

Tugas Arsitektur Perangkat Lunak

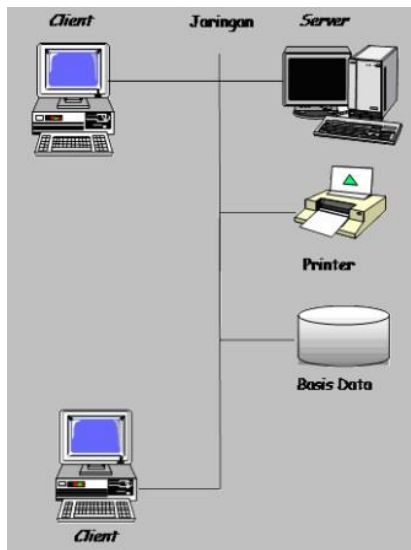
Architectural Style

1. Architectural Style Client-Server

Arsitektur jaringan Client Server merupakan model konektivitas pada jaringan yang membedakan fungsi computer sebagai Client dan Server. Arsitektur ini menempatkan sebuah komputer sebagai Server. Server ini bertugas memberikan pelayanan kepada terminal-terminal lainnya yang terhubung dalam sistem jaringan atau yang disebut Client. Server juga dapat bertugas untuk memberikan layanan berbagi pakai berkas (file server), printer (printer server), jalur komunikasi (server komunikasi).

Pada model arsitektur ini, Client tidak dapat berfungsi sebagai Server, tetapi Server dapat berfungsi menjadi Client (server non-dedicated). Prinsip kerja pada arsitektur ini sangat sederhana, dimana Server akan menunggu permintaan dari Client, memproses dan memberikan hasil kepada Client, sedangkan Client akan mengirimkan permintaan ke Server, menunggu proses dan melihat visualisasi hasil prosesnya.

Sistem Client Server ini tidak hanya diperuntukkan bagi pembangunan jaringan komputer skala luas. Sistem ini menggunakan protokol utama Transmission Control Protocol/Internet Protocol (TCP/IP), sedangkan sistem operasi yang digunakan antara lain Unix, Linux dan Windows NT.



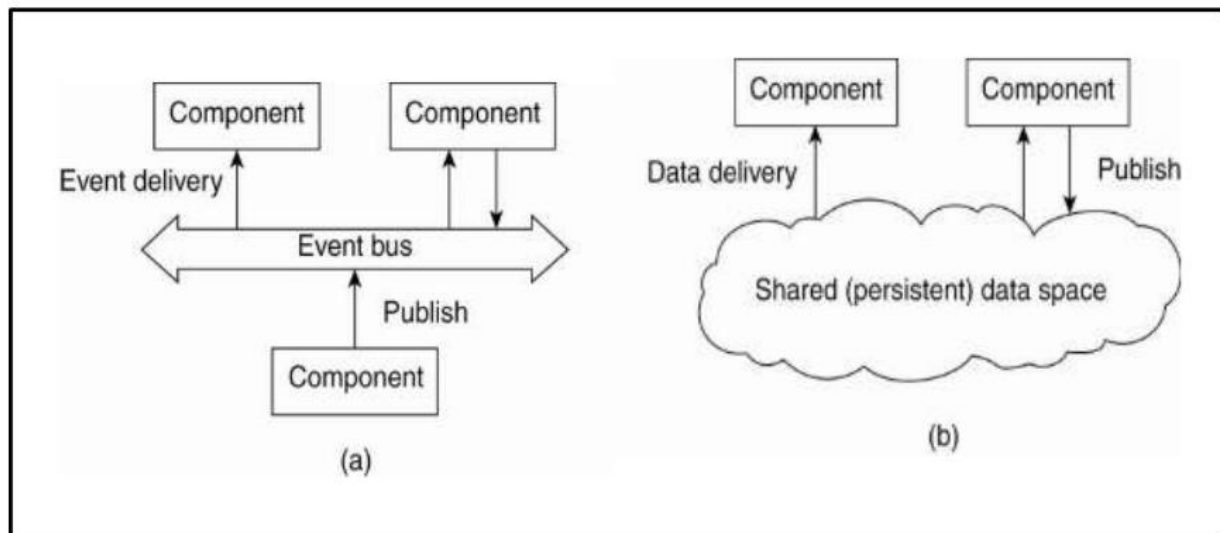
Contoh Real Aplikasi:

- Jaringan LAN sebuah kantor yang berbagi sumber daya secara bersama seperti printer, computer server, Scanner, dan perangkat lainnya..

