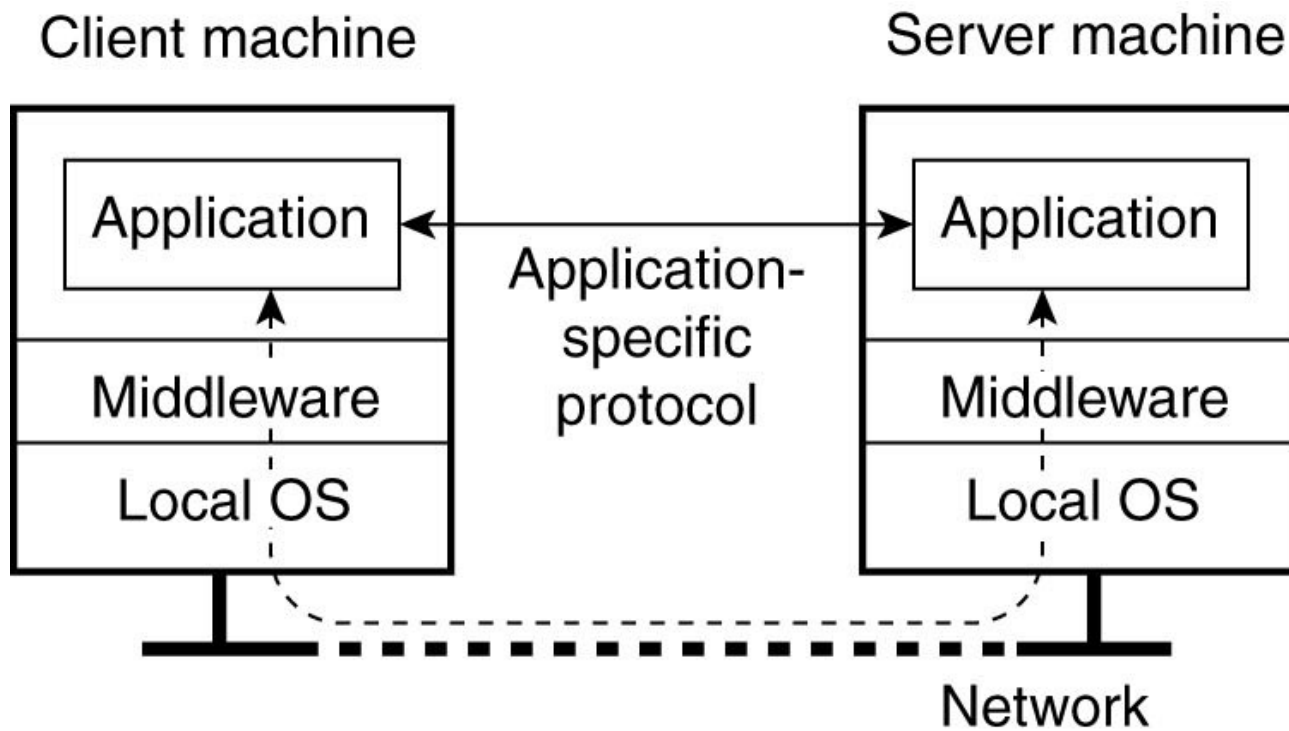


Sistemas Distribuidos

Jueves, 12 de enero

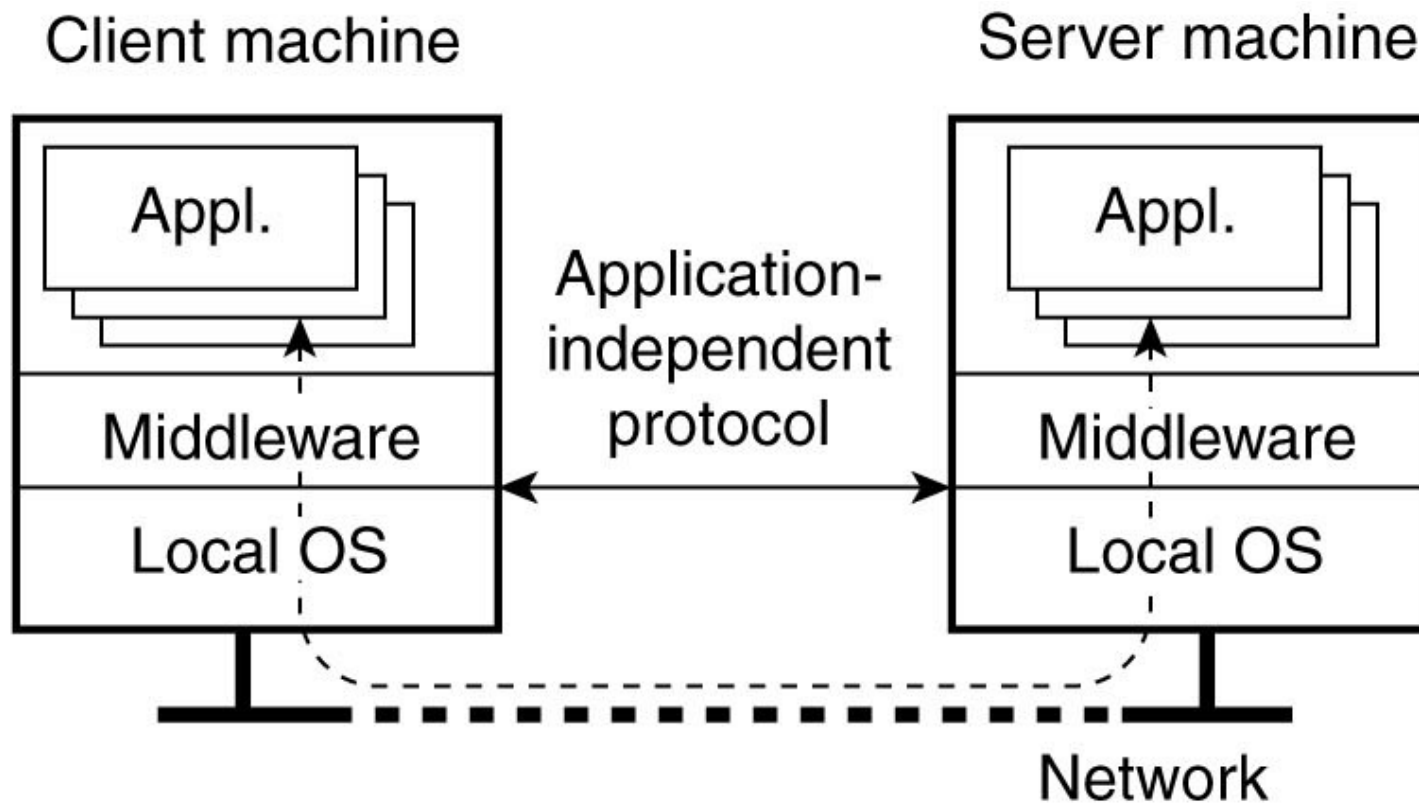
- Migración de código
