

Sistema de noticias utilizando el patrón Publicador/Suscriptor (Publisher/Subscriber)

Definiciones

El patrón o modelo de comunicación publicador/suscriptor se compone de un conjunto de aplicaciones o procesos que envían y reciben mensajes (noticias, eventos, elementos de datos, etc.) utilizando algún sistema de comunicación, de una forma especial. En este proyecto los elementos de datos o eventos a enviar serán **noticias**.

De forma más específica, los programas a desarrollar deberán hacer lo siguiente:

Los publicadores (P): publican noticias, aunque no conocen quienes son los destinatarios de dichas noticias.

Los suscriptores (S): El o los consumidores de los eventos (en este caso noticias) reciben el nombre del *suscriptor(es)*. Mediante una suscripción, el suscriptor informa al **sistema de comunicación** las noticias que son de su interés. Los suscriptores pueden recibir noticias de publicadores diferentes, pero lo hacen es a través del proceso sistema de comunicación.

Sistema de Comunicación (SC): La tarea del sistema de comunicación es hacer coincidir las suscripciones con los eventos publicados; notifica a los suscriptores de la llegada de eventos de su interés y asegura la entrega correcta de todas las noticias que los publicadores han publicado.

En la figura 1 pueden observar cómo es este patrón en general: los círculos rojos son los procesos publicadores, que pueden publicar diferentes tipos de noticias (Arte, Política, etc.). Los suscriptores, círculos de color amarillo/naranja, se suscriben a un tipo de noticia específico, por ejemplo, el suscriptor 2 (s2) solo quiere recibir noticias de política, mientras que el suscriptor 3 decide recibir noticias de ciencia y espectáculos. La figura 2 muestra una fotografía del sistema en un instante de tiempo. En ese instante solo el publicador 3 está publicando una noticia de ciencia que el sistema de comunicación envía a los suscriptores 1, 3 y 4. Note que, en ese instante de tiempo, ya solo hay en el sistema dos publicadores.

El esquema de publicador/suscriptor es bastante similar a la forma de operar de Twitter.

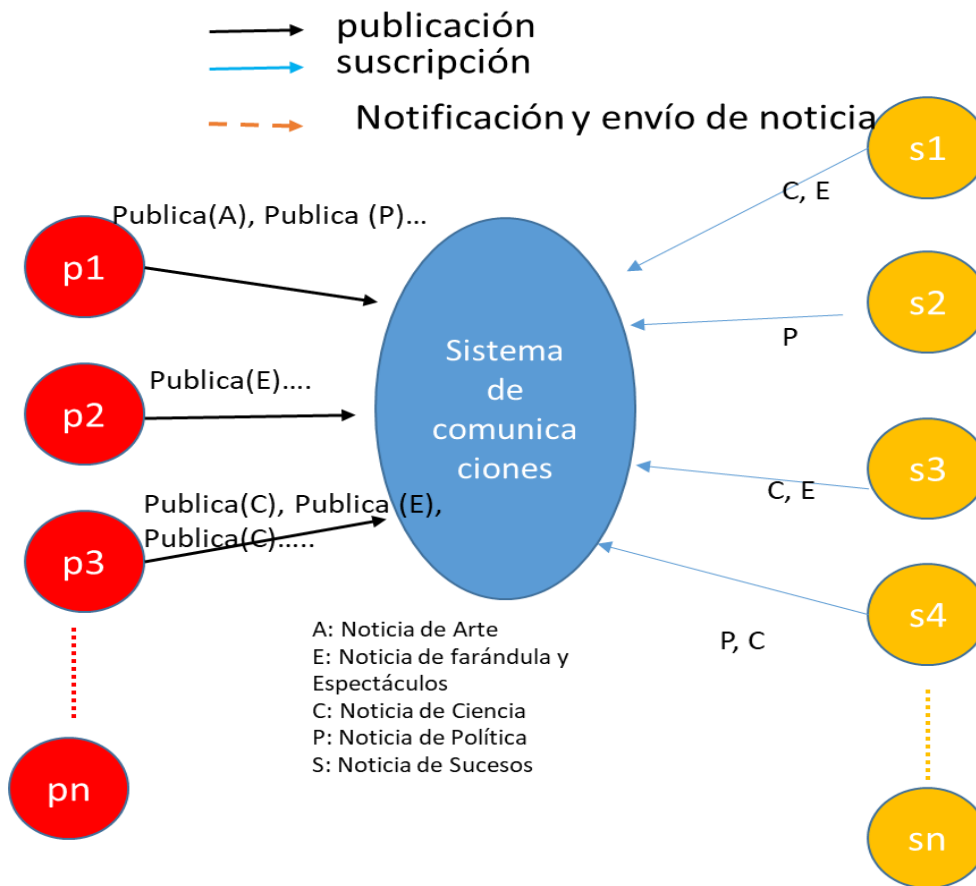


Figura 1: Sistema publicador/suscriptor a implementar en su fase de inicialización

