

LABORATORIO 7

IOT: ESTACIÓN METEOROLÓGICA





OBJETIVO

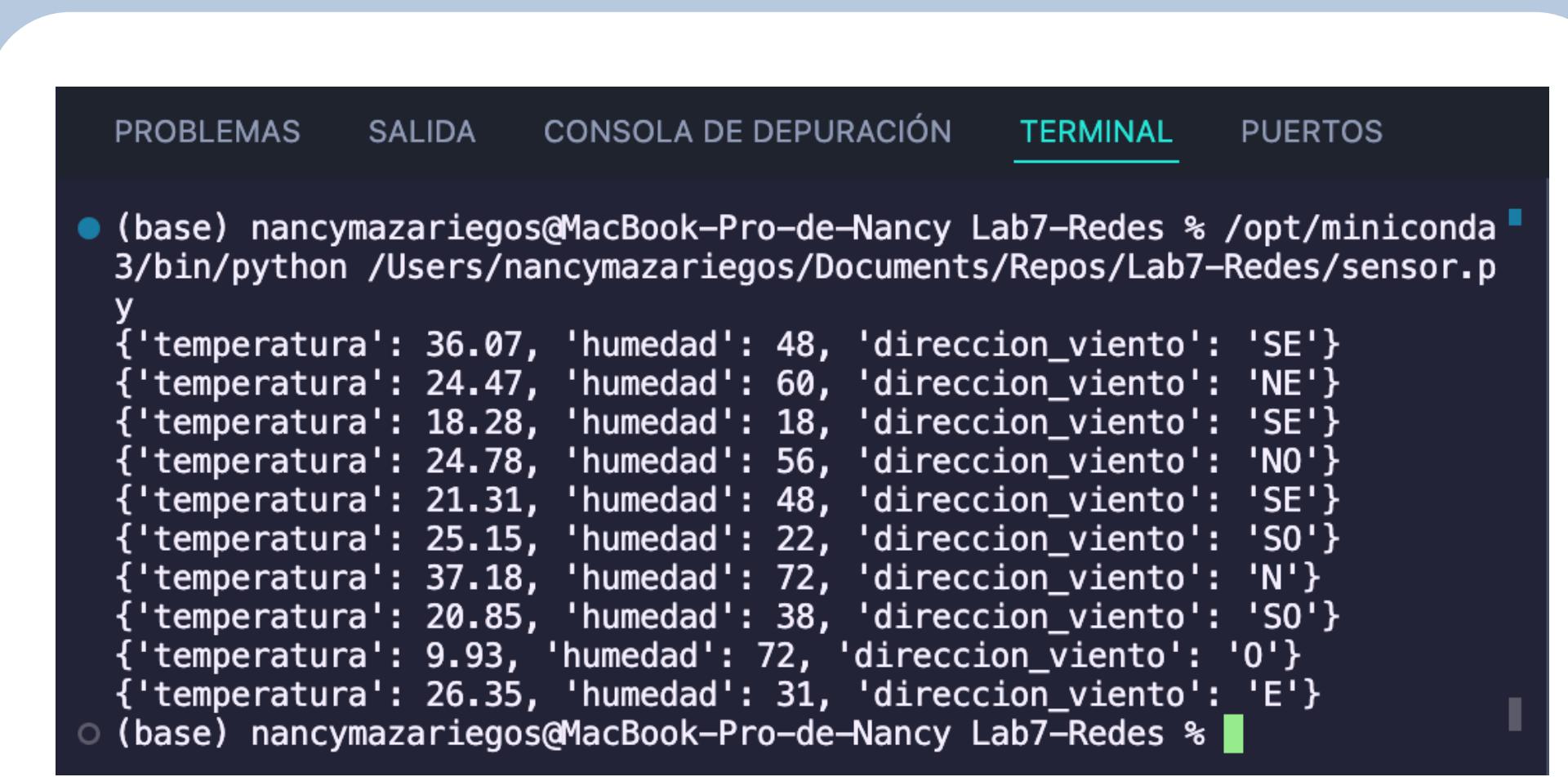
Implementar una estación meteorológica simulada que envía lecturas de temperatura, humedad y dirección del viento usando Kafka bajo un esquema Pub/Sub.

Primero se usan mensajes en formato JSON y luego un payload compacto de 3 bytes, para comparar su funcionamiento y las implicaciones de cada enfoque.

SENSOR.PY

SE IMPLEMENTÓ UNA FUNCIÓN
GENERAR_MEDICION() QUE SIMULA UNA
ESTACIÓN METEOROLÓGICA:

- GENERA UNA TEMPERATURA ALEATORIA ENTRE 0 Y 110 °C, CON DOS DECIMALES.
- GENERA UNA HUMEDAD RELATIVA ENTRE 0 Y 100 %.
- SELECCIONA UNA DIRECCIÓN DEL VIENTO ENTRE 8 POSIBLES (N, NO, O, SO, S, SE, E, NE).



The screenshot shows a terminal window with several tabs at the top: PROBLEMAS, SALIDA, CONSOLA DE DEPURACIÓN, TERMINAL (which is underlined in blue), and PUERTOS. The terminal content displays the following:

```
(base) nancymazariegos@MacBook-Pro-de-Nancy Lab7-Redes % /opt/miniconda3/bin/python /Users/nancymazariegos/Documents/Repos/Lab7-Redes/sensor.py
[...]
{'temperatura': 36.07, 'humedad': 48, 'direccion_viento': 'SE'}
{'temperatura': 24.47, 'humedad': 60, 'direccion_viento': 'NE'}
{'temperatura': 18.28, 'humedad': 18, 'direccion_viento': 'SE'}
{'temperatura': 24.78, 'humedad': 56, 'direccion_viento': 'NO'}
{'temperatura': 21.31, 'humedad': 48, 'direccion_viento': 'SE'}
{'temperatura': 25.15, 'humedad': 22, 'direccion_viento': 'SO'}
{'temperatura': 37.18, 'humedad': 72, 'direccion_viento': 'N'}
{'temperatura': 20.85, 'humedad': 38, 'direccion_viento': 'SO'}
{'temperatura': 9.93, 'humedad': 72, 'direccion_viento': 'O'}
{'temperatura': 26.35, 'humedad': 31, 'direccion_viento': 'E'}
[...]
(base) nancymazariegos@MacBook-Pro-de-Nancy Lab7-Redes %
```

SENSOR.PY

¿A QUÉ CAPA PERTENECE JSON/SOAP SEGÚN EL MODELO OSI Y PORQUÉ?

