**Implementación del algoritmo Causal Multi-Group Algorithm (CMGA)**

Dr. Saúl Pomares Hernández

El proyecto consiste en implementar al algoritmo que se muestra abajo. El algoritmo se puede programar en alguna de las siguientes plataformas:

En Java e IP Multicast.

En JSDT

En la primera plataforma el algoritmo solo puede funcionar en redes LAN, ya que IP Multicast no está habilitado de manera global.

En la segunda plataforma puede funcionar sobre redes WAN pero no utiliza una transmisión multicast, más bien utiliza una transmisión multipunto. JSDT es un middleware creado para el desarrollo de aplicaciones cooperativas distribuidas, su página web es: <http://java.sun.com/products/java-media/jsdt/index.jsp>

El algoritmo puede ser llevado a cabo en dos maneras: la primera es estática lo que significa que los grupos de procesos deben estar construidos antes de que la transmisión de mensajes inicie, y una vez que la transmisión haya comenzado ninguno de los procesos puede “salir” del o los grupos de procesos a los que pertenezca. En la segunda manera se dice que es dinámica ya que se permite que los procesos entren y salgan de los grupos de procesos a cualquier momento.

Se invita al estudiante a tomar las opciones que desee para desarrollar al algoritmo lo cual será considerado al momento de la asignación de la calificación.

**I. Initially**,

1. *VT*(*pi*)[*k*] = 0 ∀ *k*:1…Σ *g*∈*G* |*g*|
2. *CIi* ← ∅
3. *He* ← ∅

**II. For each *message* diffused by** *pi* **into** **group** ***d***

1. *VT*(*pi*)[*i*] = *VT*(*pi*)[ *i*] +1
2. **for all :** *ci*=(*k*, *x, c , ch\_dests*)
3. **if  then**
4. ****
5. ****
6. **endif**
7. **if** ∅ **then**
8. ****
9. **endif**
10. **endfor**
11. *e=( i, t=VT(pi*)**[***i*], *d*, *message ,* *He*)
12. **send**(*e*) into **the group** ***d***
13. ****

**III. For each *e* =received by** *pj*

To impose a causal delivery

*Condition of Multi-group delivery*

1. **If not** 
2. ) **then**
3. **wait**
4. **else**
5. ***Delivery*(***message***)**
6. 
7. **if**  **then**
8. \ 
9. **endif**
10. 
11. **for all** 
12. **if** (*c* ∈ *CHj*) **then**
13. **if**  **then**/\**x* ≤ *y*\*/
14. **if** *x* < *y* **then */\**** don´t do anything \*/
15. **endif**
16. **if** *x* = *y***then**
17. **if** (*c* ≠ *d*) **then**
18. *MAJ*(, *d* )
19. **else** /\* *c* = *d* \*/
20. ****
21. **endif**
22. **endif**
23. **endif**
24. **else** /\* *c* ∉ *CHj* \*/
25. **if**  **then**
26. **if** *x* < *y* **then */\**** don´t do anything\*/
27. **endif**
28. **if** *x* = *y***then**
29. *MAJ*(, *d* )
30. **endif**
31. **if** *x* > *y* **then**
32. *VT*(*pj*)[*l*] = *x*
33. \ 
34. 
35. **endif**
36. **else /\***  **\*/**
37. **if** (*VT*(*pj*)[*l*] < *x*) **then**
38. *VT*(*pj*)[*l*] = *x*
39. 
40. **endif**
41. **endif**
42. **endif**
43. **endfor**
44. **endif**

**IV. Updating**

1. *MAJ*(*cik,x,c*, *d*)
2. **if** (*c* ≠ *d*) **then**
3. ←\ *d*
4. **if**  **then**
5. ****
6. **endif**
7. **else** /\* *c* = *d \*/*
8. ****
9. **endif**

Interfase de un Pizarrón Electrónico para un Ambiente de Educación a Distancia

Dr. Saúl Pomares Hernández

El proyecto consiste en diseñar y desarrollar una interfaz interactiva de un pizarrón electrónico sobre una ambiente virtual de educación a distancia. El diseño debe considerar una funcionalidad adecuada a dicho ambiente pero considerando las limitaciones de ancho de banda que existe actualmente en las Instituciones de Educación Superior de nuestro país.

Fases del proyecto:

1. Determinar e investigar los comandos recibidos/enviados del/al dispositivo.
2. Diseño de la funcionalidad de la interfase.
3. Implementación y pruebas. (transferencia de la información)

Control Maestro para la Reproducción de

Flujos Continuos

Dr. Saúl Pomares Hernández

El proyecto consiste en diseñar y desarrollar una interfaz interactiva en la cual se pueda controlar la reproducción de 2 o más flujos de manera simultánea. El control debe considerar las diferentes funciones de reproducción como son: stop, pause, rewind, forward, play.

Consideraciones:

* El proyecto debe considerar la reproducción de manera acoplada así como des-aclopada que involucra la reproducción por separado de cada flujo.
* La sincronización al momento de pasar de una reproducción des-aclopada a aclopada.

Herramienta a utilizar: JMF (Java Media Franwork).