

Тема 10. Платформні архітектури ІС

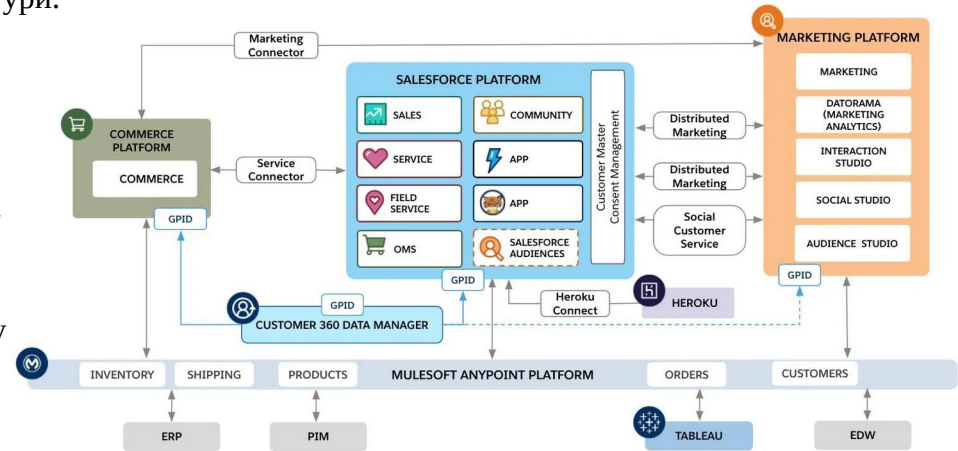
Платформні архітектури ІС

Поняття платформної архітектури виникло як результат розвитку інформаційних систем від ізольованих рішень до інтегрованих екосистем. На початкових етапах розвитку ІТ-інфраструктури підприємства створювали окремі системи для кожної функції — бухгалтерії, логістики, CRM, документообігу. Кожна з них мала власну базу даних, користувачів і інтерфейс, що ускладнювало інтеграцію й обмін інформацією.

Поступово стало очевидно, що фрагментована ІТ-структура гальмує розвиток бізнесу. Підприємства потребували єдиного середовища, де дані, користувачі й додатки могли б взаємодіяти за спільними правилами. Саме цю роль почали виконувати платформні архітектури, які інтегрують сервіси, забезпечують стандарти взаємодії і створюють спільну основу для розвитку цифрових продуктів.

Платформа, на відміну від окремої системи, не є кінцевим продуктом — це середовище для розвитку, яке дозволяє швидко створювати, масштабувати та інтегрувати нові сервіси без перебудови всієї архітектури.

- Платформа — це середовище для створення, інтеграції й управління додатками.
- Сучасні ІС дедалі частіше базуються на платформному підході.
- Це забезпечує масштабування, швидкість розробки й ефективну інтеграцію.
- **Платформна архітектура ІС** — це структура, що надає спільну інфраструктуру, сервіси й інтерфейси для побудови екосистеми взаємодіючих додатків.



Історія розвитку

Еволюція платформного підходу відбувалася поступово:

- 1990–2000-ті роки — домінування монолітних корпоративних систем, таких як SAP R/3 або Oracle E-Business Suite. Вони об'єднували ключові бізнес-процеси, але залишалися жорстко зв'язаними та складними в обслуговуванні.
- 2000–2010-ті — поява Service-Oriented Architecture (SOA). Цей підхід уперше запропонував відокремити бізнес-логіку від інтерфейсу та обмінюватися даними через стандартизовані сервіси (наприклад, SOAP).
- Після 2010 року — поширення хмарних технологій (AWS, Azure, Google Cloud), які започаткували нову модель: Platform as a Service (PaaS).
- 2020-ті роки — домінування cloud-native, microservice та AI-driven архітектур. Платформа стає не лише технологічним середовищем, а й центром управління даними, автоматизації та аналітики.

Таким чином, платформна архітектура сьогодні є кульмінацією багаторічного розвитку корпоративних IT-систем — від жорстко зв'язаних продуктів до відкритих цифрових екосистем.

Ознаки

Платформна архітектура — це структура, яка надає набір спільних сервісів (authentication, logging, analytics), інтерфейсів (API, REST, GraphQL) та інструментів розробки (SDK), що дозволяють створювати й інтегрувати додатки у межах єдиного середовища.

Платформа діє як «операційна система» для бізнесу: вона визначає стандарти, протоколи взаємодії та життєвий цикл компонентів. Таким чином, підприємства отримують не просто набір застосунків, а екосистему, де кожен модуль може бути розроблений незалежно, але працює в єдиному контексті.

Вона має наступні ознаки:

- Модульність і розширюваність
- Відкриті API
- Сервіси автентифікації, логування, аналітики
- Каталог застосунків
- Управління життєвим циклом

Проблеми, які вирішують платформні архітектури

Платформні архітектури вирішують низку системних проблем, притаманних традиційним ІС:

- Інтеграція. Вони забезпечують спільні інтерфейси (API, SDK) для обміну даними між різними компонентами.
- Швидкість інновацій. Нові сервіси можна додавати без порушення основної логіки системи.
- Масштабованість. Платформа дозволяє автоматично збільшувати ресурси при зростанні навантаження.
- Безпека. Єдиний механізм автентифікації, логування та моніторингу.
- Зниження витрат. Повторне використання компонентів замість розробки «з нуля».

Ці властивості роблять платформну архітектуру основою сучасної цифрової трансформації підприємств.

