

Основні Складнощі при Виборі Архітектури Хмарного Додатка

Наталія Сердюк

кафедра комп'ютерних інтелектуальних систем та технологій
Харківський національний університет радіоелектроніки
Харків, Україна
nataliya.serdyuk@nure.ua

The Main Difficulties in Choosing the Architecture of the Cloud Application

Nataliia Serdiuk

Department of Computer Intelligent Systems and Technologies
Kharkiv National University of Radio Electronics
Kharkiv, Ukraine
nataliya.serdyuk@nure.ua

Анотація—ІТ-індустрія живе з урахуванням хмарних обчислень. Шаблони споживання послуг дають нові можливості для користувачів, але хмари створюють ряд проблем, коли справа стосується побудови програм у хмарній парадигмі.

Abstract—The IT industry lives with the cloud in mind. Service consumption patterns provide new opportunities for users, but clouds pose a number of challenges when it comes to building applications within the cloud paradigm.

Ключові слова—хмарні технології; додатки; архітектура; розгортання, модель

Keywords—cloud technologies; applications; architecture; deployment, model

I. ВСТУП

Хмарні технології змінюють принципи проектування додатків. Хмарні додатки не монолітні, а складаються з невеликих децентралізованих служб. Ці служби взаємодіють між собою через API за допомогою асинхронного обміну повідомленнями або подіями. Додатки можна масштабувати горизонтально, додаючи нові екземпляри, якщо з'являється потреба в цьому. З цими тенденціями пов'язані нові складності. Система в

цілому повинна зберігати стійкість у разі збоїв, а розгортання повинні бути автоматизованими і передбачуваними. Для розуміння стану системи вкрай важливо налагодити ефективний моніторинг і телеметрію [1].

II. ВІДМІННОСТІ КЛАСИЧНИХ ДОДАТКІВ ВІД ХМАРНИХ

До основних особливостей класичних локальних додатків слід зарахувати такі особливості: додатки монолітні, централізовані; забезпечують передбачувану масштабованість; реляційні бази даних; сильна узгодженість; послідовна і синхронна обробка; архітектура орієнтована на уникнення відмов (основна характеристика - середній час між відмовами); рідкісні масштабні оновлення; ручне управління і централізована архітектура серверів [2,3].

На відміну від класичних додатків, сучасні хмарні додатки фрагментовані і децентралізовані; підтримують гнучке масштабування; можливе одночасне використання різних технологій зберігання даних; «Узгодженість в кінцевому рахунку»; паралельна і асинхронна обробка;



Інформаційні системи та технології ICT-2020

Секція 7.

Комунікаційні, GRID та хмарні технології. IoT.

архітектура стійка до відмов (основна характеристика - середній час відновлення); часті невеликі оновлення; автоматичне самообслуговування і незмінна інфраструктура [2,3].

III. ВИБІР АРХІТЕКТУРИ ДОДАТКІВ

На самому початку необхідно вибрати архітектуру додатку. Від першого прийнятого рішення залежить решту шляху. Яку архітектуру виберете? Це може бути архітектура на базі мікро сервісів, більш традиційний N-рівневий додаток або рішення для роботи з великими даними. Існує безліч варіантів, але найбільш використовувані сім варіантів архітектури. У кожного з них є свої переваги і недоліки. Вибір архітектури залежить від складності додатка, області застосування, його типу і завдань, для вирішення яких він призначений. Також важливо враховувати навички команди розробників і менеджерів проекту, а також наявність у додатку готової архітектури.

Вибір архітектури накладає певні обмеження на структуру додатку, обмежуючи вибір технологій і інших елементів додатку. З цими обмеженнями пов'язані як переваги, так і недоліки обраної архітектури.

IV. ВИБІР ТЕХНОЛОГІЙ ДЛЯ ОБЧИСЛЕНЬ ТА ЗБЕРІГАННЯ ДАНИХ

Вибір технологій для обчислень пов'язаний з визначенням моделі розміщення обчислювальних ресурсів, на яких запускається додаток. У максимально загальному вигляді його можна сформулювати як вибір між моделями IaaS (інфраструктура як послуга), PaaS (платформа як послуга), FaaS (функція як послуга) і всіма проміжними варіантами [4]. Щоб прийняти рішення, необхідно оцінити функції і обмеження конкретної служби, її доступність і масштабованість, вартість і можливості для управління процесом розробки. При виборі технологій для зберігання даних необхідно зрозуміти, з якими даними працює ваш додаток, які дані воно отримує і генерує і які дані створюють його користувачі. Найбільш поширені типи даних - бізнес-дані, кеші, дані Інтернету речей і телеметричні дані, а також неструктуровані журнали. Найчастіше додаток працюють з кількома типами даних. У кожного типу даних є свої вимоги до обробки, тому вам потрібно вибрати відповідне сховище для кожного з них. Деякі технології зберігання даних підтримують різні моделі зберігання. На підставі наведеної в цьому розділі інформації виберіть модель зберігання даних, яка найбільшою мірою відповідає вашим вимогам. Після цього визначте конкретний сховище даних у відповідній категорії, виходячи з таких

параметрів, як функціональні можливості, вартість та зручність управління.