- Comunicación entre procesos

Interfaces de usuario en red 1



(a)

Interfaces de usuario en red 2

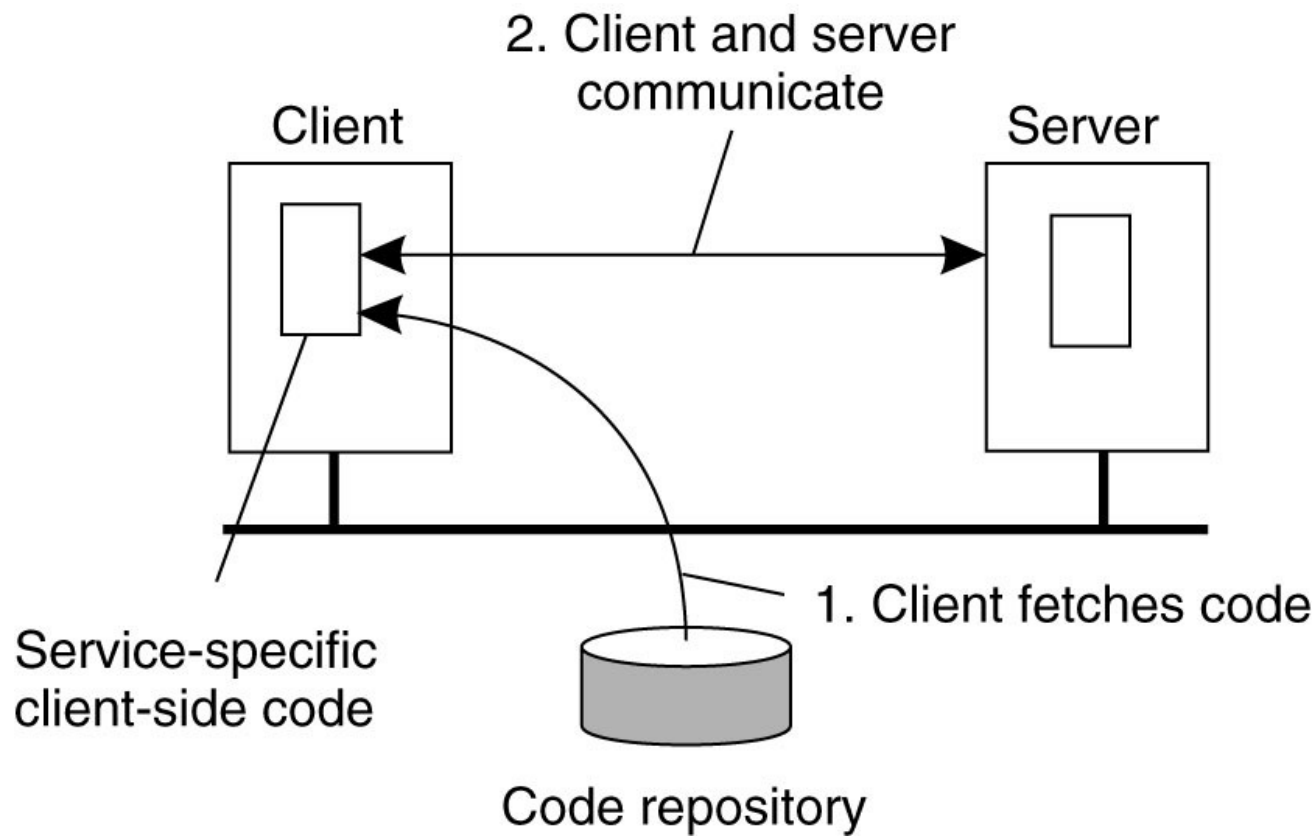


(b)

