

Comunicación

Capítulo 4. Tanenbaum

Contenido

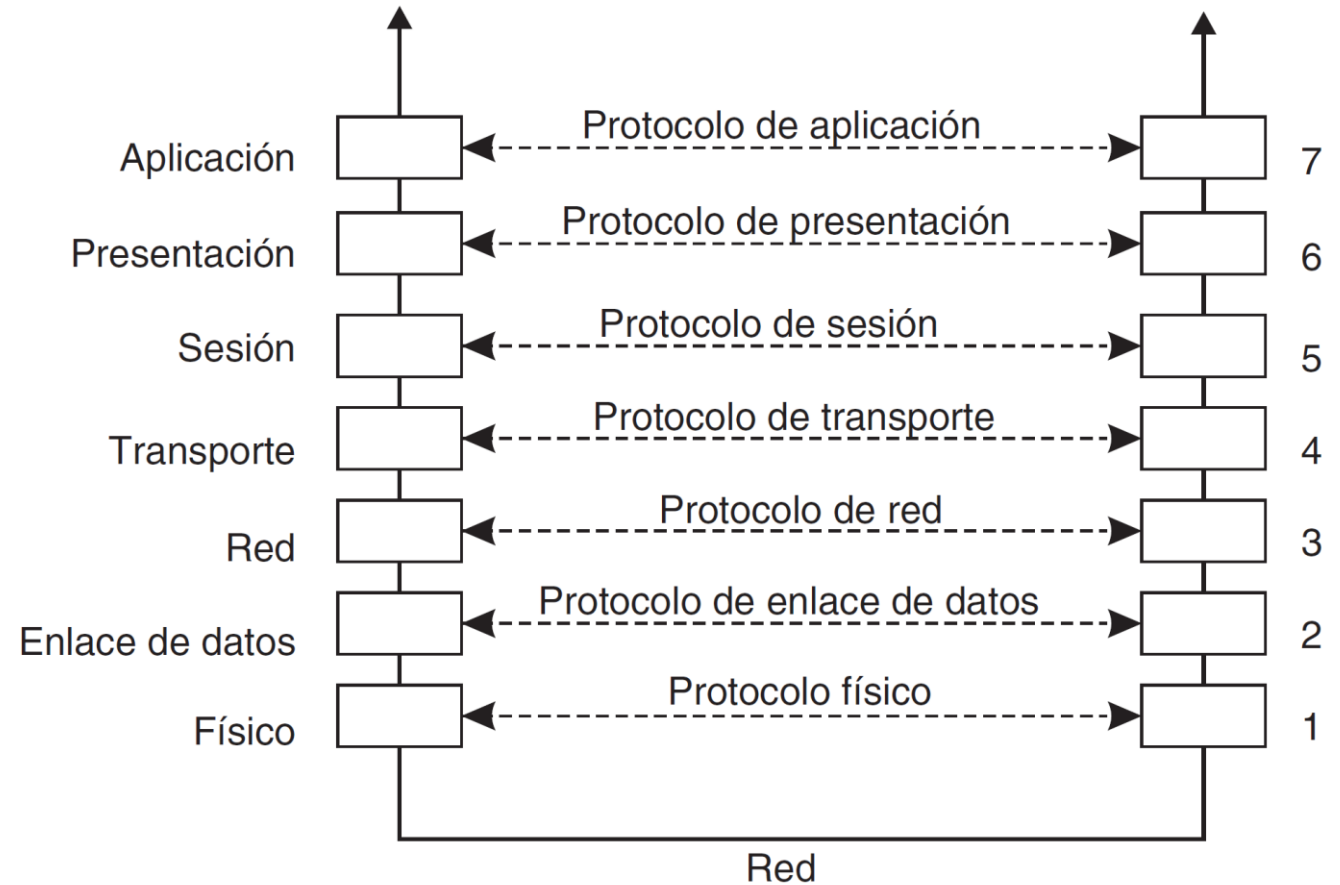
- Protocolos en capa
- Tipos de comunicación
- Llamada remota

Comunicación entre procesos

- Corazón de todo sistema distribuido
- ¿Cómo procesos en diferentes máquinas intercambian información?
 - No es una tarea fácil
- El objetivo es proveer transparencia de esta comunicación al desarrollador también

