

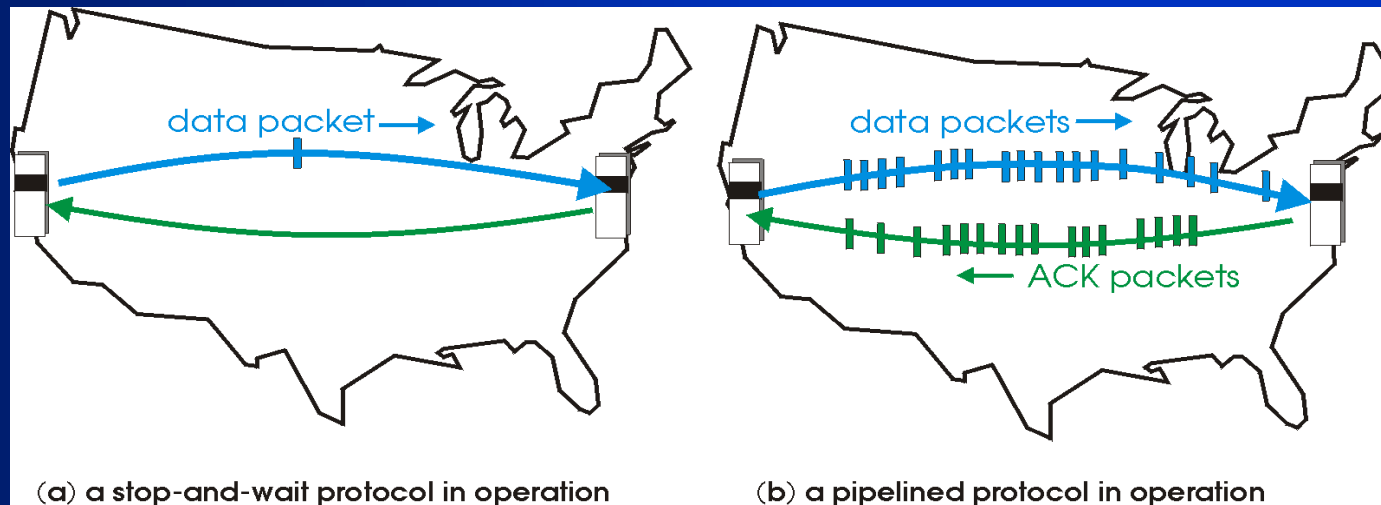
# **Tecniche ARQ**

## **(protocolli a finestra)**

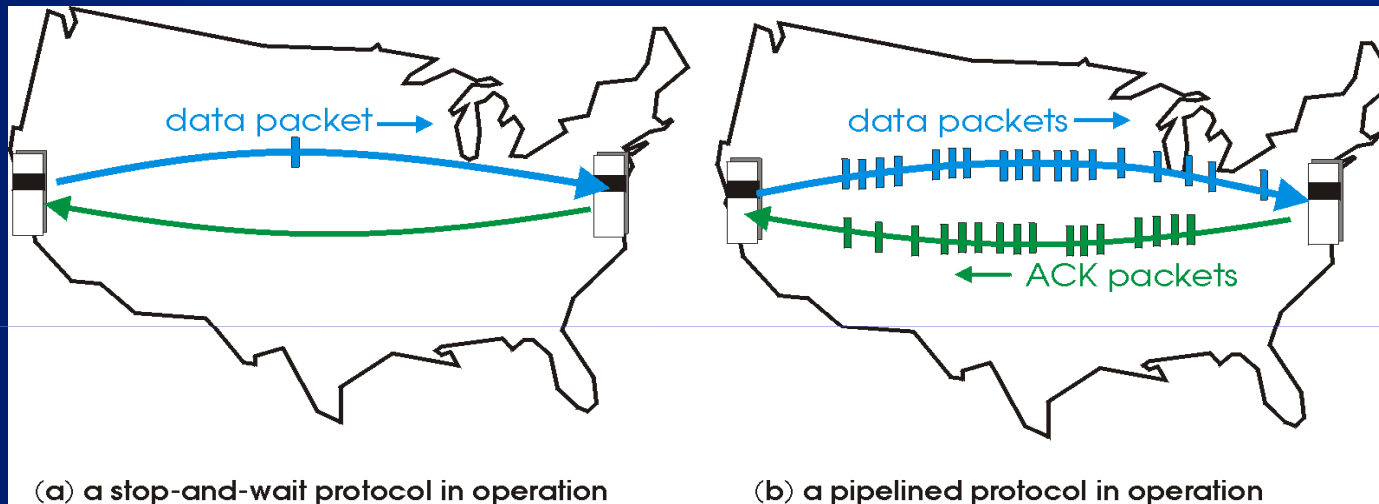
# Protocolli “pipeline”

**Pipelining:** al sender è consentito l'invio di pacchetti multipli senza che debba aspettare i riscontri

- i pacchetti in transito è come se riempissero un canale (pipeline)
- il range di numeri di sequenza aumenta
- necessità di buffering al sender e/o al receiver



# Protocolli “pipeline”



Due tipi di protocolli pipeline: *go-Back-N*,  
*selective repeat*

**Permettere la trasmissione di  
più di una PDU prima di fermarsi  
in attesa delle conferme  
migliora le prestazioni**

**Go back N**

# Go back N

## il trasmettitore

- invia fino ad N PDU facendo di ognuna una copia
- attiva un orologio per ogni PDU
- si pone in attesa delle conferme di ricezione (ACK)
- se scade un timeout prima dell'arrivo delle conferme, ripete la trasmissione di tutte le PDU non ancora confermate

# Go back N

**il ricevitore, quando riceve una PDU**

- **controlla la correttezza della PDU**
- **controlla il numero di sequenza**
- **se la PDU contiene il primo numero di sequenza non ancora ricevuto, viene consegnata ai livelli superiori, altrimenti viene scartata**
- **invia la conferma di ricezione**

# ACK Cumulativi

**La semantica associata al pacchetto di riscontro può essere:**

- **ACK individuale: si notifica la corretta ricezione di un pacchetto particolare**
- **ACK cumulativo: si notifica la corretta ricezione di tutti i pacchetti con numero di sequenza inferiore a quello specificato nell'ACK**

# Piggybacking

**Gli ACK possono anche essere inseriti negli header dei pacchetti che viaggiano in direzione opposta**



**SN: numero di sequenza della trama**

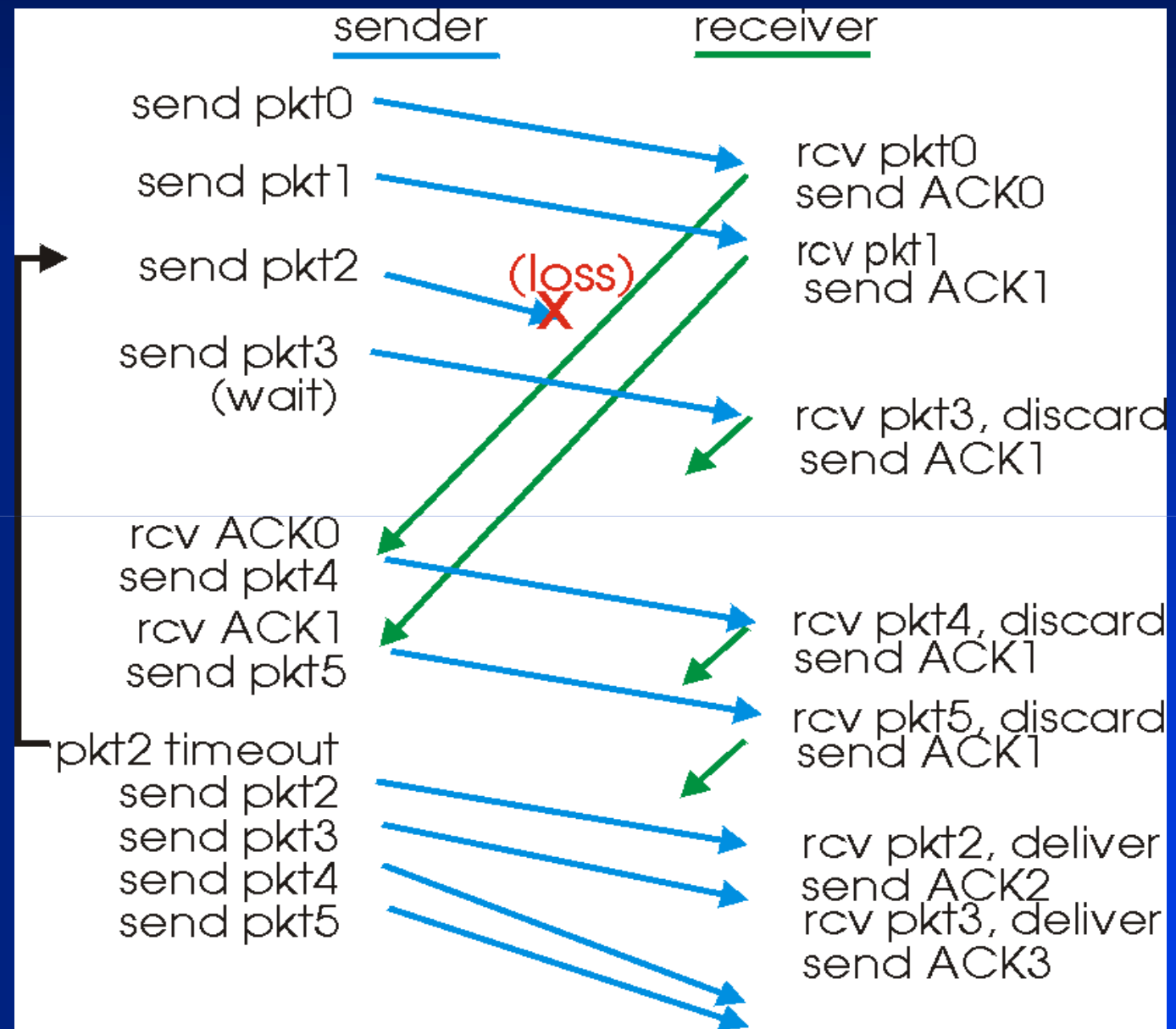
**RN: trama attesa in direzione opposta, ovvero  
riscontro cumulativo delle trame fino a RN-1**



# GBN

WT=4

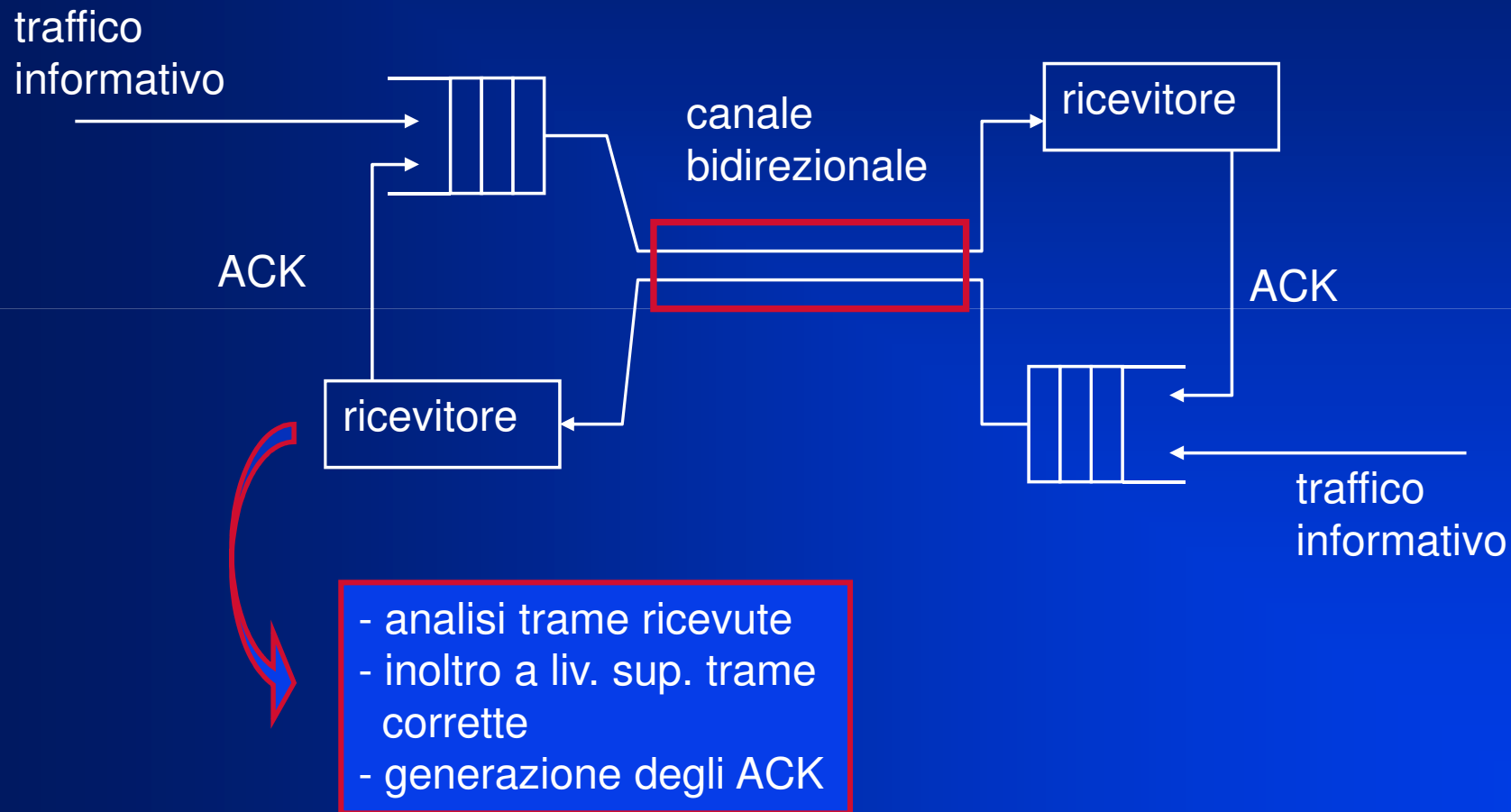
I pacchetti  
3, 4 e 5 sono  
scartati in  
attesa di  
ricevere il  
pacchetto 2  
corretto.



