

Multidifusión y paso de mensajes

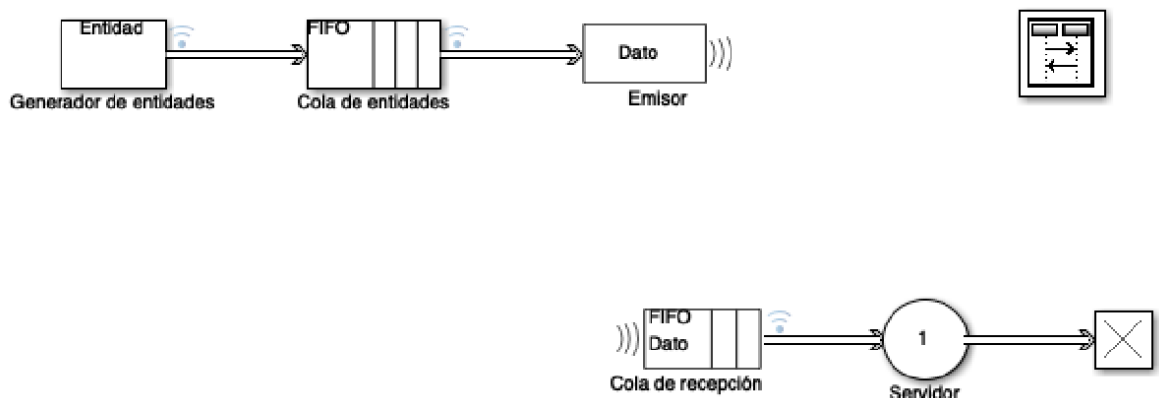
Multidifusión y paso de mensajes

Multidifusión de entidades	3
Paso de mensajes	5

Multidifusión de entidades

En múltiples ocasiones el emisor de entidades no está físicamente próximo a la entidad que tiene que dar el servicio. Esa «transmisión» de entidades puede sufrir sus propias vicisitudes (retrasos, bloqueos, retransmisiones, ...). Para estos escenarios, es necesario disponer de un Generador de entidades que nos proporcione ese modelo concreto de distancia física.

En el siguiente modelo puede verse cómo emitir entidades utilizando los bloques de multidifusión de entidades y cola de recepción de multidifusión. Se utilizará la multidifusión de entidades cuando sea necesario copiar y difundir entidades a un solo receptor o a múltiples receptores en su modelo. Una aplicación común es la creación de redes de comunicación en las que los mensajes se copian y transmiten entre los nodos de la red.

