

## **ОБҐРУНТУВАННЯ ВИКОРИСТАННЯ ХМАРНОГО ТА ВИДІЛЕНОГО СЕРВЕРІВ У МОБІЛЬНИХ ДОДАТКАХ**

О.І. Сироткіна, А.О. Козаков  
(Україна, Дніпро, НТУ «Дніпровська політехніка»)

У повсякденному житті більше мільярда людей використовують смартфони та планшети, отже існує зростаючий попит на мобільні рішення та додатки. Користувачі можуть вибрати із сотень тисяч програм, тому для того, щоб мобільні розробники мали успіх, важливо створити переконливий, цікавий та споріднений користувацький досвід. Як правило, це вимагає наявності допоміжних компонентів або служб для подання програми з відповідними даними, оповіщеннями, а також дозволяти взаємодію між користувачами тощо.

Для мобільних пристроїв та додатків часто використовують зовнішні та внутрішні шляхи розробки. Поки зовнішні шляхи визначають користувацький інтерфейс, взаємодію з користувачем та презентацію інформації, використовують бізнес-логіку, зберігання даних та безпеку. Внутрішній – це сервер або сервери, де ці дані зберігаються та передаються.

Дуже мало програм працюють без певного підключення до Інтернету, тобто вони взаємодіють з бекендом, веб-службами або API. Ці API можуть бути надані різноманітними сервісами, такими як Google, Amazon, Facebook. Також можуть бути API, які розробляються на виділеному сервері. Тобто існує два різні напрямки розробки серверів для зберігання даних та використання у мобільних додатках: хмарні технології (BaaS–Back-End-as-a-Service); розробка власного виділеного серверу.

Хмарні сервіси дозволяють керувати централізованою базою даних, яка пропонує користувачам спільно використовувати вміст через хмару. Розробнику не потрібно розробляти свій власний сервер з використанням серверної технології, такої як Ruby або PHP. Коли сторонній постачальник послуг відповідає за базову IT-інфраструктуру, розробник додатка більше не відповідає за придбання виділеного сервера або віртуальної машини. У той же час, постачальник послуг може вирішити, як ефективно використовувати свою інфраструктуру, щоб обслуговувати запити від усіх своїх клієнтів. Як наслідок, постачальник послуг зазвичай не зобов'язаний виконувати постійне навантаження для конкретного клієнта. Замість цього програма обробляє запити від усіх клієнтів одночасно, витрачаючи лише обмежений час на обробку кожного запиту конкретного клієнта. Тому такі постачальники, як правило, платять своїм клієнтам на основі загальної кількості запитів, частоти запитів протягом певного періоду або загального часу, витраченого на обслуговування всіх запитів від клієнта.

Такий самий підхід добре працює для різних типів навантажень, оскільки сторонні постачальники послуг, як правило, мають еластичні, масштабовані послуги. Коли клієнти починають користуватись сторонньою службою,

очікувана кількість запитів може бути низькою, а клієнти платять значно менше, ніж придбання, налаштування та управління власною інфраструктурою на місці або в хмарі. Для більшої кількості запитів (або очікуваних, або несподіваних піків навантаження) клієнтам не потрібно додавати сервери, щоб масштабувати їх застосування. Натомість постачальник послуг піклується про збільшення навантаження. Безумовно, обробка більшої кількості запитів обійдеться дорожче, але в більшості випадків таке рішення ще ефективніше, ніж використання спеціалізованої ІТ-інфраструктури. Відмінності хмарних технологій від виділених зображено на рис. 1.

Найважливішою перевагою BaaS є те, що вона звільняє розробників від тягаря будівництва та управління резервними системами. Це дозволяє програмісту сконцентруватися на більш важливих матеріалах, таких як розробка інтерфейсу для спілкування з користувачем, який буде фактичним фактором успіху додатка. Крім того, це допомагає розробнику уникати складних навчальних кривих, які зазвичай пов'язані з більшістю базових технологій. Таким чином, він скорочує витрати та час розробки. Розробник також отримує дешевий спосіб експериментувати з ідеями додатків та бачити, як вони працюють у реальному житті.

Найбільшим недоліком є небезпека, що постачальник BaaS-сервісу може раптово вийти з бізнесу та закрити цю службу. У такому сценарії, навіть якщо перейти на іншого постачальника, розробнику може знадобитися суттєво переробляти та перекодувати додаток, оскільки в новій службі може бути зовсім інший API. Насправді, один з найпопулярніших провайдерів BaaS, Pars, нещодавно закритий, який вплинув на багатьох розробників (хоча інфраструктура Parse була випущена під ліцензією з відкритим вихідним кодом, а нові постачальники не передбачали забезпечення Pars-сумісного BaaS).

Хоча окремі провайдери BaaS пропонують різноманітні функції за допомогою великої різноманітності моделей ціноутворення, більшість з них використовують певну модель "freemium". Це означає, що основні функції, такі як зберігання даних, аналітика користувачів, використання додатку, push-сповіщення та аутентифікація, надаються безкоштовно, до певного ліміту використання. Щойно використання перевищує цей ліміт або запитують додаткові функції, стягується плата. Це дає змогу легко створювати та запускати додатки на вільному рівні використання, а потім збільшувати їх до платного рівня, коли зростає кількість клієнтів.