**Perché scartare i pacchetti ricevuti  
non in ordine, anche se corretti?**

**Per semplificare il ricevitore e ridurre la  
necessità di buffering**

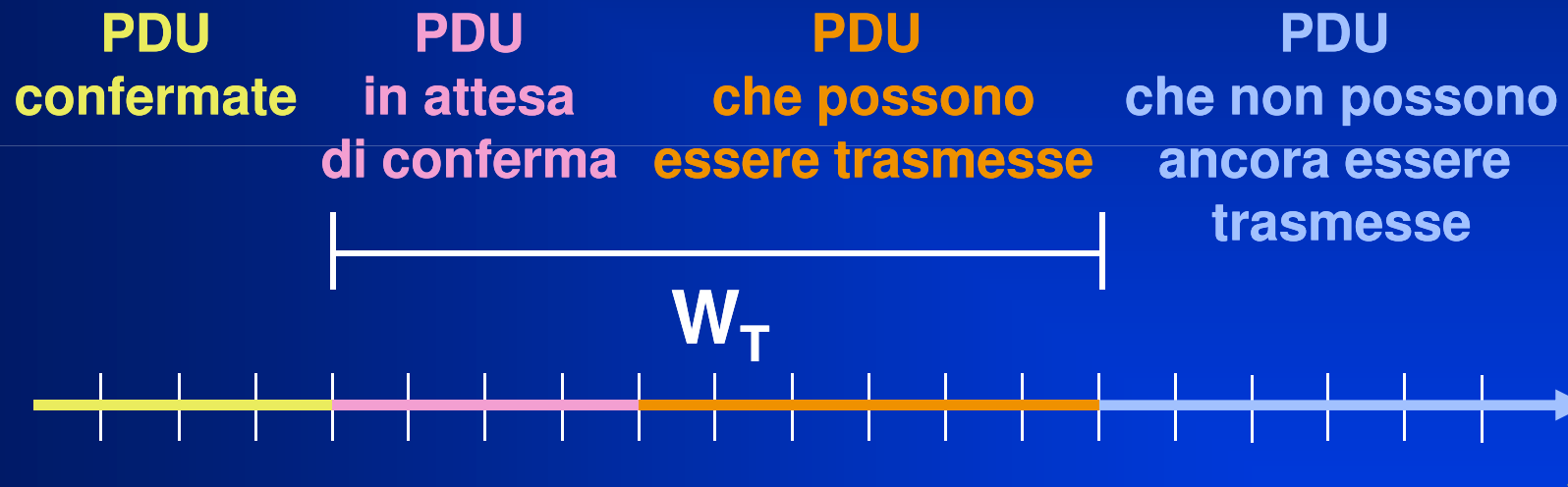
# Funzionamento full-duplex



# Protocolli a finestra

**La gamma dei numeri di sequenza permessi per i pacchetti trasmessi, ma non ancora riscontrati, può essere considerata una “finestra” di dimensione  $N$  che si apre su una gamma di numeri di sequenza.**

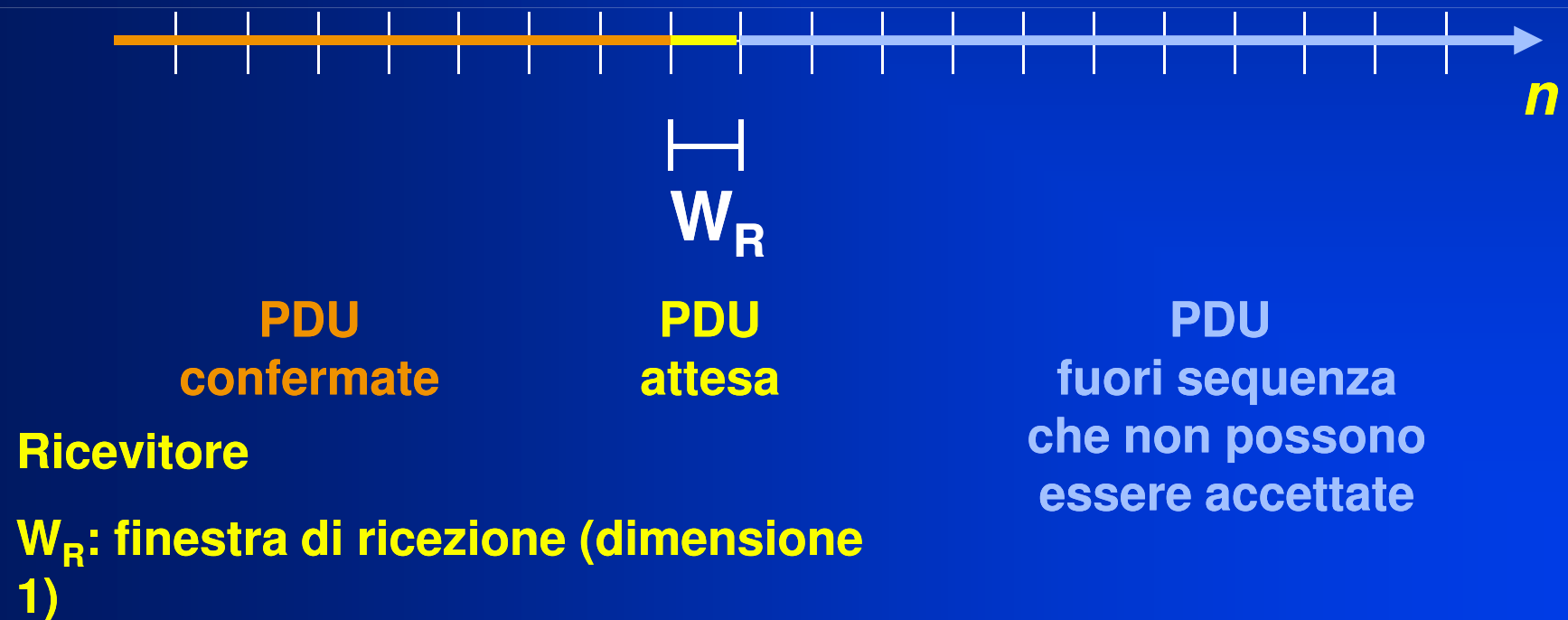
# Protocolli a finestra



**Trasmittitore**

$W_T$ : finestra di trasmissione (dimensione  $N$ )

# Protocolli a finestra



# Protocolli a finestra

**La finestra scorre in avanti sullo spazio dei numeri di sequenza ogni volta che si riceve un riscontro.**

**Si parla di protocolli a finestra scorrevole (sliding window).**

**WT avanza di una posizione ogni volta che riceve una PDU di tipo ACK relativa alla posizione più bassa della finestra.**

**Se il trasmettitore riceve un ACK-n relativo a una PDU-n che non è nella posizione più bassa, può avanzare WT fino alla posizione n, ignorando il fatto che alcuni ACK non siano ancora stati ricevuti.**

**WR avanza di una posizione ogni volta che riceve una PDU dati in sequenza e**



**La numerazione delle PDU è ciclica**

- **k bit di numerazione**
- **numerazione modulo  $2^k$**

**3 bit di  
numerazione**



3 bit di  
numerazione

$$W_R = 1$$

$$W_T = 3$$

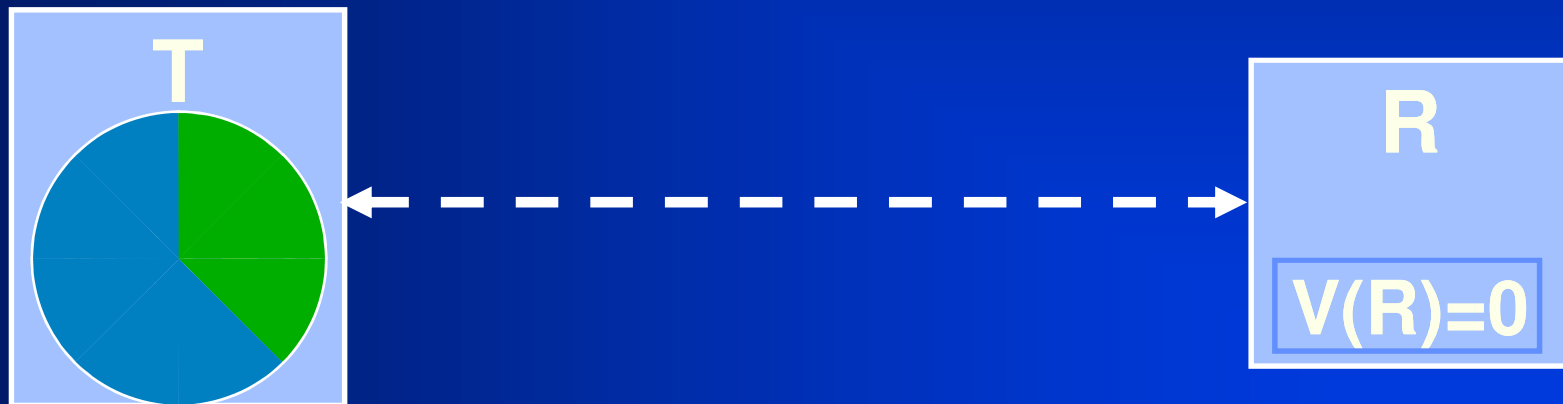


## **Go back N**

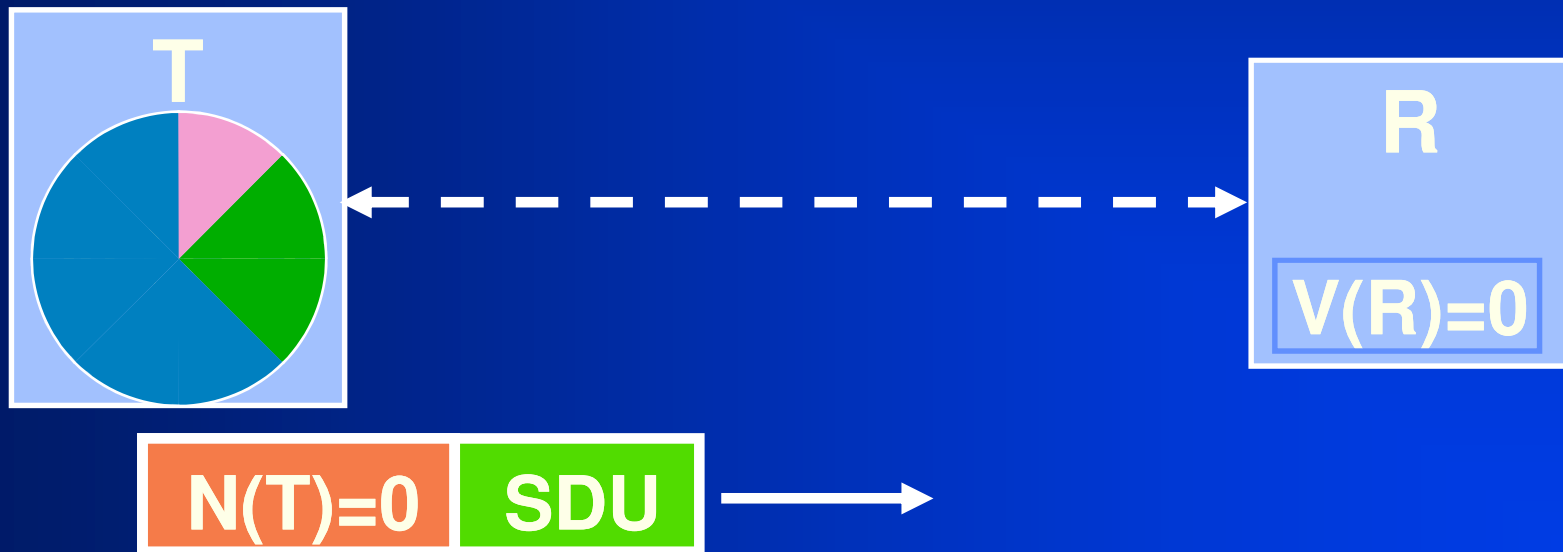
**il trasmettitore e` significativamente  
piu` complesso rispetto al caso  
dello Stop and wait**