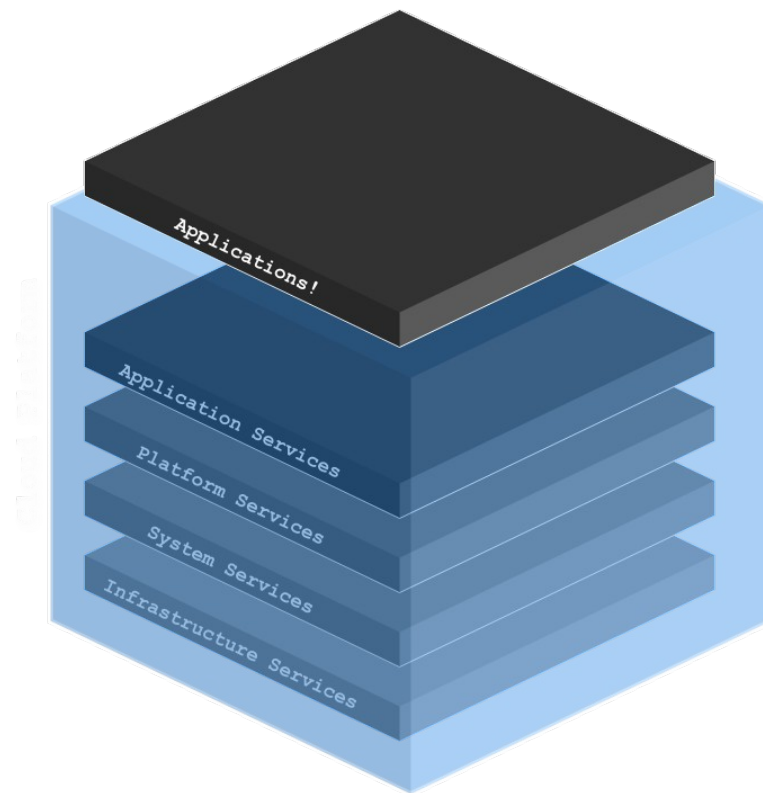


Типова архітектура платформи

Типова платформа має багаторівневу структуру:

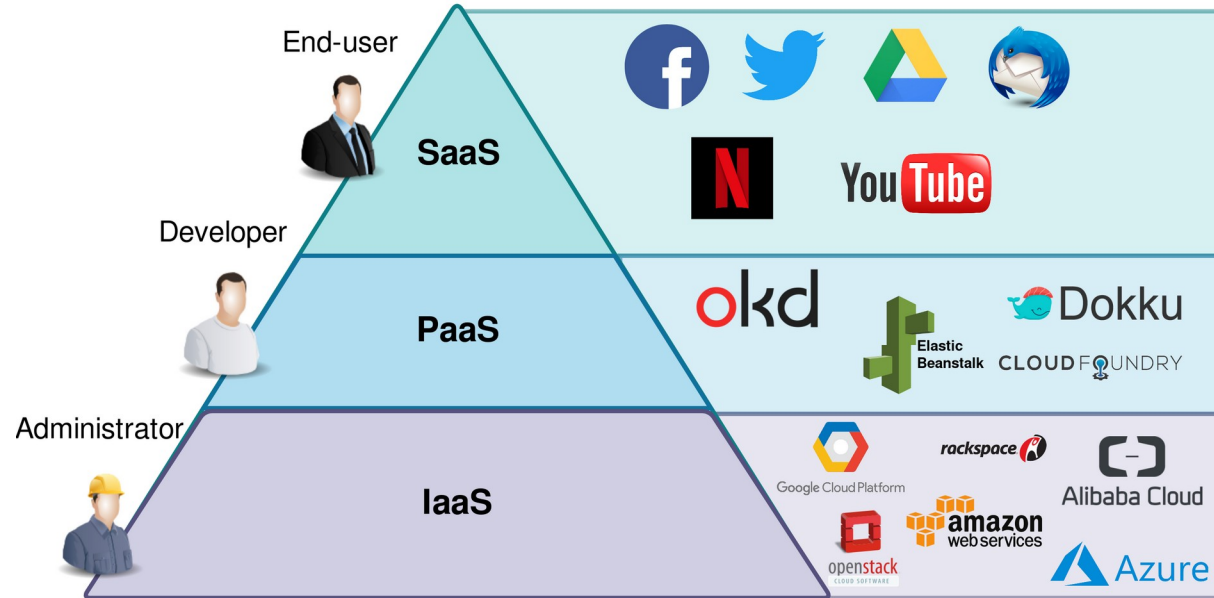
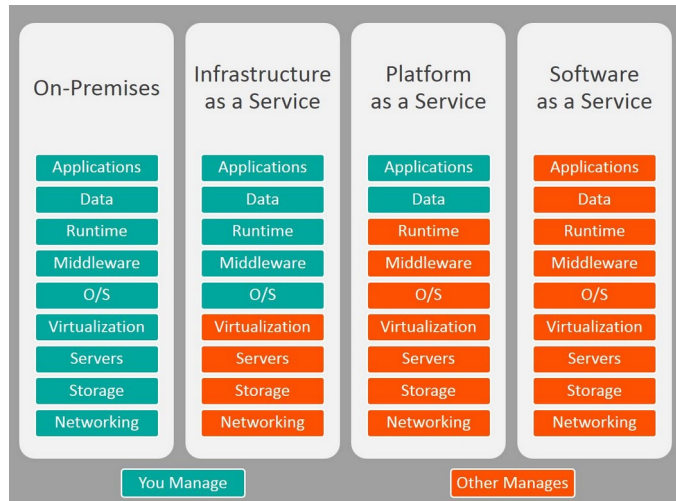
- Інфраструктурний рівень — фізичні або віртуальні сервери, контейнеризація (Docker), оркестрація (Kubernetes).
- Сервісний рівень — бази даних, черги повідомлень (RabbitMQ, SQS), сховища файлів.
- Прикладний рівень — бізнес-логіка, мікросервіси, API.
- Рівень доступу — вебінтерфейси, мобільні додатки, панелі керування.

Такий підхід забезпечує відокремлення обов'язків, гнучкість і можливість незалежного оновлення кожного шару.