JSON Y SOAP PERTENECEN A LA CAPA DE APLICACIÓN DEL MODELO OSI.

ESTO ES PORQUE DEFINEN EL FORMATO Y LA ESTRUCTURA DE LOS DATOS QUE INTERCAMBIAN LAS APLICACIONES (CÓMO SE REPRESENTAN LOS MENSAJES, QUÉ CAMPOS TIENEN, EN QUÉ ORDEN, ETC.), PERO NO SE ENCARGAN DEL TRANSPORTE, EL ENRUTAMIENTO NI EL CONTROL DE ERRORES. ES DECIR, VAN “ENCIMA” DE PROTOCOLOS COMO HTTP/TCP Y SON USADOS DIRECTAMENTE POR EL SOFTWARE DE APLICACIÓN.

¿QUÉ BENEFICIOS TIENE UTILIZAR UN FORMATO COMO JSON/SOAP?

- INTEROPERABILIDAD: PERMITEN QUE SISTEMAS ESCRITOS EN LENGUAJES Y PLATAFORMAS DIFERENTES SE ENTIENDAN ENTRE SÍ.
- ESTANDARIZACIÓN: SIGUEN REGLAS BIEN DEFINIDAS, POR LO QUE MUCHAS HERRAMIENTAS Y LIBRERÍAS YA LOS SOPORTAN.
- LEGIBILIDAD: ESPECIALMENTE JSON ES FÁCIL DE LEER Y DEPURAR POR HUMANOS.
- EXTENSIBILIDAD: SE PUEDEN AGREGAR NUEVOS CAMPOS SIN ROMPER NECESARIAMENTE A LOS CLIENTES EXISTENTES, SIEMPRE QUE SE MANTenga COMPATIBILIDAD.

PRODUCER_JSON.PY

EL PRODUCER SE CONECTA AL BROKER KAFKA
IOT.REDESVG.CLOUD:9092 Y PUBLICA LOS
DATOS SIMULADOS EN EL TOPIC 22513 USANDO
MENSAJES EN FORMATO JSON.

EL PRODUCER ENVÍA UNA LECTURA CADA 15–30
SEGUNDOS CON LOS CAMPOS:

- TEMPERATURA
- HUMEDAD
- DIRECCION_VIENTO

```
PROBLEMAS SALIDA CONSOLA DE DEPURACIÓN TERMINAL PUERTOS

(base) nancymazariegos@MacBook-Pro-de-Nancy Lab7-Redes % /opt/miniconda 3/bin/python /Users/nancymazariegos/Documents/Repos/Lab7-Redes/producer_json.py
[2025-11-16 16:51:12.157200] Producer JSON iniciado. Enviando a topic '22513'...
Presiona Ctrl+C para detener.

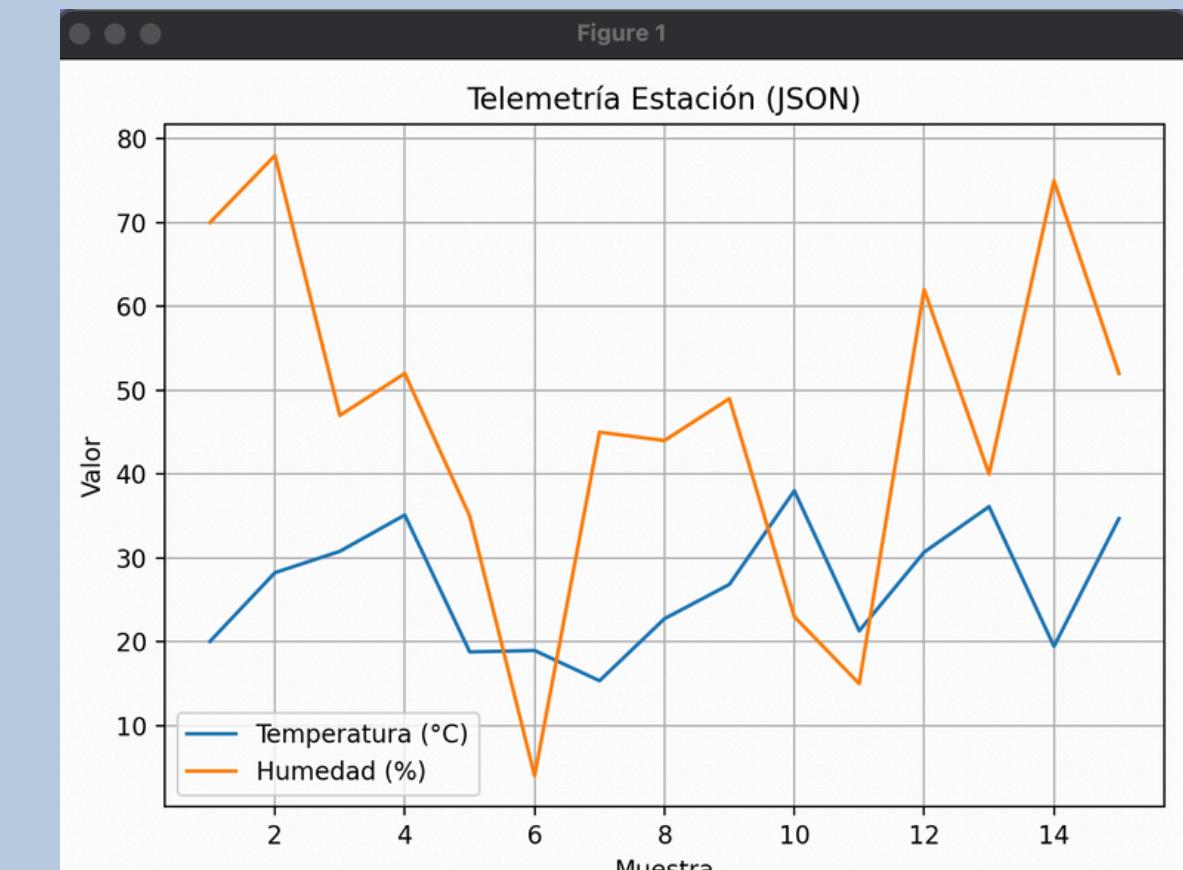
[2025-11-16 16:51:12.479118] Enviado a 22513: {'temperatura': 39.77, 'humedad': 44, 'direccion_viento': 'S'}
[2025-11-16 16:51:40.481331] Enviado a 22513: {'temperatura': 30.78, 'humedad': 84, 'direccion_viento': 'O'}
[2025-11-16 16:52:07.488275] Enviado a 22513: {'temperatura': 13.37, 'humedad': 52, 'direccion_viento': 'E'}
[2025-11-16 16:52:31.492433] Enviado a 22513: {'temperatura': 22.68, 'humedad': 43, 'direccion_viento': 'NO'}
[2025-11-16 16:52:53.493849] Enviado a 22513: {'temperatura': 18.16, 'humedad': 79, 'direccion_viento': 'SO'}
```