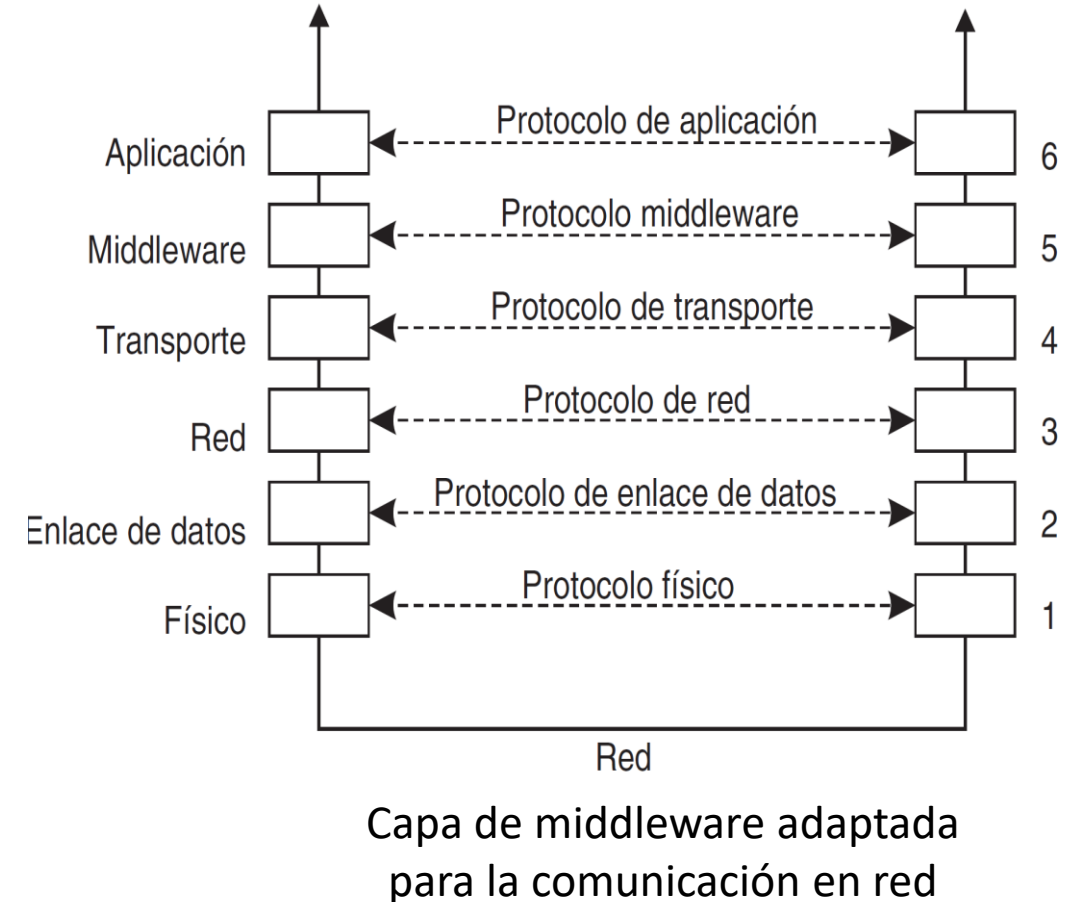
Protocolos en capas

- Protocolos de comunicación de redes son la base de cualquier SD.
- Debido a la ausencia de memoria compartida: comunicación por mensajes



Capa de Middleware

- Capa de software que es situada lógicamente entre la capa de aplicación y la de transporte
- Interfaz es única
- Varios protocolos para soportar servicios de middleware:
 - Autenticación: No relacionados a una aplicación
 - Protocolos de confirmación (atomicidad)
 - Comunicación: evitar accesos simultáneos a un recurso.