Migración de código

- Justificación
- Antecedentes
 - Migración de procesos
 - De máquinas cargadas a otras más desocupadas
- Carga de trabajo
 - Tamaño de la cola en CPU
 - Utilización de CPU
- Sin embargo
 - Minimizar comunicaciones podría ser más importante
- Agentes móviles

Razones para migrar código



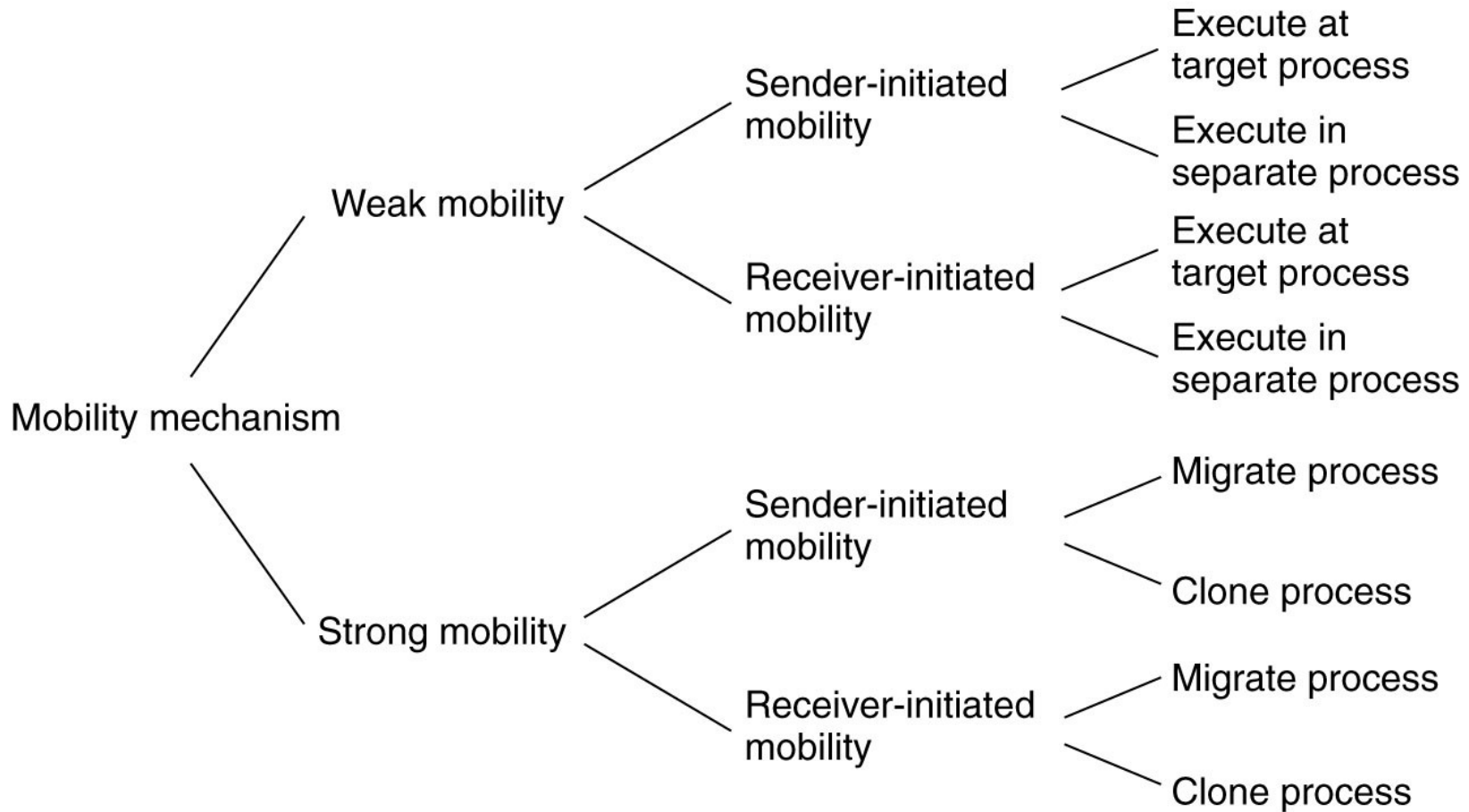
Razones para migrar código

- Rendimiento
- Flexibilidad
- Enfoque tradicional de los SD
 - Particionar tareas
- Ventaja
 - Clientes no tienen que tener todas las aplicaciones preinstaladas para interactuar con el servidor
- Principal desventaja
 - Problemas de seguridad