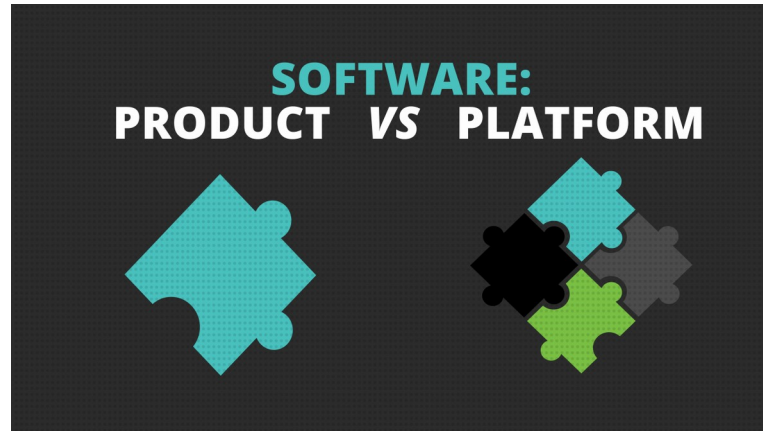
Види платформ

1. Інфраструктурні (AWS, Azure, GCP)
2. Платформи розробки (Heroku, OpenShift)
3. Бізнес-платформи (Salesforce, SAP)
4. Галузеві (BIM 360)
5. Інтеграційні (MuleSoft, Boomi)



Платформа і продукт

- Продукт — виконує конкретну функцію.
- Платформа — забезпечує розвиток екосистеми.
- Користувач продукту — кінцевий споживач, користувач платформи — розробники.

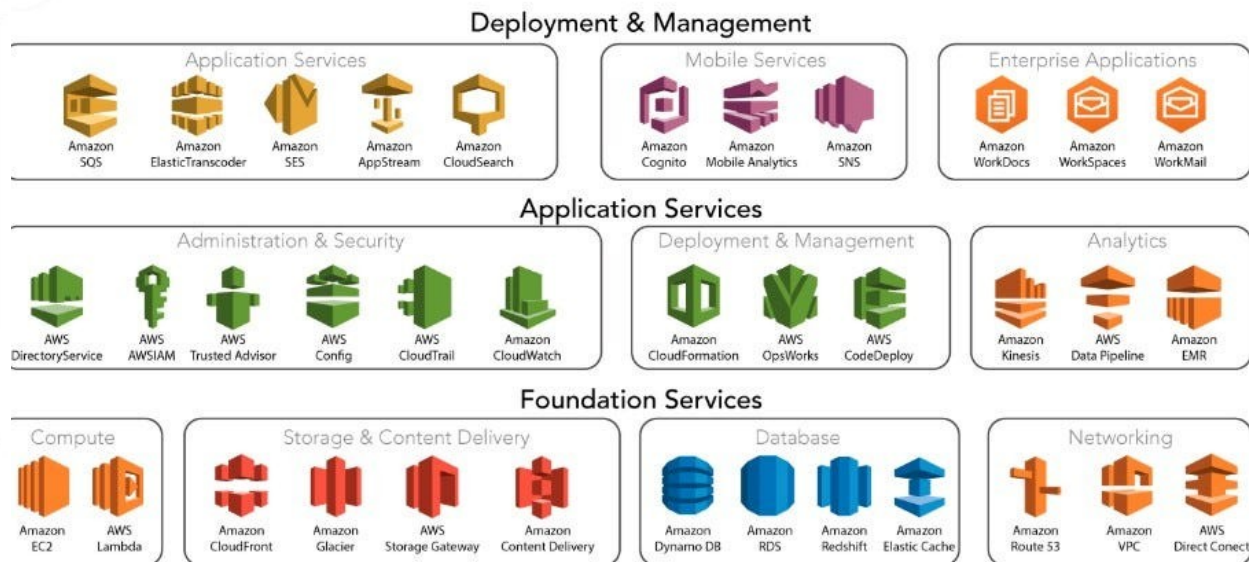
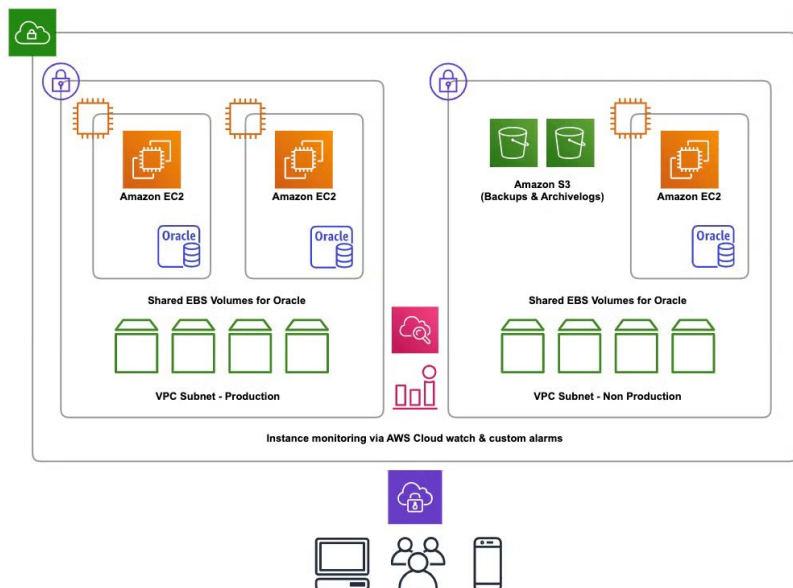


Приклади сучасних платформ

- **AWS (Amazon Web Services)** — одна з найвідоміших інфраструктурних платформ. Вона поєднує IaaS, PaaS і SaaS-рівні, дозволяючи будувати додатки будь-якого масштабу.
- **Salesforce** — бізнес-платформа для управління клієнтськими відносинами, що переросла у середовище створення власних додатків (PaaS) з відкритими API.
- **Kubernetes** — технологічна платформа для оркестрації контейнерів, яка є стандартом для cloud-native архітектур.
- **SAP Business Technology Platform** — приклад корпоративної платформи, що об'єднує аналітику, автоматизацію та інтеграцію.
- **BIM 360** — це платформна система управління життєвим циклом будівельних проєктів, розроблена компанією Autodesk. Вона поєднує інструменти для проєктування, координації, будівництва й експлуатації об'єктів у єдиному хмарному середовищі.

