

Universidad Técnica Federico Santa María  
Departamento de Informática  
Héctor Labraña  
David Rojas  
Fecha: Noviembre 2018

Campus San Joaquín  
Sistemas Distribuidos  
201373535-5  
201573502-6

## Protocolo de comunicación para torres de control aéreas (TAP)

### Resumen

Se realiza un protocolo de comunicación para torres de control aéreas en específico para coordinar el paso de mensajes entre la torre de control con las solicitudes de despegue y aterrizaje de un determinado avión y la solicitud de datos por parte de una pantalla de información.

El protocolo de comunicación TAP considera una arquitectura distribuida y con paso de mensajes mediante llamadas remotas a procedimientos (RPC).

### Tabla de Contenidos

1. Descripción del Problema .....	2
1.1 Requisitos del Sistema.....	2
1.2 Restricciones.....	3
2. Flujo de comunicación .....	3
2.1 Diagrama de Flujo.....	3
2.2 Variables.....	4
3. Protocolo de Comunicación.....	5
3.1 Diagrama de Secuencia Torre - Avión.....	5
2.5 Diagrama de Secuencia Torre - Pantalla de Información.....	6

## 1. Descripción del Problema

El protocolo debe trabajar en una aplicación distribuida, específicamente emular de forma simplificada la comunicación que se realiza entre el controlador de la torre de control y el piloto del avión, a través de mensajes usando RPC. Para esto se propone la utilización de múltiples servidores, correspondientes a una torre de control por aeropuerto. Cada torre de control opera de manera independiente a las demás y tendrá que controlar los vuelos en su propio aeropuerto, además de pantallas que actuarán como clientes que consultan información.

### 1.1 Requisitos del Sistema

#### 1.1.1 Torre de Control:

- Indicar la altura a la que el avión debe esperar antes de aterrizar.
- Indicar la pista de aterrizaje de un vuelo entrante.
- Indicar la pista de despegue de un vuelo saliente, en que lugar en la cola de espera se encuentra el avión e indicando cuál avión está antes (Indicar aerolínea y número de vuelo).
- Antes de finalizar la comunicación con el avión, comunicar la dirección IP de la torre de control destino al avión saliente y la altura a la que debe despegar para evitar colisiones.

#### 1.1.2 Aviones:

- Pedir instrucciones de aterrizaje: la torre de control debe indicar las instrucciones necesarias mencionadas anteriormente, si no quedan pistas libres deberá encolarse.
- Pedir instrucciones de despegue: la torre de control debe indicar si puede despegar, esto dependerá de si el peso máximo que soporta el avión es excedido o si existe déficit de combustible, el avión esperará la instrucciones necesarias y además la torre de control actual indicará la dirección IP de la torre de destino para que el avión pueda iniciar una nueva conexión al llegar a destino.
- Indicar la cantidad de pasajeros a bordo estando en la gate de abordaje, lugar desde el cual puede cargar combustible al mismo tiempo de ser necesario.

#### 1.1.3 Pantallas de Información:

- Listar los aviones que acaban de llegar (Arrivals).
- Listar los aviones listos para salir (Departures).

## 1.2 Restricciones

- Para comunicar cada instancia de los componentes debe utilizar los métodos y procedimientos entregados por gRPC, haciendo uso de protocol buffers.
- Cada componente (Torre de control, Avión y Pantalla de información) debe estar programado en un lenguaje distinto al de los demás, pero debe estar dentro de la lista de lenguajes soportados por gRPC.
- Cada avión puede comunicarse sólo con una torre de control a la vez.
- Cada instancia de los elementos debe poder correr sobre nodos diferentes, por ejemplo: los aviones y pantallas podrían correr sobre dispositivos móviles (si utilizan lenguajes como Android-Java y Objective-C) y las torres de control en distintos servidores.

## 2. Flujo de comunicación

### 2.1 Diagrama de Flujo

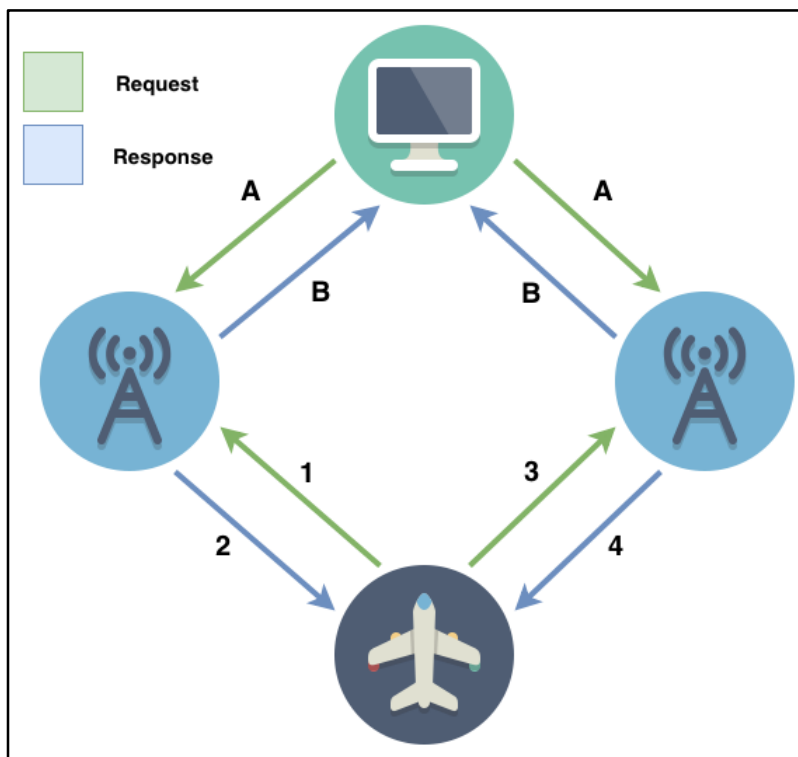


Figura 1: Flujo de comunicación del protocolo TAP. La secuencia de números representa el flujo de comunicación de un avión con una torre de origen y aterrizaje, en cambio, la secuencia alfabética representa el flujo de comunicación de una pantalla de información con dos torres de control, observar que el flujo de comunicación es el mismo en cada par torre-pantalla.