- memoria**
- orologi**
- algoritmi**

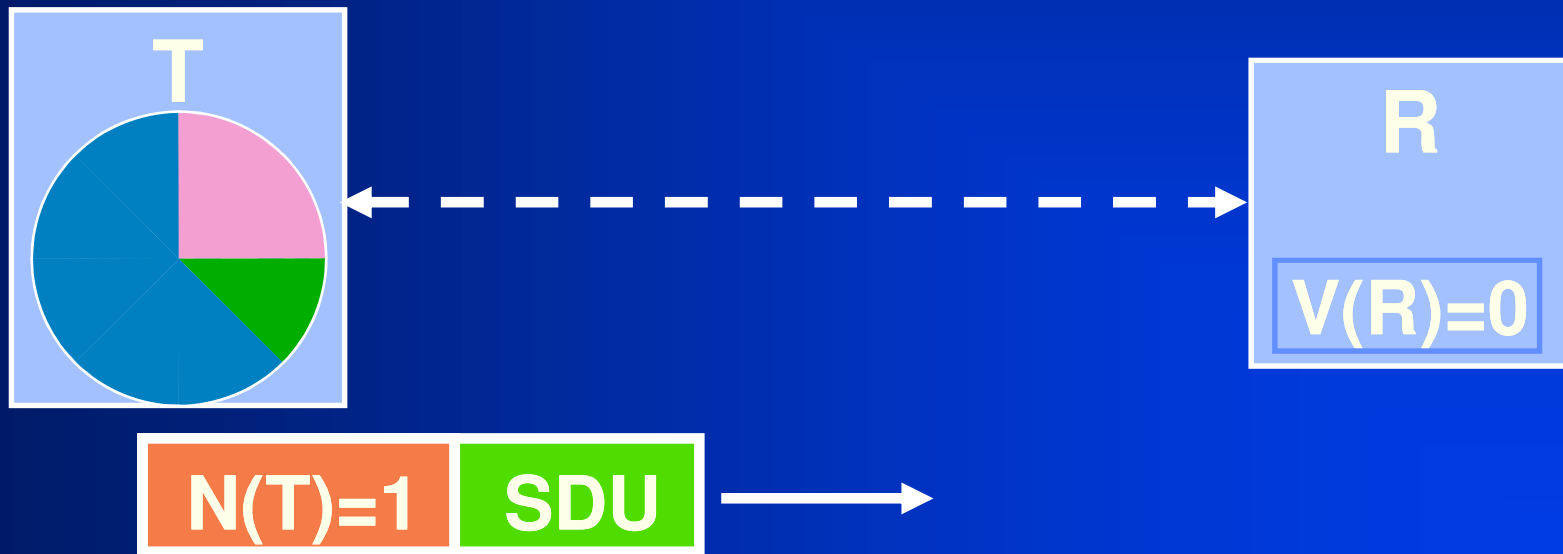
# Inizializzazione



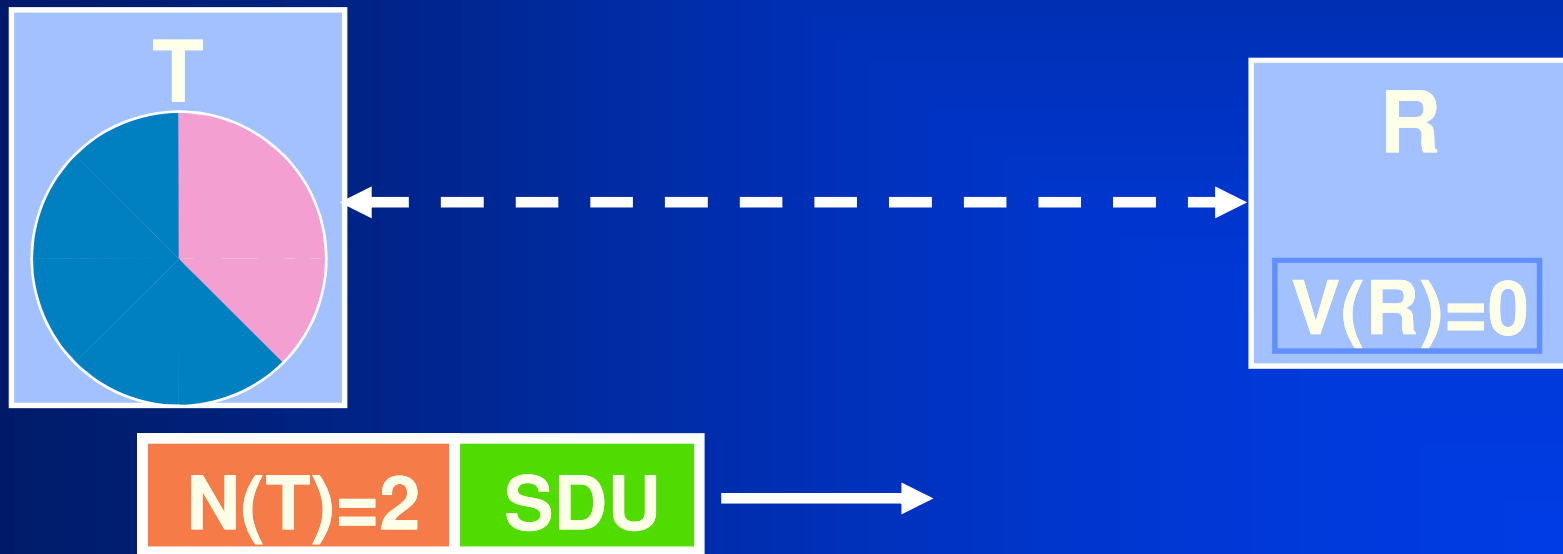
## Trasmissione PDU



## Trasmissione PDU

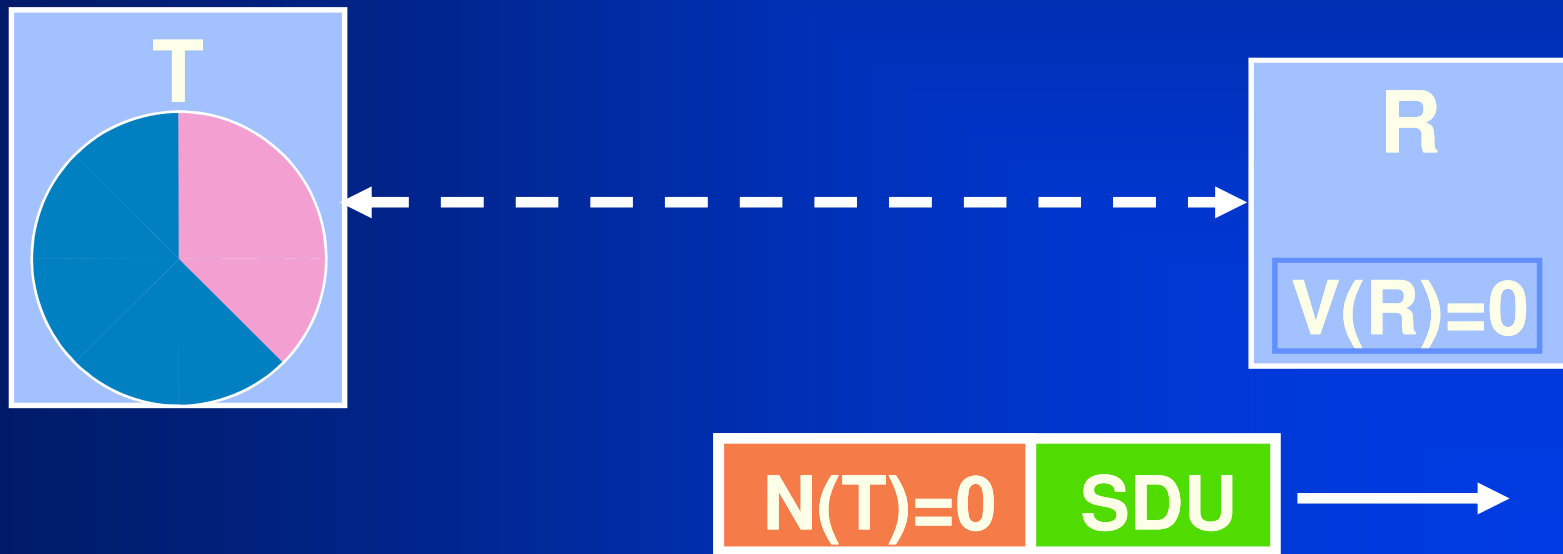


## Trasmissione PDU

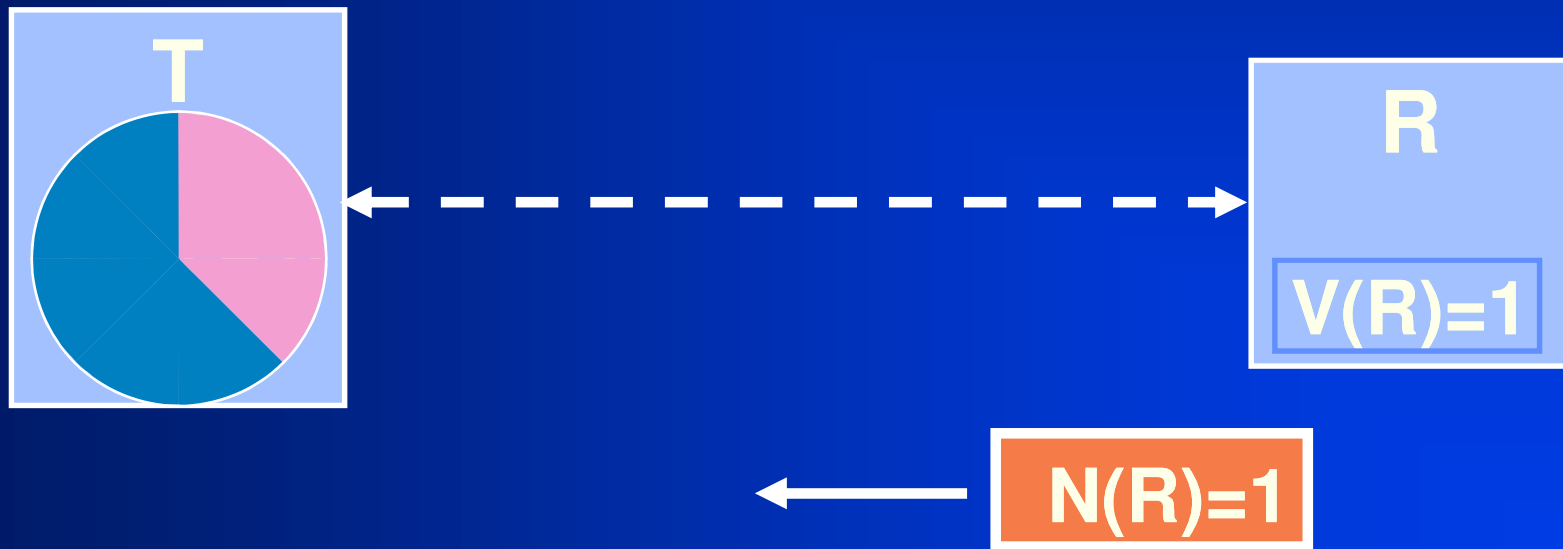




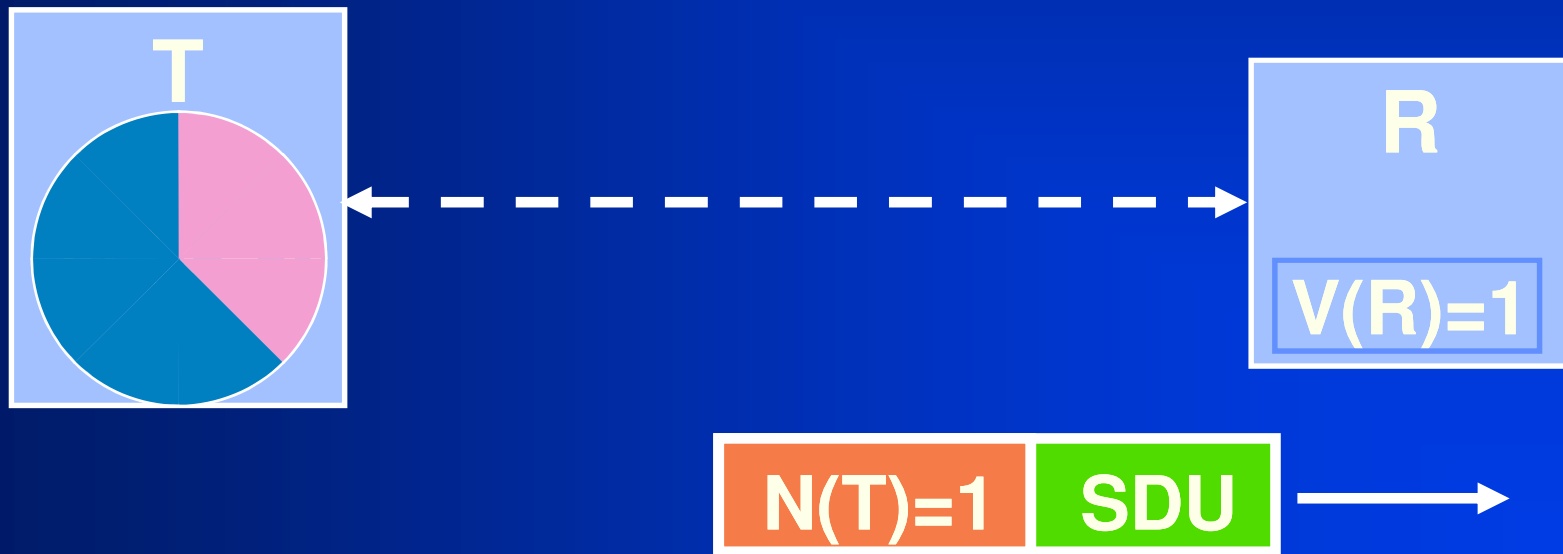
## Ricezione PDU



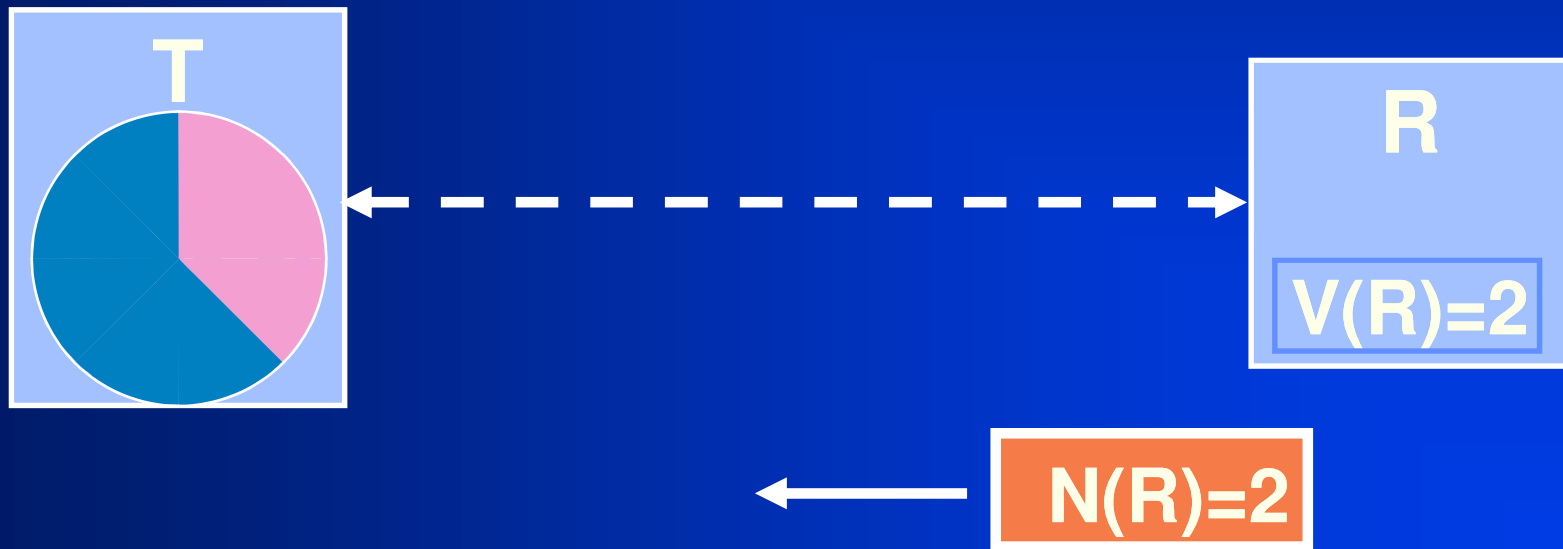
## Trasmissione ACK



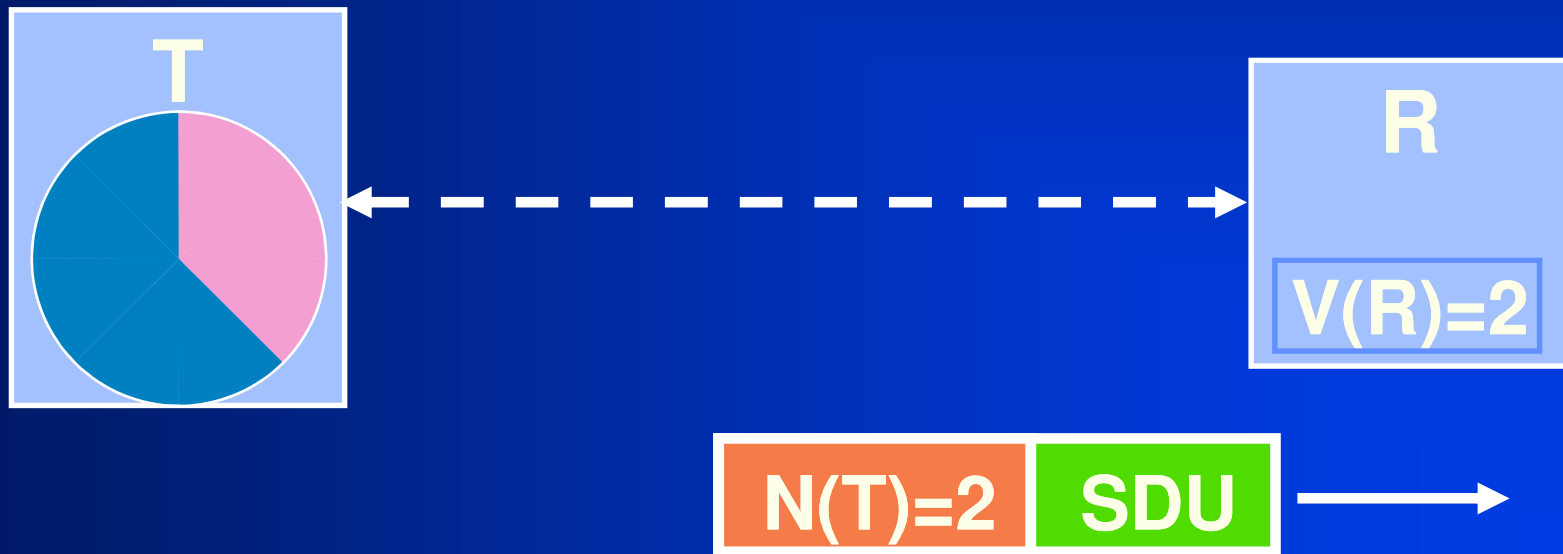
