**Phần 1: Kiến trúc đơn khối (Monolithic) - Kiến trúc phân tán (Distributed)**

**1. Monolithic Architecture**

✅ **Ưu điểm**:

* Thiết kế đơn giản, dễ phát triển, triển khai, debug và test.
* Hiệu suất cao với ứng dụng ít phức tạp do không có độ trễ mạng.

❌ **Nhược điểm**:

* Phát triển chậm khi ứng dụng lớn.
* Khó mở rộng, dễ bị lỗi dây chuyền.
* Rào cản công nghệ, thiếu linh hoạt.
* Mọi thay đổi yêu cầu deploy toàn bộ hệ thống.

**2. Distributed Architecture**

✅ **Ưu điểm**:

* Chịu lỗi tốt, mở rộng dễ dàng.
* Giảm rủi ro triển khai, phạm vi ảnh hưởng nhỏ khi thay đổi.
* Độ trễ thấp, phản hồi nhanh hơn.

❌ **Nhược điểm**:

* Tăng độ phức tạp, cần deploy tự động.
* Hiệu suất tổng thể thấp hơn, tốn tài nguyên hơn.
* Khó debug, quản lý giao dịch phân tán phức tạp.
* Chi phí vận hành cao.

**Phần 2: Các mẫu kiến trúc phổ biến**

**1. Layered Architecture (Kiến trúc phân lớp)**

✅ **Ưu điểm**: Dễ quản lý, phù hợp với dự án monolithic, chi phí thấp.  
❌ **Nhược điểm**: Logic phải truyền qua từng lớp dù không cần xử lý, gây lãng phí tài nguyên.  
📌 **Dùng khi**: Dự án đơn giản, có ngân sách và thời gian hạn chế.

**2. Client-Server Architecture**

✅ **Ưu điểm**: Dễ quản lý, mở rộng, tối ưu tài nguyên.  
❌ **Nhược điểm**: Dễ quá tải server, dễ bị tấn công DoS.  
📌 **Dùng khi**: Hệ thống cần quản lý tập trung, bảo mật cao.

**3. Pipeline Architecture**

✅ **Ưu điểm**: Dễ bảo trì, nâng cấp, mở rộng.  
❌ **Nhược điểm**: Các filter cần tuân thủ định dạng dữ liệu chung.  
📌 **Dùng khi**: Xử lý dữ liệu qua nhiều công đoạn độc lập.

**4. Event-Driven Architecture (EDA)**

✅ **Ưu điểm**: Linh hoạt, mở rộng tốt, giảm tải hệ thống.  
❌ **Nhược điểm**: Khó debug, cần message broker, có thể gây độ trễ.  
📌 **Dùng khi**: Xử lý dữ liệu theo sự kiện, yêu cầu mở rộng linh hoạt.

**5. Service-Oriented Architecture (SOA)**

✅ **Ưu điểm**: Tái sử dụng cao, tích hợp tốt, dễ bảo trì.  
❌ **Nhược điểm**: Giao tiếp mạng tốn băng thông, cần ESB, độ trễ cao.  
📌 **Dùng khi**: Cần tích hợp với hệ thống cũ (legacy systems).

**6. Microservices Architecture**

✅ **Ưu điểm**: Dễ mở rộng, linh hoạt, sử dụng công nghệ đa dạng, tăng độ ổn định.  
❌ **Nhược điểm**: Quản lý phức tạp, khó giao tiếp giữa services, yêu cầu DevOps mạnh, chi phí cao.  
📌 **Dùng khi**: Hệ thống lớn, cần mở rộng linh hoạt và triển khai nhanh.