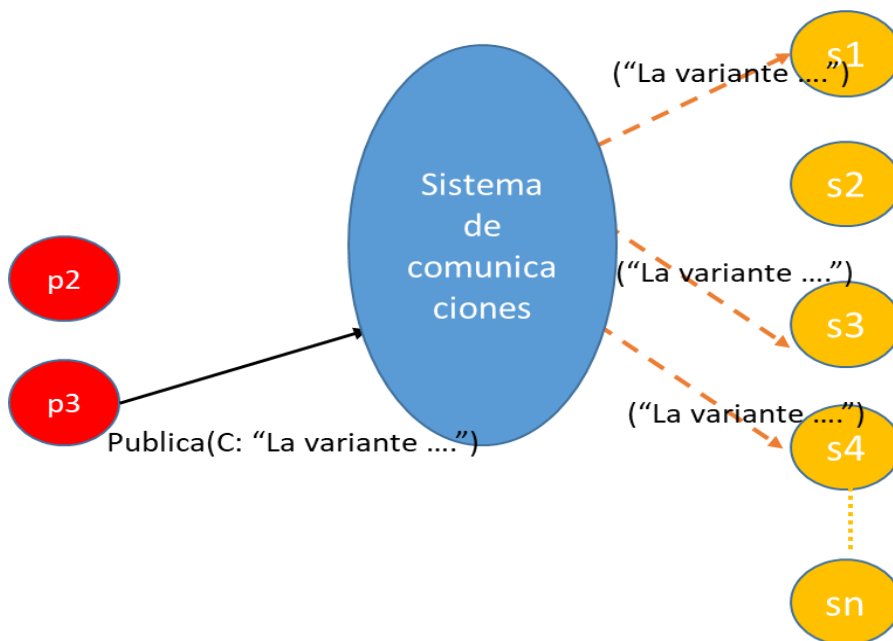


Figura 2: Fotografía del sistema en un instante determinado de su operación

Cómo debe implementar el proyecto

A continuación, se describe la funcionalidad de cada proceso, entradas y salidas (cuando aplique) y los mecanismos de comunicación que se van a usar. Hay aspectos que no están explícitos, porque forman parte de lo que deben diseñar e implementar los estudiantes, como parte del proyecto. **Recuerde que usted no puede modificar ninguna de las llamadas de los procesos (publicador, suscriptor y sistema) que aparecen en este enunciado, sin antes contar con el aval de su profesor.**

Publicador

Son los procesos que envían las noticias al sistema de comunicaciones. Los procesos se invocarán desde el shell de la siguiente forma:

\$ publicador -p pipePSC -f file -t timeN Donde:

Los **flags**, **-p**, **-f** y **-t** indican el tipo de argumento que viene a continuación, esto permite que se cambie el orden de los argumentos, siempre que este se anteceda con el flag correspondiente.

pipePSC: es el pipe nominal por donde los publicadores envían información al Sistema de Comunicación. *Este corresponde al pipe de color violeta en la figura 3*

file: es el nombre de un archivo donde están las diferentes noticias que el publicador enviará al Sistema de Comunicaciones. *Este archivo tiene un formato específico. Cada una de las líneas del archivo contendrá una noticia. La primera letra, que debe estar en mayúscula, debe representar alguno de los 5 tipos de noticia que se especifican en la página 4 de este enunciado y luego debe estar un símbolo de dos puntos y el resto de la línea corresponderá al texto de la noticia, terminando en un punto. Las líneas podrán tener espacios adicionales en blanco siempre que respeten los detalles que se dan aquí del formato. Si alguna noticia no está en el formato adecuado se debe indicar por la pantalla al iniciar el programa publicador.*

Por ejemplo, algunos ejemplos del formato de estos archivos pueden ser los siguientes:

Publicador 1	Publicador 2
A: El 25 se presenta la orquesta de cámara. P: Se aproximan las elecciones. P: Hay un nuevo candidato en Alianza Popular. E: Fue exitosa la presentación de Maluma.	A: XXXXXXXX. P: El presidente dio declaraciones. S: Fue robada por un motorizado. S: YYYYYYYY. C: El telescopio captó un nuevo exoplaneta.