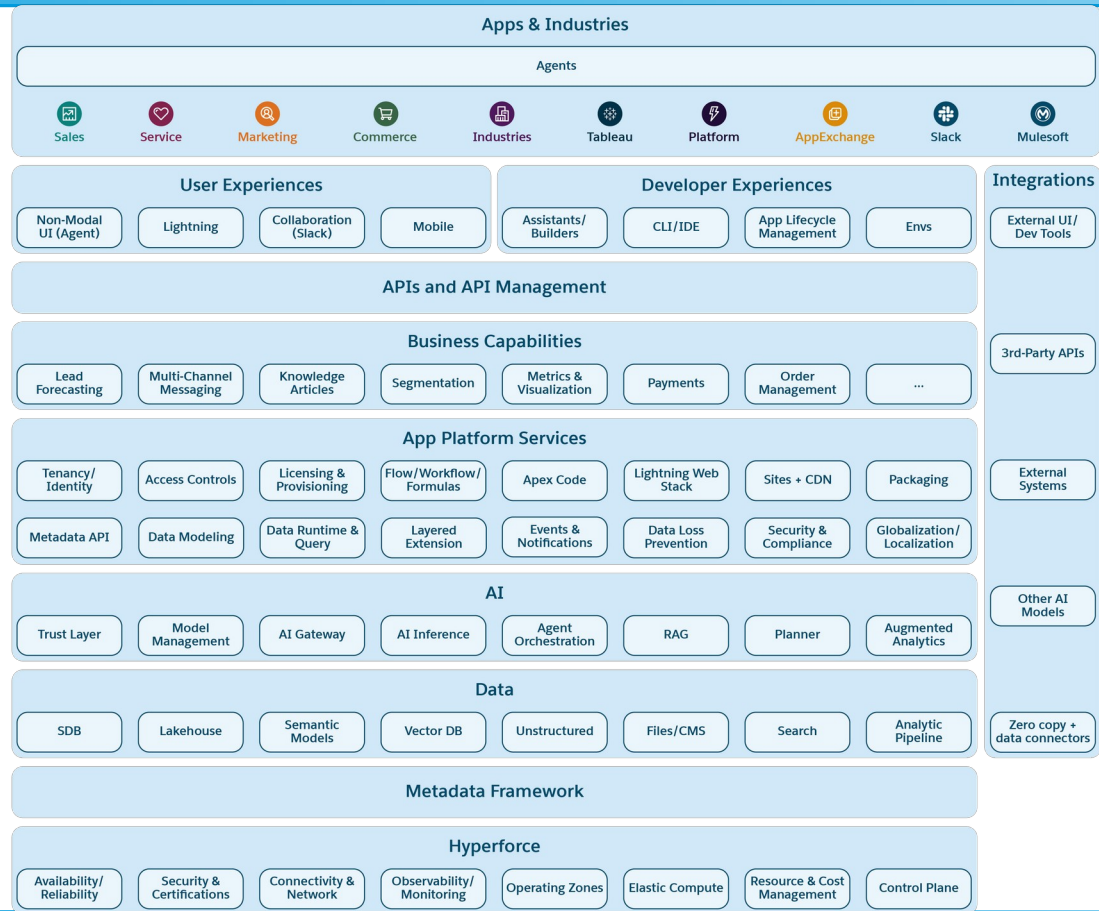
AWS як приклад

AWS — глобальна хмарна платформа, що надає інфраструктуру (IaaS), сервіси (PaaS) та аналітичні інструменти (SaaS). Вона об'єднує сотні сервісів — від віртуальних машин (EC2) до безсерверних функцій (Lambda) і черг повідомлень (SQS), формуючи повноцінну екосистему для побудови та масштабування будь-яких ІС.



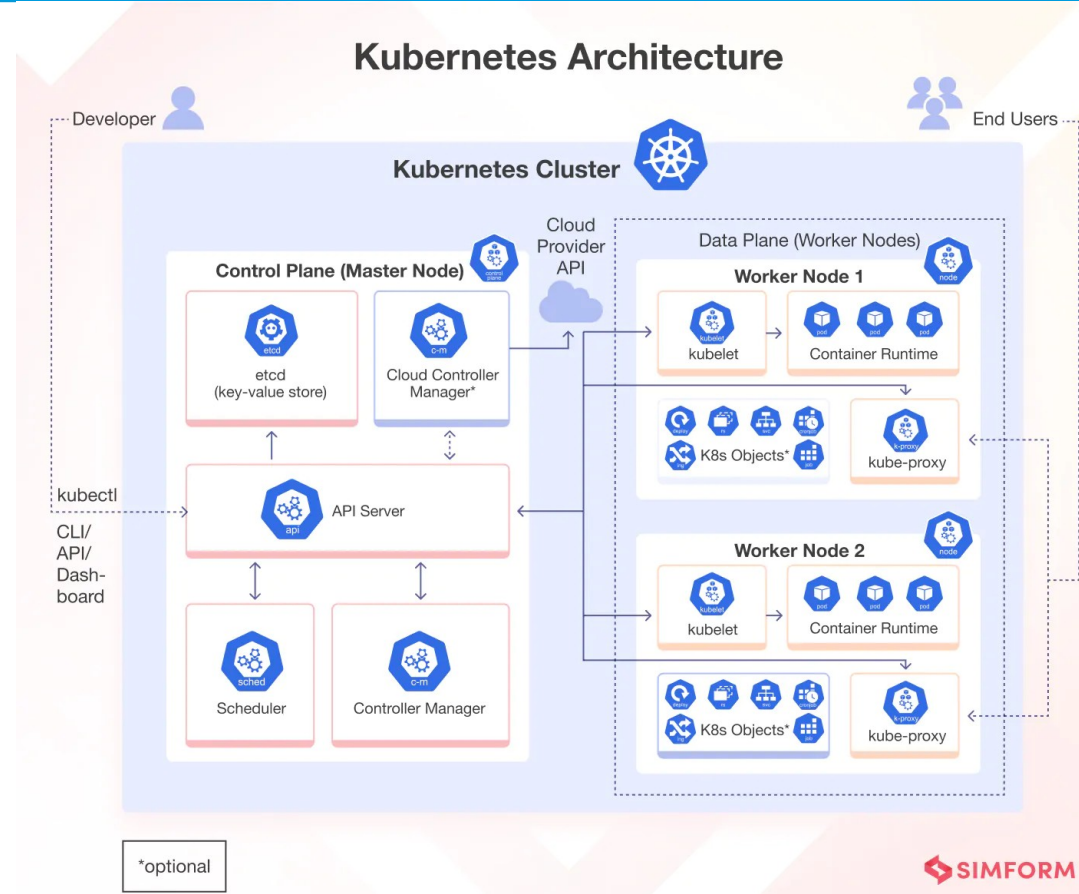
Salesforce Platform

Salesforce — це бізнес-платформа для управління відносинами з клієнтами (CRM), яка еволюціонувала у повноцінне середовище для розробки корпоративних додатків. Її архітектура побудована на метаданих (metadata-driven), підтримує low-code інструменти, аналітику, інтеграцію та автоматизацію бізнес-процесів.



