



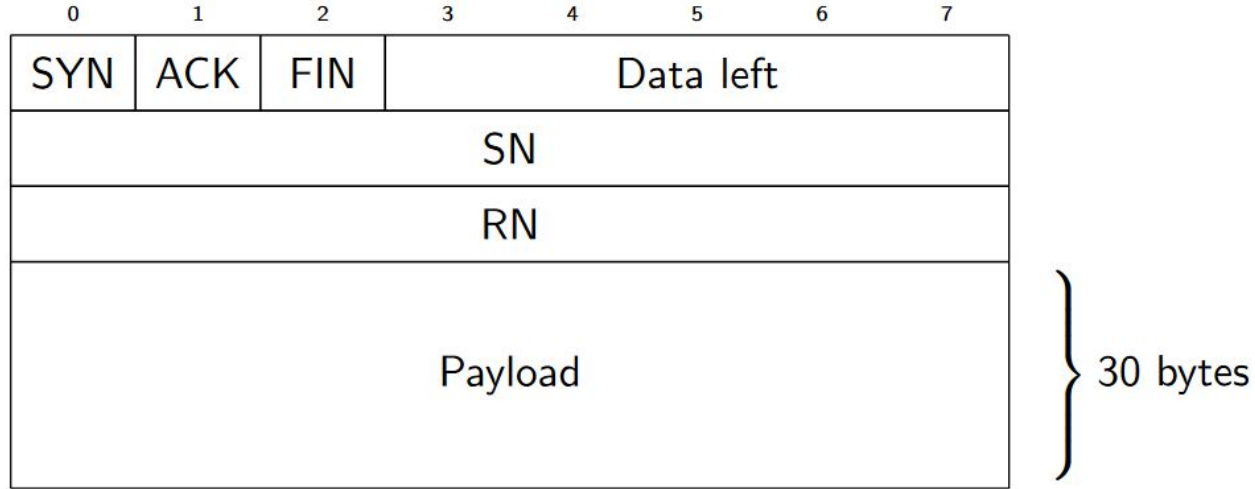
# Proyecto Fase II

Caracoracle

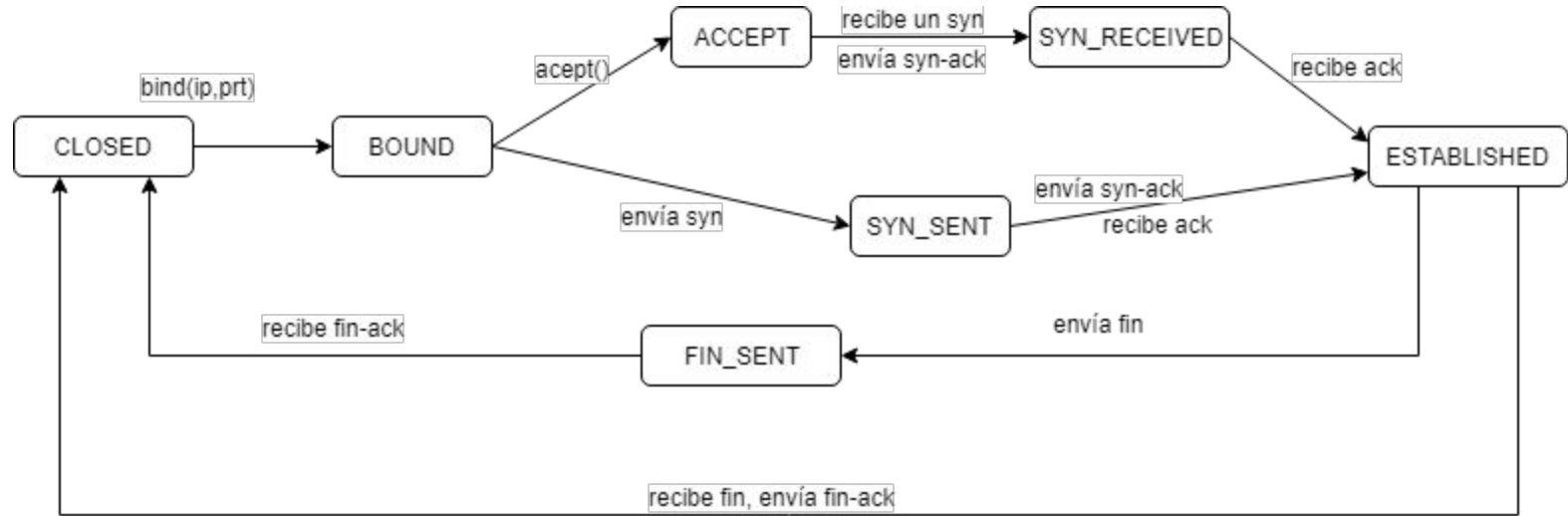
Fabián Leandro  
Carlos Azofeifa



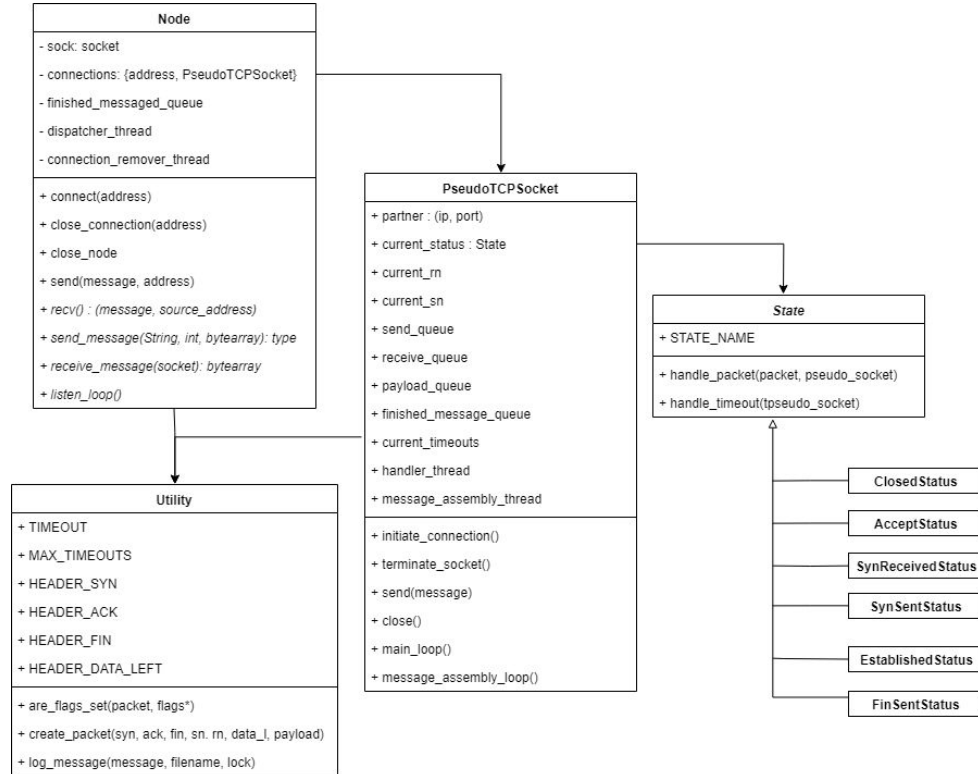
# Estructura de los paquetes



# Máquina de estados



# Diseño general



# Flujo: Node

- Inicia un thread que funciona como dispatcher, escuchando en el socket y enviando los mensajes a la conexión asociada
- Cuando recibe un SYN aloca los recursos necesarios para abrir la conexión, lo cual consiste en instanciar un PseudoTCPSocket
- Los métodos connect(), send, close() deben indicar la conexión pertinente. recv() devuelve el mensaje y la dirección origen.
- Otra thread espera cierres de conexiones asíncronicos para eliminar su entrada en la tabla.
- Al ejecutar close\_node() se inicia el cierre de todas las conexiones, se espera a que terminen, y se detienen las dos threads mencionadas.

# Flujo: Node

- Dos threads:
  - Uno consume los mensajes que el nodo envía a la conexión, enviando y recibiendo payloads y ACKs. Inserta los payloads en una cola, indicando con un ítem especial cuándo termina un mensaje.
  - Otro consume la cola de payloads, ensamblando el mensaje y lo pasa a la capa superior cuando termina.
- `send()` fragmenta el mensaje y llena una cola de salida, consumida por el thread principal para enviar el mensaje.
- Cuando el nodo finaliza el proceso de cierre, notifica al nodo para que elimine su entrada en la tabla de conexiones.

# Notas

- Los nodos siempre hacen `bind()` y utilizan solo un socket
  - Problemas de eficiencia
- Con el comando `ping`, se midió un RTT de  $>1\text{ms}$  entre dos computadoras conectadas por ethernet. Se eligió un timeout de 5ms.
- Como se usan solo 5 bits para codificar el tamaño del payload, su tamaño máximos es 30 bytes.