Middleware de Comunicación

- Servicios de comunicación que pueden ser adicionados a la capa de middleware
- Tipos de middlewares de comunicación.:
 - Capa de procedimientos remoto
 - Comunicación orientada a mensajes
 - Comunicación orientada a flujo

Tipos de Comunicación (Persistencia)

- Persistente:
 - Almacena el Mensaje durante el tiempo que sea necesario para entregarlo al receptor
- Transitoria:
 - Almacena un mensaje solo mientras las aplicaciones remitente y destinatario se ejecutan
- Los mensajes serán almacenados o no por el middleware

Tipos de comunicación (sincronización)

- Asíncrona
 - Remitente continua su ejecución inmediatamente después de enviar el mensaje
- Síncrona
 - Remitente es bloqueado hasta que se sabe que su petición es aceptada
- Puntos de sincronización:
 - Remitente bloquea hasta que el middleware avise
 - Remitente bloquea hasta que su petición llegue al receptor
 - Remitente bloquea hasta que el receptor responde

Tipos de comunicación (granularidad)

- Discreta:
 - Comunicación por mensajes, donde cada uno de ellos es una unidad de información completa
- Flujo:
 - Comunicación por varios mensajes que están relacionados unos con otros
 - Relacionados por el orden o por la relación temporal.

Llamada de Procedimiento Remoto (RPC)

- Es un middleware de comunicación que “permite a los procesos llamar procedimientos localizados en otras máquinas”
- La idea es hacer que una llamada de procedimiento remoto se parezca a una llamada local (transparencia).
- Problema: Paso de mensajes en espacios de direccionamiento diferentes, caída de la máquina, etc.

RPC y los Stubs

- La transparencia es alcanzada a través de los stubs (apéndices):
- Stub del cliente:
 - Empaqueta los parámetros en un mensaje y lo envía para la máquina del servidor
 - Desempaqueta y compatibiliza la arquitectura.
- Stub del servidor
 - Desempaqueta los parámetros e invoca al procedimiento correcto pasando los parámetros
 - Empaqueta los parámetros en el mensaje de respuesta
 - Ambos son compilados antes

Pasos para RPC



Pasos para un RPC

- Empaquetar parámetros en un mensaje es conocido como marshalling de parámetro.
- RPC soporta:
 - Paso por valor
 - Paso por referencia

Comunicación orientada a mensaje

- Middle RPC y RMI pueden ser inadecuados
- Receptor siempre activo?
- El comportamiento de sincronismo y bloqueo puede ser inadecuado en ambientes con dispositivos volátiles
 - Red de sensores inalámbrica
 - IoT
 - Computación ubicua
- ¿Cómo enfrentar las limitaciones?
 - Orientación a mensajes

Interfaz de Intercambio de Mensajes (MPI)

- Aumento de arquitecturas paralelas y data centers
- Objetivo: Escribir con facilidad aplicaciones para arquitecturas paralelas.
- Ideal: independencia de hardware y de arquitectura que soporten comunicación orientada a mensajes

Interfaz de Intercambio de Mensajes (MPI)

- Porque no usar sockets?
 - Herramienta genérica para modelar comunicación remota para TCP/IP (sincronismo y buffer limitados)
- MPI : Considerado un padrón de intercambio de mensajes para clusters.
 - Modelo para aplicaciones paralelas (ej. funciones)
- Comunicación transitoria: Mensaje es almacenado en el sistema mientras el emisor y el receptor están activos

MPI: Características

- La comunicación ocurre dentro de un grupo conocido de procesos
- Cada grupo tiene un identificador propio
- Cada proceso dentro de un grupo recibe un identificador (local)
- Par (grupoID, procesoID) identificación fuente o destino único de un mensaje
- Varios grupos de procesos pueden hacer parte de un cálculo

Middleware Orientada a Mensaje (MOM)

- Soporte para comunicación asíncrona y persistente
- Idea: Aplicaciones se comunican retirando e insertando mensajes en filas específicas
- Mensajes serán entregadas eventualmente al receptor
- Comunicación flacamente acoplada

