

UNIDAD 8 --> Actividades solucionadas por alumnos

ACTIVIDAD 1 OBJETIVOS: Caracterizar los mecanismos de comunicación. Responda brevemente a las siguientes preguntas (con un máximo de 50 palabras por respuesta):

a) ¿Qué diferencias existen entre los mecanismos que utilizan primitivas básicas de comunicación y los que utilizan construcciones del lenguaje de programación?

Los basados en primitivas básicas de comunicación usan operaciones de envío y recepción, mientras que los que usan construcciones del lenguaje de programación proporcionan un nivel de abstracción superior, el envío y recepción de los mensajes se realiza de manera transparente.

b) ¿Cómo puede ser la estructura de los mensajes en un mecanismo de comunicación cualquiera?

Pueden no estar estructurados, tener únicamente una cabecera estructurada, o bien tener una estructura transparente al programador.

c) En los mecanismos de comunicación el contenido de los mensajes puede ser en bytes o bien en texto. ¿Qué ventajas presentan los mensajes de texto frente a los mensajes en bytes?

Los mensajes en texto tienen la ventaja de que son compatibles para cualquier arquitectura y lenguaje de programación, a diferencia de los mensajes en bytes.

d) ¿Qué diferencias existen entre el direccionamiento directo y el indirecto?

Con el directo el ordenador emisor envía directamente al receptor, mientras que con el indirecto el emisor envía los mensajes a un intermediario o bróker que los lleva al receptor.

e) ¿Qué tipos de sincronización existen en los mecanismos de comunicación? ¿Podrías dar ejemplos de cada uno?

Puede ser asincrónica, donde el emisor solo recibe la confirmación del middleware; sincrónica respecto a la entrega, en este caso el emisor recibe la confirmación en cuanto el receptor recibe el mensaje; o sincrónica respecto a la respuesta, donde el emisor recibe una confirmación después de que el receptor procese los datos.

Asincrónica: colas de mensajes

Sincrónicas (entrega): sockets TCP

Sincrónicas (entrega): RCP, ROI

f) ¿Qué diferencias existen entre comunicación persistente y no persistente? ¿Podrías dar ejemplos de mecanismos de comunicación para cada tipo de persistencia?

Para la comunicación persistente el middleware tendrá la capacidad de almacenar los mensajes, por lo que el receptor no requerirá estar activo cuando se envía el mensaje y el emisor podrá acabar su ejecución antes de que el mensaje sea recibido. Por otra parte, la comunicación no persistente requiere que ambos estén activos en el momento de transmitir el mensaje.

Persistente: colas de mensajes

No persistente: sockets, RCP/ROI

ACTIVIDAD 2 OBJETIVOS: Clarificar conceptos sobre mecanismos de comunicación. Explique brevemente los siguientes conceptos (con un máximo de 50 palabras cada uno):

- a) Sockets: sistema a través del cual dos programas o procesos (en principio situados en ordenadores distintos) pueden intercambiar un flujo de datos
- b) Servicios web: tecnología que permite a varias aplicaciones intercambiar datos en la red
- c) Colas de mensajes: sistema a través del cual una serie de mensajes se puede almacenar hasta que el receptor de los mismos pueda o decida acceder a ellos
- d) RPC: (Remote Procedure Call) es un mecanismo que permite y gestiona de manera transparente al programador la llamada a un catálogo de procedimientos en un ordenador remoto
- e) XML: es un lenguaje de marcado creado con el objetivo de transmitir datos a partir de varios sistemas en Internet
- f) JSON: (JavaScript Object Notation) es un formato de texto sencillo creado para el intercambio de datos
- g) Serialización: proceso por el cual un objeto se empaqueta para poder ser enviado a través de la red

ACTIVIDAD 3 OBJETIVOS: Caracterizar el mecanismo de Invocación a Objeto Remoto (ROI). Ordene, según el orden en que tienen lugar, los siguientes pasos de una ROI. ¿Falta algún paso para que todo el mecanismo ROI funcione? Si es así, descríbalos.

- C. El proceso cliente invoca el método del proxy local relacionado con el objeto remoto
- E. El proxy empaqueta los argumentos y, utilizando la referencia al objeto, llama al ORB
- F. El ORB gestiona la invocación, haciendo que el mensaje llegue al esqueleto.
- *. El esqueleto desempaqueta los argumentos e invoca el método deseado
- A. El método llamado finaliza y se desbloquea el esqueleto
- D. El esqueleto empaqueta los resultados y llama al ORB, el cual hace llegar el mensaje al proxy
- B. El proxy desempaqueta los resultados y los devuelve al proceso cliente