

---

# Programación Distribuida y Tiempo Real

---

Estilos arquitecturales



# Arquitecturas...

- De hardware
  - Inicialmente independientes
  - Cada vez más asociadas al software
    - Infraestructura/“operaciones”
    - DevOps: infraestructura “a la par” (continuous...)
  - Virtualización ==> Con código (vagrant-docker...)
  - No va a ser el “centro” de PDyTR
    - DevOps - Continuous - ... ==> Ing. de soft.



# Arquitecturas...

- De software (van Steen - Tanenbaum, 3Ed.)

“The organization of distributed systems is mostly about the software components that constitute the system. These software architectures tell us how the various software components are to be organized and how they should interact.

...

For our discussion, the notion of an architectural style is important. Such a style is formulated in terms of components, the way that components are connected to each other, the data exchanged between components, and finally how these elements are jointly configured into a system. A component is a modular unit with well-defined required and provided interfaces that is replaceable within its environment... Replacing a component can be done only if its interfaces remain untouched.”



# Arquitecturas...

- De software (Van Steen - Tanenbaum, 3Ed.)

“The organization of distributed systems is mostly about the software components that constitute the system. These software architectures tell us how the various software components are to be organized and how they should interact.

...

For our discussion, the notion of an architectural style is important. Such a style is formulated in terms of components, the way that components are connected to each other, the data exchanged between components, and finally how these elements are jointly configured into a system. A component is a modular unit with well-defined required and provided interfaces that is replaceable within its environment... Replacing a component can be done only if its interfaces remain untouched.”



# ¿“Arquitecturas” o “Estilos Arquit.”?

- Consulta:

Dear profs. van Steen and Tanenbaum,  
I've obtained a digital (personalized) copy of your book  
Distributed Systems 3rd edition (2017)

...

From your book, page 5.6, section 2.1:  
...(lo que está en la diapositiva anterior)...

As far as I'm able to understand,  
"software architectures"  
and  
"architectural styles"  
would be exactly the same, right?



# ¿“Arquitecturas” o “Estilos Arquit.”?

- Respuesta prof. van Steen:

Hi Fernando,

Software architectures is the umbrella term; an architectural style guides the way toward a more specific software architecture.

Maarten



# ¿“Arquitecturas” o “Estilos Arquit.”?

- Respuesta prof. van Steen:

Hi Fernando,

Software architectures is the umbrella term; an architectural style guides the way toward a more specific software architecture.

Maarten

- Mi respuesta a la respuesta....:

Thank you. I understand your answer, but I do not understand that from the text... maybe I'm losing something, but so far it's good enough for me.



# Arquitecturas...

- De software (van Steen-Tanenbaum)

...the logical organization of a distributed system into software components, also referred to as its software architecture [Bass et al., 2003].

...

A component is a modular unit with well-defined required and provided interfaces that is replaceable within its environment [OMG, 2004b].

- “Estilos Arquitecturales” ...

the notion of an architectural style is important. Such a style is formulated in terms of components, the way that components are connected to each other, the data exchanged between components, and finally how these elements are jointly configured into a system.





---

# Respetando a los Autores...

## Estilos arquitecturales



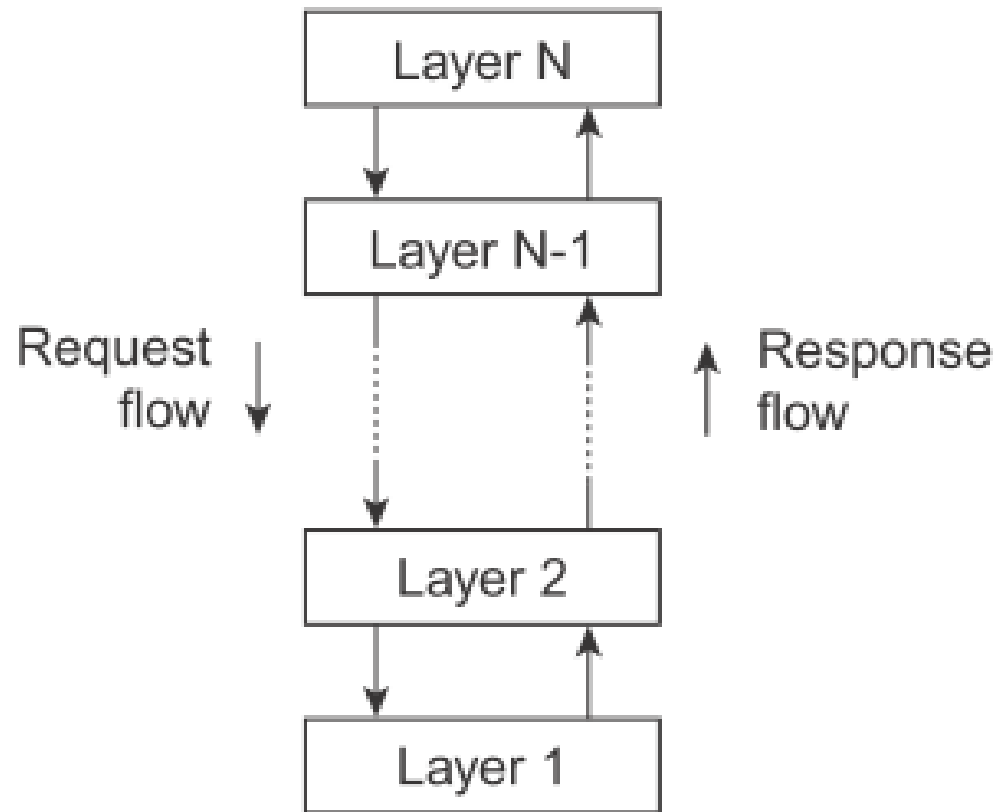
# Estilos Arquitecturales

- Los más importante para Sistemas Distribuidos
  - Layered architectures
  - Object-based architectures
  - Resource-centered architectures
  - Event-based architectures
- Cada uno de ellos afectará
  - La programación
  - Los aspectos de tiempo real involucrados (Si los hubiera)...



