

Progetto Dronazon

Sistemi Distribuiti e Pervasivi – Appello Settembre 2021

Marco Cutecchia

Matricola: 983828

Info varie

- Ogni 2 secondi, dopo essere entrati nel sistema, tutti i droni provano a mandare un messaggio di heartbeat al master per essere sicuri che sia ancora vivo
- Tutte le comunicazioni gRpc hanno una deadline di 2 secondi, unica eccezione è la richiesta del lock per l'area di ricarica che non ha una deadline
- Gli endpoint RESTful del server admin sono:

```
GET /drones
POST /drones/{droneId}
DELETE /drones/{droneId}
GET /stats/last/{howMany}
GET /stats/after/{ts1}/before/{ts2}
GET /stats/average-deliveries/after/{ts1}/before/{ts2}
GET /stats/average-kms-travelled/after/{ts1}/before/{ts2}
POST /stats/
```

Fase di avvio

Server Admin



Drone in entrata



Droni nel sistema



POST /drones/4
Ciao, sono <ID=4, ip=localhost, port=9004>



Fase di avvio

Server Admin



Drone in entrata



Droni nel sistema



POST /drones/4
Ciao, sono <ID=4, ip=localhost, port=9004>

Benvenuto. Ti trovi a (x=2, y=7)
Droni nella rete: [
 <ID=4, ip=localhost, port=9004>,
 <ID=1, ip=localhost, port=9001>,
 <ID=2, ip=localhost, port=9002>,
 <ID=3, ip=localhost, port=9003>
]

Fase di avvio

Server Admin



Benvenuto. Ti trovi a (x=2, y=7). Droni nella rete: [...]

Drone in entrata



Droni nel sistema



Sono <ID=4, ip=localhost, port=9004>

Sono <ID=4, ip=localhost, port=9004>

Sono <ID=4, ip=localhost, port=9004>

Fase di avvio

Server Admin

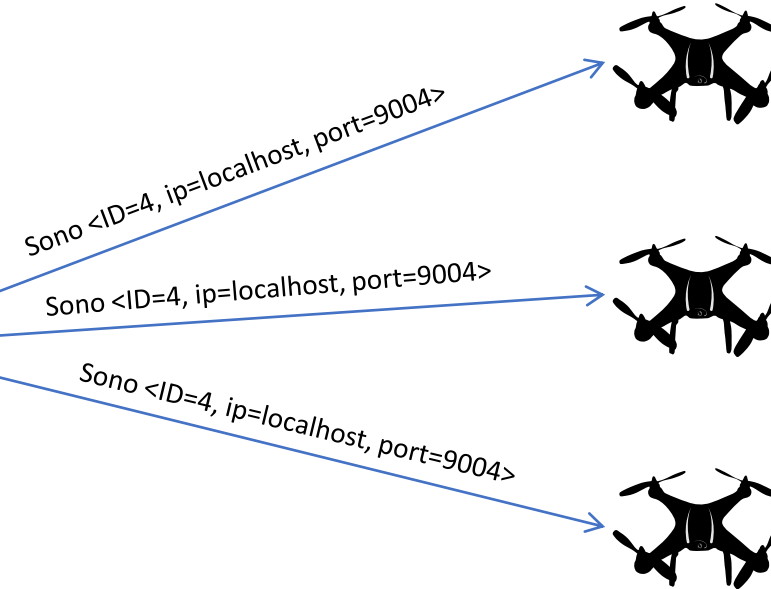


Drone in entrata



Droni nel sistema

```
message DroneIdentifierPacket {  
    int32 id = 1;  
    string address = 2;  
    int32 port = 3;  
}  
  
message CityPointPacket {  
    int32 x = 1;  
    int32 y = 2;  
}  
  
message DroneJoinMessage {  
    DroneIdentifierPacket sender = 1;  
    CityPointPacket starting_position = 2;  
}
```



Fase di avvio – Master esistente

Server Admin



Benvenuto. Ti trovi a (x=2, y=7). Droni nella rete: [...]



Drone in entrata



Droni nel sistema



- Tutti i droni che hanno risposto vengono salvati.
- Se un drone ha risposto dicendo di essere il master viene segnato