Інший недолік полягає в тому, що налаштування базової інфраструктури в BaaS часто обмежується. Це може означати, що деяка функціональність, яку розробник хоче впровадити для свого додатка, недоступна.

Альтернативою використання хмарних технологій є використання окремого бекенд серверу, який цілком розробляється для використання в конкретному додатку з певними функціями, запитами та API. Для розробки даного рішення програміст повинен освоїти додаткові навички та компоненти такі, як HTTP, REST, JSON, та ін. Виділений сервер – це фізичний сервер, який придбаний або орендований повністю для власних потреб бізнесу

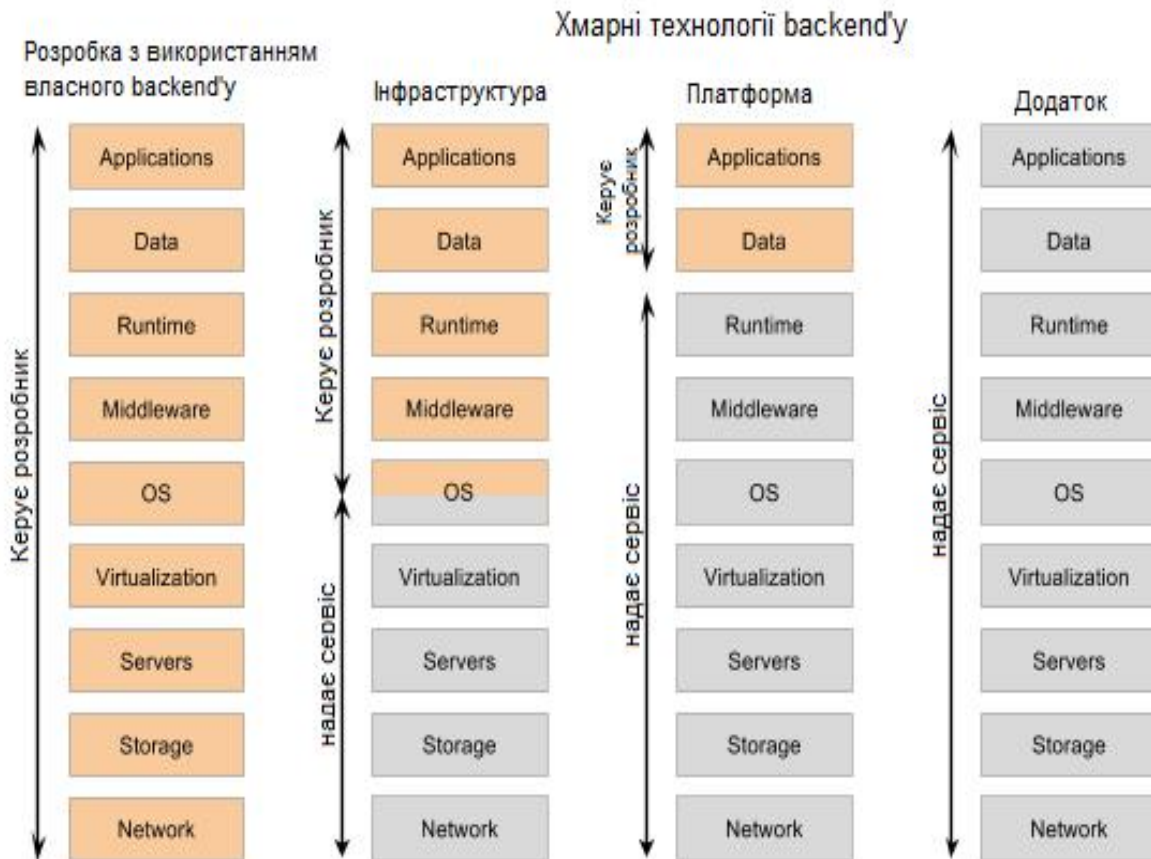


Рис. 1. Відмінності хмарних технологій сервера від виділеного

Виділені сервери відмінно підходять для великих підприємств, організацій, які потребують надзвичайно високого рівня безпеки даних, або організаціям, які мають високий попит, вимагаючи від усіх своїх серверів, що працюють щодобову роботу. Компанії, що використовують програми з великою кількістю користувачів, такі як бази даних та платформи BigData, також мають значну цінність.

Вибір правильного сервера та правильного підходу є важливим рішенням для розробки мобільних додатків. Вибір правильної технології, буде залежати від бізнес-цілей і цілей, а також від першочергових та довгострокових вимог до ІТ.

#### ПЕРЕЛІК ПОСИЛАНЬ:

1. MartinMonroe. TheGospelofMBaaS. InfoQ, pages 1-16, 2013.
2. KinLane. RiseofMobileBackendas a service (MBaaS) API Stacks. API Evangelist, pages 13-21, 2012.
3. MichaelFacemire. EnterpriseMobility: howfastcandevlopmentgo. Forrester.compage 2, 2012.