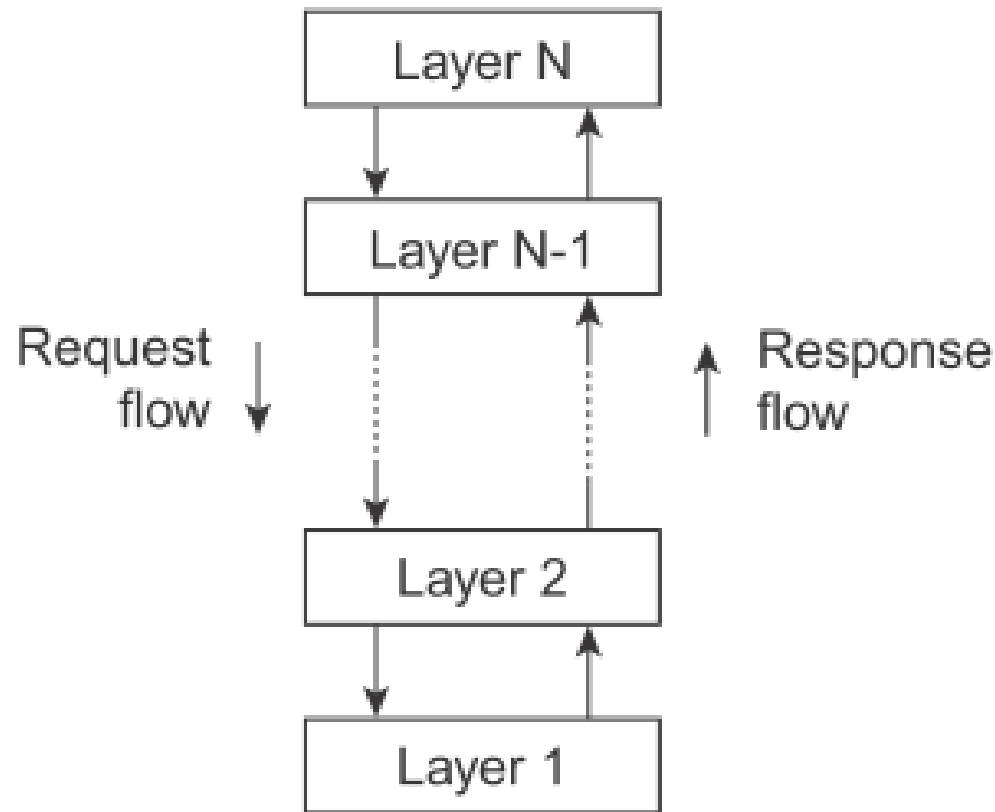
# Estilos Arquitecturales

- Layered architectures (capas)



# Estilos Arquitecturales

- Layered architectures (capas)



Componentes: capas

Interfaces: c/s

Organización: c/s, quizás  
“niveles de abstracción”

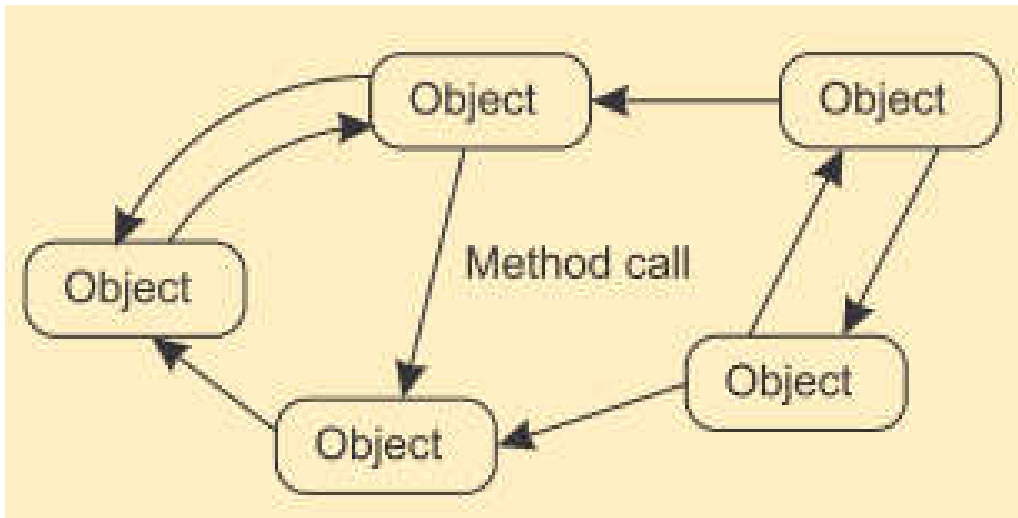
Comunicaciones:

- requerimientos
- respuestas



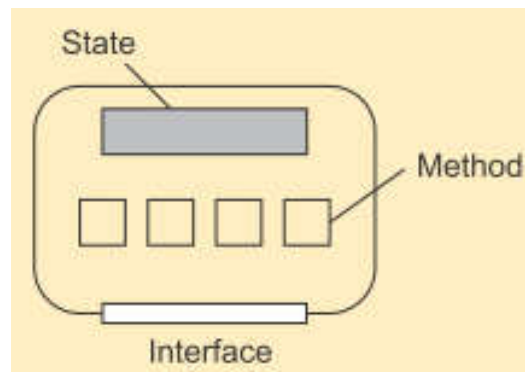
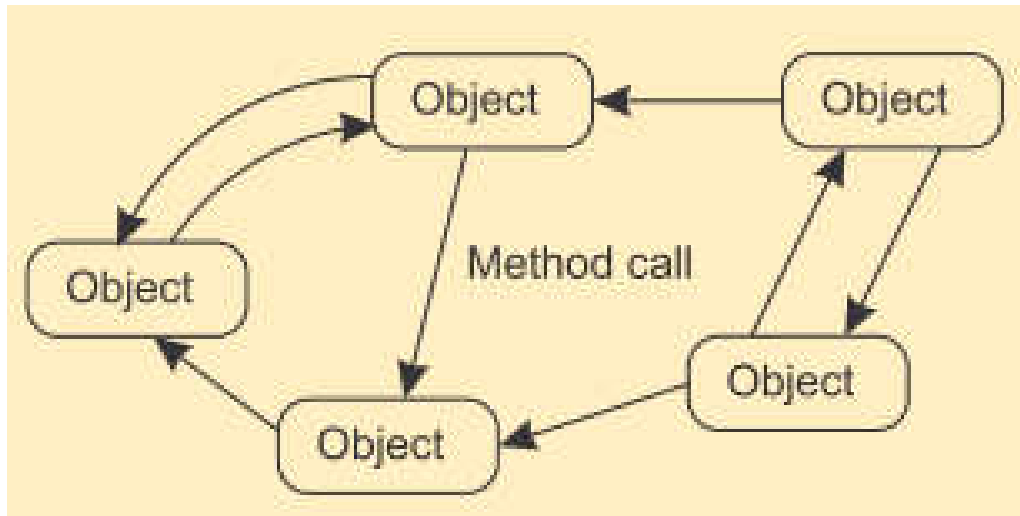
# Estilos Arquitecturales

- Object-based (objetos distribuidos)



# Estilos Arquitecturales

- Object-based (objetos distribuidos)



Componentes: objetos

Interfaces: métodos

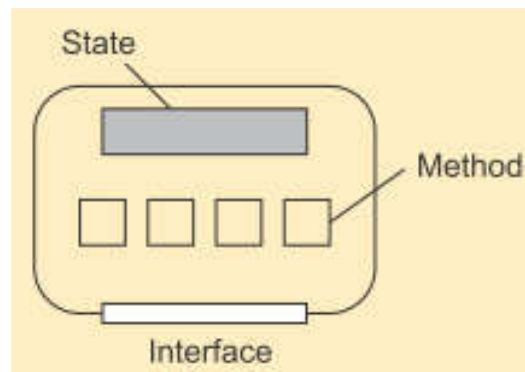
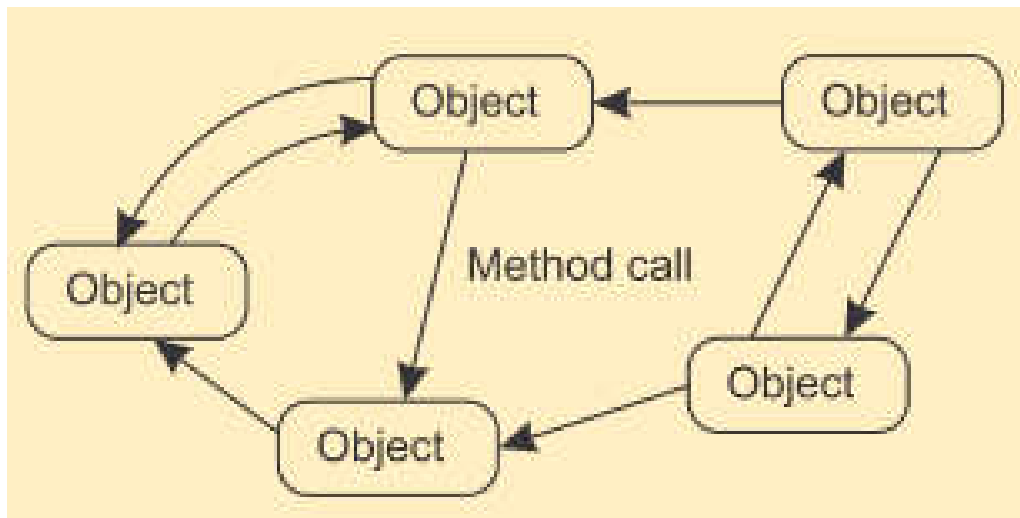
Organización: distribuir  
objetos en el sistema

Comunicaciones:  
métodos



# Estilos Arquitecturales

- Object-based (objetos distribuidos)



Componentes: objetos

Interfaces: métodos

Organización: distribuir objetos en el sistema

Comunicaciones: métodos

Detalles en la práctica, pero no es “trivial” distribuir un ambiente de objetos



# Estilos Arquitecturales

- Resource-centered architectures (acceso a recursos)

## RESTful architectures

- 1 Resources are identified through a single naming scheme
- 2 All services offer the same interface
- 3 Messages sent to or from a service are fully self-described
- 4 After executing an operation at a service, that component forgets everything about the caller

Componentes: manejadores (¿”servidores”?) de recursos

Interfaces: servicios, los mismos para todos, sin estados

Organización: distribuir manejadores/recursos en el sistema

Comunicaciones: interfaces, servicios





# Estilos Arquitecturales

- Resource-centered architectures (acceso a recursos)

## RESTful architectures

- 1 Resources are identified through a single naming scheme
- 2 All services offer the same interface
- 3 Messages sent to or from a service are fully self-described
- 4 After executing an operation at a service, that component forgets everything about the caller

Component

Interfaces: s

Organizació

Comunicació

Operation	Description
PUT	Create a new resource
GET	Retrieve the state of a resource in some representation
DELETE	Delete a resource
POST	Modify a resource by transferring a new state

Figure 2.7: The four operations available in RESTful architectures.