Sono <ID=1, ip=..., port=9001, NON MASTER>

Sono <ID=2, ip=..., port=9002, NON MASTER>

Sono <ID=3, ip=..., port=9004, MASTER>

Fase di avvio – Master esistente

Server Admin

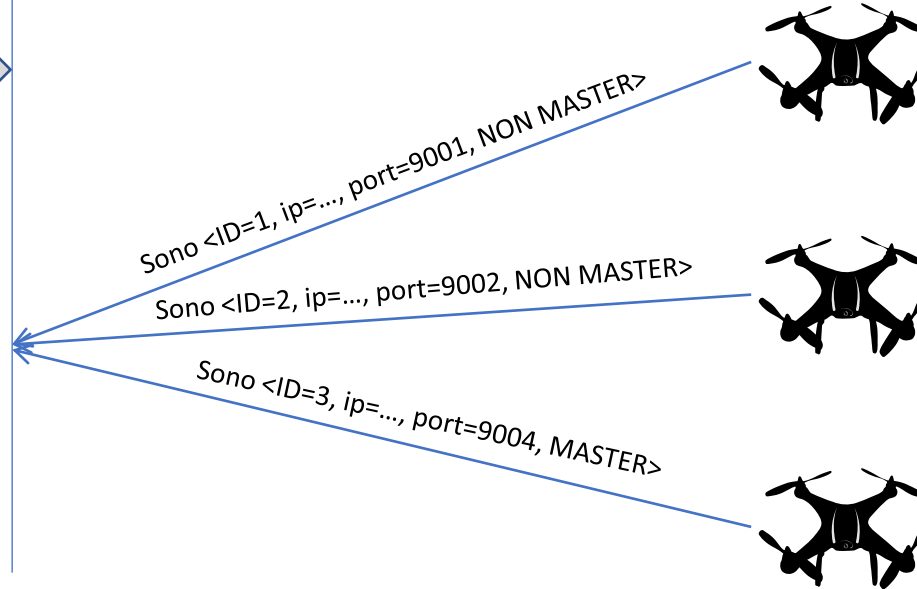


```
message DroneIdentifierPacket {  
    int32 id = 1;  
    string address = 2;  
    int32 port = 3;  
}  
  
message DroneJoinResponse {  
    DroneIdentifierPacket sender = 1;  
    bool is_master = 2;  
}
```

Drone in entrata



Droni nel sistema



Fase di avvio – Master mancante

Server Admin



Benvenuto. Ti trovi a $(x=2, y=7)$. Droni nella rete: [...]



Drone in entrata



Droni nel sistema

- Tutti i droni che hanno risposto vengono salvati
- Il drone va avanti senza conoscere un master
- Inizierà un elezione solo quando avrà bisogno di comunicare con il master

Sono <ID=1, ip=..., port=9001, NON MASTER>

Sono <ID=2, ip=..., port=9002, NON MASTER>

Sono <ID=3, ip=..., port=9004, NON MASTER>



Elezione – Chang and Roberts

- Ogni drone mantiene una lista contenente tutti gli altri droni nella rete. Questa lista è utilizzata per capire chi è il prossimo drone nell'anello
- Ogni volta che una comunicazione con il drone master fallisce (o non è possibile) una elezione viene avviata

Due tipi di messaggi vengono scambiati durante un'elezione: ELECTION ed ELECTED

```
message ElectionMessage {  
    DroneIdentifierPacket candidate_leader  
= 1;  
    int32  
candidate_leader_battery_percentage = 2;  
}
```

```
message ElectedMessage {  
    DroneIdentifierPacket new_leader = 1;  
}
```

Elezione – Casi limite durante l'uscita di un drone

- *Durante l'invio dei messaggi ELECTION o ELECTED il drone futuro leader abbandona l'anello, portando gli altri droni a scambiarsi all'infinito il messaggio senza mai finire*
- *Il drone futuro master riceve il suo messaggio ELECTION ma abbandona il sistema prima di mandare il messaggio ELECTED, lasciando in stallo tutti gli altri droni*

SOLUZIONE: Se un drone ha ricevuto un messaggio ELECTION allora non può completare lo spegnimento fino alla fine dell'elezione.

Fine dell'elezione vuol dire:

- Per i droni master quando hanno ricevuto indietro tutti i messaggi ELECTED che hanno inviato
- Per i droni follower quando hanno ricevuto ed inoltrato un messaggio ELECTED

Elezione – Un caso limite durante l'entrata di un drone

Un drone entra durante una elezione, il futuro master risponde di non essere lui il master perché non ha ancora ricevuto indietro il suo messaggio ELECTION. Il drone entrante perde il messaggio ELECTED perché il drone precedente nell'anello ha processato prima il messaggio ELECTED del suo messaggio.

SOLUZIONE:

Se un drone già master riceve un messaggio ELECTION allora ferma il messaggio ed invia un messaggio ELECTED con se stesso per sicurezza.

I droni che ricevono un messaggio ELECTED conoscendo già il master inoltrano senza far nulla.

