

Estilo	COMPONENTES	CONECTORES	DATOS	TOPOLOGIA	RESTRIC	CUALIDADES	USOS TIPICOS	PRECAUCIONES
Sistemas en Capas	Capas	Llamadas Parametros de llamada y valores de retorno	Parametros de llamada y valores de retorno	Lineal o grafo dirigido aciclicos		Clara dependencia en la estructura, cambios en una capa C no afectan a capas no conectadas con C	: Diseño de Sistemas Operativos, Maquinas Virtuales y Protocolos de Red	Diseños con muchos niveles pueden ser ineficientes
Cliente-Servidor: Analisis	Cientes y Servidores	Llamadas remotas, protocolos de red	Parámetros de llamada y valores de retorno	Dos niveles, con multiples clientes conectados con uno o mas servidores	Los clientes no se pueden conectar entre si	Centraliza la computación y los datos en el servidor. Un servidor poderoso puede atender a varios clientes	Aplicaciones donde se necesita centralizar datos/procesos, o donde el computo requiere una maquina de alta capacidad y los clientes solo realizan tareas simples de interface	Cuando esta limitado el ancho de banda y hay mucho clientes
Pipes and Filters	Programas independientes (filtros)	Routers explícitos de streams de datos	Stream de datos	Pipeline (conexiones en T son posibles)•		Filtros mutuamente independientes. Estructura simple de streams de entrada/salida facilitan la combinación de componentes. Flexibilidad: Agregar, eliminar, cambiar y reusar filtros•	Aplicaciones sobre sistemas operativos. Procesamiento de audio, video	Cuando estructuras de datos complejos deben ser pasadas entre filtros. Cuando se requiere interacción entre filtros
Eventos	Generadores y/o consumidores de eventos independientes y concurrentes. Buses de Eventos (Podría existir	Conexiones con los Event-buses	Eventos	Los componentes se comunican con el event-bus, no directamente entre ellos.		Altamente escalable, Fácil de evolucionar, Efectivo para aplicaciones heterogéneas altamente distribuidas.	Software de UI y Aplicaciones de área amplia que involucran partes independientes (mercados financieros, logística, redes de sensado)	No existen garantías que un evento sea procesado, ni cuando lo será

	más de uno)							
Peer-to-Peer	Peers- componentes independient es que tienen su propio estado y control	Protocolos de red, generalment e propietarios.	Mensajes de red.	Red que puede tener conexione s redundant es entre peer. Puede variar arbitraria y dinámicam ente.		Altamente robusto de cara a la falla de un nodo. Escalable, en términos de acceso a los recursos y poder de cómputo. Computación distribuida	Cuando las fuentes de información y operación están distribuidas: File sharing, Mensajería instantánea, Gridcomputing, etc.	Cuando la recuperación de la información es critica en el tiempo y no puede hacer frente a la latencia propuesta por el protocola

Sistemas basados en capas

Resumen: Consiste de una secuencia ordenada de capas. Cada capa provee servicios que pueden ser accedidos por subprogramas o componentes de una capa superior.

Cliente – Servidor

Los clientes piden servicios a los servidores los cuales realizan las computaciones adecuadas y contestan con la información requerida. La comunicación se inicia por el cliente

Pipes and Filters

Programas separados y ejecutados, potencialmente de manera concurrente. Datos son pasados como un stream de un programa al siguiente.

Eventos

Resumen: Componentes independientes que asincrónicamente emiten y reciben eventos comunicados a través de event-buses.

Peer-to-Peer

Estado y comportamientos son distribuidos entre pares que pueden actuar tanto como clientes como servidores