P.18-34.
5. Bhaskar Prasad Rimal A Taxonomy and Survey of Cloud Computing Systems / Bhaskar Prasad Rimal, Eummi Choi, Ian Lumb // Proceedings of the Fifth International Joint Conference on INC, IMS and IDS. – 2009. – P.44-51.
6. Luis M. Vaquero A Break in the Clouds: Towards a Cloud Definition / Luis M. Vaquero, Luis Rodero-Merino, Juan Caceres, Maik Linder // ACM SIGCOMM Computer Communication Review. – 2009. – Volume 39. – P.50-55.
7. Mike Piech Platform-as-a-Servise Private Cloud with Oracle Fusion Middleware / Mike Piech. – An Oracle White Paper. – 2009. – 20 p.
8. Walter F. Witt Keep Your Feet on the Ground When Moving Software into the Cloud / Walter F. Witt // International Journal of Digital Content Technology and its Applications. – 2010. – Volume 4, Number 2. – P.10-17.
9. Локазюк В.М. Надійність, контроль, діагностика і модернізація ПК: Навчальний посібник для вузів / Локазюк В. М., Савченко Ю. Г. – Київ: “Академія” (Альма-матер), 2004. – 375с.

Надійшла 20.8.2011 р.

УДК 004.75

Д.М. МЕДЗАТИЙ, В.В. ДУДКО, Р.М. ХОМЕНКО

Хмельницький національний університет

ОПТИМІЗАЦІЯ ВИДІЛЕННЯ ОБЧИСЛЮВАЛЬНИХ РЕСУРСІВ У CLOUD-СИСТЕМАХ

В роботі проведено класифікацію cloud-систем з метою визначення їх архітектурних особливостей. Проведено аналіз відомих методів вибору обчислювальних ресурсів для розв'язання конкретних задач. Запропоновано підхід до вибору обчислювальних ресурсів у cloud-системах за додатковим описом запиту, що дозволить підвищити ефективність опрацювання запитів та зменшить кількість відмов. Наведено приклад формування опису запиту для автоматизованої системи розподілу обчислювальних ресурсів.

In the article the classification and architecture specifics of cloud-systems have been considered. Analysis of known methods of computing resources choosing in cloud computing systems was conducted. The new approach for computing resources choosing, which use accessory description of user request was developed. New approach increase efficiency of request processing in the cloud systems.

Ключові слова: обчислювальні ресурси, розподілені системи.

Вступ

Сучасні комп'ютерні мережі дозволяють об'єднувати різнотипні обчислювальні засоби у єдину систему, здатну розв'язувати трудомісткі задачі, максимально ефективно використовуючи усі наявні вільні ресурси. Наявні високошвидкісні канали передачі даних дозволяють користувачам, незалежно від ступеня їх віддаленості отримати доступ до таких систем і користуватись необхідними ресурсами через мережу Інтернет.