Procesos

- Migración
 - Programas para instalar
 - Programas en ejecución
- Segmento de código
- Segmento de recursos
- Segmento de ejecución

Modelos



Migración y recursos locales

Resource-to-machine binding

	Unattached	Fastened	Fixed	
Process-to-resource binding	By identifier	MV (or GR)	GR (or MV)	GR
	By value	CP (or MV,GR)	GR (or CP)	GR
	By type	RB (or MV,CP)	RB (or GR,CP)	RB (or GR)

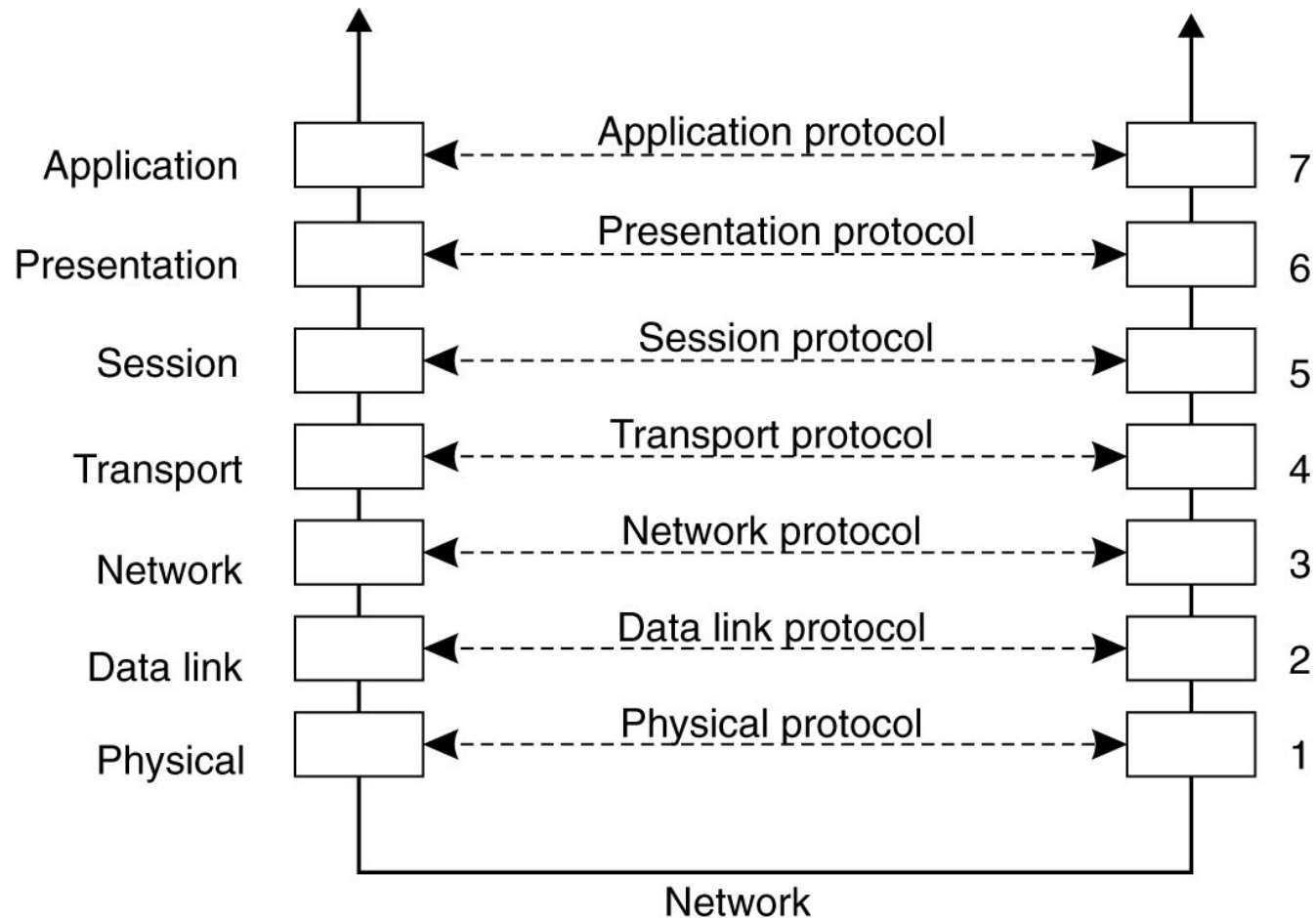
GR Establish a global systemwide reference