## Ricezione PDU



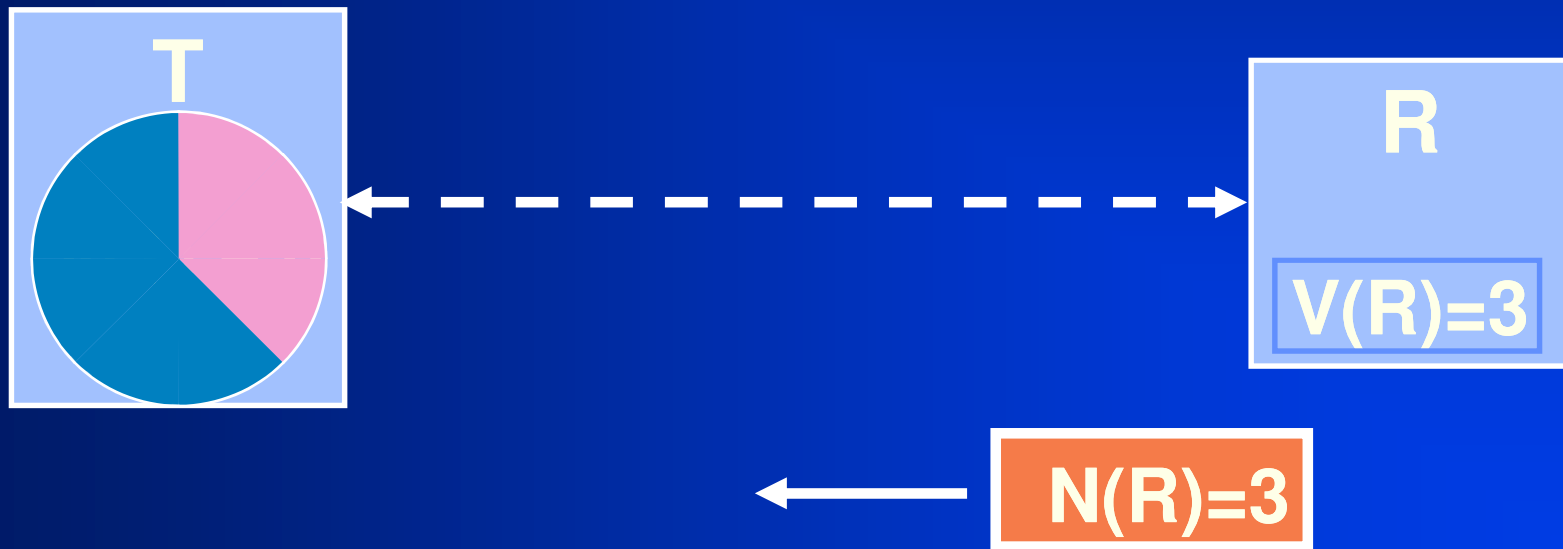
## Trasmissione ACK



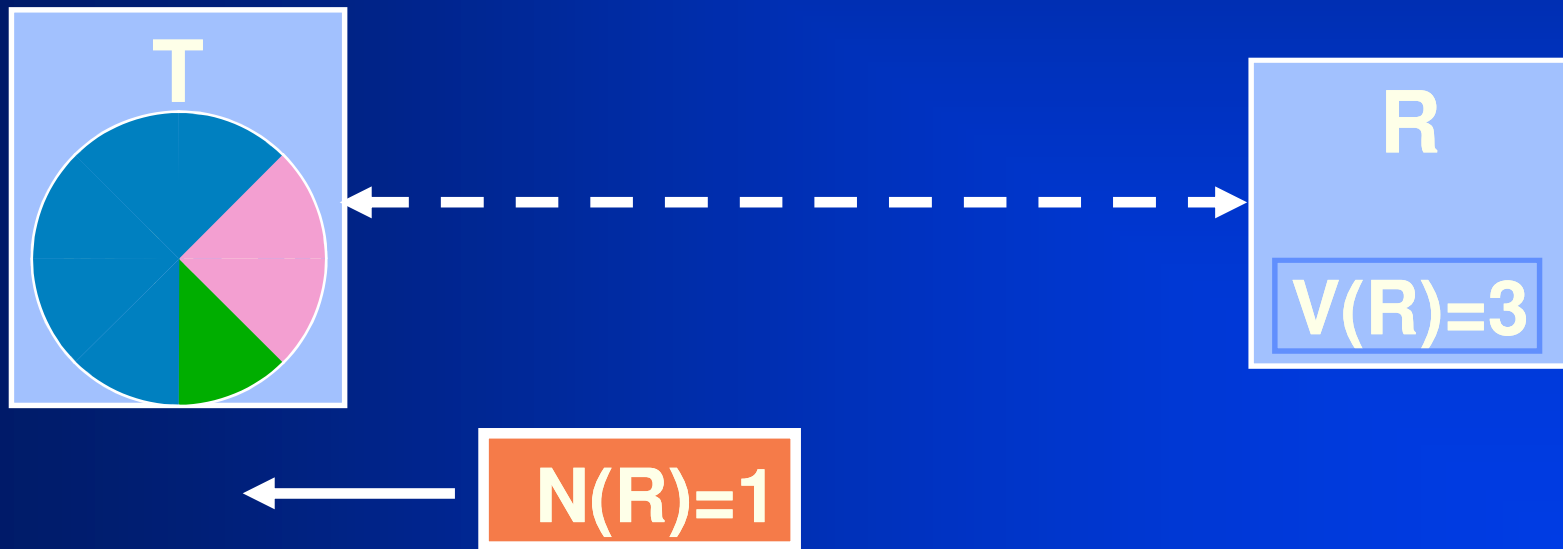
## Ricezione PDU



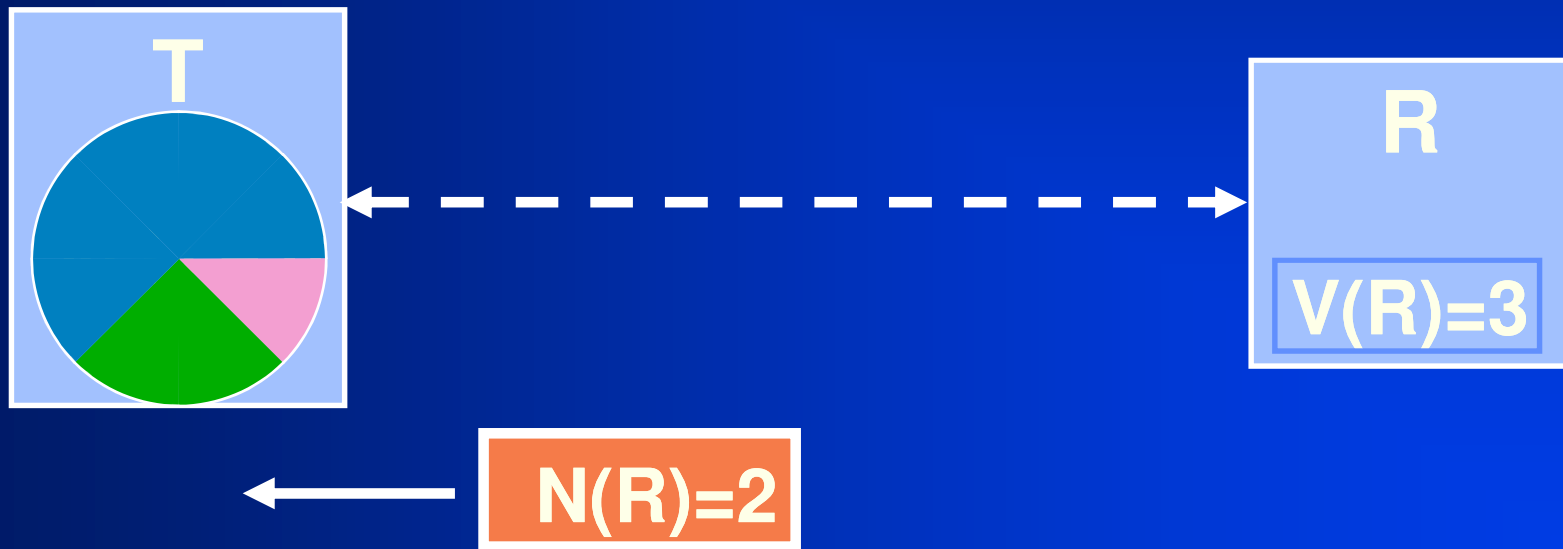
## Trasmissione ACK



## Ricezione ACK

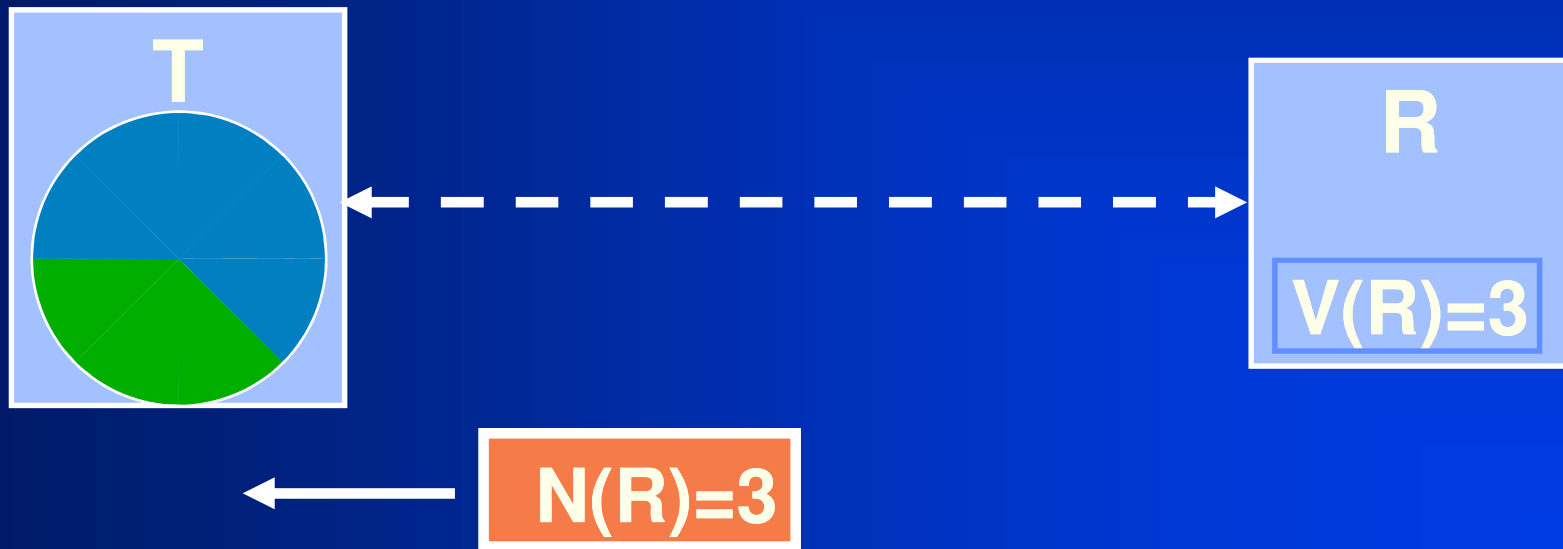


## Ricezione ACK





## Ricezione ACK

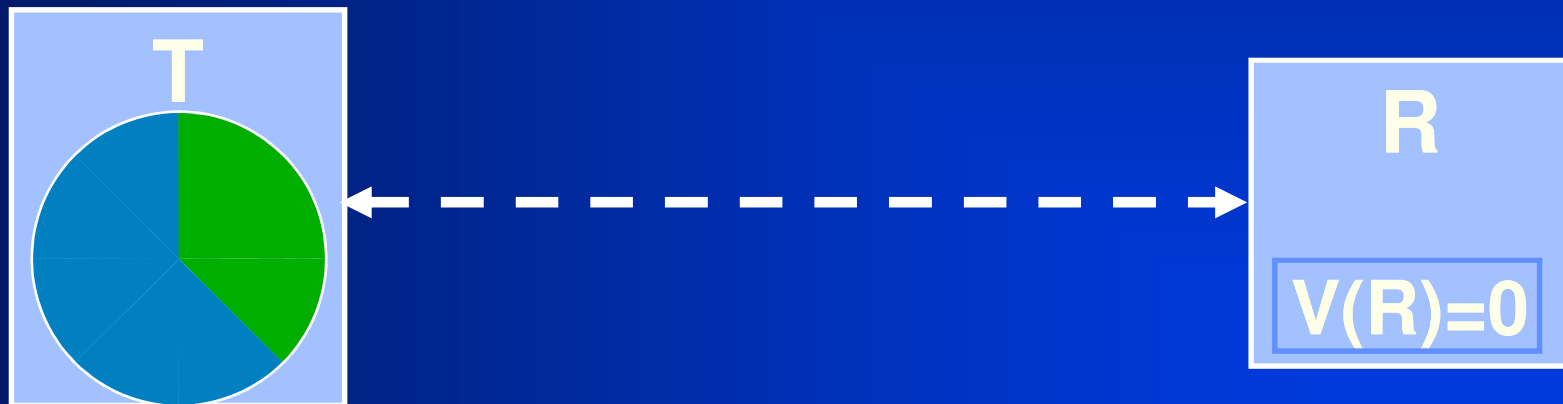


**Si possono usare conferme cumulative**

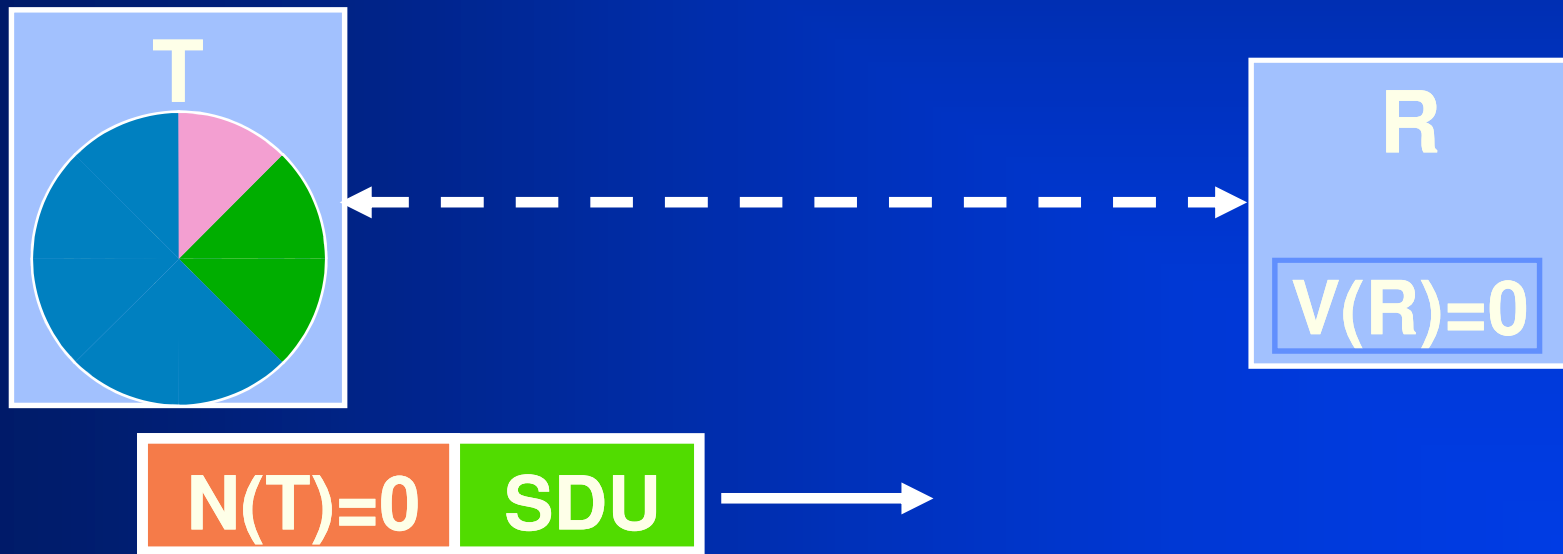
**Si possono usare conferme cumulative**