---

# Estilos Arquitecturales

- Publish-Subscribe (publicación-suscripción de eventos)
- Enfocado directamente en eventos



# Estilos Arquitecturales

- Publish-Subscribe (publicación-suscripción de eventos)
- Enfocado directamente en eventos
  - Un sistema no cambia a menos que suceda algo
    - Evento
  - Se deberían identificar los eventos
    - Cómo se genera
    - Qué “significa”
    - Qué datos involucra
    - I/O, propios, etc.



# Estilos Arquitecturales

- Publish-Subscribe (publicación-suscripción de eventos)
- Enfocado directamente en eventos
  - Un sistema no cambia a menos que suceda algo
    - Evento
  - Se deberían identificar los eventos
    - Cómo se genera
    - Qué “significa”
    - Qué datos involucra
    - I/O, propios, etc.

} Heterogeneidad



# Estilos Arquitecturales

- Publish-Subscribe (publicación-suscripción de eventos)
- Enfocado directamente en eventos
  - Un sistema no cambia a menos que suceda algo
    - Evento
  - Se deberían identificar los eventos
  - Componentes del sistema que *generan* eventos
    - Publicadores
  - Componentes del sistema que *manejan* eventos
    - Suscriptores
  - Todo el sistema se construye a partir de esos dos tipos de componentes



---

# Estilos Arquitecturales

- Publish-Subscribe (publicación-suscripción de eventos)
- Enfocado directamente en eventos
- Como mínimo
  - Desacopla el sistema en espacio (SD)
  - Orienta la identificación de requerimientos



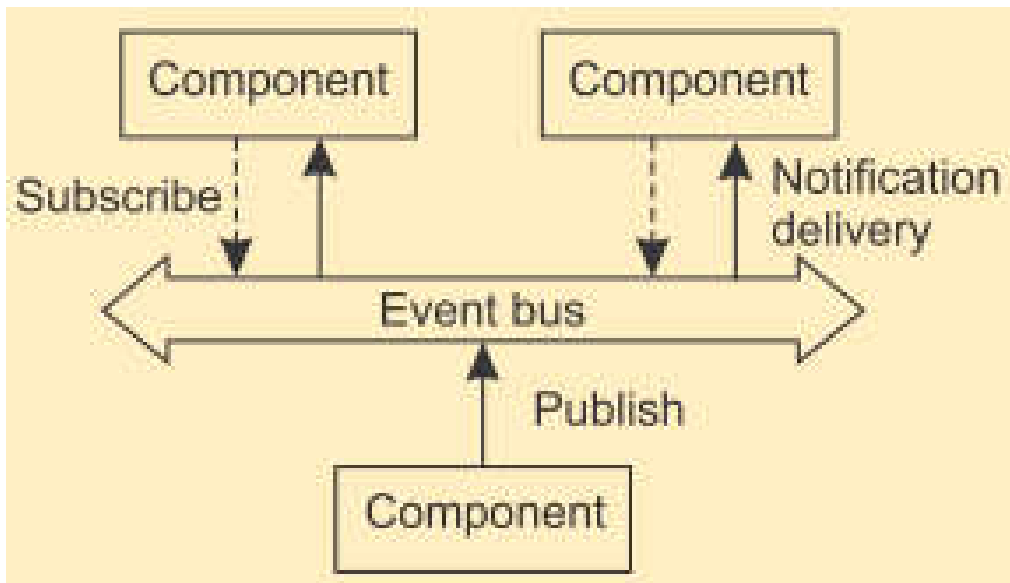
# Estilos Arquitecturales

- Publish-Subscribe (publicación-suscripción de eventos)
- Enfocado directamente en eventos
- Como mínimo
  - Desacopla el sistema en espacio (SD)
  - Orienta la identificación de requerimientos
- Evento - Nivel de Abstracción



# Estilos Arquitecturales

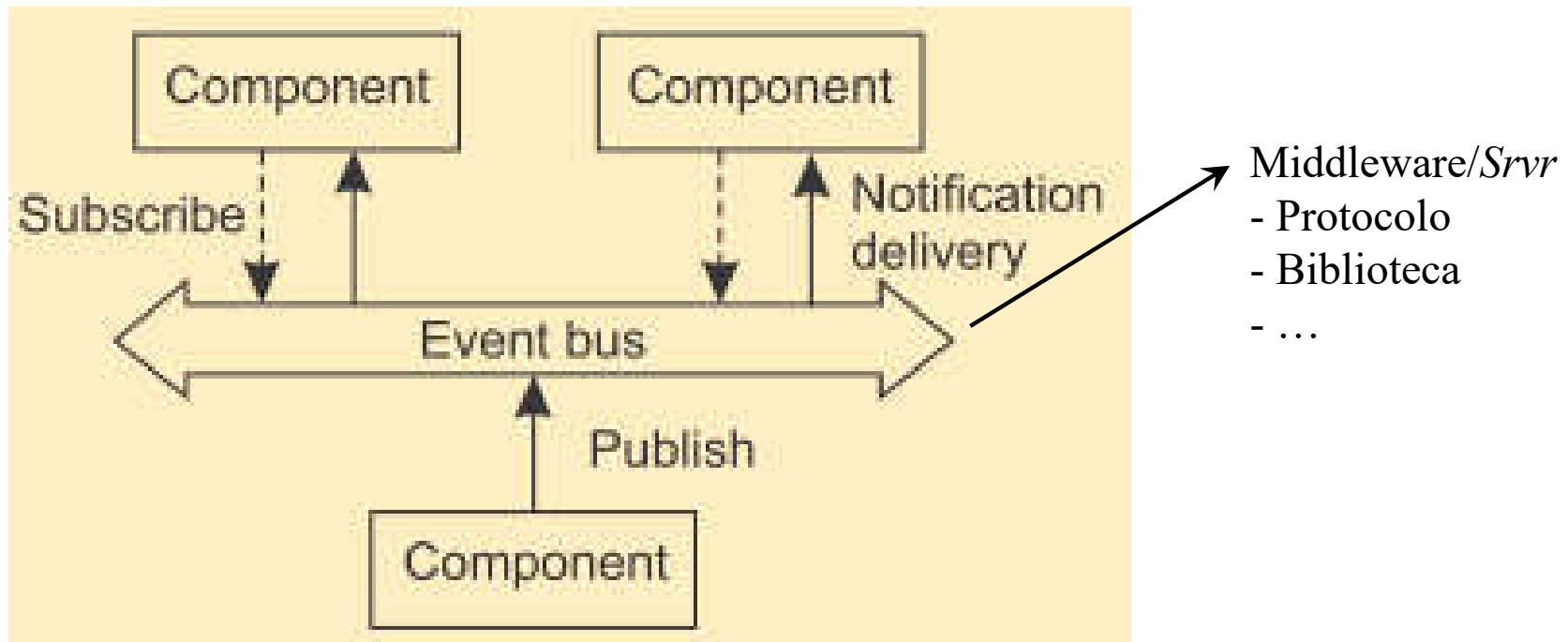
- Publish-Subscribe (publicación-suscripción de eventos)