MV Move the resource

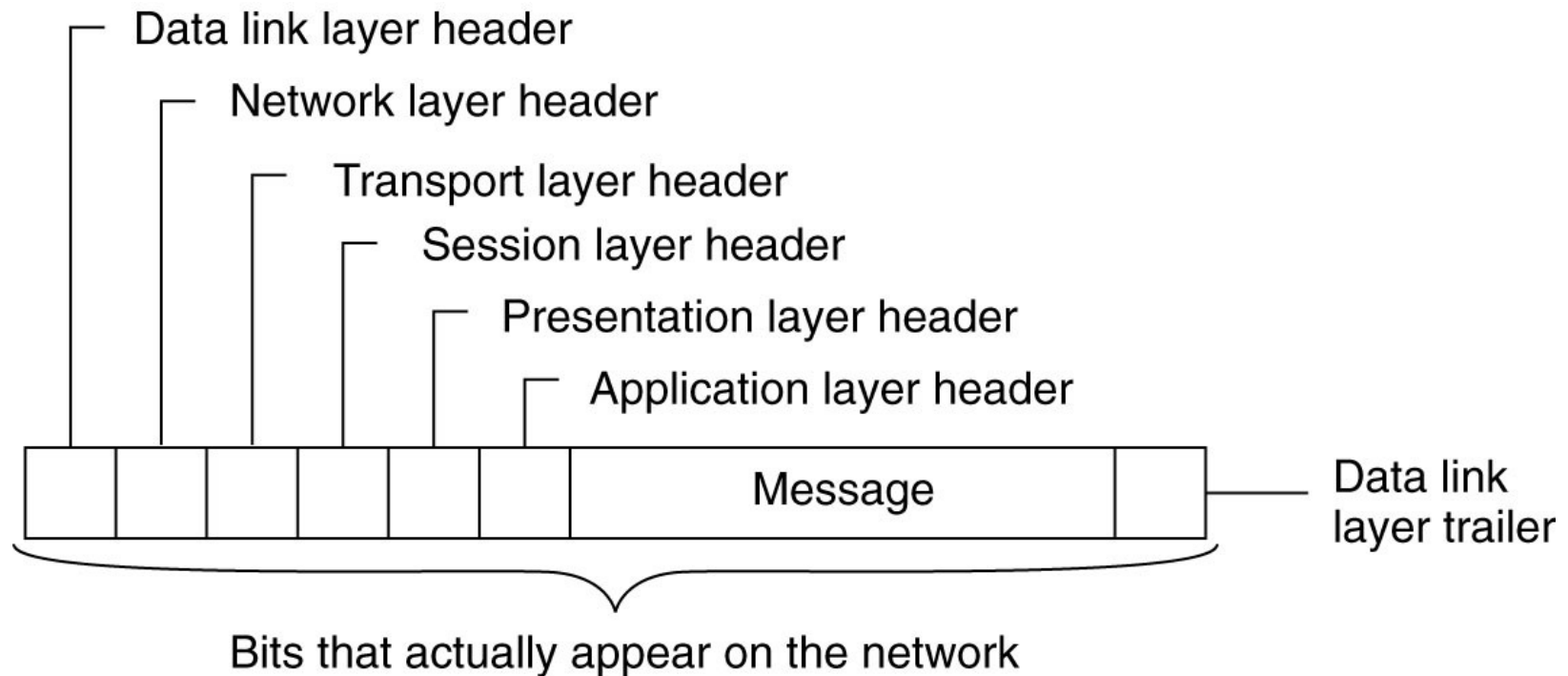
CP Copy the value of the resource

RB Rebind process to locally-available resource

Comunicación



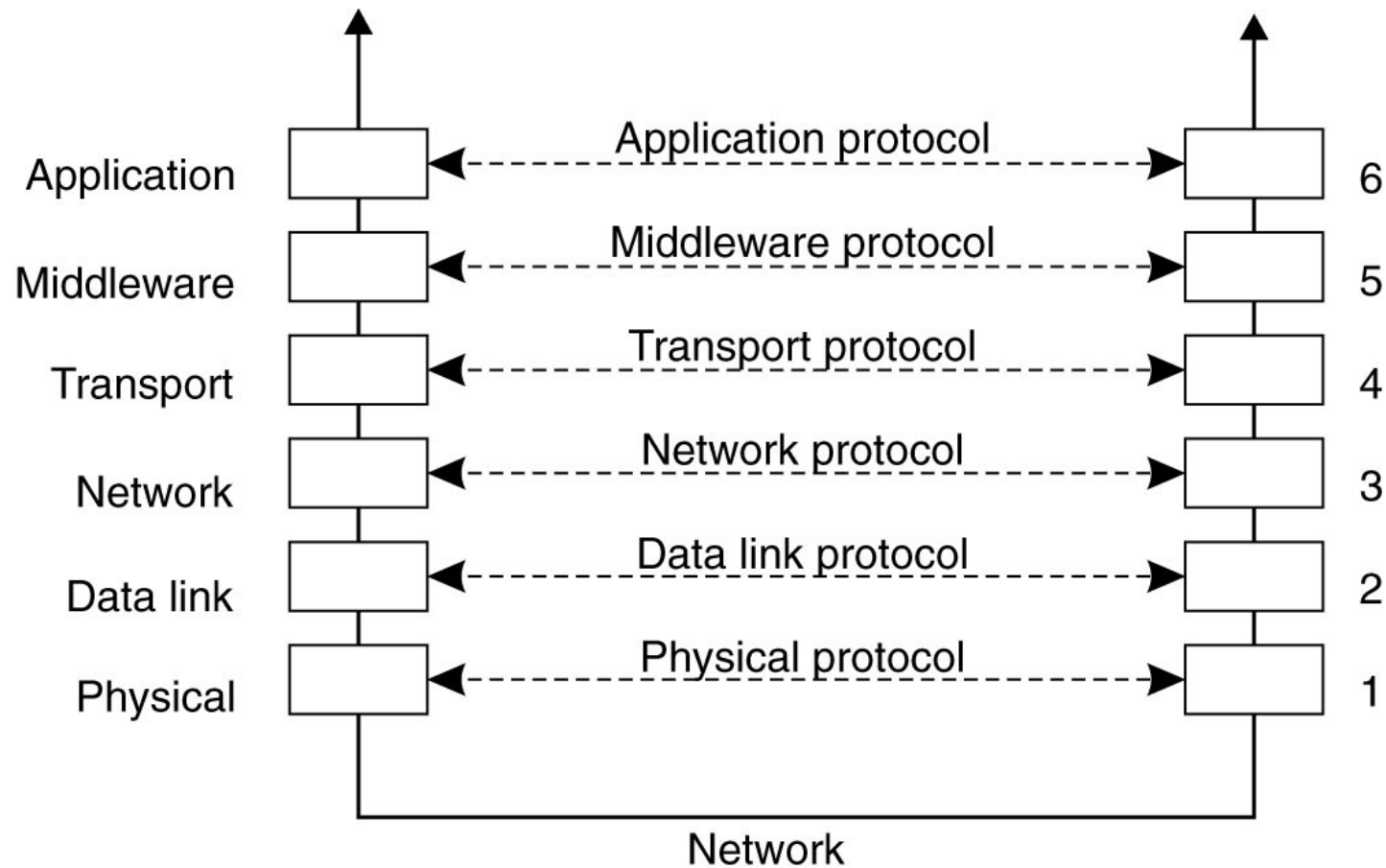
Comunicación