Share Lín. 9, col. 15

PRODUCER + CONSUMER JSON

SE EJECUTARON SIMULTÁNEAMENTE EL PRODUCER Y EL CONSUMER USANDO MENSAJES EN FORMATO JSON.

- EL PRODUCER PUBLICÓ LECTURAS SIMULADAS DE TEMPERATURA, HUMEDAD Y DIRECCIÓN DEL VIENTO EN EL TOPIC 22513, MIENTRAS QUE EL CONSUMER LAS RECIBIÓ EN TIEMPO REAL Y ACTUALIZÓ UNA GRÁFICA CON AMBAS VARIABLES.



PROBLEMAS SALIDA CONSOLA DE DEPURACIÓN TERMINAL PUERTOS

```
(base) nancymazariegos@MacBook-Pro-de-Nancy Lab7-Redes % python consumer_json_plot.py
[2025-11-16 16:25:48.049130] Consumidor JSON escuchando en topic '22513'.
..
Abre también producer_json.py para empezar a recibir datos.

[2025-11-16 16:26:00.932770] Recibido #1: {'temperatura': 20.05, 'humedad': 70, 'direccion_viento': 'SE'}
[2025-11-16 16:26:28.942862] Recibido #2: {'temperatura': 28.24, 'humedad': 78, 'direccion_viento': 'S'}
[2025-11-16 16:26:56.949714] Recibido #3: {'temperatura': 30.79, 'humedad': 47, 'direccion_viento': 'N'}
[2025-11-16 16:27:24.952765] Recibido #4: {'temperatura': 35.12, 'humedad': 52, 'direccion_viento': 'S'}
[2025-11-16 16:27:49.063366] Recibido #5: {'temperatura': 18.79, 'humedad': 35, 'direccion_viento': 'O'}
[2025-11-16 16:28:10.958772] Recibido #6: {'temperatura': 18.96, 'humedad': 4, 'direccion_viento': 'SE'}
[2025-11-16 16:28:32.966457] Recibido #7: {'temperatura': 15.35, 'humedad': 45, 'direccion_viento': 'O'}
[2025-11-16 16:28:48.979899] Recibido #8: {'temperatura': 22.75, 'humedad': 44, 'direccion_viento': 'S'}
[2025-11-16 16:29:10.982962] Recibido #9: {'temperatura': 26.84, 'humedad': 49, 'direccion_viento': 'S'}
[2025-11-16 16:29:36.984202] Recibido #10: {'temperatura': 38.01, 'humedad': 23, 'direccion_viento': 'E'}
[2025-11-16 16:30:06.986934] Recibido #11: {'temperatura': 21.3, 'humedad': 15, 'direccion_viento': 'N'}
[2025-11-16 16:30:33.992486] Recibido #12: {'temperatura': 30.69, 'humedad': 62, 'direccion_viento': 'SE'}
[2025-11-16 16:31:01.007697] Recibido #13: {'temperatura': 36.12, 'humedad': 40, 'direccion_viento': 'N'}
[2025-11-16 16:31:17.998656] Recibido #14: {'temperatura': 19.45, 'humedad': 75, 'direccion_viento': 'N'}
[2025-11-16 16:31:42.010741] Recibido #15: {'temperatura': 34.69, 'humedad': 52, 'direccion_viento': 'NO'}
```

+ - | X

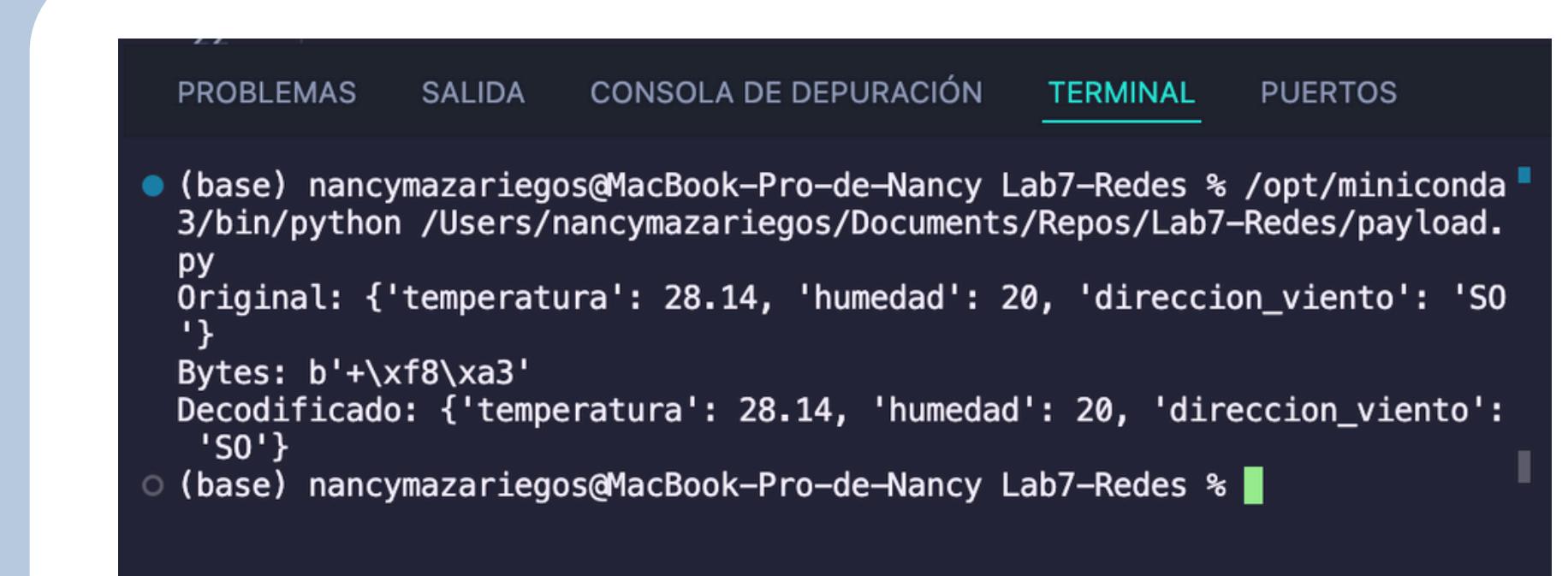
zsh
zsh
python3...
python3...