En la tabla hay solo dos publicadores porque es un ejemplo, en las sustentaciones se podrá crear un número mayor de publicadores. La letra indica el tipo de noticia (solo las explícitas en la figura 1) y a continuación el texto.

timeN: es el tiempo en segundos entre el envío de una noticia y el envío de la siguiente al Sistema de Comunicación (esto se usa en la sustentación para que se entremezclen las publicaciones de varios publicadores)

Adicionalmente tenga en cuenta que:

- En cualquier momento, antes de que finalice el funcionamiento del sistema completo, se puede crear un proceso publicador.
- Los publicadores terminan luego de que el archivo que contiene las noticias llega a su fin y que transcurra el tiempo **timeN**.

Suscriptor

Puede imaginar que son procesos que se van a ejecutar en las pantallas de una cafetería. El dueño de la cafetería, por supuesto, sabe los intereses de sus clientes, por tanto, se suscribe a varios tópicos.

Cada proceso suscriptor se invocará de la siguiente manera:

\$ suscriptor -s pipeSSC

Donde:

pipeSSC: es el pipe de comunicación de los suscriptores con el Sistema de Comunicación. Es el pipe verde en la figura 3. Por ahí solo se envía los temas o tópicos a los cuales se suscribe un determinado suscriptor.

Una vez invocado el proceso, se le preguntará al usuario en la pantalla por los tópicos a los que desea suscribirse. Debe suscribirse al menos a un tópico. El suscriptor envía esta información al Sistema de Comunicaciones y luego se queda esperando por las noticias de interés. Cada vez que llegan una o varias noticias deben aparecer por la pantalla.

En cualquier momento de la corrida del sistema se puede crear o levantar un suscriptor.

Los suscriptores terminan cuando el Sistema de Comunicación les avisa (leer abajo la descripción de este componente)

Sistema de Comunicaciones (SC)

Es el proceso encargado de: aceptar las noticias de los publicadores, suscripciones de los clientes o suscriptores, hacer el match y enviar la información a quien corresponda. El proceso se va a invocar desde la consola de la siguiente forma:

\$ sistema -p pipePSC -s pipeSSC -t timeF

Donde:

pipePSC: nombre del pipe nominal de comunicación entre los publicadores y el SC. Este nombre debe coincidir con el nombre dado al invocar a los publicadores, ya que se trata del mismo pipe nominal.

pipeSSC: nombre del pipe nominal de comunicación entre los suscriptores y el SC. Este nombre debe coincidir con el nombre dado al invocar a los suscriptores, ya que se trata del mismo pipe nominal.

timeF: Una vez que han terminado todos los procesos publicadores el SC esperará este **timeF** en segundos (por si se crea otro publicador). Una vez concluido este tiempo, envía un mensaje o notificación a los suscriptores indicando el fin de la emisión de noticias y muestra un mensaje por su pantalla, indicando que ha terminado su trabajo.

Los procesos publicadores y suscriptores envían la información relacionada con noticias y suscripciones a través de 2 pipes nominales (uno para cada tipo de proceso), tal y como se muestra en la figura 3.

Ud. debe diseñar el resto de las funcionalidades del Sistema de Comunicaciones: estructuras de datos para guardar la información de publicadores y suscriptores, pipes adicionales para enviar las noticias, hilos o señales (si los considera necesario). Este diseño lo debe exponer en la primera sustentación del proyecto (también con ilustraciones, como se muestran en este enunciado).