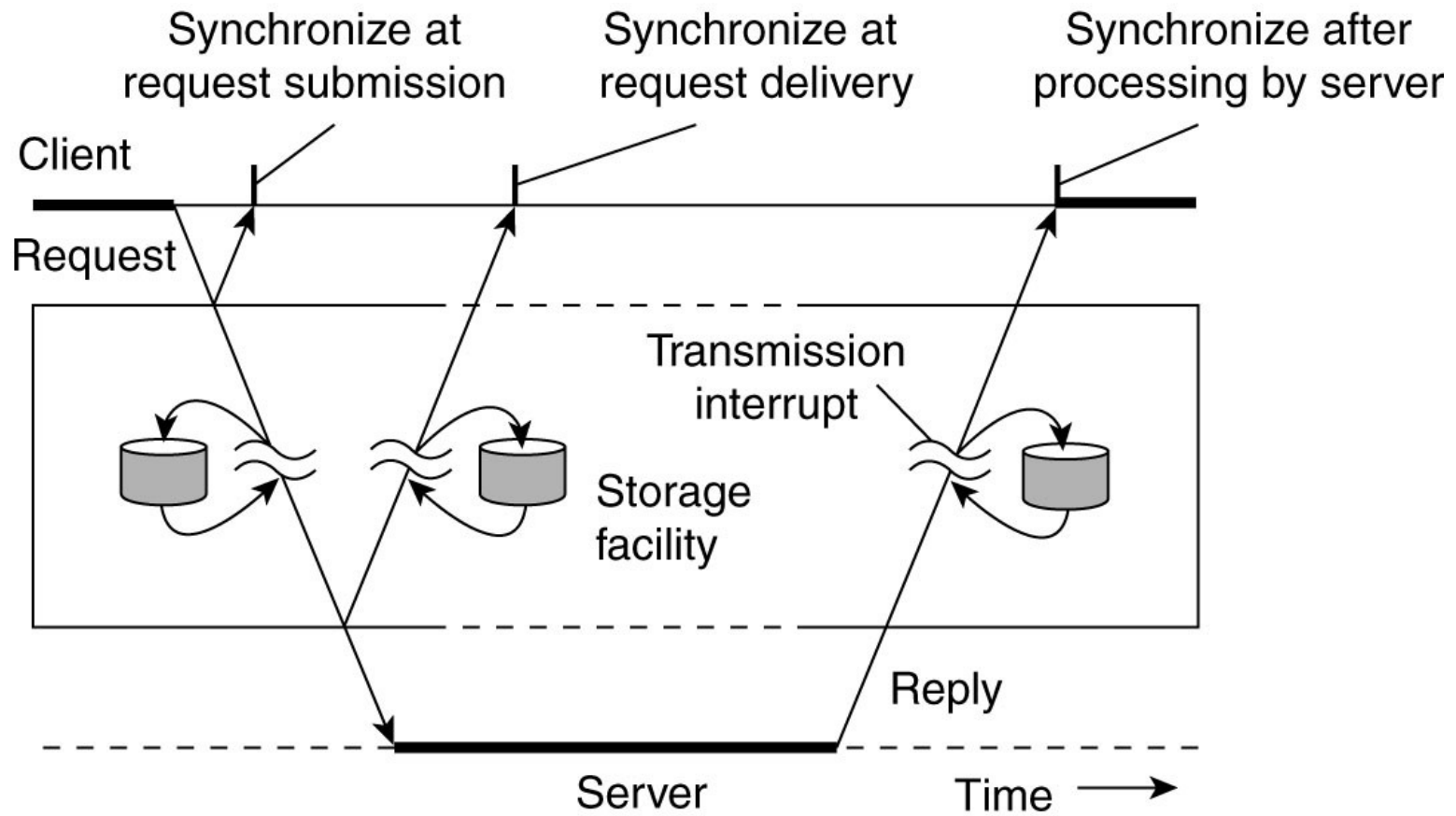
Protocolos de middleware

- Autenticación
 - Usuarios
- Autorización
 - Permisos
- Commit
 - Atomicidad en transacciones
- Bloqueo
 - Protección contra accesos simultáneos