PAYLOAD.PY

SE DISEÑÓ UN FORMATO DE 24 BITS (3 BYTES) PARA ENVIAR LA MISMA INFORMACIÓN USANDO MENOS ESPACIO:

- BITS 23–10 (14 BITS): TEMPERATURA $\times 100$ (0–110.00 °C → 0–11000).
- BITS 9–3 (7 BITS): HUMEDAD ENTERA (0–100 %).
- BITS 2–0 (3 BITS): DIRECCIÓN DEL VIENTO (ÍNDICE 0–7).

DE ESTA FORMA, TODA LA LECTURA DE LA ESTACIÓN CABE EN SOLO 3 BYTES, SACRIFICANDO UN POCO DE SIMPLICIDAD A CAMBIO DE AHORRAR ANCHO DE BANDA.



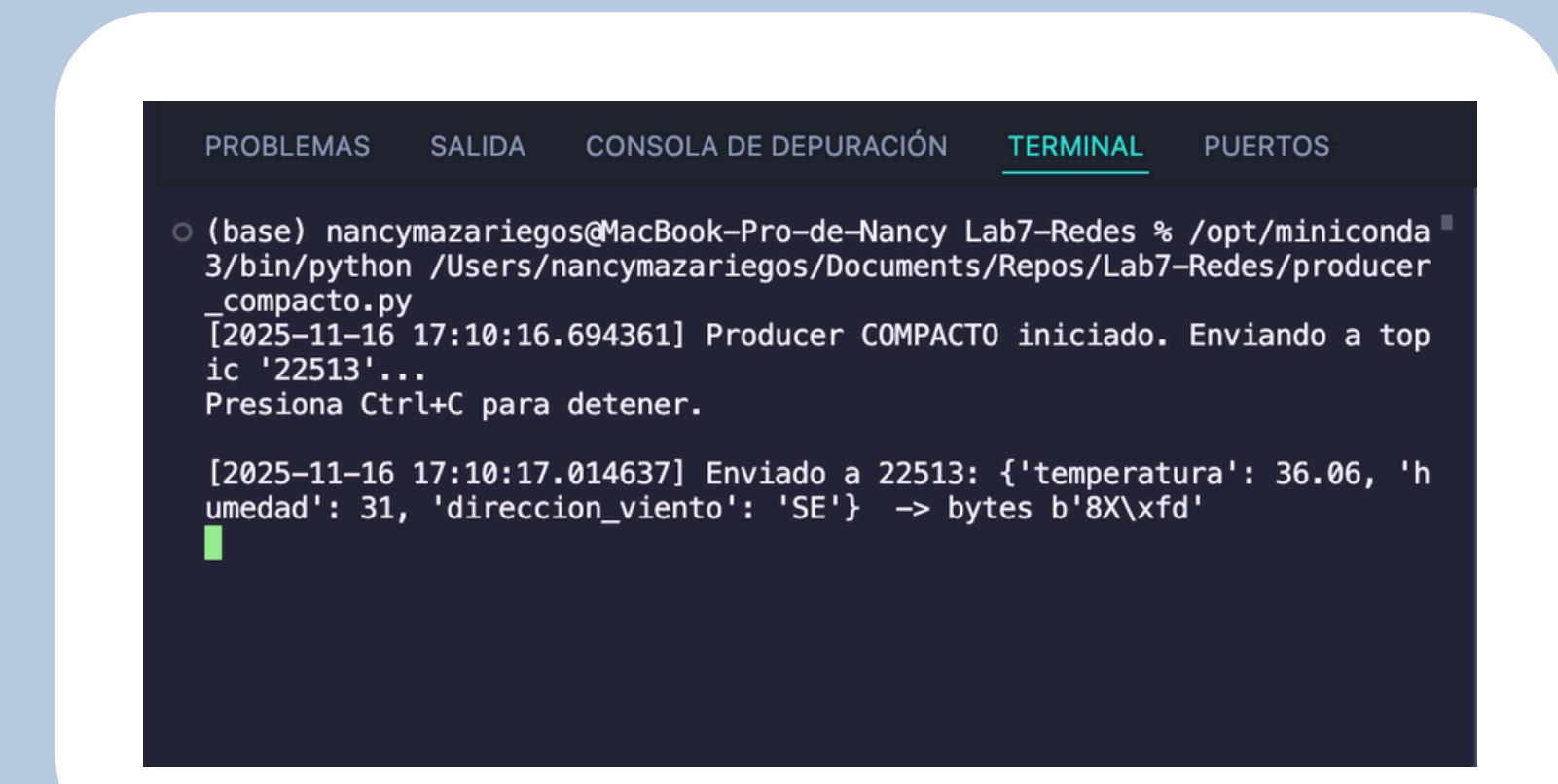
The screenshot shows a terminal window with several tabs at the top: PROBLEMAS, SALIDA, CONSOLA DE DEPURACIÓN, TERMINAL (which is underlined in blue), and PUERTOS. The terminal content is as follows:

```
(base) nancymazariegos@MacBook-Pro-de-Nancy Lab7-Redes % /opt/miniconda3/bin/python /Users/nancymazariegos/Documents/Repos/Lab7-Redes/payload.py
Original: {'temperatura': 28.14, 'humedad': 20, 'direccion_viento': 'S0'}
Bytes: b'\xf8\x80\x83'
Decodificado: {'temperatura': 28.14, 'humedad': 20, 'direccion_viento': 'S0'}
```

PRODUCER_COMPACTO.PY

EN ESTE ARCHIVO USE PRODUCER_COMPACTO PARA PRODUCIR LOS DATOS Y ENVIARLOS.
LAS MEDICIONES SE GENERAN CON EL ARCHIVO SENSOR.PY, LUEGO SE CODIFICAN A UN PAYLOAD DE TRES BYTES CON PAYLOAD_COMPACTO.PY Y FINALMENTE SE ENVIAN AL TOPIC 22513 EN EL BROKER DE KAFKA.