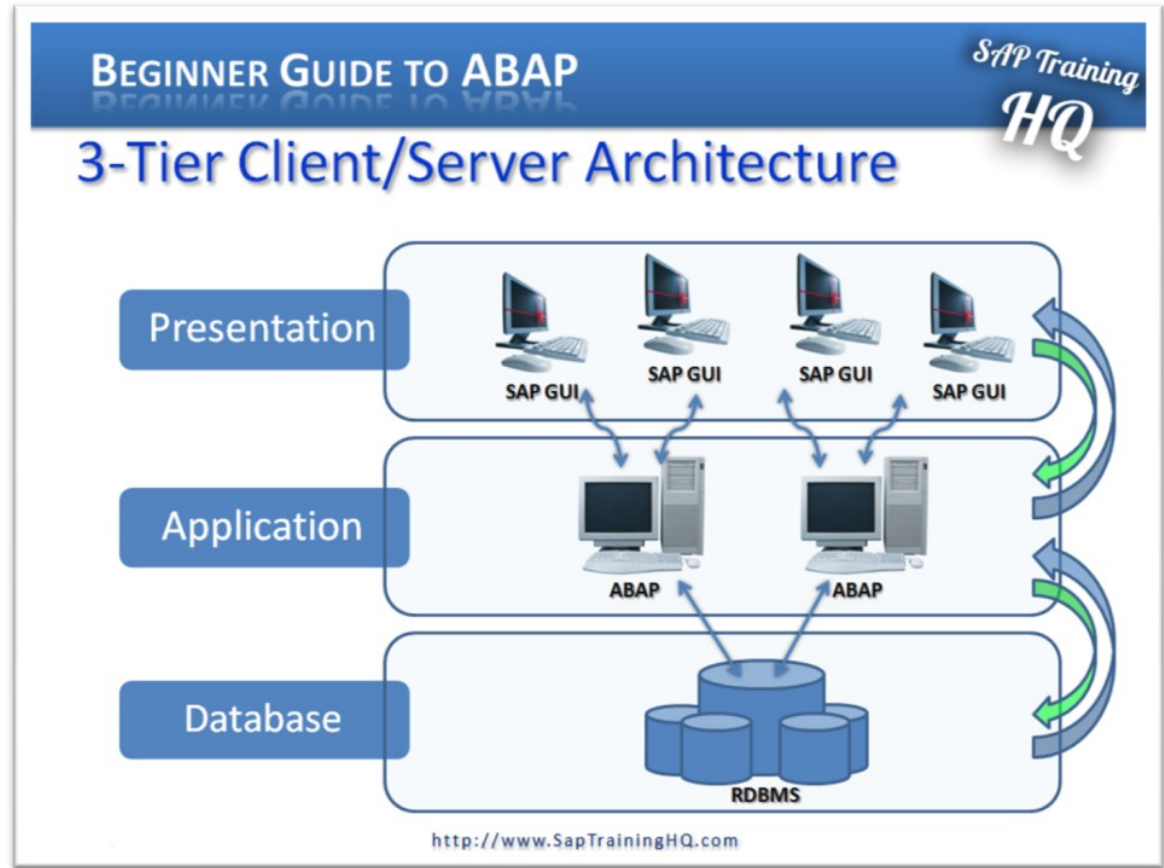
Kubernetes

Kubernetes — відкрита платформа оркестрації контейнерів, яка автоматизує розгортання, масштабування та управління мікросервісами. Вона є фундаментом cloud-native архітектур і дозволяє реалізувати незалежність компонентів, стійкість до збоїв та ефективне використання ресурсів.



SAP Business Technology Platform (BTP)

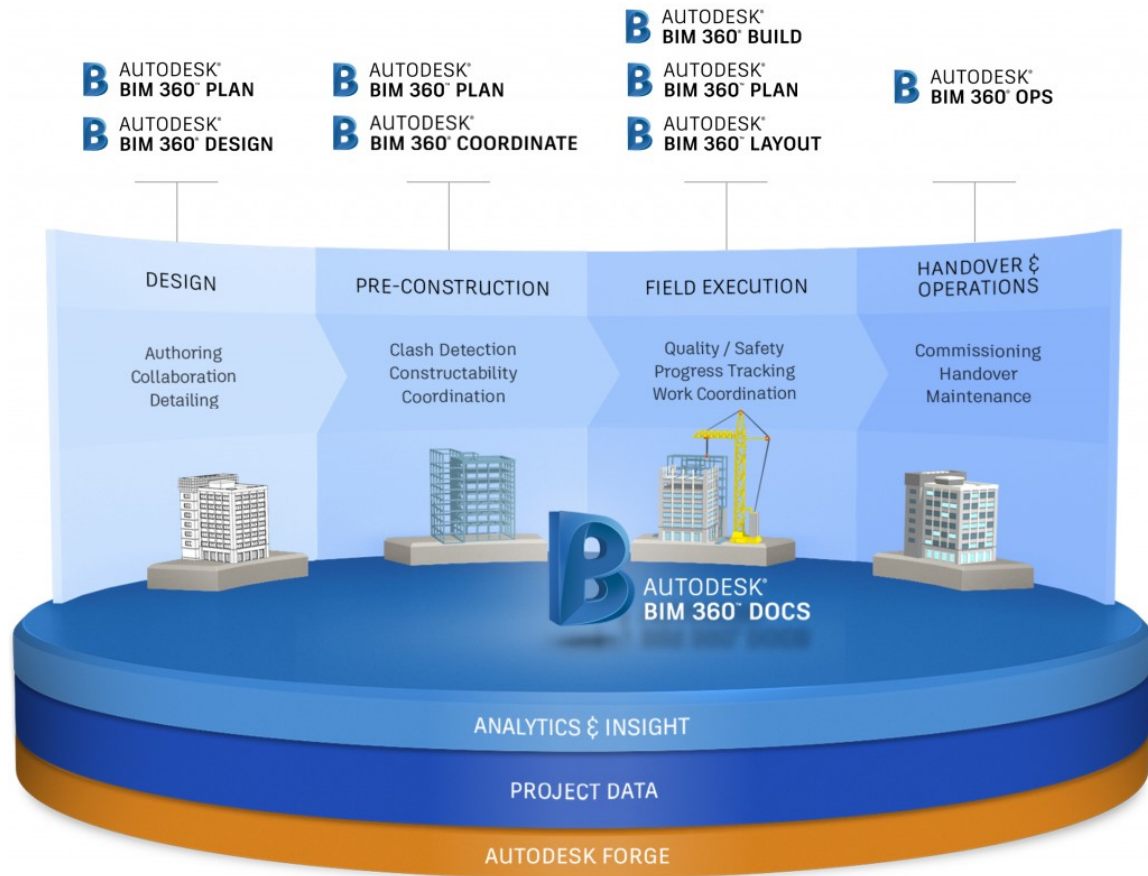
SAP BTP — це корпоративна платформа, що об'єднує аналітику, управління даними, інтеграцію й автоматизацію процесів. Вона забезпечує API-центричну взаємодію між бізнес-додатками SAP і сторонніми сервісами, підтримує хмарні сценарії та інтелектуальні сервіси на базі AI.