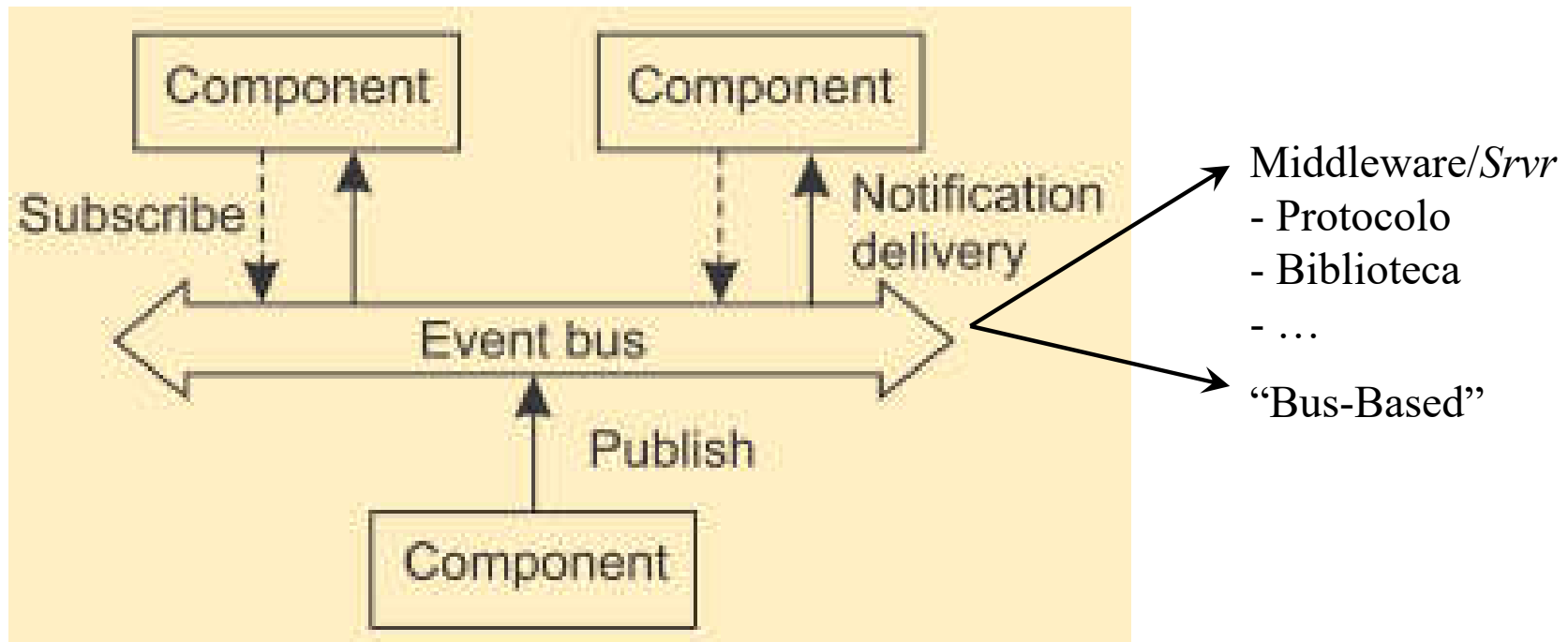
# Publicación/Suscripción

- Esquemáticamente



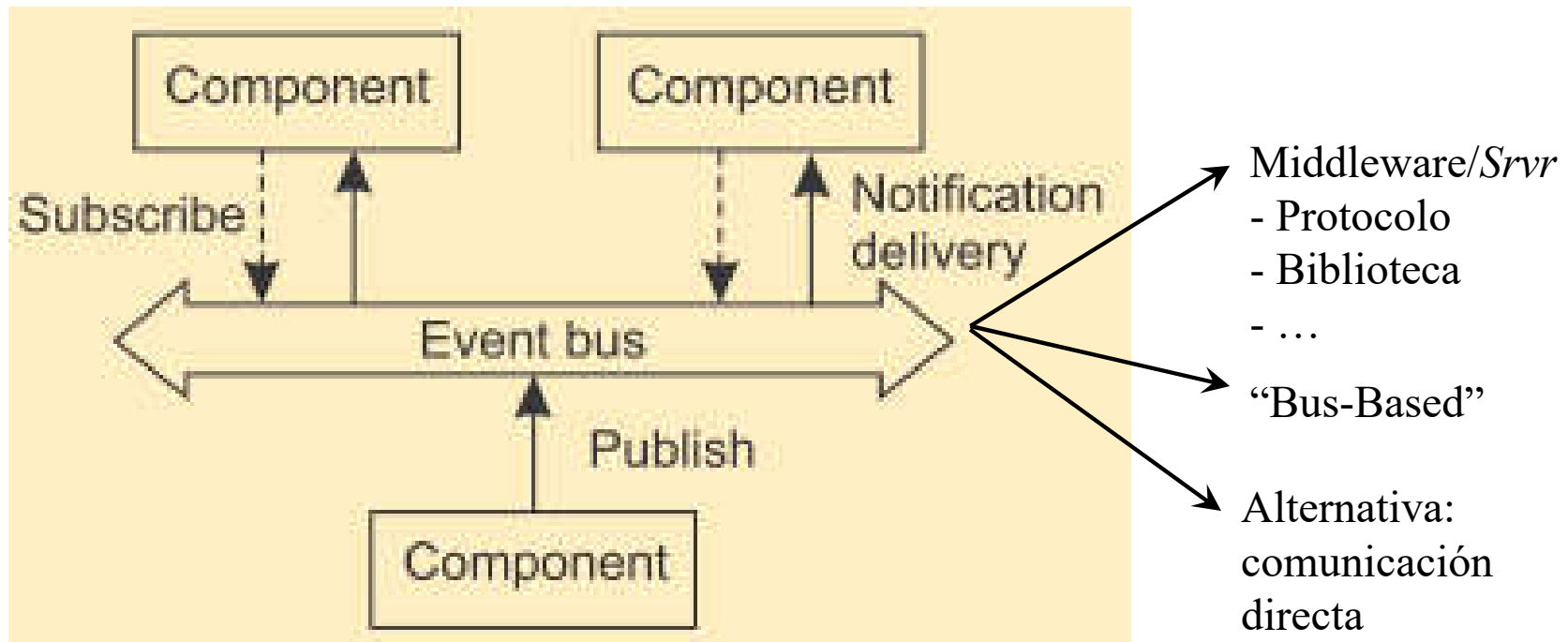
# Publicación/Suscripción

- Esquemáticamente



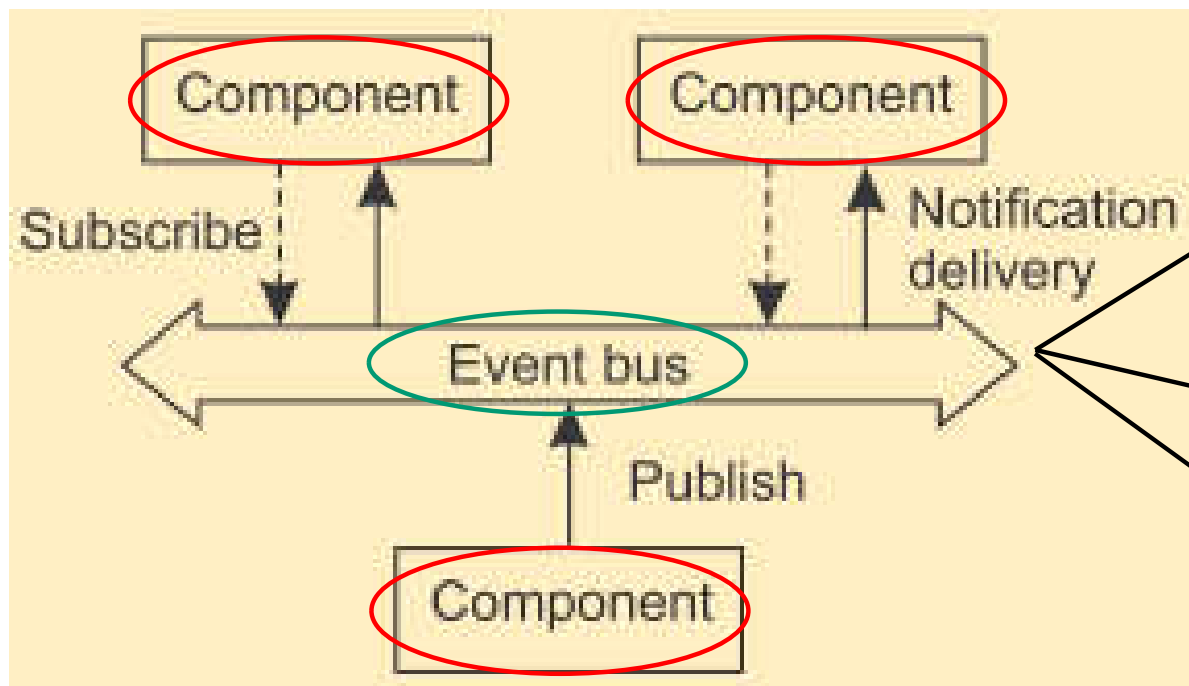
# Publicación/Suscripción

- Esquemáticamente



# Publicación/Suscripción

- Esquemáticamente



Middleware/*Srvr*  
- Protocolo  
- Biblioteca  
- ...

“Bus-Based”

Alternativa:  
comunicación directa  
(vs. “Invocación  
Implícita”)