Cuatro combinaciones en el modo de ejecución

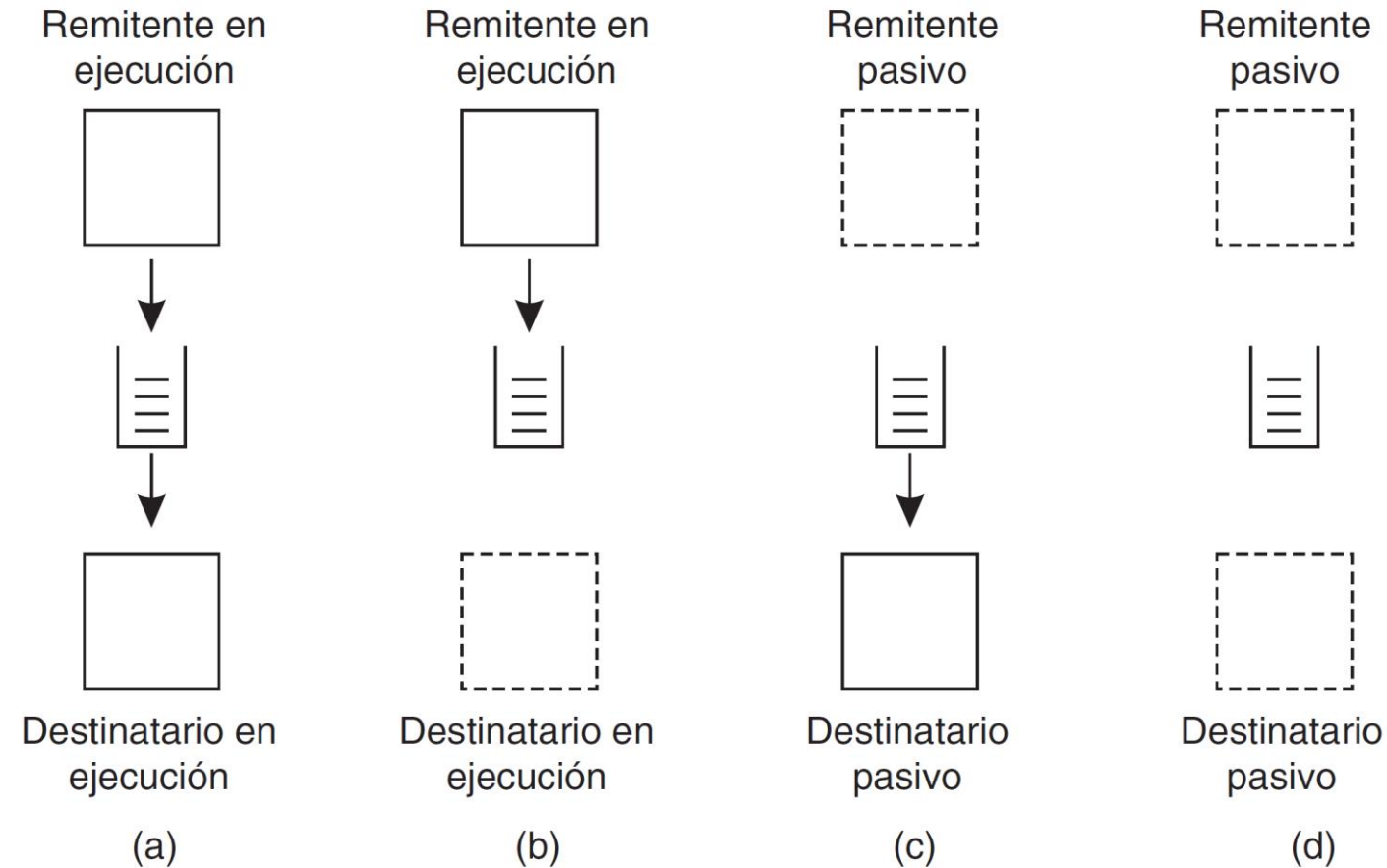


Figura 4-17. Cuatro combinaciones de comunicaciones muy poco acopladas mediante el uso de colas.

Preguntas

1. El término middleware y sistemas distribuidos son equivalentes? Si no, cuál es la diferencia?
2. Porque el objetivo de RPC es que se de la transparencia?
3. Qué es marshaling y stub: Qué pasos involucran, explique como actúan en RMI, cómo es el proceso de comunicación en RMI
4. Cuáles son los pasos involucrados en la implementación de un RPC
- 5.Cuál es la diferencia entre sockets y MPI?