Assegnamento degli ordini

Synchronized

Coda ordini pendenti

1	2	3	4
---	---	---	---

1. Assegno ordini pendenti con i droni disponibili
2. Mando in parallelo queste richieste

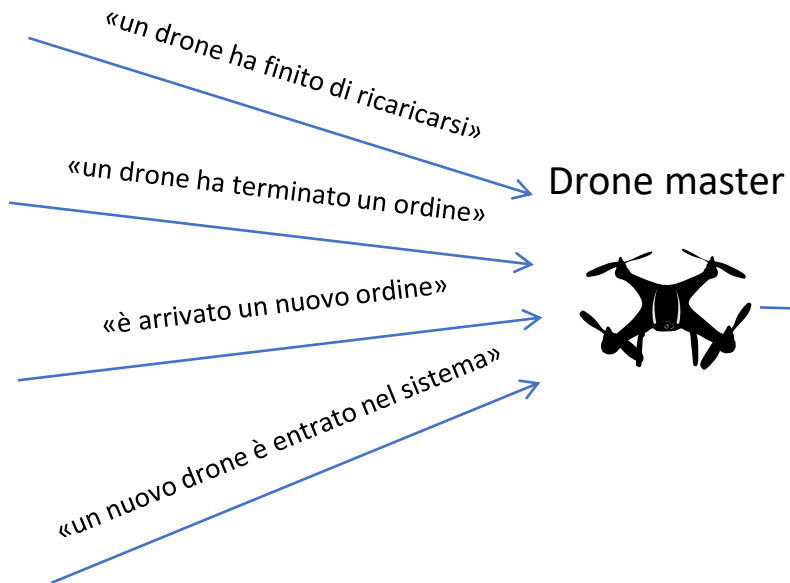
Non è detto che assegno tutti gli ordini!

Rimettere un ordine nella coda riattiva questo processo

Se il drone accetta la consegna:
Mi segno che il drone è impegnato e non proverò più ad assegnargli ordini fino alla conferma di consegna

Se il drone rifiuta la consegna:
Rimetto l'ordine nella coda. Mi segno che il drone non è disponibile per le consegne fino a suo aggiornamento.

Se il drone non è raggiungibile:
Rimetto l'ordine nella coda. Cancello il drone dalla lista dei droni nel sistema



Assegnamento degli ordini

```
message OrderPacket {  
    int32 id = 1;  
    CityPointPacket starting_point = 2;  
    CityPointPacket delivery_point = 3;  
}  
  
message AssignOrderMessage {  
    OrderPacket order = 1;  
}  
  
message AssignOrderResponse {  
    bool accepted = 1;  
}  
  
message CompletedDeliveryMessage {  
    uint64 timestamp = 1;  
    DroneIdentifierPacket sender = 2;  
    OrderPacket completed_order = 3;  
  
    double travelled_kms = 4;  
    repeated double pollution = 5;  
    int32 battery_percentage = 6;  
}
```

Synchronized

Coda ordini pendenti



1. Assegno ordini pendenti con i droni disponibili
2. Mando in parallelo queste richieste

Non è detto che assegno tutti gli ordini!

Rimettere un ordine nella coda
riattiva questo processo

Se il drone accetta la consegna:

Mi segno che il drone è impegnato e non proverò più ad assegnargli ordini fino alla conferma di consegna

Se il drone rifiuta la consegna:

Rimetto l'ordine nella coda. Mi segno che il drone non è disponibile per le consegne fino a suo aggiornamento.

Se il drone non è raggiungibile:

Rimetto l'ordine nella coda. Cancello il drone dalla lista dei droni nel sistema



Mutua esclusione per la ricarica

- Quando si richiede il lock vengono aperte più chiamate sincrone (bloccanti) verso tutti gli altri droni su diversi thread e viene fatta una join su tutti i thread
- I droni che ricevono la richiesta del lock rispondono immediatamente se vogliono concedere il lock, oppure chiamano *wait()* se hanno la precedenza ed useranno *notifyAll()* quando non sono più interessati al lock
- Unica chiamata gRpc senza alcuna deadline: se il drone è irraggiungibile è come se avesse concesso il lock

Mutua esclusione per la ricarica

- Un drone a cui viene concesso il lock inizierà a rifiutare tutti i futuri ordini
- Non è detto che inizi a ricaricarsi subito! Potrebbe dover completare un ordine assegnato mentre aspettava il lock
- Inizierà ricaricarsi subito dopo aver segnalato al master che ha terminato la consegna
 - Questo potrebbe implicare una o più elezioni nel frattempo
- Una volta completata la ricarica segnala al master di aver finito di ricaricarsi e ricomincia ad accettare gli ordini

Fase di shutdown

- 1) Aspetta che il drone non stia consegnando ordini
- 2) Ferma l'ascolto di ordini dal broker MQTT
- 3) Aspetta che tutti gli ordini siano stati assegnati ed accettati
- 4) Aspetta che il drone non stia consegnando un ordine
- 5) Aspetta che non sia in corso un'elezione
- 6) Spegni il middleware gRpc
- 7) Manda le statistiche al server admin
- 8) Avvisa il server admin che questo drone sta uscendo
- 9) Spegni il programma

Fase di shutdown

CASO LIMITE: Il master vuole spegnersi, ci sono ordini pendenti ma nessun drone disponibile per consegnarli

In questo caso, se il master ha abbastanza batteria, proverà lui stesso a consegnarli. Altrimenti resterà in attesa che altri droni entrino nel sistema.

CASO LIMITE: Viene richiesto di ricaricarsi dopo che è stato avviato lo shutdown del drone

In questo caso la richiesta di ricaricarsi è ignorata