2. Architectural Style Event-Based

Event Based Architectural pada dasarnya berdasarkan propagasi event. Proses mengeluarkan event setelah Middleware memberikan kepastian hanya proses itu saja yang bisa di subscribe untuk event yang diterima. Keuntungan EBA adalah proses bersifat loosely coupled.

Sebagai contoh, dalam sistem Bidang, alat-alat seperti editor dan variabel monitor mendaftarkan untuk breakpoint peristiwa debugger. Ketika debugger berhenti di breakpoint, itu mengumumkan suatu peristiwa yang memungkinkan sistem untuk secara otomatis memanggil metode alat tersebut terdaftar. Metode ini mungkin sebuah gulir editor untuk garis sumber yang tepat atau menampilkan kembali nilai dipantau variabel. Dalam skema ini, debugger hanya mengumumkan suatu peristiwa, tetapi tidak tahu lain alat apa (jika ada) prihatin dengan peristiwa itu, atau apa yang mereka akan lakukan ketika peristiwa yang diumumkan.



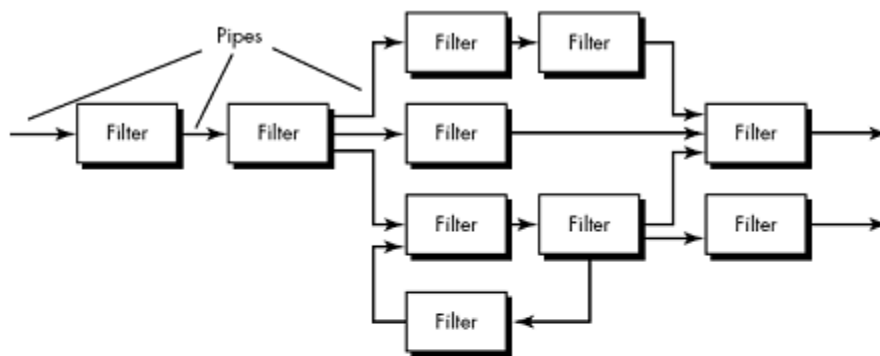
3. Architectural Style Pipes and Filter

Pada model ini masing-masing memiliki satu set input dan satu set output. Komponen membaca aliran data pada input dan menghasilkan aliran data outputnya, memberikan contoh lengkap hasilnya dalam urutan standar. Hal ini biasanya dicapai dengan menerapkan local transformasi untuk memasukkan aliran dan komputasi bertahap sehingga output input dimulai sebelum dikonsumsi. Oleh karena itu komponen yang disebut "filter". Konektor gaya ini berfungsi sebagai medium untuk sungai, transmisi output satu filter untuk masukan lain. Oleh karena itu konektor ini disebut "pipa".

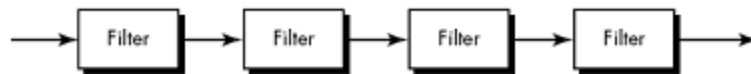
Di antara invariants penting dari gaya, filter harus independen entitas: khususnya, mereka tidak harus berbagi negara dengan filter lainnya. Lain invariant penting adalah bahwa filter tidak mengetahui identitas mereka hulu dan hilir filter. Spesifikasi mereka mungkin membatasi apa yang muncul pada masukan pipa atau membuat jaminan tentang apa yang muncul pada pipa

output, tetapi mereka tidak dapat mengidentifikasi komponen-komponen di ujung pipa tersebut. Selanjutnya, kebenaran output dan menyaring jaringan pipa tidak boleh bergantung pada urutan filter yang melakukan pemrosesan tambahan mereka-meskipun penjadwalan wajar dapat diasumsikan.

Spesialisasi umum dari gaya ini meliputi saluran pipa, yang membatasi topologi untuk urutan linear filter, pipa berikat yang membatasi jumlah data yang dapat berada pada pipa, dan diketik pipa, yang mengharuskan data yang melewati antara dua filter memiliki tipe yang didefinisikan dengan baik.



(a) Pipes and filters



(b) Batch sequential