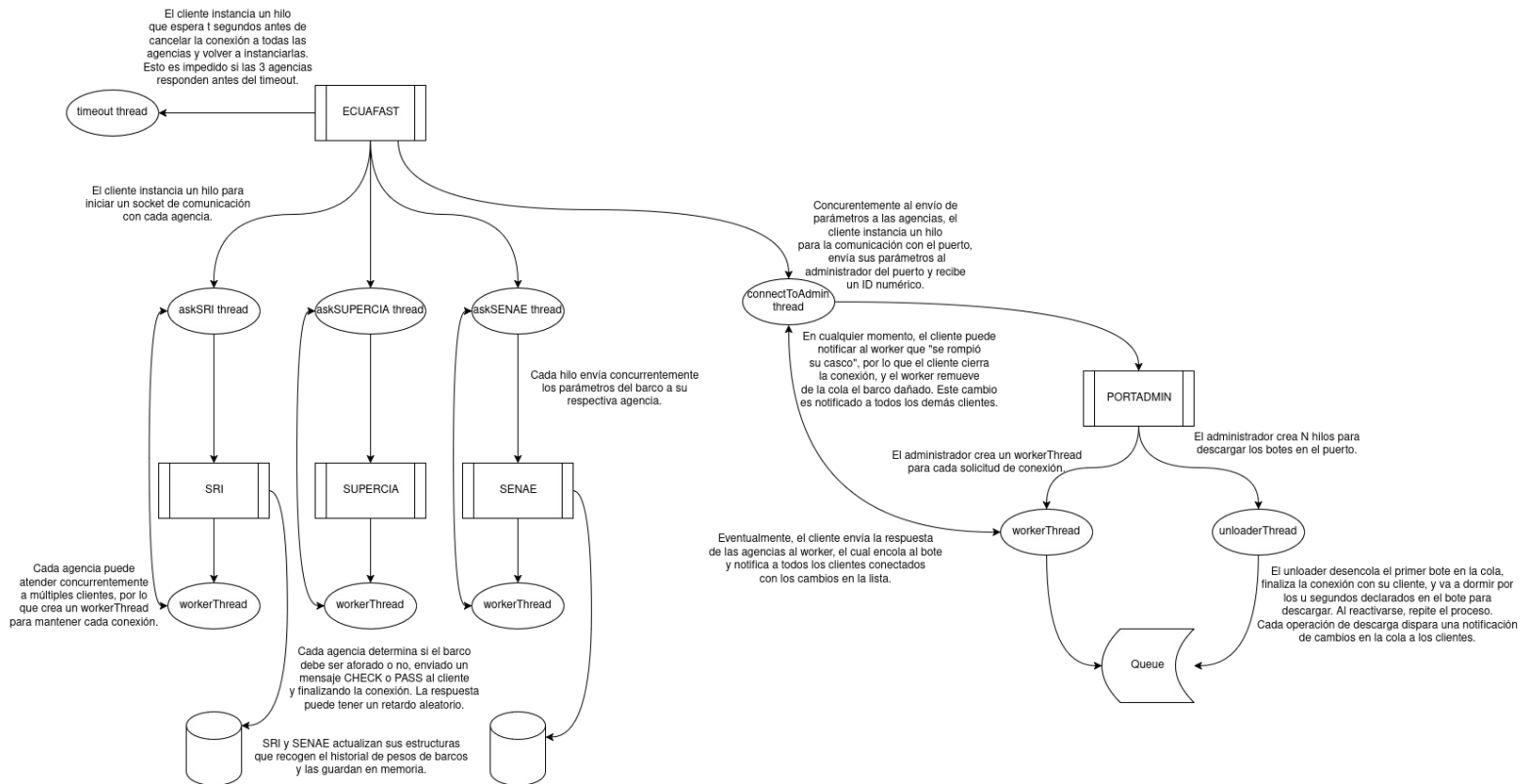


# PROYECTO FINAL SSOO – DOCUMENTACIÓN

Nombre: José Julio Suárez



## Detalles de implementación:

- Para notificar a las agencias si deben guardar o descartar el barco, ecuafast ejecuta la función commit() o rollback(), las cuales envían un string con la orden a cada agencia.
- SRI maneja la persistencia de sus datos con un memory-mapped file, truncado al tamaño de la estructura WeightBuffer. Esta estructura tiene un arreglo con índice circular de 20 espacios para almacenar el peso de los últimos 20 barcos.
- En SRI, la complejidad del cálculo del peso promedio es  $O(1)$ , pues al último promedio se le resta la diferencia entre el peso del nuevo barco y el peso a ser sobrescrito, ponderado a 20 elementos.
- SENA maneja la persistencia de sus datos con 5 memory-mapped files. Esto es así debido a que la estructura principal SENABuffer contiene punteros a 2 heaps, y cada heap contiene un puntero al arreglo de sus datos. Por lo tanto, cada estructura referenciada debe ser resuelta con su propio archivo, o las referencias serían inválidas.
- En SENA, la complejidad del cálculo del 3er cuartil es  $O(\log n)$ . Esto se logra al dividir la distribución entre un maxHeap que contiene al primer 75% de la distribución, y un minHeap que contiene al 25% superior.

El algoritmo de inserción funciona tal que:

- Si no hay elementos, agregar el nuevo elemento al maxHeap.
- Si los hay, comparar el nuevo elemento con la raíz del minHeap.
- Si el nuevo elemento es mayor, agregarlo al minHeap. Caso contrario, agregarlo al maxHeap.
- Incrementar el tamaño total de la muestra en 1.

Una vez insertado, hay que mantener la proporción de 75% - 25% de la distribución entre los heaps:

- Mientras el tamaño del maxHeap sea mayor al tamaño de la distribución  $\times 0.75$ , remover la raíz del maxHeap e insertarla en el minHeap.
- Mientras el tamaño del minHeap sea mayor al tamaño de la distribución  $\times 0.25$ , remover la raíz del minHeap e insertarla en el maxHeap.

Así, el tamaño del maxHeap será siempre el piso de  $0.75(n+1)$ , lo que permitirá encontrar al 3er cuartil entre la raíz del maxHeap y la raíz del minHeap.

- En PORTADMIN, la cola (lista enlazada) y sus atributos son protegidos mediante una versión modificada a la solución del bounded-buffer. En el problema original, el buffer es finito, por lo que su estructura implementa un semáforo contador “empty” inicializado con el tamaño del buffer. Sin embargo, una lista enlazada crece indiscriminadamente, por lo que no se necesita llevar cuenta de espacios disponibles.