BIM 360 (Autodesk Construction Cloud Platform)

BIM 360 — це платформна система управління життєвим циклом будівельних проєктів, розроблена компанією Autodesk. Вона поєднує інструменти для проєктування, координації, будівництва й експлуатації об'єктів у єдиному хмарному середовищі.

Ключовою особливістю BIM 360 є спільна робота з цифровими моделями (BIM — Building Information Modeling) у реальному часі. Платформа забезпечує доступ до актуальних даних усім учасникам проєкту — від архітекторів і інженерів до підрядників та замовників.



Cloud-native підхід

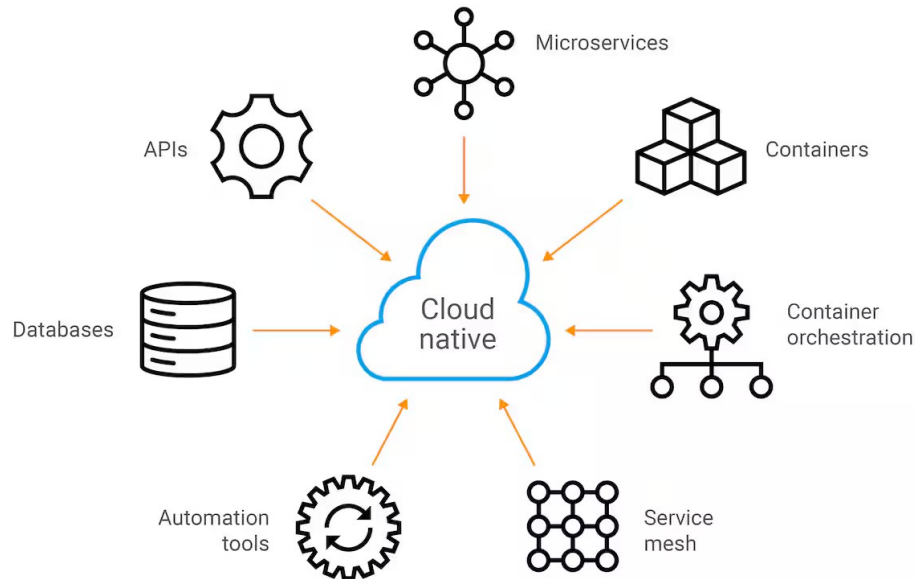
Cloud-native — це сучасна парадигма побудови платформ, що базується на контейнеризації, мікросервісах та DevOps-підходах.

Основні принципи:

- Масштабування через автоматичне розподілення навантаження.
- Безперервність завдяки CI/CD-конвеєрам.
- Стійкість — система залишається працездатною при відмові окремих компонентів.
- Портативність — розгортання у будь-якому середовищі: локальному або хмарному.

Cloud-native платформи забезпечують мінімальний час між розробкою і впровадженням, що критично для інноваційних компаній.

What is cloud native architecture?



API-економіка

API стали ключовим механізмом взаємодії між компаніями, сервісами та пристроями.

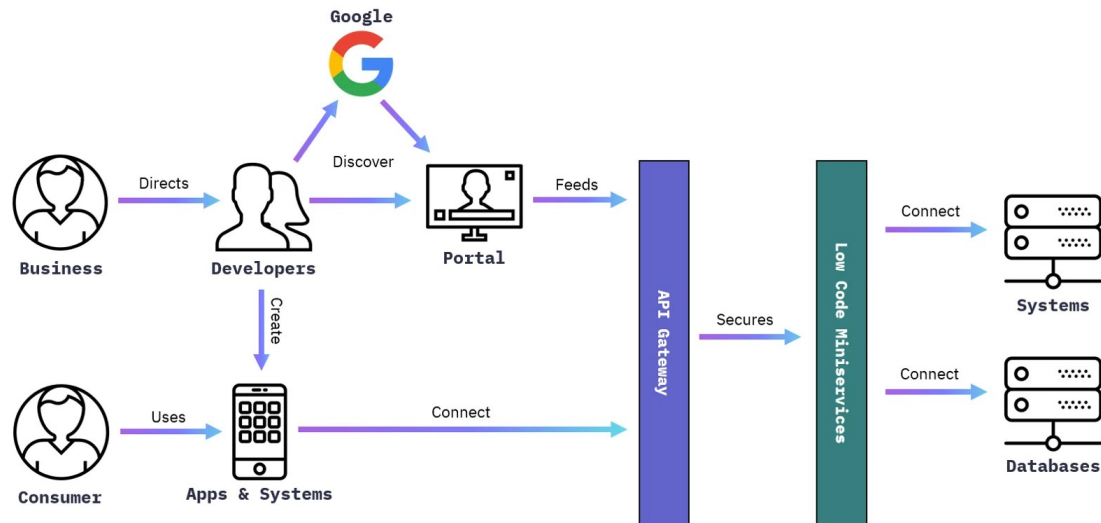
Платформа з відкритими API дозволяє стороннім розробникам створювати нові сервіси, розширюючи екосистему.