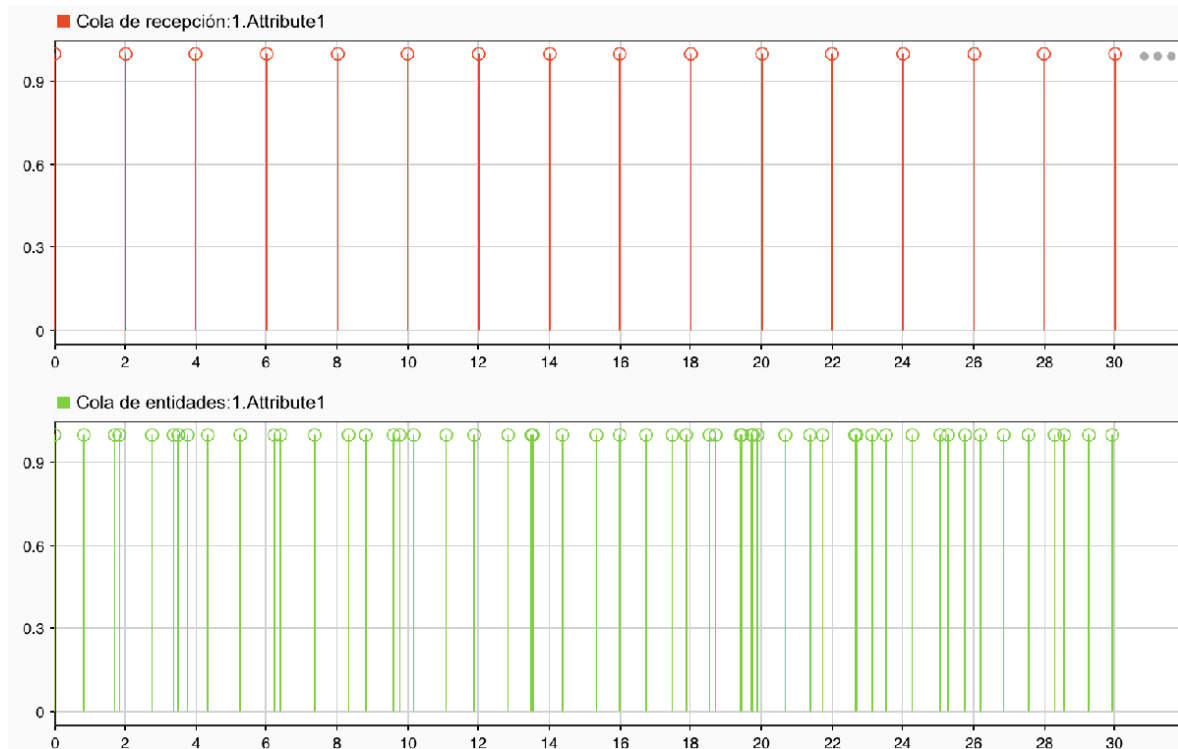


La configuración de los nuevos componentes es sencilla:

- El Generador de entidades se establece con **Time source - MATLAB.action** a un valor de `dt = rand(1,1);`
- La Cola de entidades se deja con los valores por defecto.

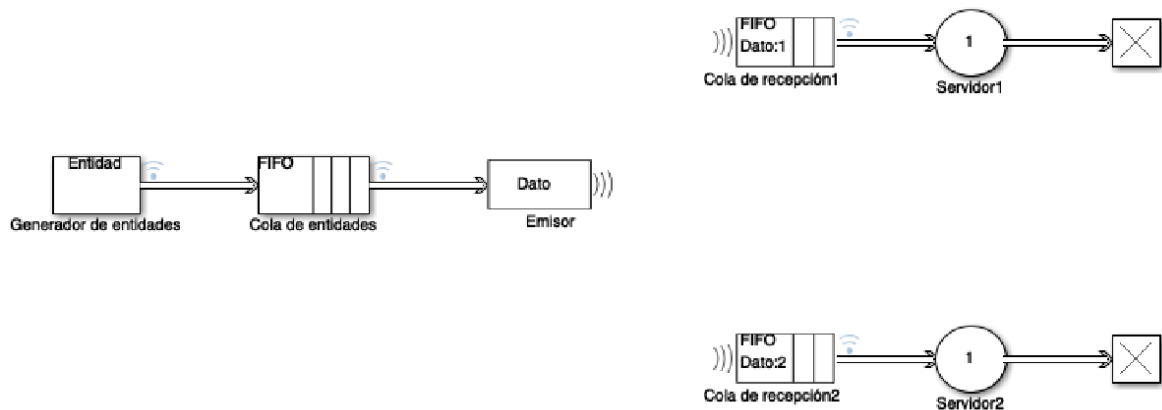
- El Emisor se etiqueta en **Multicast tag** con un nombre conveniente. En nuestro caso, Dato.
- La cola de recepción debe contener en **Multicast tag** el mismo nombre que hayamos puesto en el emisor (Dato)
- El Servidor se establece con un valor de tiempo de servicio de 2.
- Se puede añadir el **Visor de Secuencias** para analizar los resultados, aunque es opcional.

Es interesante observar el comportamiento del modelo utilizando el Data Inspector. Aquí mostraremos la cola de entidades y la cola de recepción. (Nótese que la señal del emisor no tiene la posibilidad de ser inspeccionada):