## 2.2 Variables

Para poder generar el flujo de comunicación en una llamada remota a procedimientos (RPC), es necesario, definir los atributos de cada llamada remota en base al flujo descrito en la figura 1. Para ello se utiliza protocol buffers como IDL para serializar los datos en el sistema distribuido.

En particular, los atributos necesarios para poder realizar el paso de mensajes y construir el archivo .proto se definen en el siguiente apartado.

```
[1]    string destino;
        string aerolinea;
        string codigo;

[2]    bool estado;
        string Ipdestino;
        string mensaje;
        int32 pista;
        int32 altura;

[3]    string aerolinea;
        string codigo;
        int32 PesoMax;
        int32 CapTanque;
        string IptorreActual;

[4]    bool estado;
        string mensaje;
        int32 pista;
        int32 puerta;
        int32 altura;

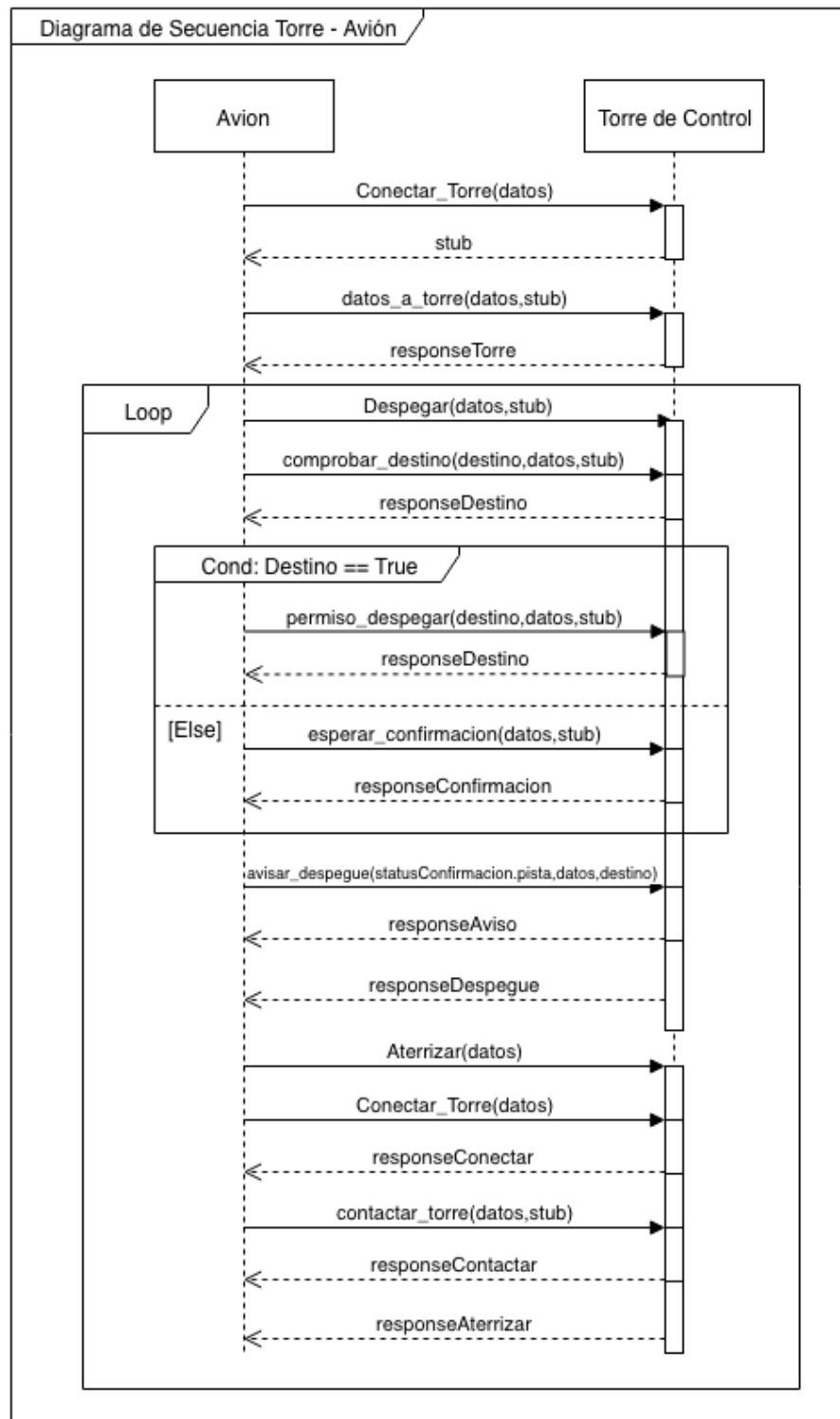
[A]    string aeropuerto;

[B]    repeated ObjectAvion; -> ObjectAvion{
                                string codigo;
                                string lugar;
                                int32 pista;
                                string hora; }
```

**IMPORTANTE:** Se debe compilar el archivo proto según las instrucciones del lenguaje de programación utilizado en cada elemento del sistema (avion, pantalla de información y torre de control).

### 3. Protocolo de Comunicación

#### 3.1 Diagrama de Secuencia Torre - Avión



## 3.2 Diagrama de Secuencia Torre - Pantalla de Información