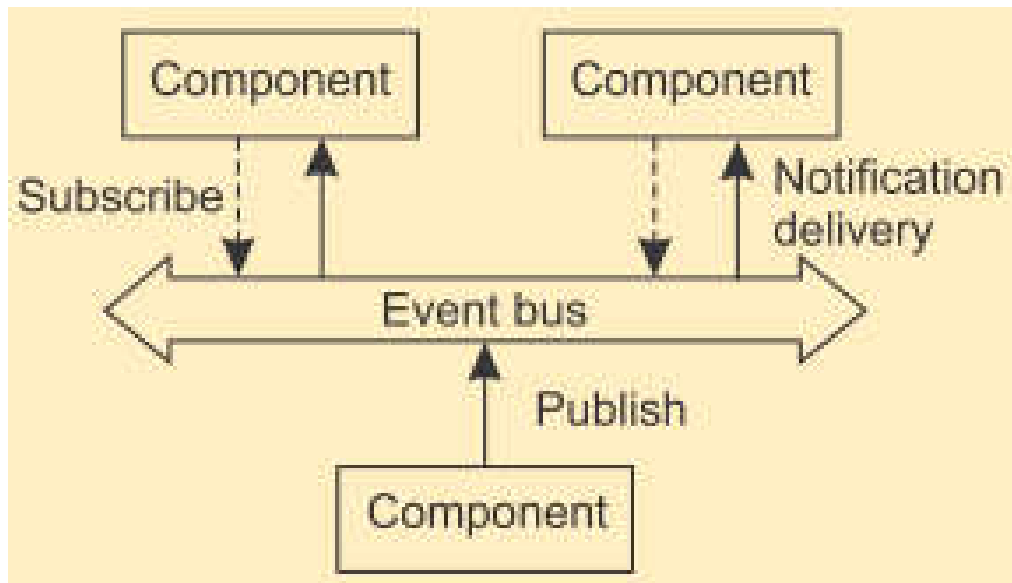
  Pueden ser distribuidos en diferentes sitios o pueden estar en un mismo sitio

  Comunicación con todos los componentes



# Estilos Arquitecturales

- Publish-Subscribe (publicación-suscripción de eventos)



Componentes: procesos o módulos o hilos

Interfaces: publicación y suscripción de eventos

Organización: distribuir procesos + p-s

Comunicaciones:  
definidas por el  
middleware para/de p-s



# Publicación/Suscripción

- Partes de la Arquitectura
  - Componentes que publican eventos
  - Componentes que se suscriben a eventos
  - Interfaz: definición y comunicación de eventos
    - *Unidireccional* (publica: *envía*, suscribe: *recibe*)
    - *Comunicación Asíncrona*



# Publicación/Suscripción

- Partes de la Arquitectura
  - Componentes que publican eventos
  - Componentes que se suscriben a eventos
  - Interfaz: definición y comunicación de eventos
    - *Unidireccional* (publica: *envía*, suscribe: *recibe*)
    - Comunicación *Asincrónica*
  - Interacciones: cada componente con el middleware o servicio de transporte de eventos, no entre sí
    - Esto es lo que *desacopla* en espacio



# Publicación/Suscripción

- Partes de la Arquitectura
  - Componentes que publican eventos
  - Componentes que se suscriben a eventos
  - Interfaz: definición y comunicación de eventos
    - *Unidireccional* (publica: *envía*, suscribe: *recibe*)
    - Comunicación *Asincrónica*
  - Interacciones: cada componente con el middleware o servicio de transporte de eventos, no entre sí
    - Esto es lo que *desacopla* en espacio
  - Acoplados en *tiempo*





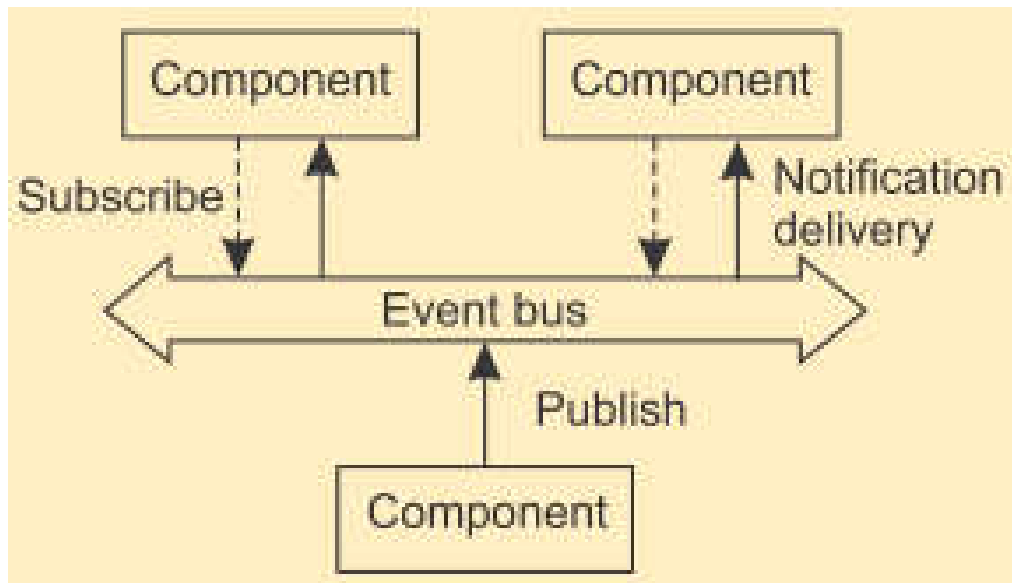
# Publicación/Suscripción

- Partes de la Arquitectura:
  - Publicadores – Procesos – Sitios
    - Publicador: 1 evento o varios
    - Proceso: 1 publicador o varios
    - Sitio (en el caso de SD): 1 proceso o varios
  - Suscriptores – Procesos – Sitios
    - Idem anterior
  - Combinaciones...