- EN LA SALIDA SE PUEDE VER TANTO LA MEDICION ORIGINAL COMO LOS TRES BYTES ENVIADOS, LO QUE CONFIRMA QUE EL PROCESO DE CODIFICACION Y ENVIO ESTA FUNCIONANDO BIEN.



```
PROBLEMAS SALIDA CONSOLA DE DEPURACIÓN TERMINAL PUERTOS

(base) nancymazariegos@MacBook-Pro-de-Nancy: ~ /opt/miniconda3/bin/python /Users/nancymazariegos/Documents/Repos/Lab7-Redes/producer_compacto.py
[2025-11-16 17:10:16.694361] Producer COMPACTO iniciado. Enviando a topic '22513'...
Presiona Ctrl+C para detener.

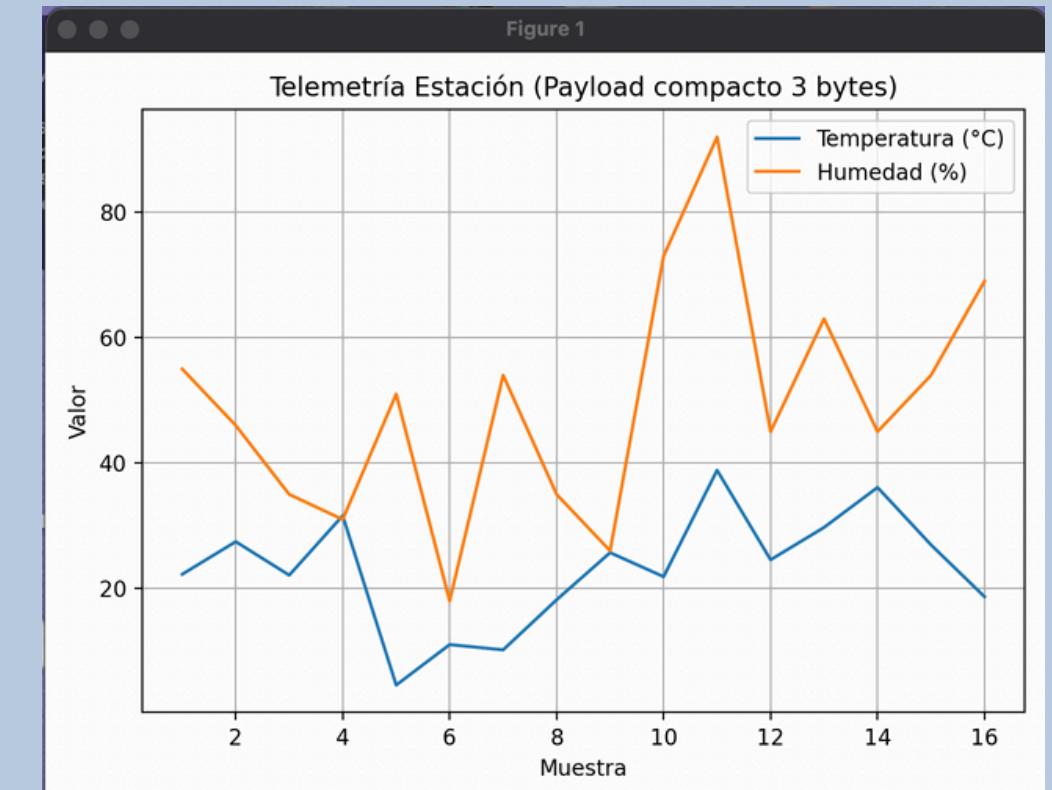
[2025-11-16 17:10:17.014637] Enviado a 22513: {'temperatura': 36.06, 'humedad': 31, 'direccion_viento': 'SE'} -> bytes b'8X\xfd'
```

PRODUCER + CONSUMER COMPACTO

EN ESTA PARTE DEL LABORATORIO SE CORRÍO EL PRODUCER COMPACTO Y EL CONSUMER COMPACTO AL MISMO TIEMPO.

EL PRODUCER YA NO ENVIA UN MENSAJE JSON, AHORA SOLO MANDA TRES BYTES POR CADA MEDICION.

EL CONSUMIDOR TOMA ESOS TRES BYTES, LOS DECODIFICA Y VUELVE A OBTENER LA TEMPERATURA, LA HUMEDAD Y LA DIRECCION DEL VIENTO.



PROBLEMAS SALIDA CONSOLA DE DEPURACIÓN TERMINAL PUERTOS

```
(base) nancymazariegos@MacBook-Pro-de-Nancy Lab7-Redes % python consumer.py
[2025-11-16 16:38:06.081812] Consumer COMPACTO escuchando en topic '22513'...
Abre también producer_compacto.py para empezar a recibir datos.

[2025-11-16 16:38:20.050649] Recibido #1: bytes=b'\x99\xbc' -> {'temperatura': 22.26, 'humedad': 55, 'direccion_viento': 'S'}
[2025-11-16 16:38:40.042816] Recibido #2: bytes=b'*\xe9u' -> {'temperatura': 27.46, 'humedad': 46, 'direccion_viento': 'SE'}
[2025-11-16 16:39:08.030656] Recibido #3: bytes=b'\x8d\x19' -> {'temperatura': 22.11, 'humedad': 35, 'direccion_viento': 'NO'}
[2025-11-16 16:39:26.034363] Recibido #4: bytes=b'1\xf8' -> {'temperatura': 31.6, 'humedad': 31, 'direccion_viento': 'N'}
[2025-11-16 16:39:44.045366] Recibido #5: bytes=b'\x07)\x98' -> {'temperatura': 4.58, 'humedad': 51, 'direccion_viento': 'N'}
[2025-11-16 16:40:00.048274] Recibido #6: bytes=b'\x10\x93' -> {'temperatura': 11.04, 'humedad': 18, 'direccion_viento': 'S0'}
[2025-11-16 16:40:23.050950] Recibido #7: bytes=b'\x0fx\x1fb3' -> {'temperatura': 10.2, 'humedad': 54, 'direccion_viento': 'S0'}
[2025-11-16 16:40:38.044149] Recibido #8: bytes=b'\x1cm\x1b' -> {'temperatura': 18.19, 'humedad': 35, 'direccion_viento': 'S0'}
[2025-11-16 16:41:03.056980] Recibido #9: bytes=b'(\,xd4' -> {'temperatura': 25.71, 'humedad': 26, 'direccion_viento': 'S'}
[2025-11-16 16:41:28.060264] Recibido #10: bytes=b'"I' -> {'temperatura': 21.87, 'humedad': 73, 'direccion_viento': 'NO'}
[2025-11-16 16:41:46.095321] Recibido #11: bytes=b'<\xb6\xe5' -> {'temperatura': 38.85, 'humedad': 92, 'direccion_viento': 'SE'}
[2025-11-16 16:42:09.068321] Recibido #12: bytes=b'&i' -> {'temperatura': 24.57, 'humedad': 45, 'direccion_viento': 'NO'}
[2025-11-16 16:42:27.077127] Recibido #13: bytes=b',\xf9' -> {'temperatura': 29.73, 'humedad': 63, 'direccion_viento': 'NO'}
[2025-11-16 16:42:53.079492] Recibido #14: bytes=b'8mI' -> {'temperatura': 36.11, 'humedad': 45, 'direccion_viento': 'NO'}
[2025-11-16 16:43:13.086086] Recibido #15: bytes=b'\x1d\xb0' -> {'temperatura': 26.95, 'humedad': 54, 'direccion_viento': 'N'}
```