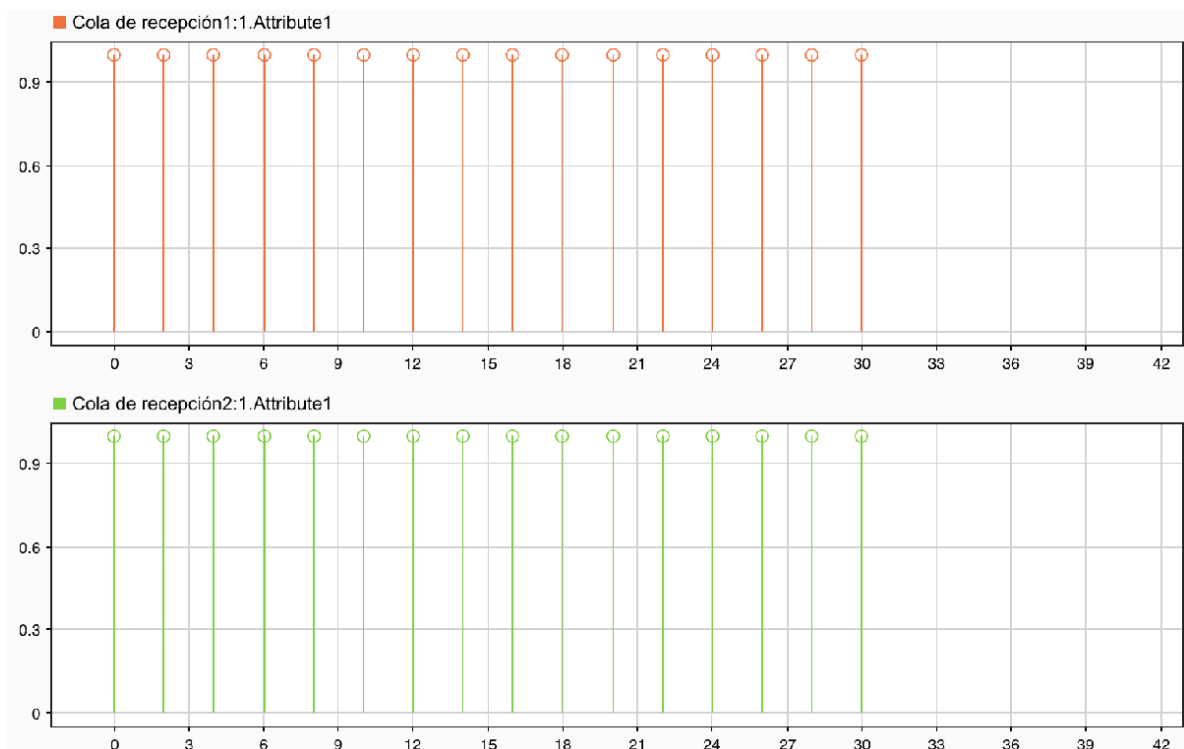


Merece la pena observar que la cola de recepción produce una especie de «suavizado» u «homegeinización» en las entidades que llegan al servidor, esto es, retiene las entidades que van llegando, bloqueando si es preciso su generación, y las envía al servidor según la demanda de este.

Este modelo se puede extender a varias líneas de recepción como se muestra en el modelo siguiente:

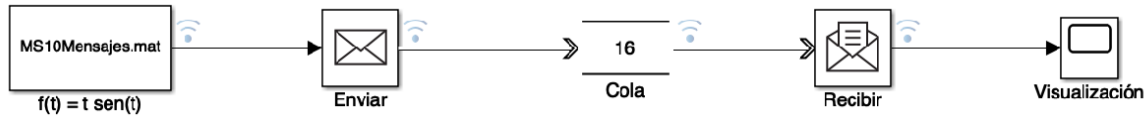


Las secuencias en los receptores resultarán exactamente las mismas, como debía esperarse:



Paso de mensajes

En esta ocasión estudiaremos cómo se envían, reciben y visualizan mensajes y veremos la herramienta de animación (**Animation**) y el inspector de almacenamiento (**Storage Inspector**). El modelo con el que vamos a trabajar es el siguiente:



Donde la función $f(t) = t \sin t$ se envía a través de una cola que la discretiza y se recibe en el bloque correspondiente