- **orologio al ricevitore**

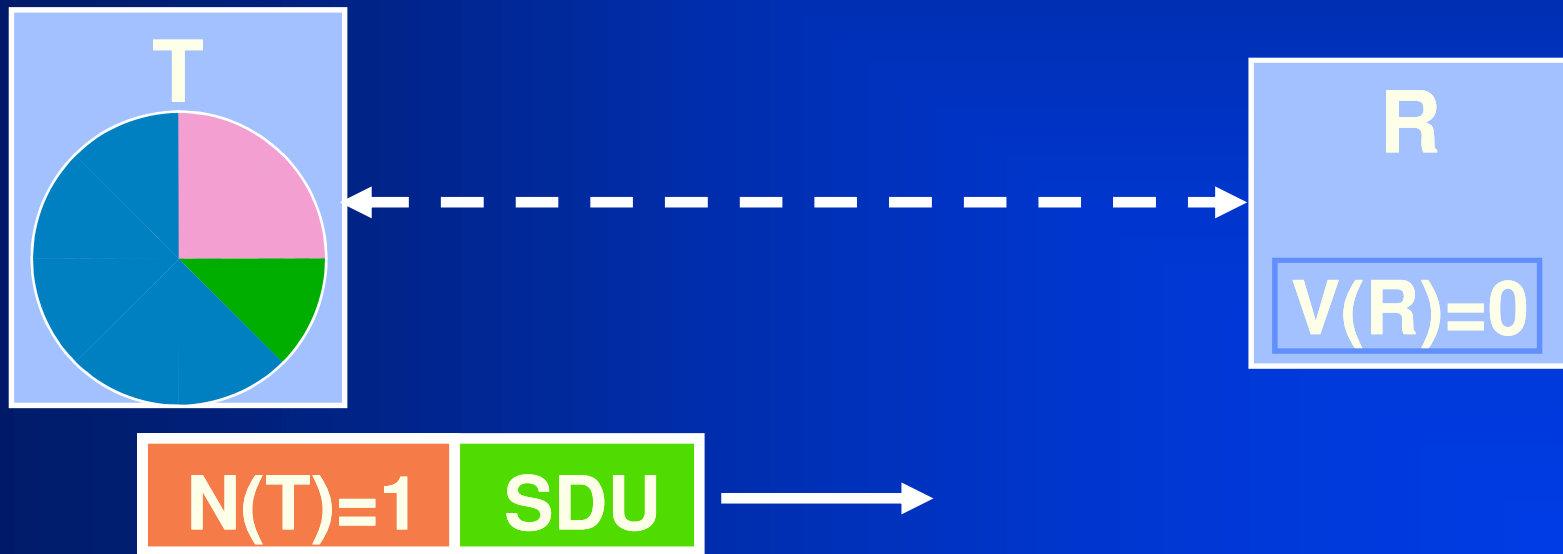
# Inizializzazione



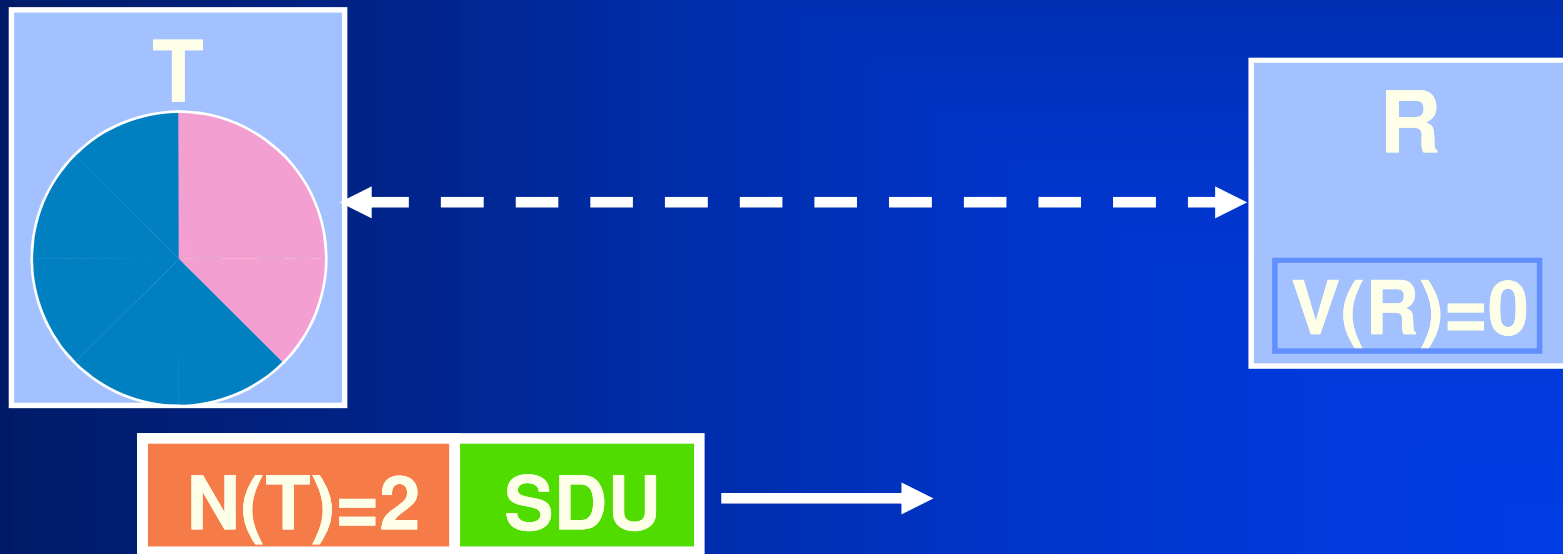
## Trasmissione PDU



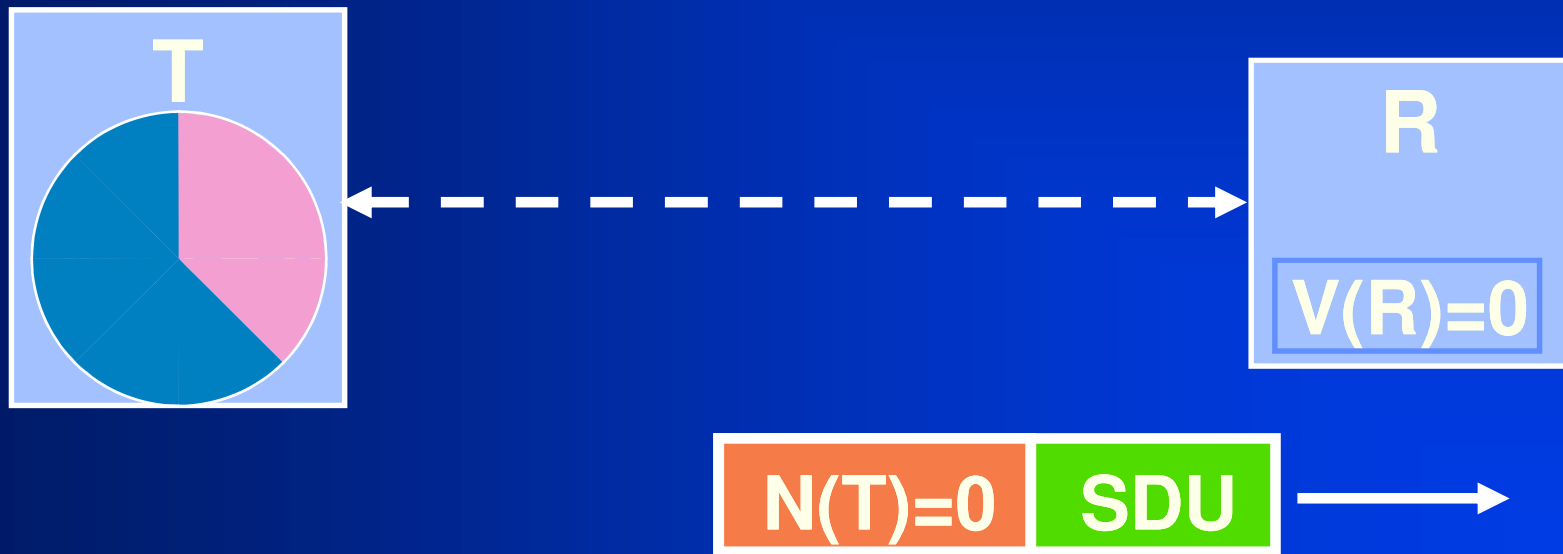
## Trasmissione PDU



## Trasmissione PDU

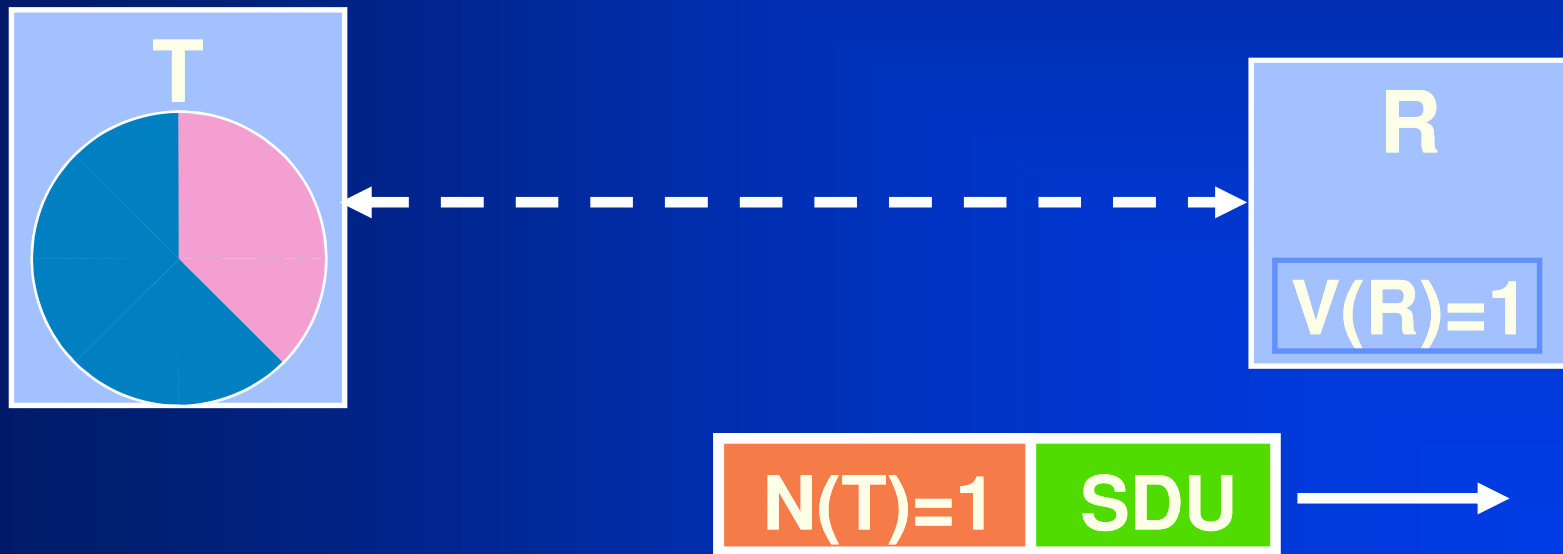


## Ricezione PDU

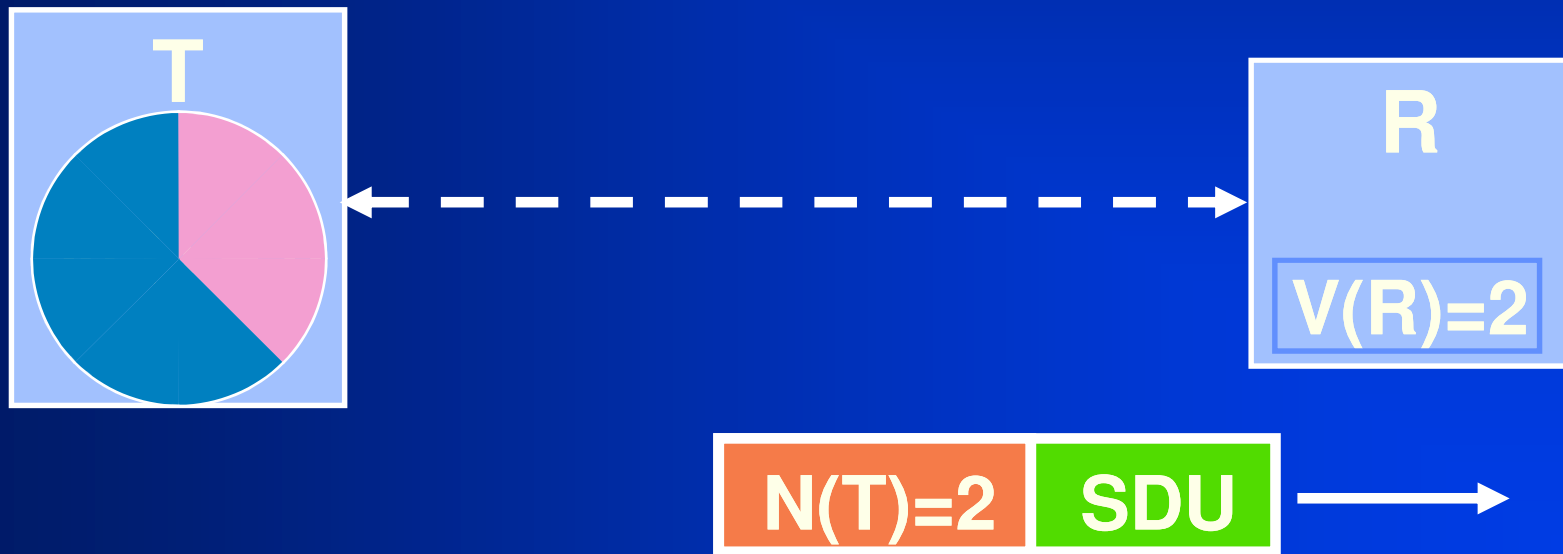




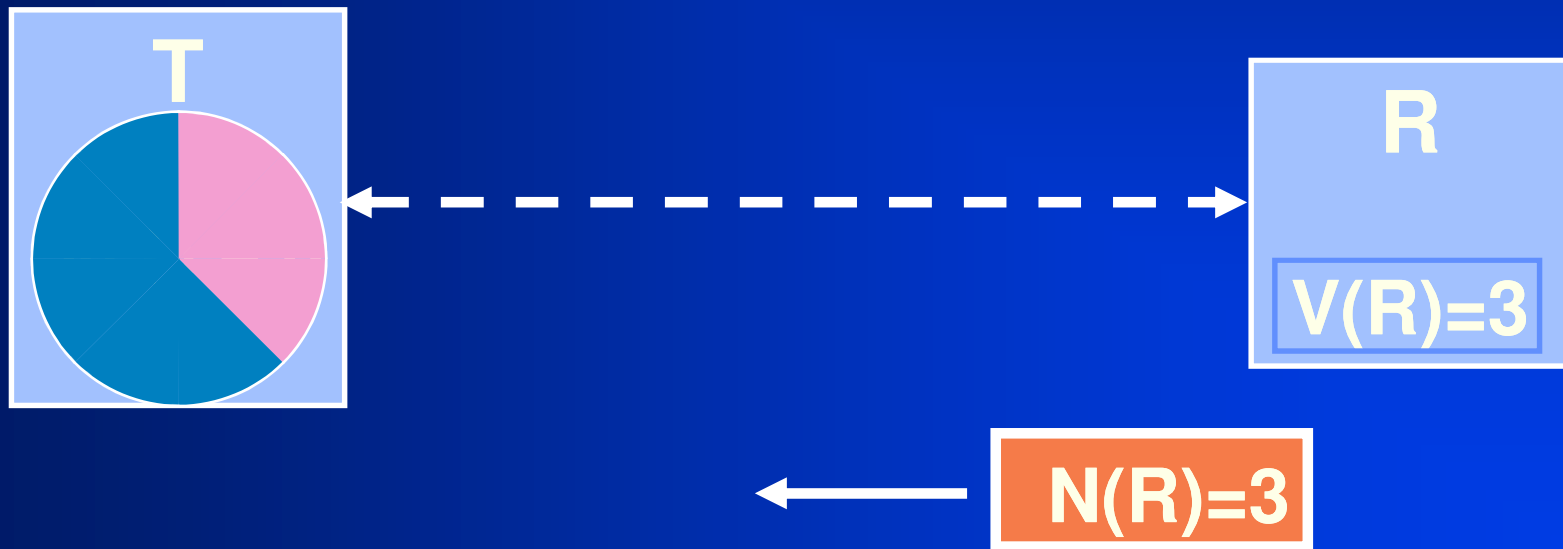