zsh
python3...
python3...

Lín. 28, col. 48 Espacios: 4 UTF-8 LF Python Finish Setup base (3.13.7) Go Live Prettier

PREGUNTAS

¿A QUÉ CAPA PERTENECE JSON/SOAP SEGÚN EL MODELO OSI Y PORQUÉ?

JSON Y SOAP PERTENECEN A LA CAPA DE APLICACIÓN DEL MODELO OSI.

ESTO ES PORQUE DEFINEN EL FORMATO Y LA ESTRUCTURA DE LOS DATOS QUE INTERCAMBIAN LAS APLICACIONES (CÓMO SE REPRESENTAN LOS MENSAJES, QUÉ CAMPOS TIENEN, EN QUÉ ORDEN, ETC.), PERO NO SE ENCARGAN DEL TRANSPORTE, EL ENRUTAMIENTO NI EL CONTROL DE ERRORES. ES DECIR, VAN “ENCIMA” DE PROTOCOLOS COMO HTTP/TCP Y SON USADOS DIRECTAMENTE POR EL SOFTWARE DE APLICACIÓN.

¿QUÉ BENEFICIOS TIENE UTILIZAR UN FORMATO COMO JSON/SOAP?

- INTEROPERABILIDAD: PERMITEN QUE SISTEMAS ESCRITOS EN LENGUAJES Y PLATAFORMAS DIFERENTES SE ENTIENDAN ENTRE SÍ.
- ESTANDARIZACIÓN: SIGUEN REGLAS BIEN DEFINIDAS, POR LO QUE MUCHAS HERRAMIENTAS Y LIBRERÍAS YA LOS SOPORTAN.
- LEGIBILIDAD: ESPECIALMENTE JSON ES FÁCIL DE LEER Y DEPURAR POR HUMANOS.
- EXTENSIBILIDAD: SE PUEDEN AGREGAR NUEVOS CAMPOS SIN ROMPER NECESARIAMENTE A LOS CLIENTES EXISTENTES, SIEMPRE QUE SE MANTenga COMPATIBILIDAD.

¿QUÉ VENTAJAS Y DESVENTAJAS TIENE EL ENFOQUE PUB/SUB DE KAFKA?

VENTAJAS:

- DESACOPLA PRODUCTOR Y CONSUMIDOR.
- ESCALA FÁCIL (MÁS PRODUCTORES/CONSUMIDORES).
- PERMITE PROCESAR DATOS EN TIEMPO REAL Y RE-LEER EVENTOS.

DESVENTAJAS:

- INFRAESTRUCTURA MÁS COMPLEJA DE MONTAR Y MANTENER.
- CURVA DE APRENDIZAJE ALTA.
- ES EXAGERADO PARA SISTEMAS MUY PEQUEÑOS.

¿PARA QUÉ APLICACIONES SÍ Y PARA CUÁLES NO USAR KAFKA?

SÍ TIENE SENTIDO:

- TELEMETRÍA E IOT (SENSORES), LOGS, MÉTRICAS.
- DASHBOARDS Y ANÁLISIS EN TIEMPO REAL.
- ARQUITECTURAS DE MICROSERVICIOS BASADAS EN EVENTOS.

NO TIENE TANTO SENTIDO:

- APPS PEQUEÑAS CON POCAS PETICIONES.
- CASOS SIMPLES DE REQUEST–RESPONSE (UN API CHICO).
- CUANDO NO QUIERES MANTENER UN CLÚSTER ADICIONAL.

¿QUÉ COMPLEJIDADES INTRODUCE UN PAYLOAD PEQUEÑO?

- HAY QUE EMPAQUETAR DATOS A NIVEL DE BITS.
- OBLIGA A DECIDIR CUÁNTA PRECISIÓN GUARDAR.
- CODIFICACIÓN/DECODIFICACIÓN MÁS COMPLEJA Y MÁS DIFÍCIL DE DEPURAR.

¿CÓMO HACER QUE LA TEMPERATURA QUEPA EN 14 BITS?