При виборі технології для обчислень необхідно враховувати наступні обставини:

1. Модель розміщення.
2. Управління процесом розробки.
3. Масштабованість.
4. Доступність.
5. Вартість.

Крім вартості власне служби необхідно врахувати операційні витрати на управління побудованим на базі цієї служби рішенням. Зокрема операційні витрати на вирішення IaaS, як правило, бувають більш високими. Які загальні обмеження кожної служби? Які архітектури додатків підходять для цієї чи іншої служби?

На одному кінці набору можливих рішень знаходиться модель IaaS (інфраструктура як послуга). В рамках IaaS готуються необхідні віртуальні машини, а також пов'язані мережеві ресурси і компоненти для зберігання даних. Після цього на них можна розгорнути будь-яке програмне забезпечення і додатки. Така модель найбільше нагадує традиційну локальну середу [4].

У моделі PaaS (платформа як послуга) ви отримуєте керувану середу розміщення, в якій можна розгорнути свій додаток без необхідності керувати віртуальними машинами і мережевими ресурсами [5].

Між IaaS і PaaS є безліч інших проміжних варіантів розгортання.

У моделі FaaS (функція як послуга) ідея відмови від необхідності думати про середовище проживання реалізована в ще більшій мірі. Замість того щоб створювати обчислювальні екземпляри і розгортати на них код, просто розгортається код, а потім служба автоматично запускає його. Не потрібно адмініструвати обчислювальні ресурси. У цих службах застосовується без серверна архітектура, що забезпечує просте масштабування відповідно до потреб в обробці трафіку.

IaaS забезпечує максимальний контроль, гнучкість і переносимість, FaaS - простоту, гнучку масштабованість і потенційну економію фінансів, оскільки оплачується тільки час, протягом якого виконувався код.

PaaS займає проміжне положення. У загальному випадку чим більш гнучкою є служба, тим більший обсяг обов'язків по конфігурації і управління ресурсами лягає на



користовуча. Послуги FaaS автоматично керують багатьма аспектами середовища виконання програми, а рішення IaaS дозволяють готувати, налаштовувати і контролювати створювані віртуальні машини і мережеві компоненти.

Вибір технології зберігання даних, що відповідають вимогам, є одним з ключових рішень, прийнятих на етапі проектування. Можна вибирати з сотень технологій баз даних SQL і NoSQL. Сховища даних зазвичай класифікують за структурою даних і за типами операцій, які вони підтримують.

Не всі технології зберігання даних в рамках однієї категорії володіють одним і тим же набором функцій. Більшість таких технологій підтримують серверні механізми запитів і обробки даних. В одних випадках ці функції вбудовані в систему управління сховищем, в інших - функції зберігання і обробки даних відокремлені один від одного, і можна вибирати між різними технологіями обробки і аналізу. Крім того, технології зберігання даних підтримують різні інтерфейси програмування і управління [6].

У загальному випадку спочатку треба вирішити, яка модель зберігання даних найкраще відповідає вимогам. Після цього визначається конкретне сховище даних у

відповідній категорії, виходячи з таких параметрів, як функціональні можливості, вартість та зручність управління.

V. ВИСНОВОК

Поява і розвиток нових тенденцій, вимог користувачів і можливостей дає можливість постійно вдосконалювати архітектури. Вибір правильної архітектури для застосування, підбір найбільш підходящих технологій для обчислювальних систем і систем зберігання даних дозволяє переходити до проектування та розробки хмарного додатка.

ЛІТЕРАТУРА REFERENCES

- [1] Gillam, Lee, Nick Antonopoulos «Cloud Computing: Principles, Systems and Applications», *Computer Communications and Networks* Springer, 2010, ISBN 9781849962407, 379 p.
- [2] Amazon S3 Available: http://ru.wikipedia.org/wiki/Amazon_S3
- [3] Windows Azure. Available: http://en.wikipedia.org/wiki/Windows_Azure
- [4] Orlando D. “Modeli servisnykh oblachnykh vycherpan’: infrastruktura yak servis” Available: <http://www.ibm.com/developerwork/ru/library/cloudservices1iaas/>.
- [5] Orlando D. “Modeli servisnykh oblachnykh vypadiv: platforma yak servis”, Available: <http://www.ibm.com/developerworks/ru/library/cl-cloudservices2paas/>.
- [6] Miller R. Who “Has the Most Web Servers?”, Available: <http://www.datacenterknowledge.com/archives/2009/05/14/whos-got-the-most-web-servers/>

