

05/02/2024

Desarrollo De Aplicaciones Distribuidas

Investigación:

Socket

Integrantes:

Perea Valdez Leonardo Alberto



Reflexión sobre los distintos tipos de servidores y clientes:

Servidor TCP en Python y Cliente TCP en Java:

- Ventajas: Conexión confiable al tener confirmación de entrega de datos. Lógica de flujo de bytes más simple al usar streams.
- Desventajas: Quizás un poco más complejo de implementar. Requiere que el servidor esté escuchando constantemente para aceptar nuevas conexiones de clientes. Si la conexión se interrumpe hay que volver a conectarse.

Servidor UDP en Python y Cliente UDP en Java:

- Ventajas: Más simple de implementar, sólo envío y recepción de datagramas. Permite broadcasting de mensajes. Menos sobrecarga.
- Desventajas: No hay confirmación de entrega o reordenamiento de datagramas. Probabilidad de pérdida de mensajes. Lógica más compleja para el desarrollador.

Servidor TCP en Python y Cliente UDP en Java:

- NO ES POSIBLE. TCP y UDP son protocolos incompatibles entre sí. Requieren lógicas de comunicación completamente distintas.
- Se podría crear alguna capa intermedia que convierta entre TCP y UDP, pero sería algo muy complejo y con varias desventajas de performance y latencia.

En conclusión, la opción 1 de usar el mismo protocolo (TCP o UDP) para el servidor y todos los clientes es la más recomendada, por simplicidad y consistencia en la lógica de comunicaciones de red entre los procesos.