Protocolos de middleware



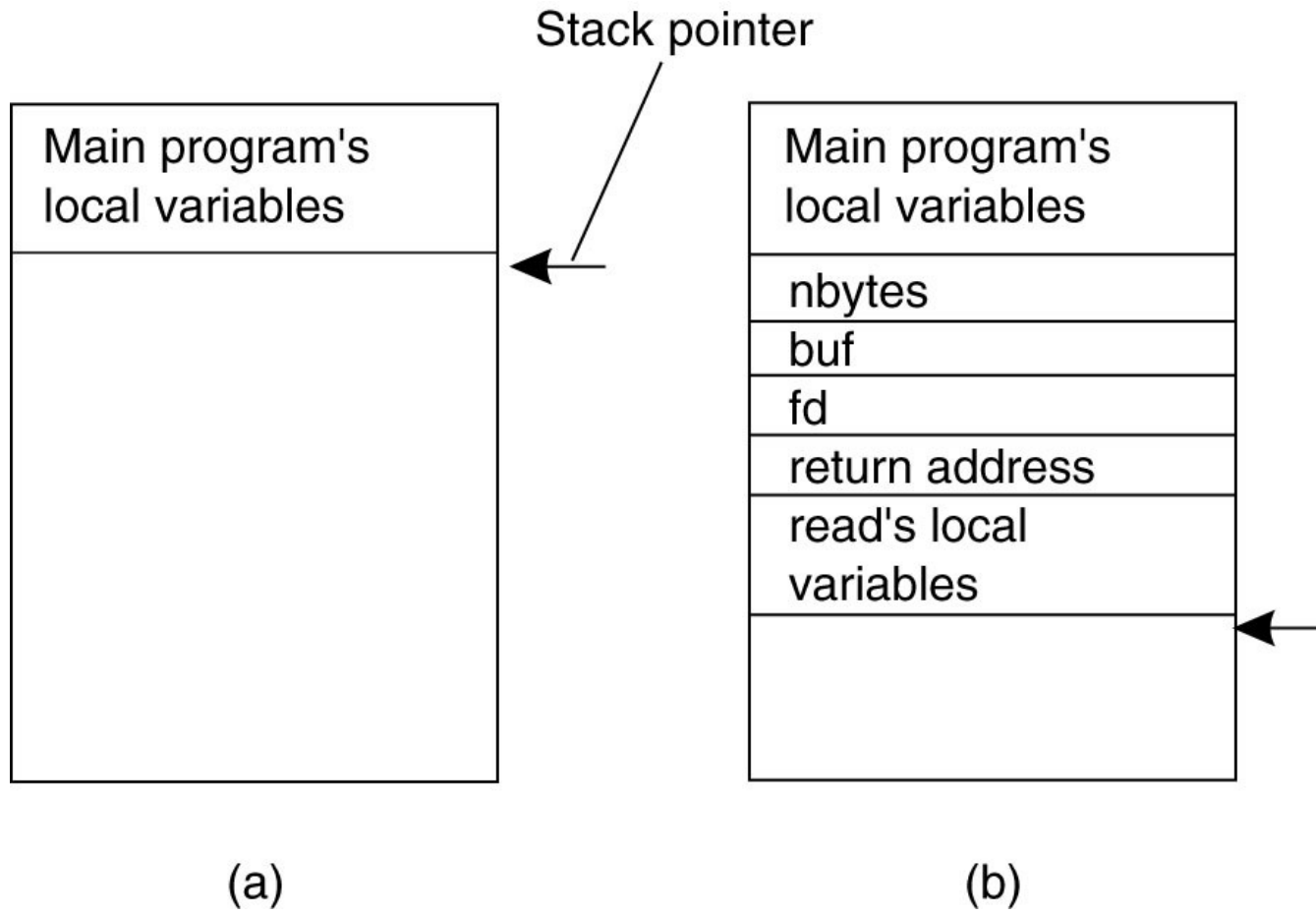
Tipos de comunicación



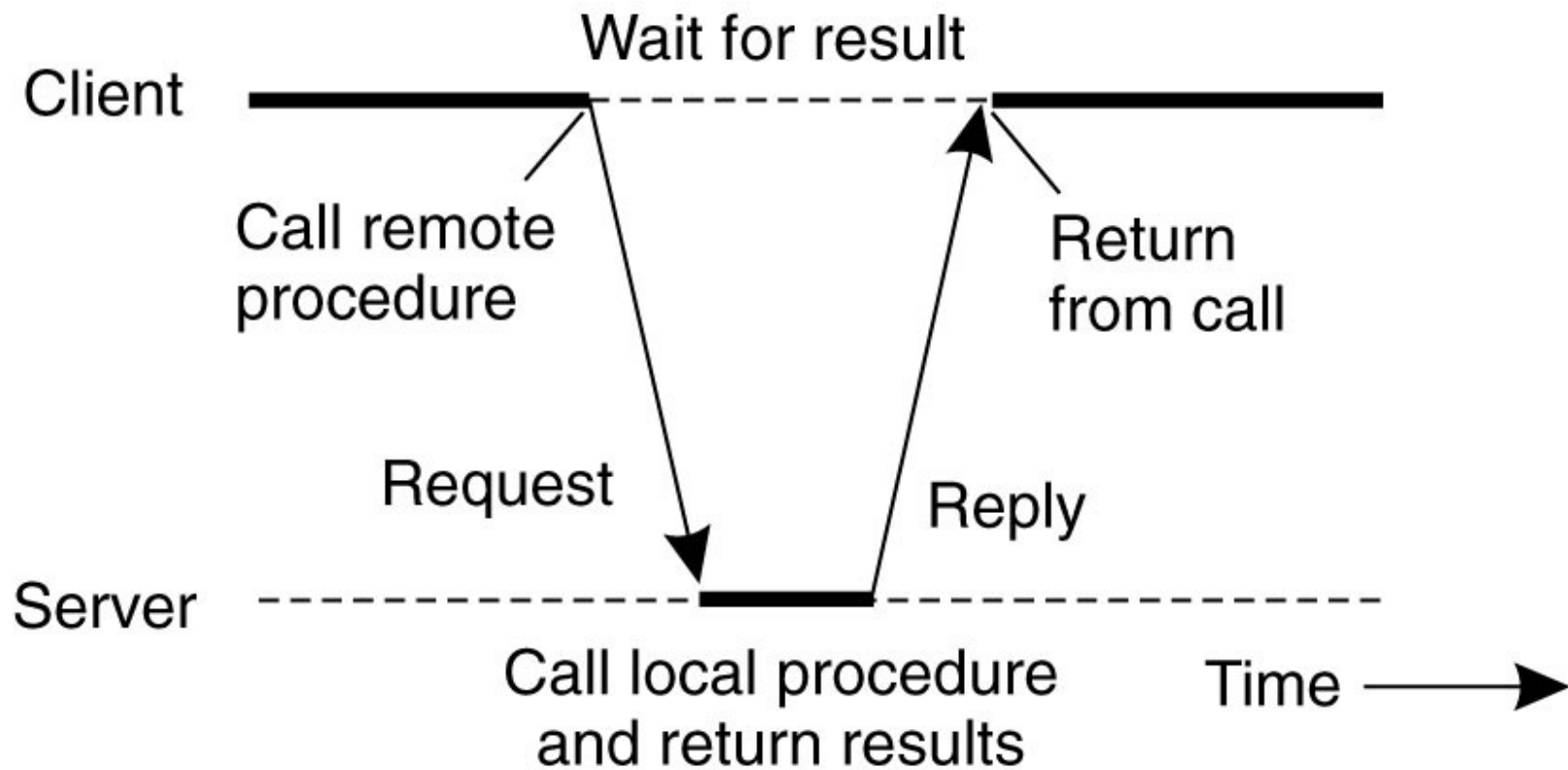
Tipos de comunicación

- Persistente o transitoria
- Sincrónica o asincrónica
- Discreta o continua (streaming)

Procedure Call convencional



Stub Cliente-Servidor



Remote Procedure Calls 1

1. El procedimiento del cliente llama al stub del cliente de forma normal.
2. El stub del cliente construye un mensaje y llama al SO local.
3. El SO del cliente envía el mensaje al SO remoto.
4. El SO remoto le pasa el mensaje al stub del servidor.
5. El stub del servidor desempaca los parámetros y llama al servidor.

Remote Procedure Calls 2

6. El servidor realiza el trabajo y regresa el resultado al stub.
7. El stub del servidor lo empaca en un mensaje y llama al SO local.
8. El SO del servidor envía el mensaje al SO del cliente.
9. El SO del cliente entrega el mensaje al stub del cliente.
10. El stub del cliente desempaca el resultado y regresa al cliente.