## Ricezione PDU



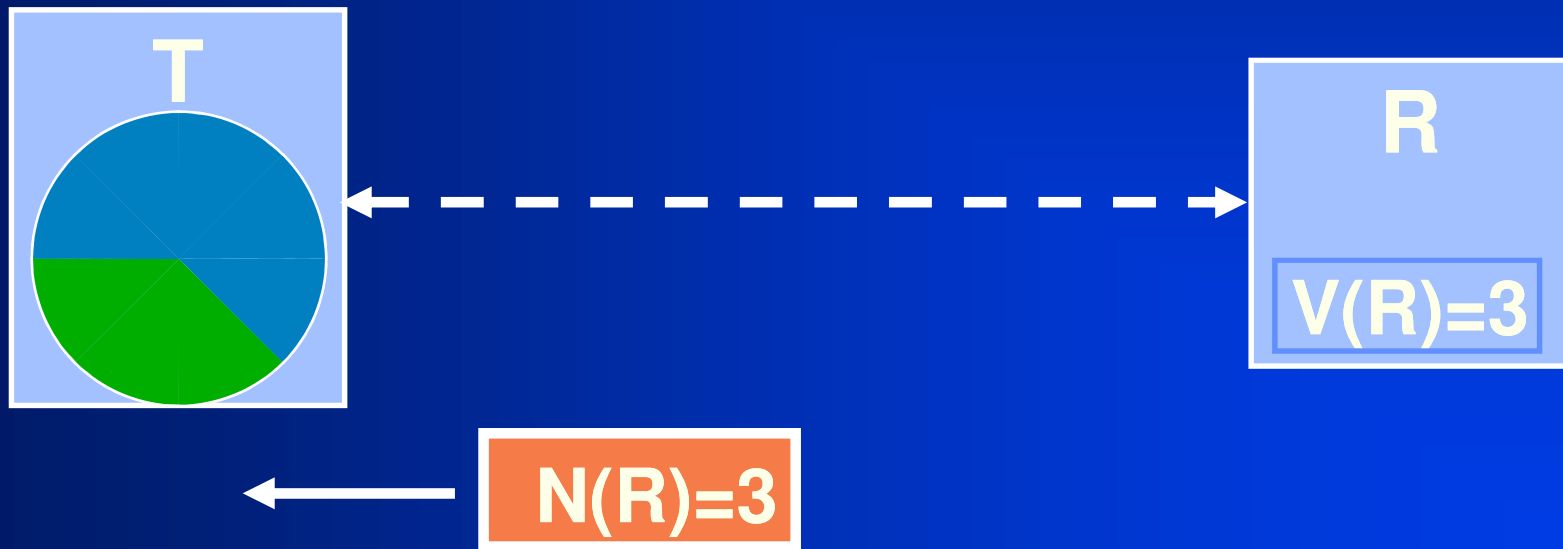
## Ricezione PDU



## Trasmissione ACK

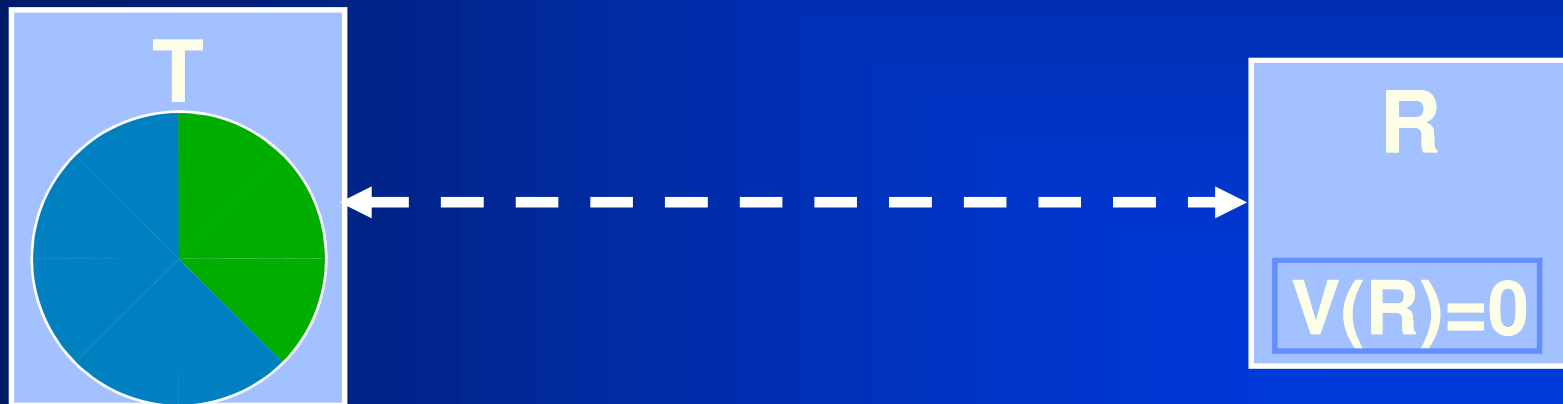


## Ricezione ACK

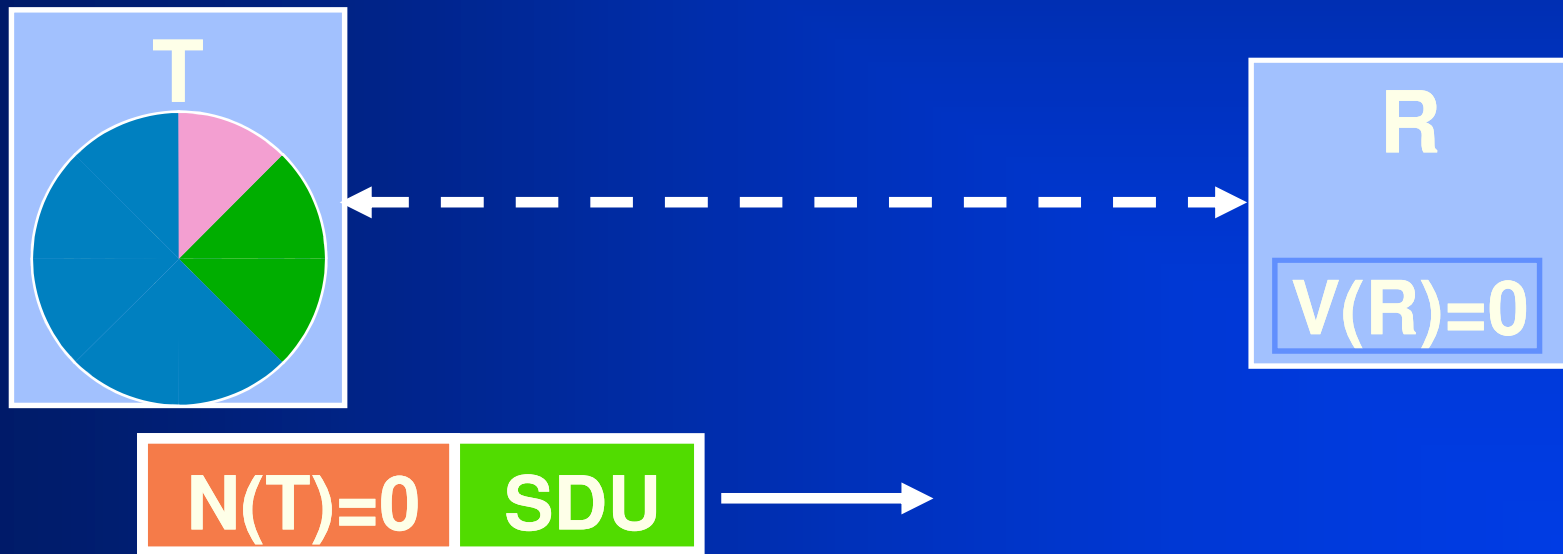


**Ricezione di una PDU errata**

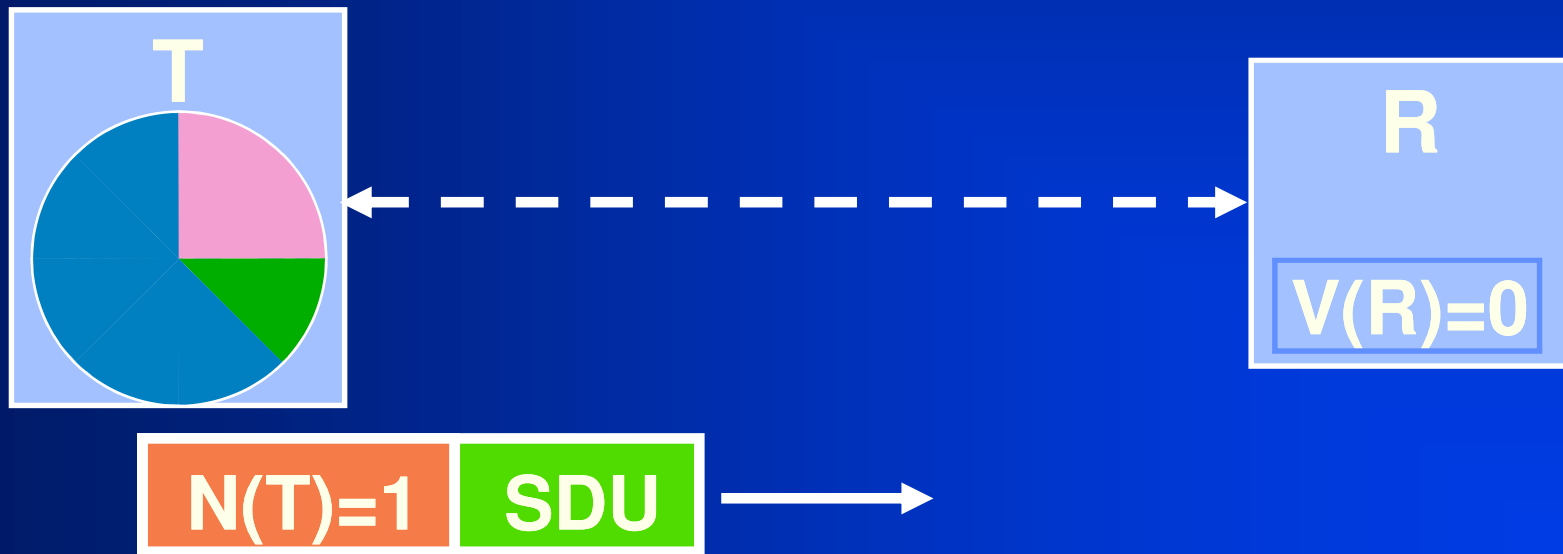
# Inizializzazione



## Trasmissione PDU

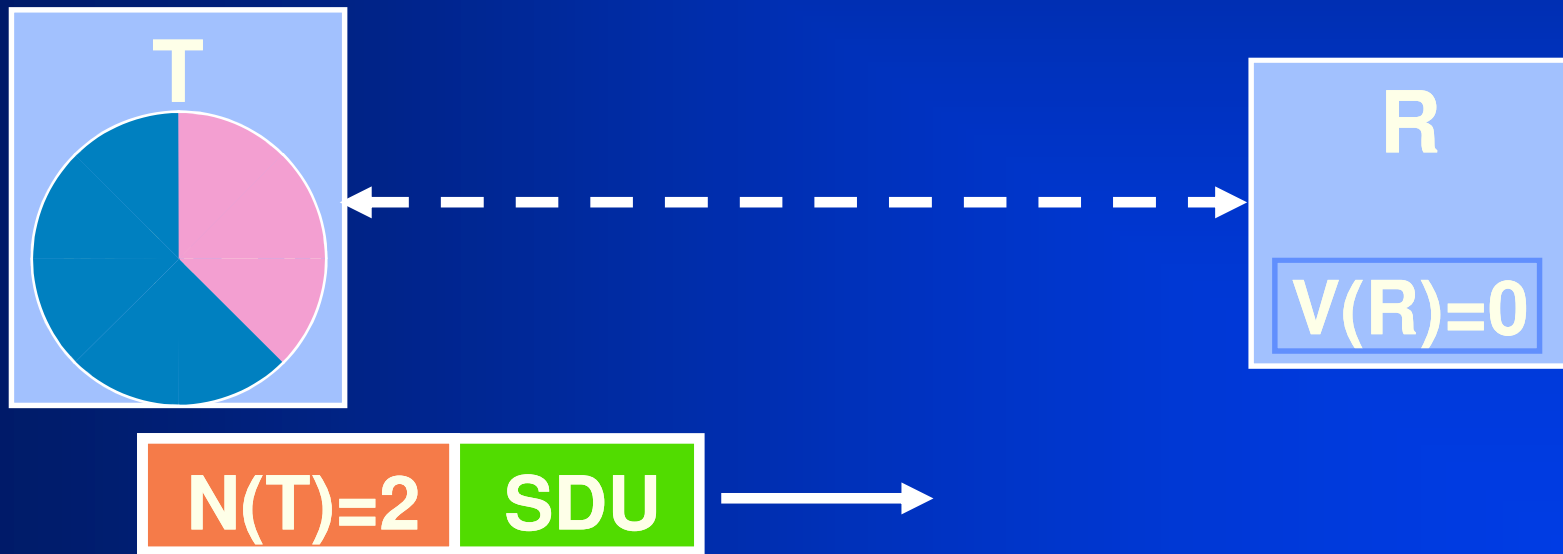


## Trasmissione PDU

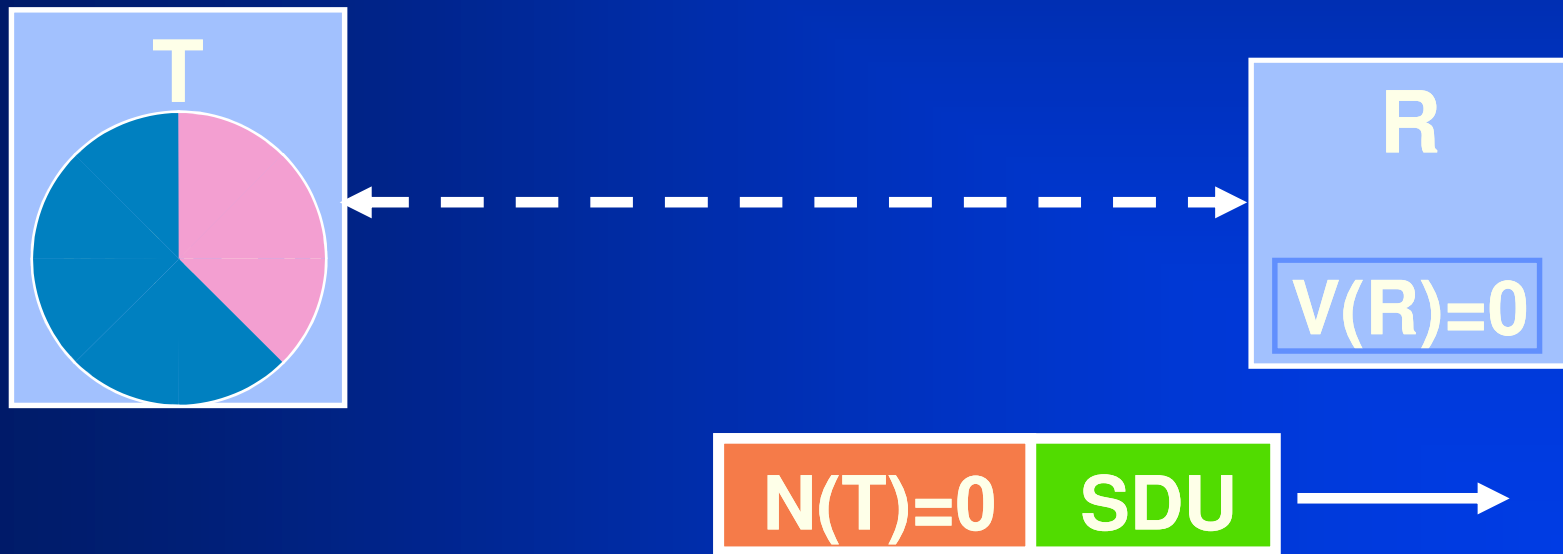