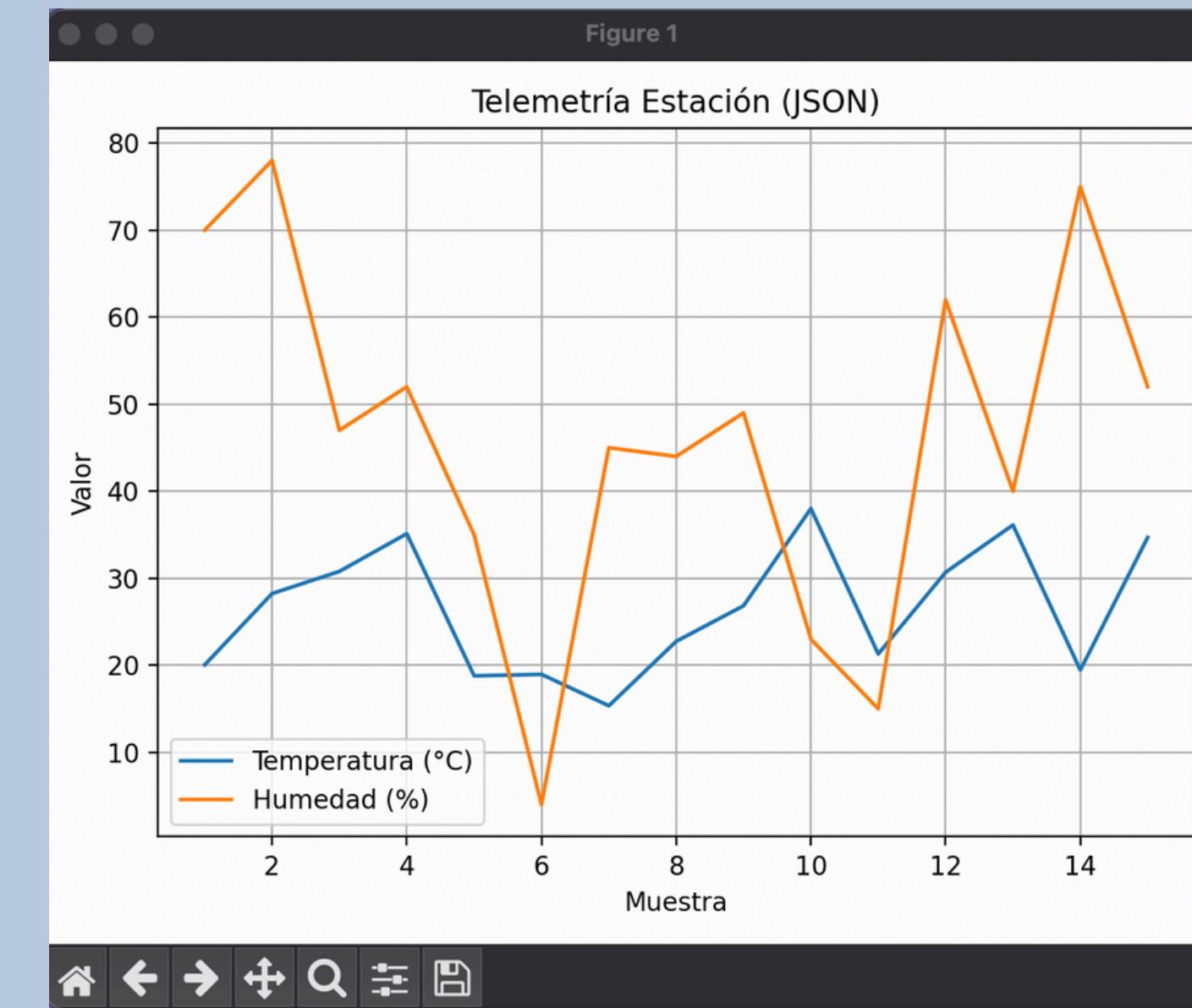
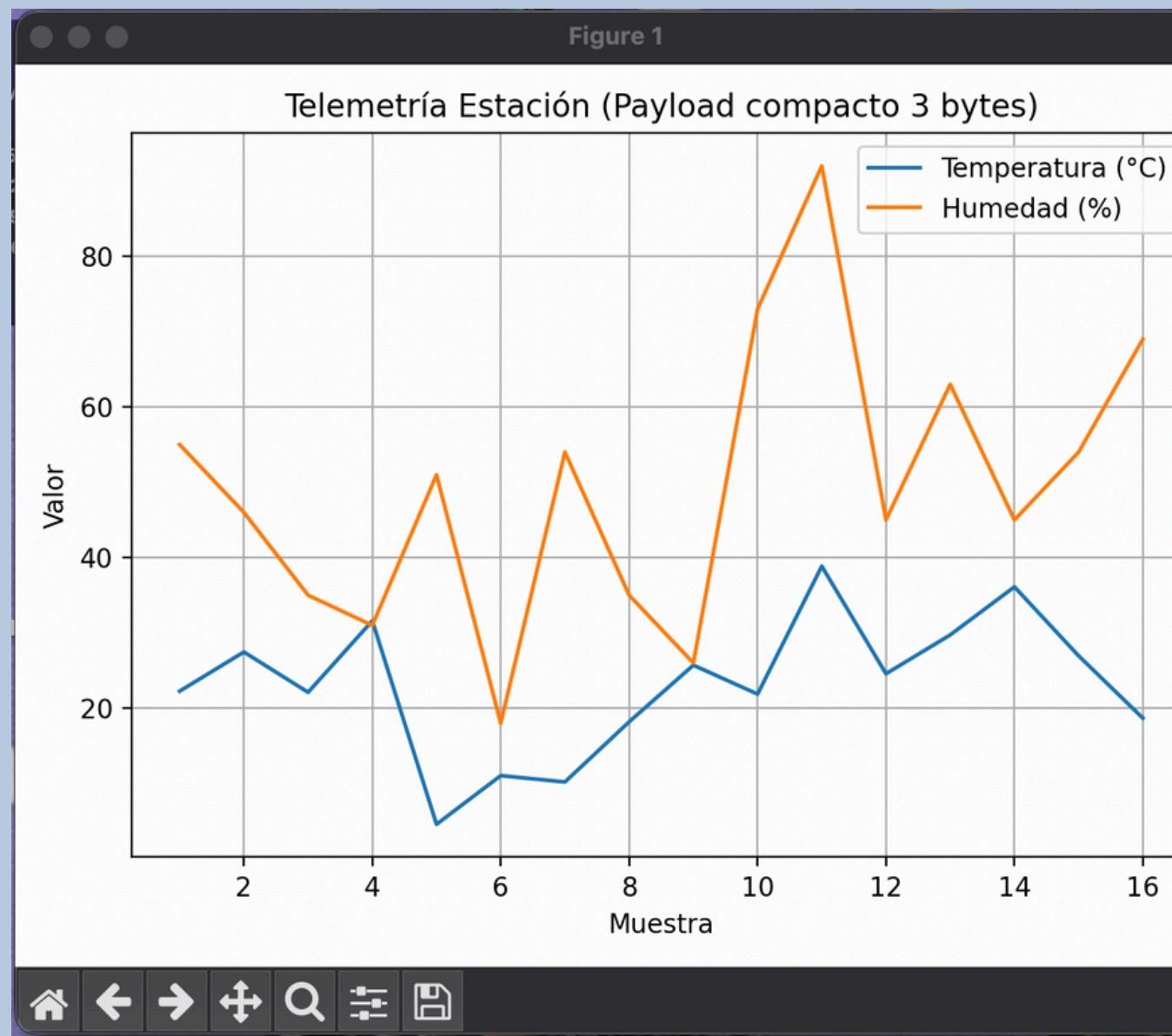
- RANGO: 0–110 °C.
- GUARDAR TEMP_INT = ROUND(TEMP * 100) → 0–11000.
- 14 BITS PERMITEN HASTA 16383, ASÍ QUE 0–11000 CABE SIN PROBLEMA.
- AL DECODIFICAR: TEMPERATURA = TEMP_INT / 100.