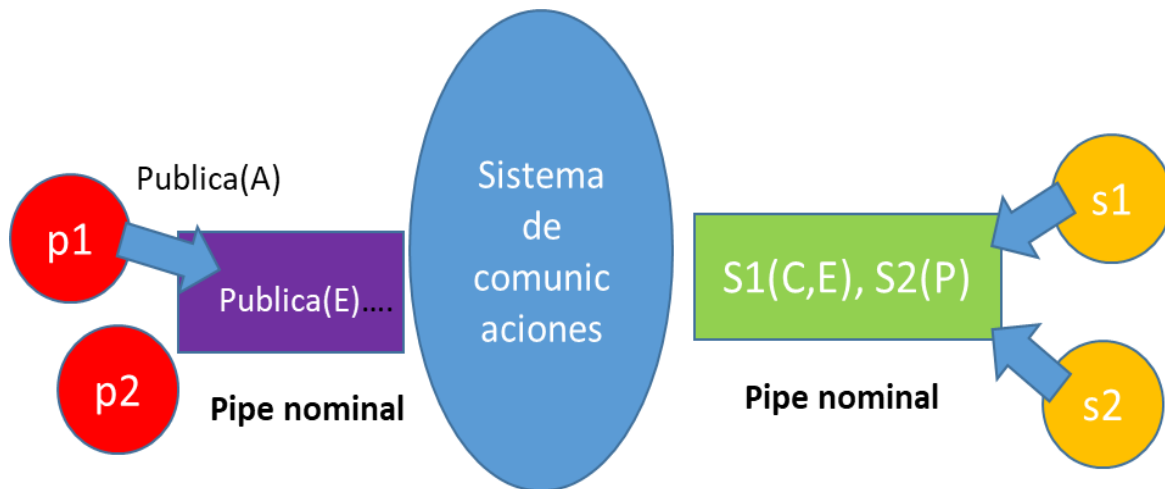


Figura 3: Comunicación con el sistema de comunicaciones

Noticias

Solo se manejarán 5 categorías de noticias:

A: Noticia de Arte

E: Noticia de Farándula y Espectáculos

C: Noticia de Ciencia

P: Noticia de Política

S: Noticia de Sucesos

Cada noticia publicada tendrá un máximo de 80 caracteres.

Lo que debe entregar en la primera parte

El proyecto se debe entregar en la semana 11: jueves 7

Valor de la entrega 10% (del 30% total del proyecto)

Debe sustentar:

1. El diseño completo del sistema:
 - Estructuras de datos para guardar la información en cada uno de los tres tipos de procesos.
 - Pipes adicionales a los especificados en el enunciado para enviar las noticias

- Formato de **todos** los mensajes que se envían entre todos los procesos
- ¿Cómo se hace el “match” de mensajes publicados con los suscriptores, internamente en el SC? ¿Cómo es el diseño de este proceso para recibir la comunicación de los dos tipos de procesos? Se debe entregar un diagrama de secuencia de los que se usan en UML.
- ¿Cómo deja de funcionar todo el sistema? ¿Cómo lo implementarán, con mensajes señales, etc.?
- Implementación de los tres programas, en la fase de suscripción de sus funciones, la publicación de noticias se evaluará en la segunda entrega.

Todo esto lo explicarán en una exposición de máximo 10 minutos, luego el profesor o la profesora realizará las preguntas y observaciones que considere pertinentes.

Deberán implementar y mostrar para esta primera entrega:

- Al menos un publicador enviando varios tipos de noticias.
- Al menos dos suscriptores “suscribiéndose” a un tipo de noticia
- El SC imprime lo publicado y decide qué noticia va para que suscriptor y lo avisa en su pantalla.

Los profesores enviarán previamente las noticias que van a publicar los publicadores y cuáles serán las suscripciones realizadas por el suscriptor.

La entrega que deben realizar en la plataforma del aula virtual consiste en: códigos fuente de las funcionalidades implementadas, un archivo *makefile* para generar los procesos ejecutables, un archivo **pdf** con la presentación utilizada en la sustentación. Todo debe venir en un archivo en formato tar.gz.

Entrega final del proyecto

La entrega final será en la semana 17 (24 de mayo para todos los grupos). La sustentación será los días 25 y 26. La entrega final contiene: el proyecto implementado en su totalidad, los códigos fuente del proyecto y un archivo makefile; todos los archivos deben estar en un archivo en formato tar.gz. Sin excepción, los proyectos que no se entreguen el día 24 de mayo no serán calificados (si tienen algún problema deben hablar previamente con el o la profesora responsable del curso).

Recuerde que los profesores entregarán los casos de prueba para el día de la sustentación.

Observaciones adicionales

- El proyecto lo deben realizar en grupos de como máximo 3 estudiantes.
- Los estudiantes pueden intercambiar ideas sobre el proyecto, pero al momento de codificar cada grupo debe trabajar de forma autónoma, sin intercambiar códigos con otros grupos. Cualquier sospecha de copia o fraude académico será elevada a las instancias correspondientes dentro de la Facultad de Ingeniería.
- El proyecto lo deben hacer en lenguaje ANSI C, es decir que funcione en el lenguaje C disponible en cocalc.com, ya que allí será corregido el proyecto.