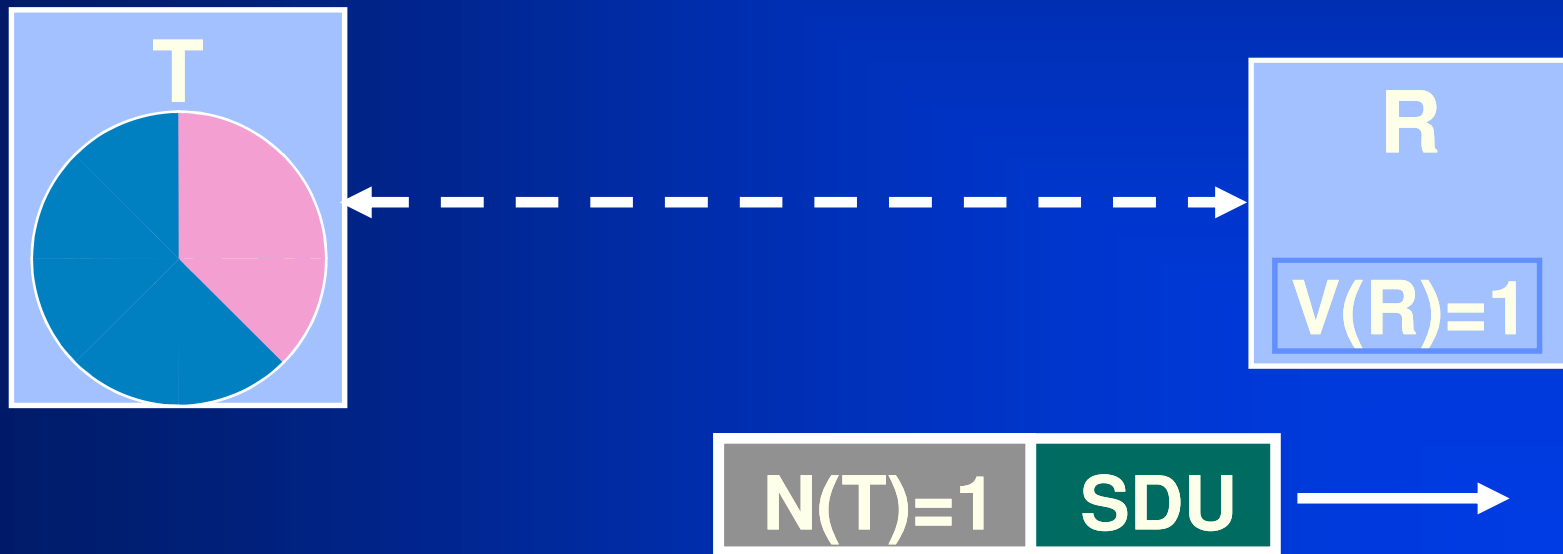
## Trasmissione PDU



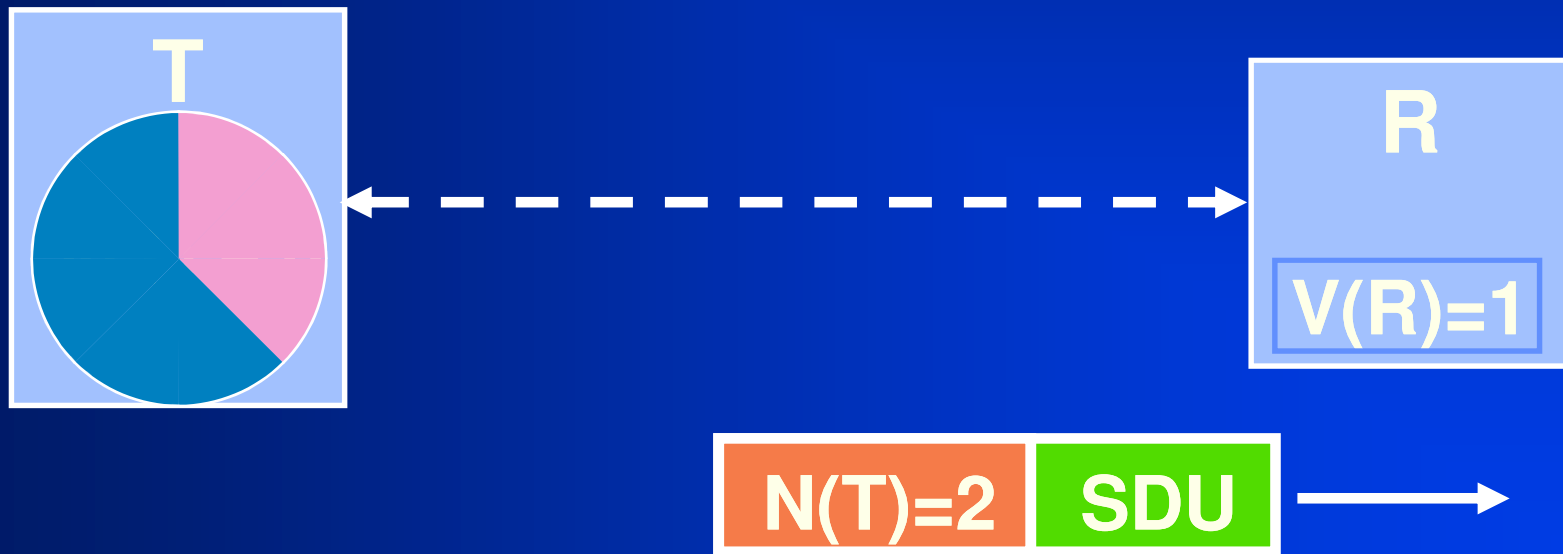
## Ricezione PDU



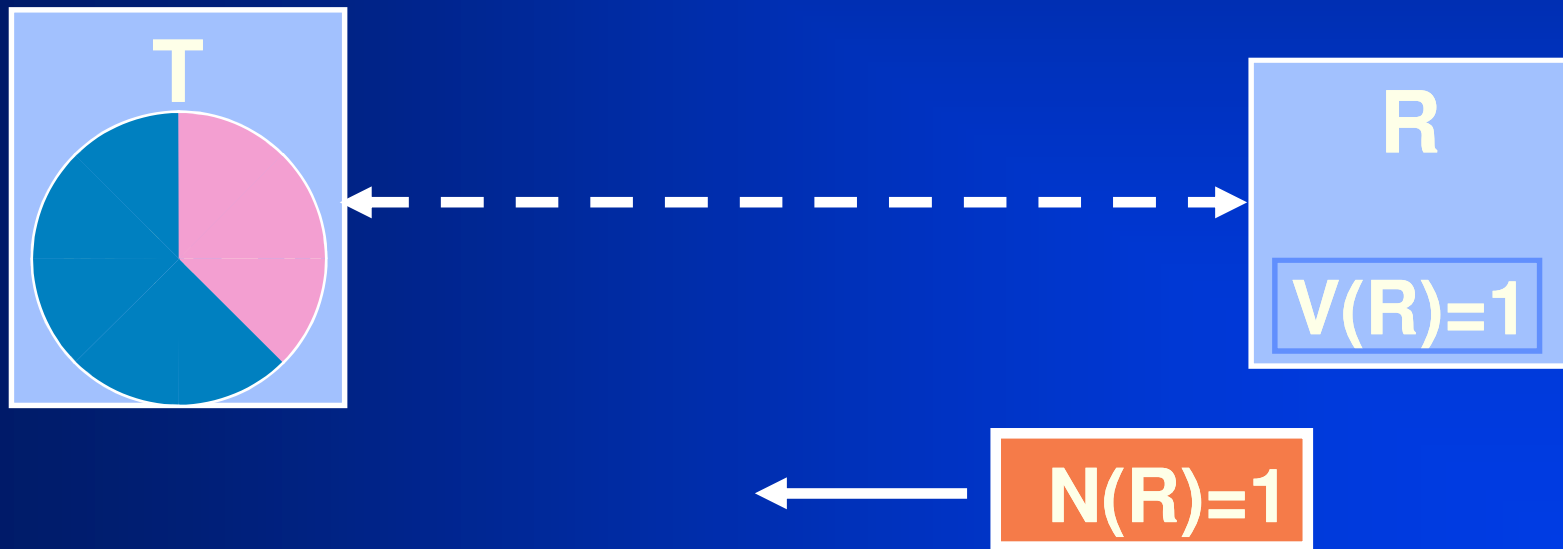
## Ricezione PDU errata



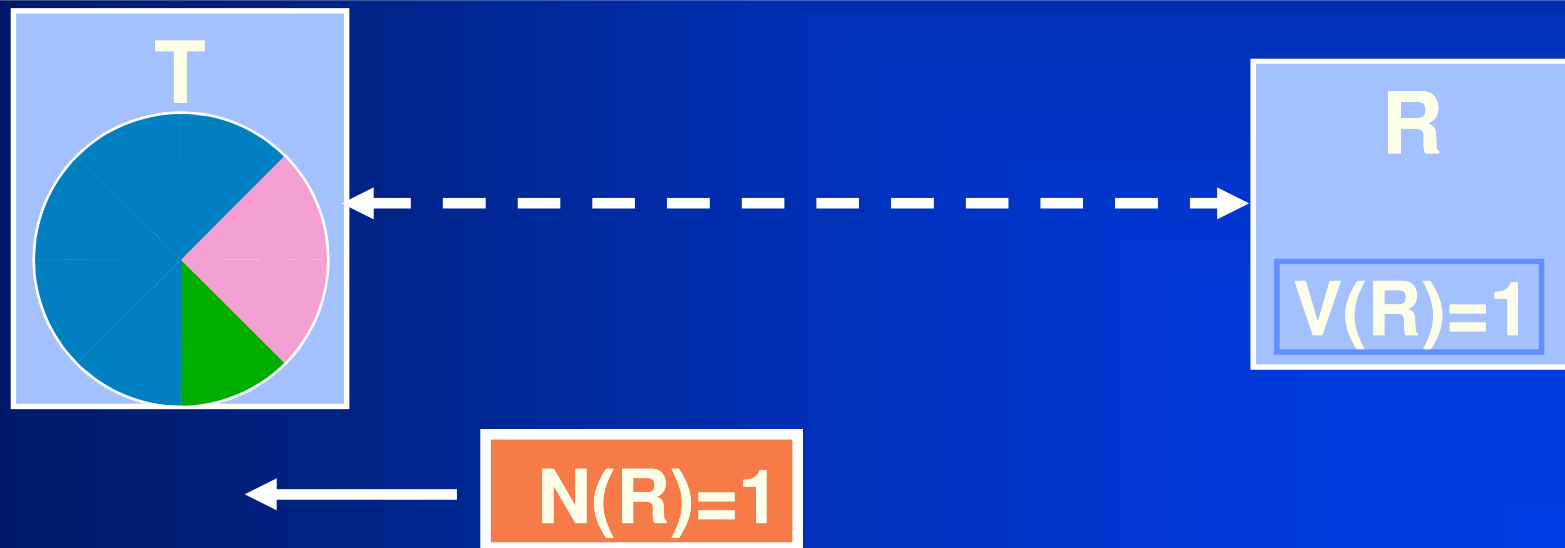
## Ricezione PDU fuori sequenza



## Trasmissione ACK

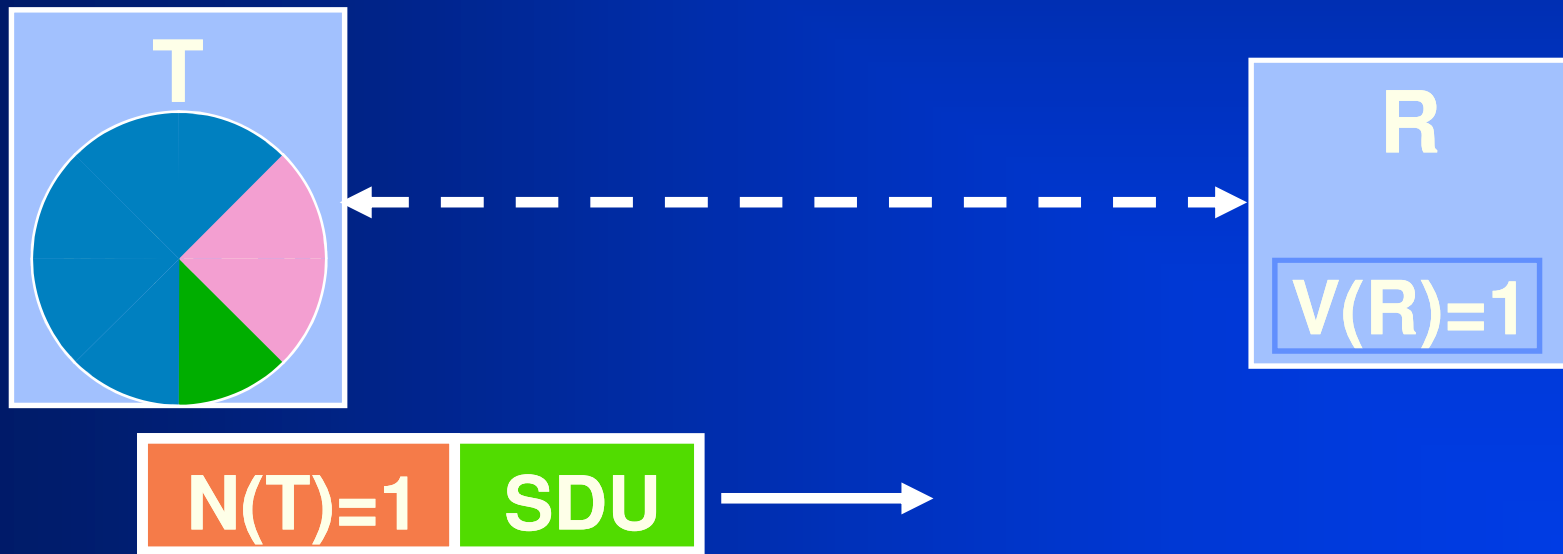


## Ricezione ACK



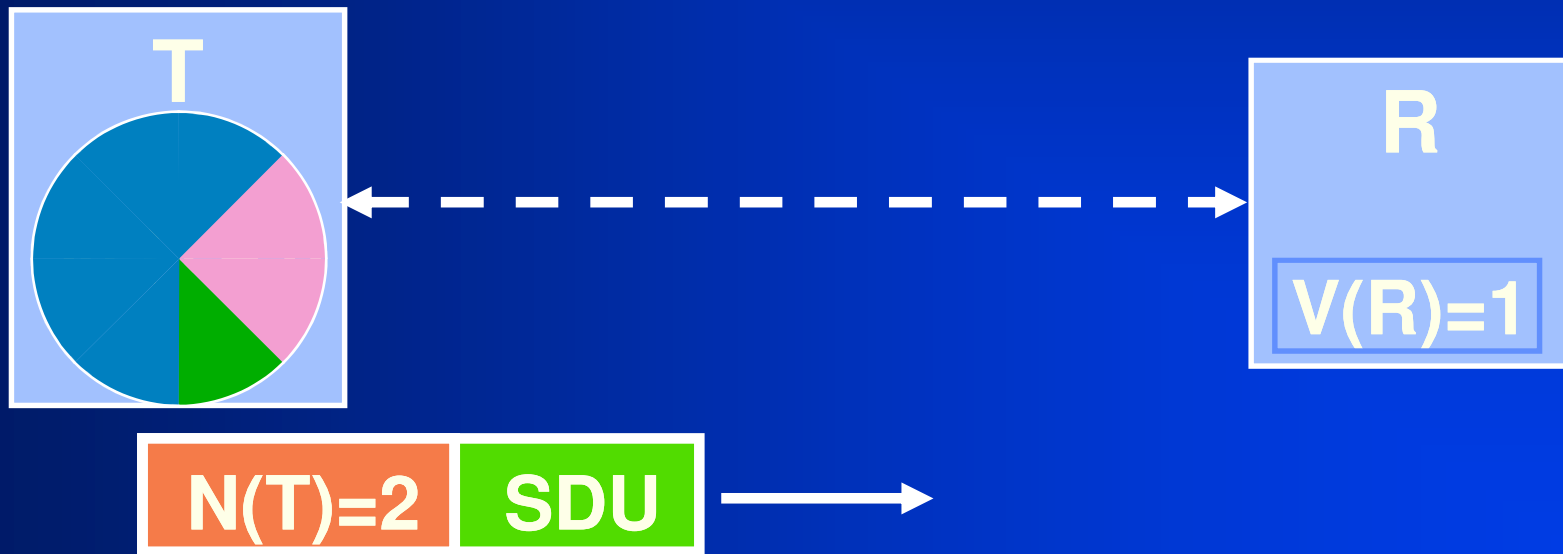