¿Y SI LA HUMEDAD TAMBIÉN FUERA FLOAT CON UN DECIMAL?

- HABRÍA QUE GUARDAR 0–100.0 COMO 0–1000 (MULTIPLICANDO POR 10).
- ESO NECESITA 10 BITS, PERO SOLO TIENES 7 AHORA.
- TENDRÍAS QUE:
 - BAJAR PRECISIÓN (SIN DECIMALES),
 - ROBAR BITS A OTRA VARIABLE, O
 - AUMENTAR EL TAMAÑO DEL PAYLOAD.

¿QUÉ DE KAFKA AYUDA SI LAS RESTRICCIONES FUERAN AÚN MÁS FUERTES?

- COMPRESIÓN DE MENSAJES (COMPRESSION.TYPE = GZIP, LZ4, ETC.).
- AGRUPAR MENSAJES EN BATCHES (BATCH.SIZE, LINGER.MS).
- PROCESAR Y RESUMIR DATOS ANTES (KAFKA STREAMS) PARA ENVIAR MENOS INFORMACIÓN CRUDA.



CONCLUSIONES

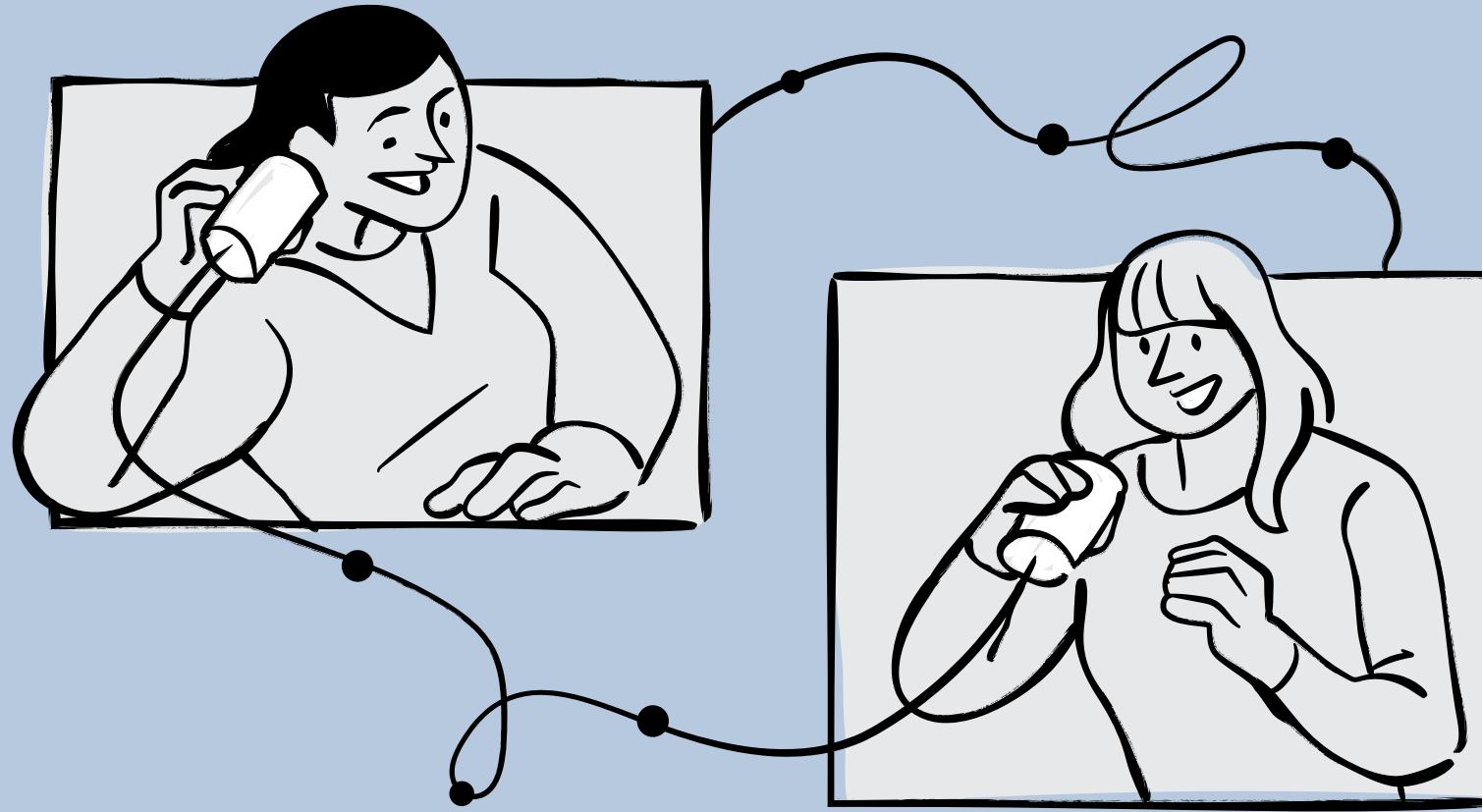
El uso de Kafka con el modelo de publicacion y suscripcion permite desacoplar al productor y al consumidor y facilita el manejo de flujos de datos en tiempo real

El payload compacto de tres bytes reduce de forma importante el tamano del mensaje, a costa de una codificacion y decodificacion mas compleja

El formato JSON es muy util para depurar y entender los mensajes, pero genera mensajes mas grandes en la red

Se demostró que es posible mantener la información esencial de la estación meteorológica aun cuando el mensaje es muy pequeño





¡MUCHAS GRACIAS!