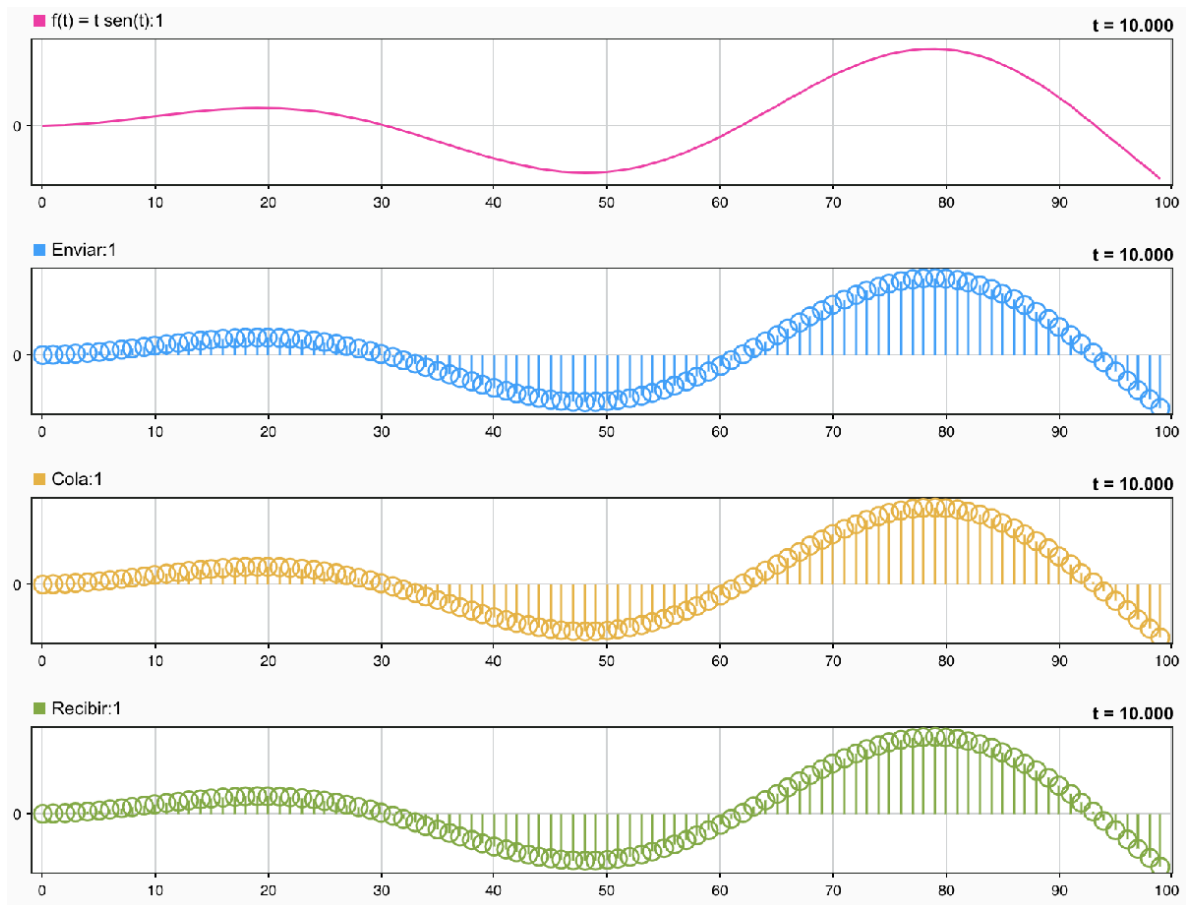
La configuración de bloque es la siguiente:

- From File - La fuente de la señal. En este caso se genera una señal con la función que se expresa en el siguiente código:

```
t = 0:0.1:10;
y = t.*sin(t);
funcion = y';
save('MS10Mensajes', 'funcion')
```

- El bloque lee el fichero creado con los datos de la función y lo envía al bloque Enviar, que lo discretiza. El tiempo de muestreo se establece a 0.1, aunque realmente serán los datos de la función los que manden sobre la resolución de los paquetes a enviar.
- Enviar - Convierte las señales de datos y envía los mensajes. El valor especificado para el parámetro Tiempo de muestreo del bloque Onda sinusoidal determina la velocidad a la que el bloque Enviar envía mensajes. Por lo tanto, el bloque Enviar envía un mensaje cada 0.1 de tiempo de simulación.
- Cola - Almacena los mensajes. Observe la línea de mensajes entre el bloque Enviar y el bloque Cola. La capacidad por defecto de la cola es de 16, lo que significa que el bloque Cola puede almacenar como máximo 16 mensajes. El comportamiento de ordenación de mensajes por defecto es LIFO, lo que significa que los mensajes entrantes se ordenan según la política de último en entrar, primero en salir. Por defecto, la casilla Sobrescribir el elemento más antiguo si la cola está llena está seleccionada. Cuando la cola está llena, un mensaje entrante sobrescribe el mensaje más antiguo de la cola.
- Recibir - Recibe mensajes y los convierte en datos de señal. En el bloque, el parámetro Tiempo de muestreo se establece en 0.1. El bloque Recibir recibe un mensaje cada 0.1 de tiempo de simulación.
- Visualización - Visualiza los mensajes recibidos por el bloque Recibir.

Si ejecutamos la simulación y abrimos la ventana de **Data inspector**, veremos los resultados:



Para un conocimiento más detallado de los patrones de paso de mensajes y sus diferentes combinaciones, ver [Modeling Message Communication Patterns with SimEvents](#)