Цей підхід лежить в основі API есоному, де дані й функціональність самі по собі стають бізнес-продуктом.

Приклади: Google Maps API, Stripe, OpenAI API.

Ключові технології: OpenAPI, GraphQL, gRPC.

API Economy – The Big Picture



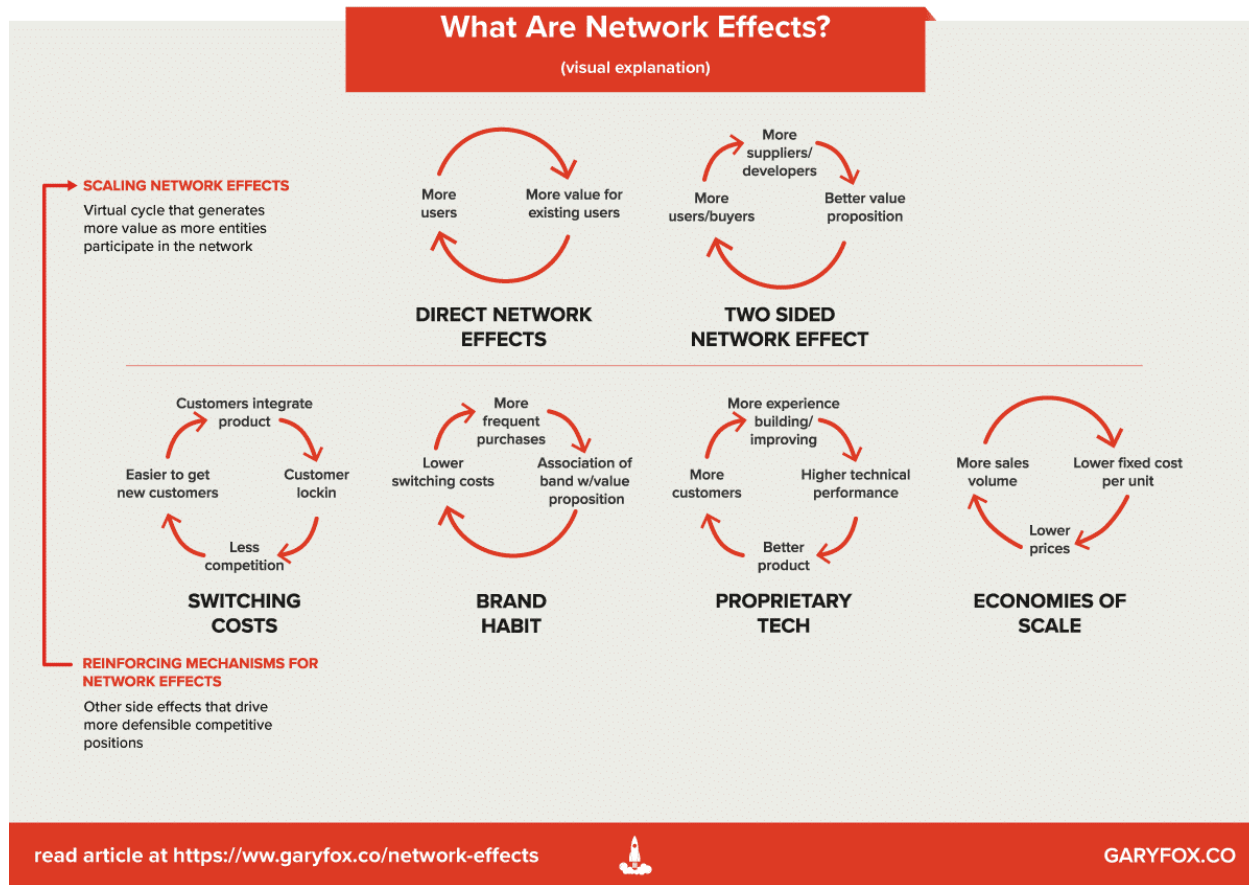
Прикладна екосистема і мережевий ефект

Цінність платформи зростає з кількістю користувачів і розробників.

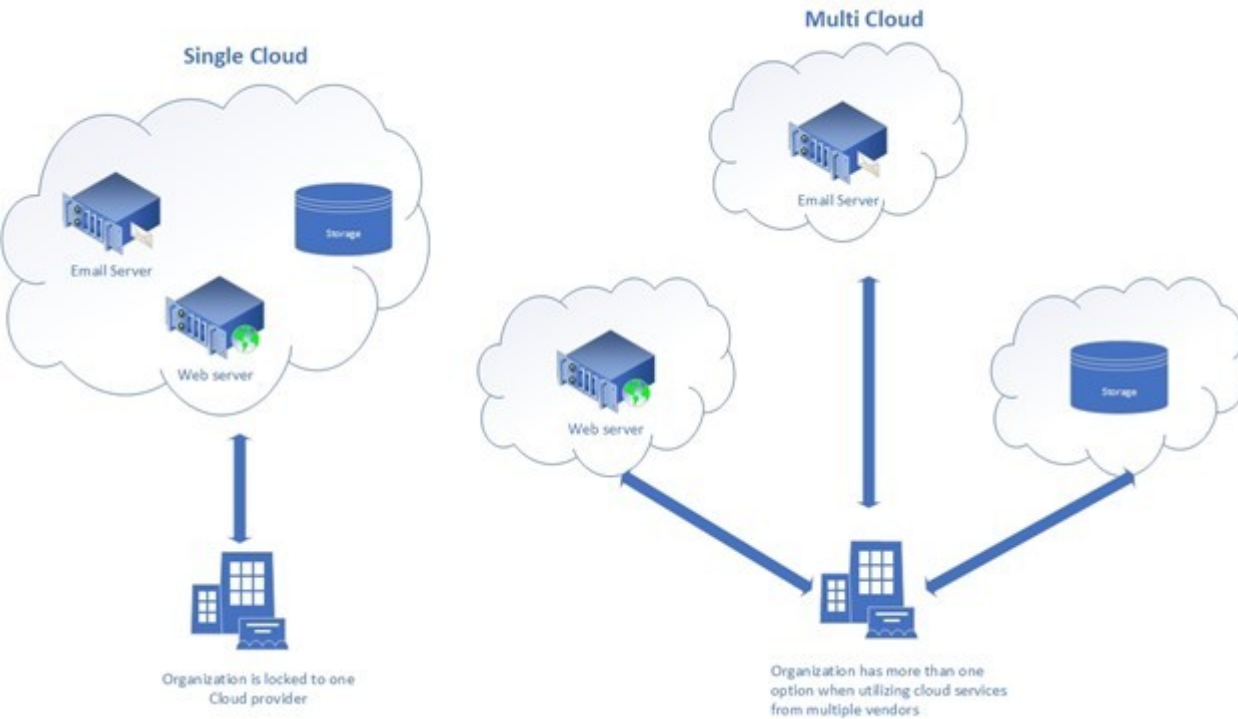
Це явище називають мережевим ефектом.