# Estilos Arquitecturales

- Publish-Subscribe (publicación-suscripción de eventos)



Desacoplado en el espacio

Acoplado en el tiempo

Componentes: procesos o módulos o hilos

Interfaces: publicación y suscripción de eventos

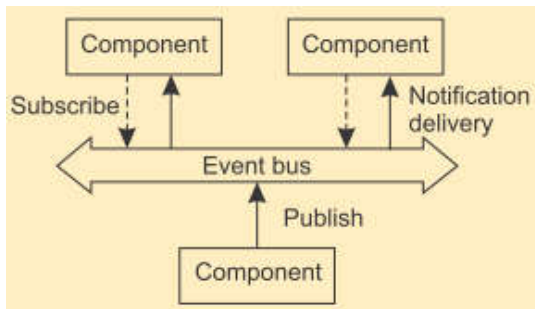
Organización: distribuir procesos + p-s

Comunicaciones: definidas por el middleware para/de p-s



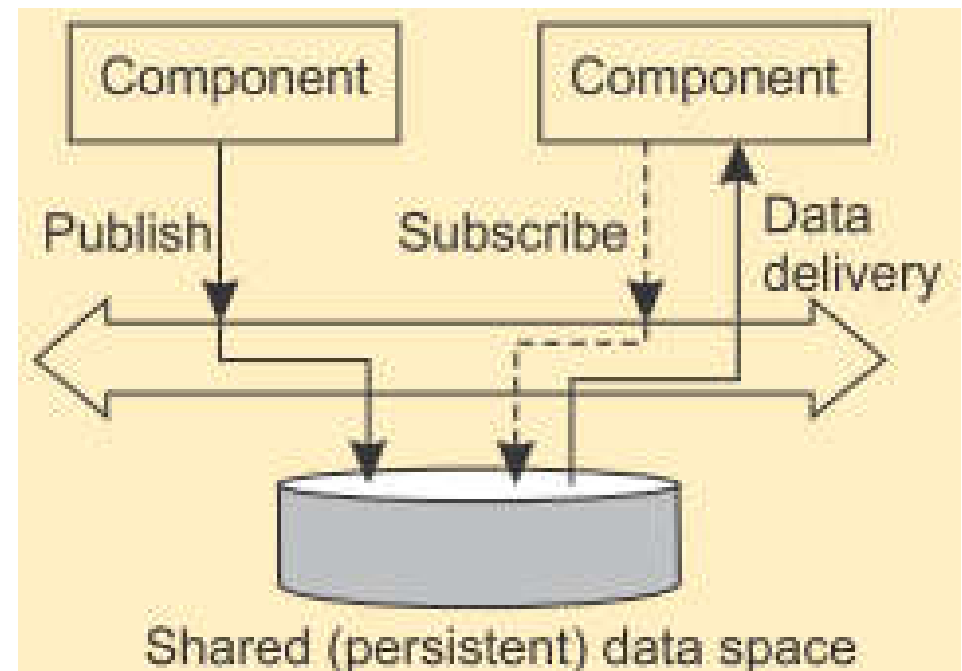
# Estilos Arquitecturales

- Publish-Subscribe (publicación-suscripción de eventos)



Desacoplado en el espacio

Acoplado en el tiempo



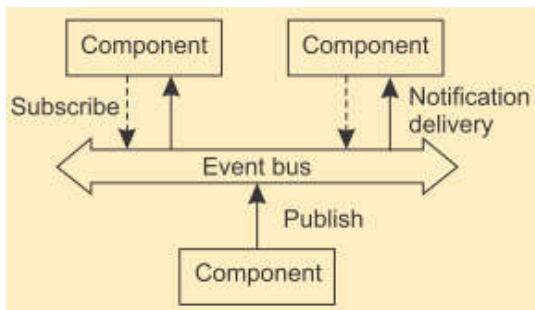
Desacoplado en el espacio

Desacoplado en el tiempo



# Estilos Arquitecturales

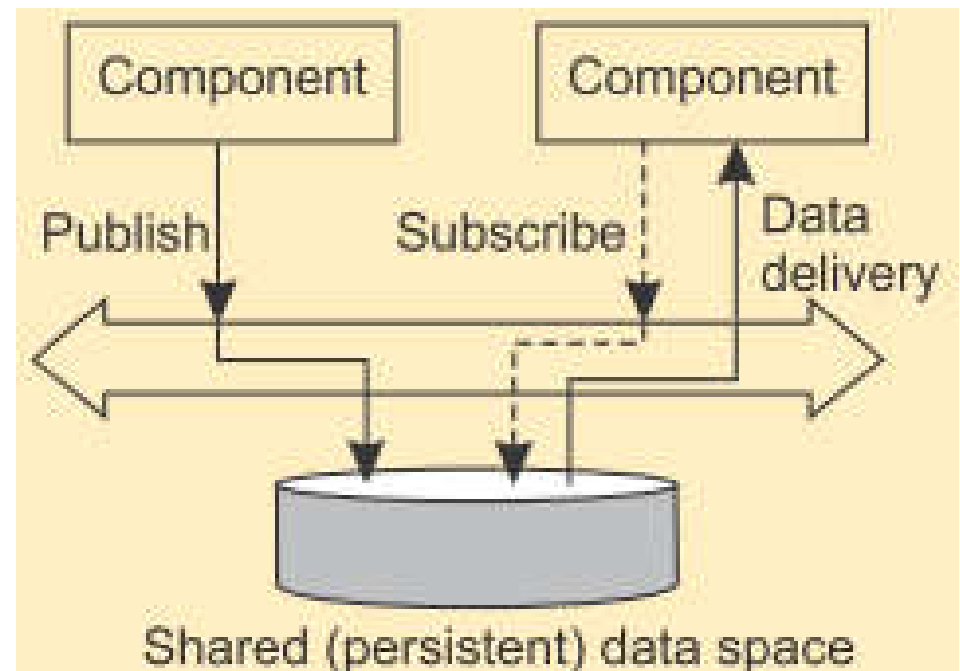
- Publish-Subscribe (publicación-suscripción de eventos)



Desacoplado en el espacio

Acoplado en el tiempo

Recursos y tolerancia a fallas



Desacoplado en el espacio

Desacoplado en el tiempo



# Estilos Arquitecturales

- Publish-Subscribe (publicación-suscripción de eventos)
  - Todo el sistema “en función de” eventos
  - Publicación
  - Suscripción
  - Ambas funciones
- Opciones
  - Desacoplado en espacio y acoplado en tiempo
  - Desacoplado en tiempo y espacio (necesariamente con persistencia de eventos)



# Dudas/Consultas

- Plataforma Ideas