Приклади — Android, Shopify, AWS Marketplace, де відкритість API і можливість створення розширень формують потужну спільноту навколо продукту.

Таким чином, платформа стає не лише технологічним, а й економічним явищем — ядром бізнес-моделі.



Виклики платформного підходу



Платформна архітектура має також свої обмеження:

Vendor lock-in — залежність від постачальника технології (наприклад, AWS або SAP).

Безпека даних — централізація вимагає жорстких політик доступу.

Керування версіями API — підтримка сумісності з попередніми додатками.

Баланс між стандартизацією та гнучкістю — надмірна централізація може сповільнювати інновації.

Тому архітектори мають знаходити баланс між стабільністю і швидкістю розвитку екосистеми.

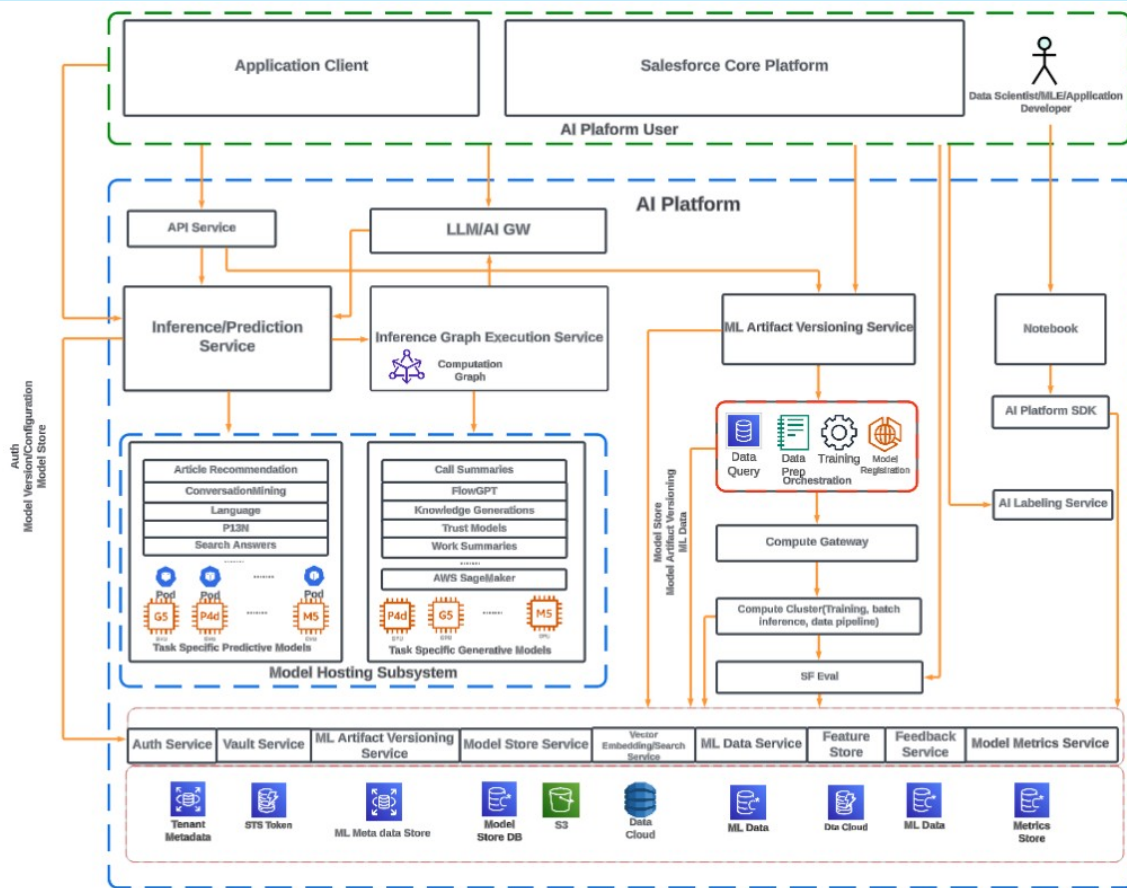
Тенденції

- **Інтеграція штучного інтелекту (AI)** — оптимізація інфраструктури та аналітики.
- **Low-code / No-code платформи** — створення додатків без програмування (PowerApps, AppSheet).
- **Edge computing** — обробка даних ближче до джерела (IoT, промислові системи).
- **Self-service архітектури** — користувач самостійно створює або налаштовує сервіси.

Платформи еволюціонують у напрямку інтелектуальних екосистем, де всі процеси — від збору даних до прийняття рішень — взаємопов'язані.

Аналітика, автоматизація, машинне навчання й адаптивне управління ресурсами утворюють основу для самонавчальних систем.

Сучасні тренди вказують на поєднання платформності, штучного інтелекту та розподіленої обробки даних.



Висновки

Платформні архітектури формують основу сучасних інформаційних систем.

Вони забезпечують інтеграцію, масштабування та інновації, дозволяючи підприємствам перетворювати ІТ-інфраструктуру на стратегічну перевагу.

Тенденції розвитку показують, що майбутнє ІС — це платформна, гнучка, інтелектуальна екосистема, де технології, дані та люди працюють у єдиному цифровому континуумі.

Дякую за увагу!