**Scade il timeout**

## Ritrasmissione PDU

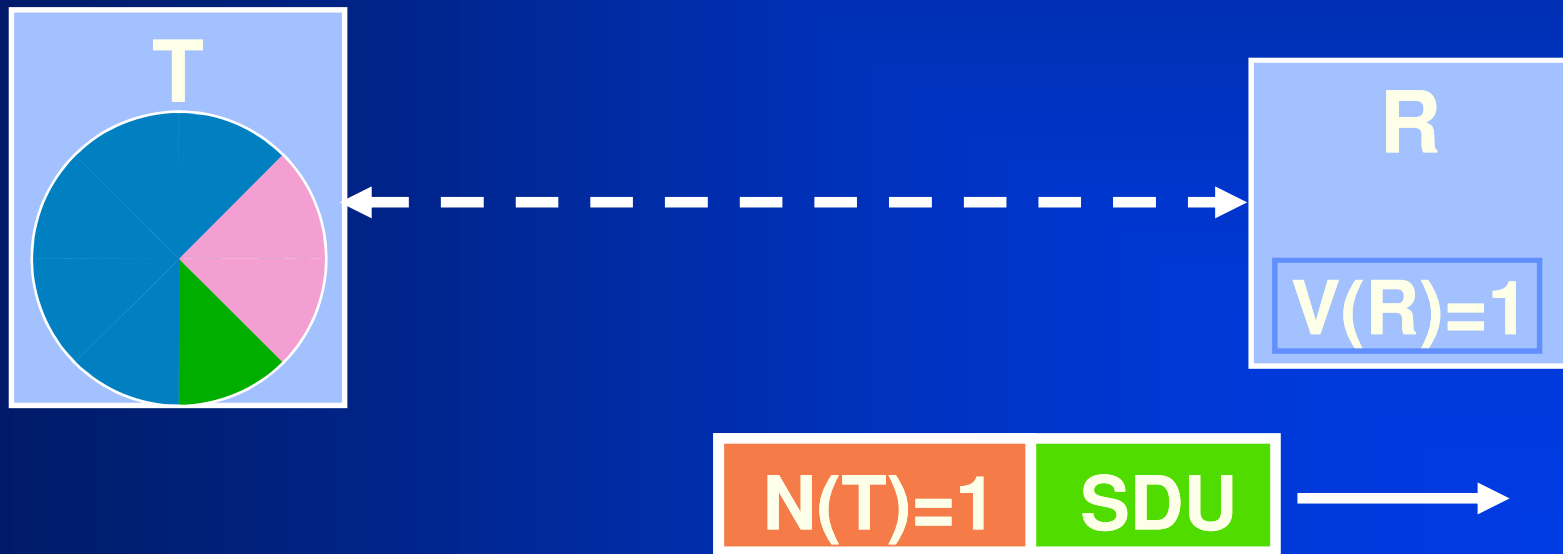




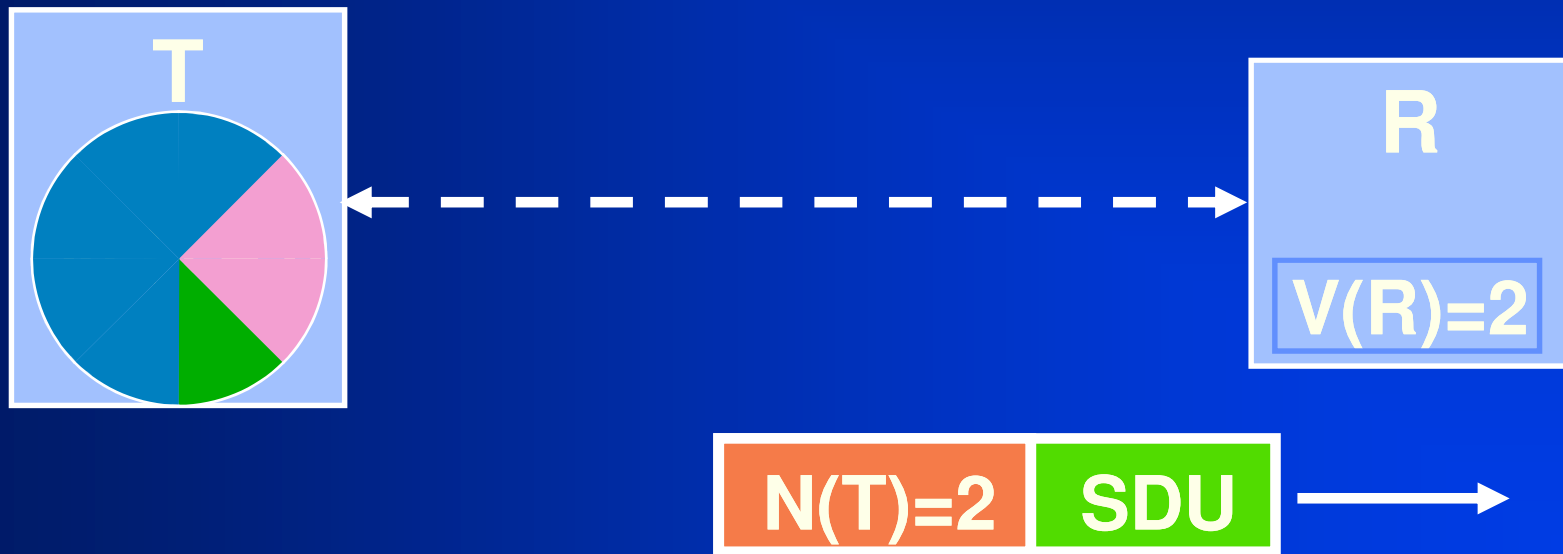
## Ritrasmissione PDU



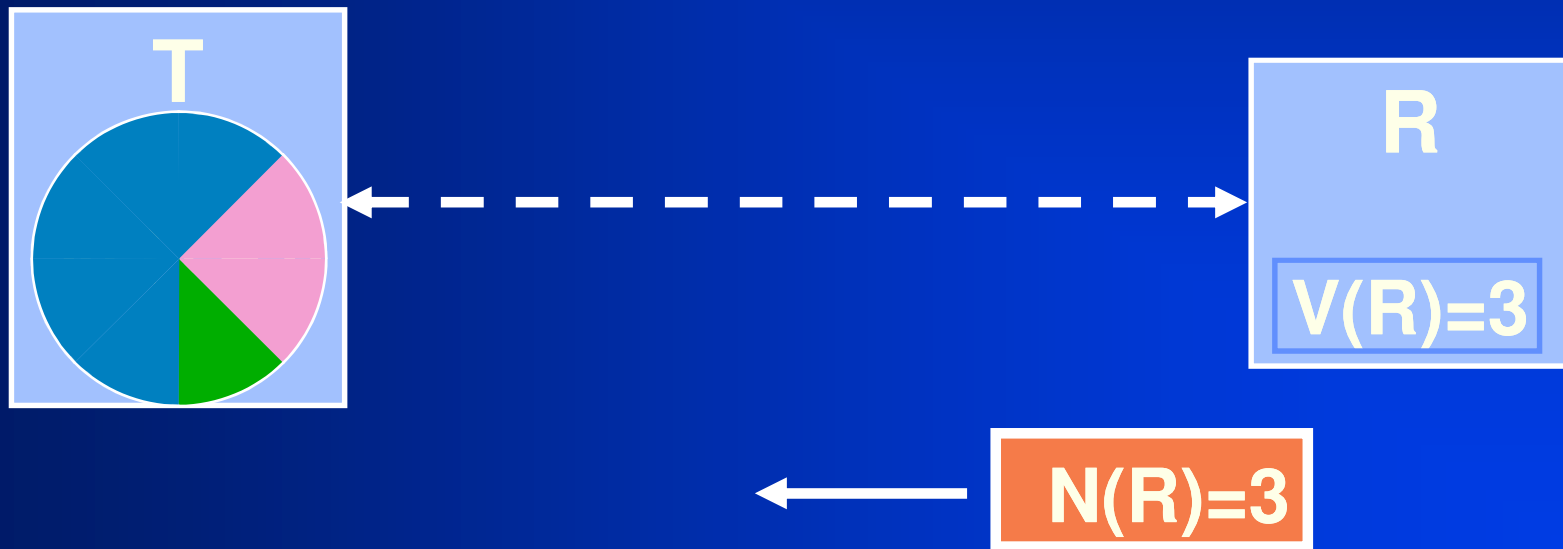
## Ricezione PDU



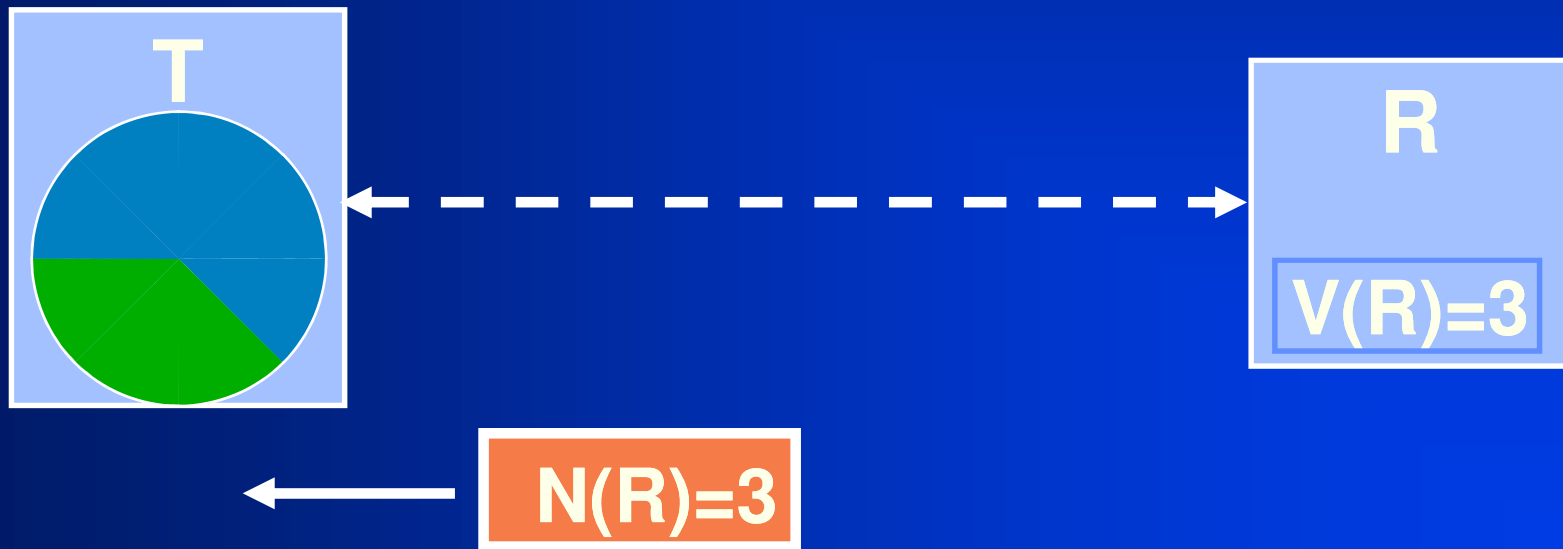
## Ricezione PDU



## Trasmissione ACK

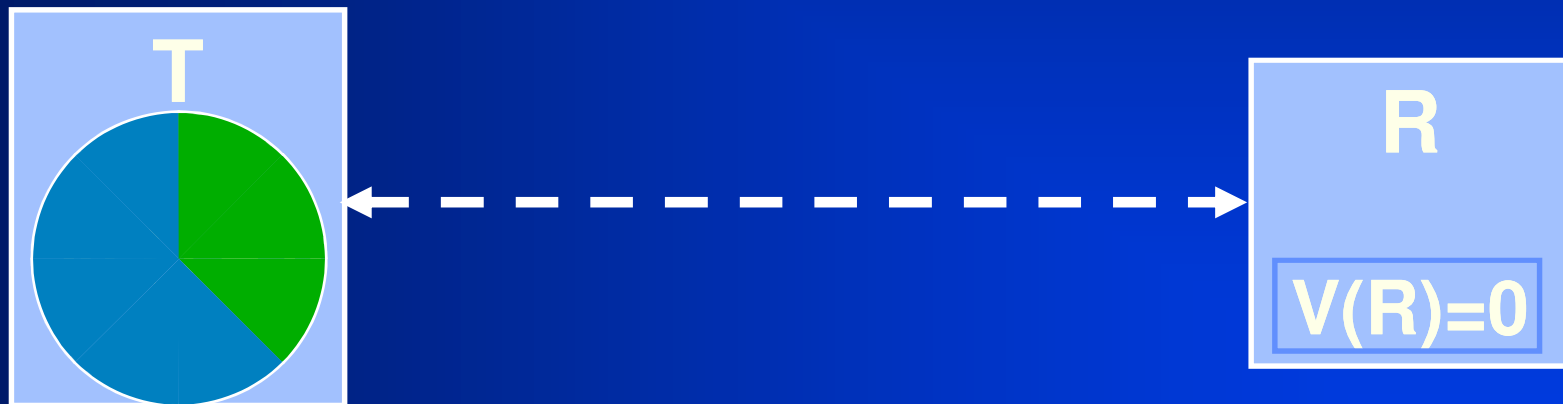


## Ricezione ACK

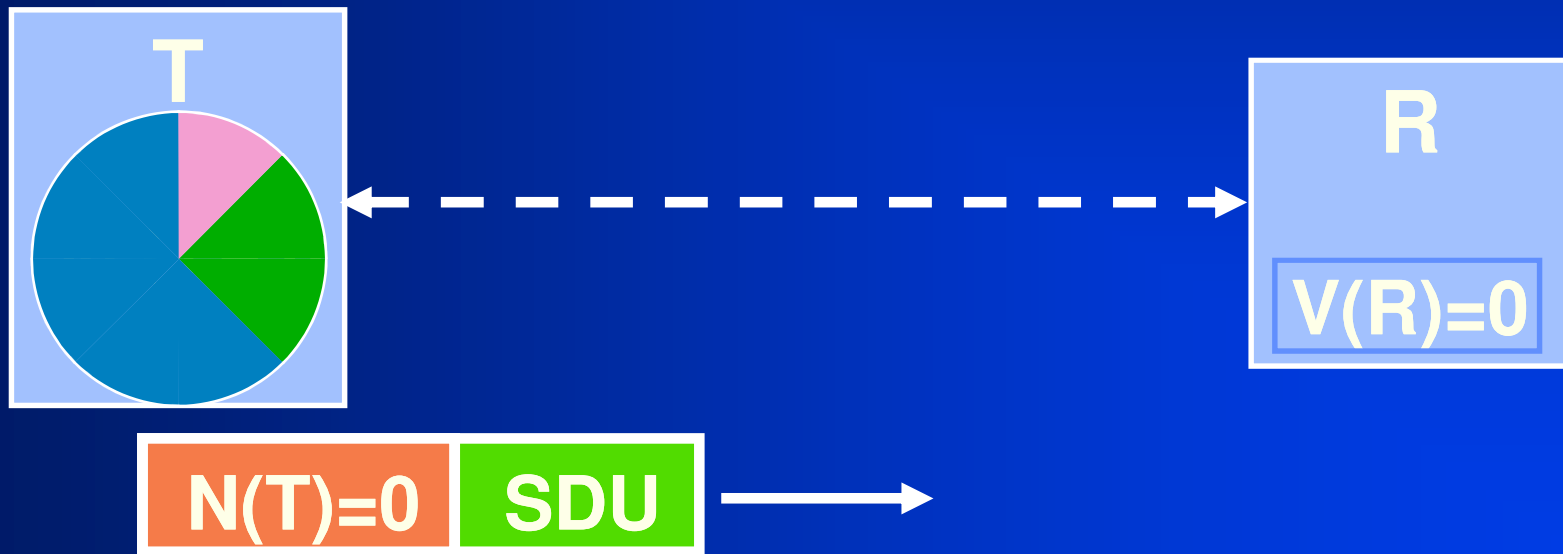


**Ricezione di una conferma errata**

# Inizializzazione

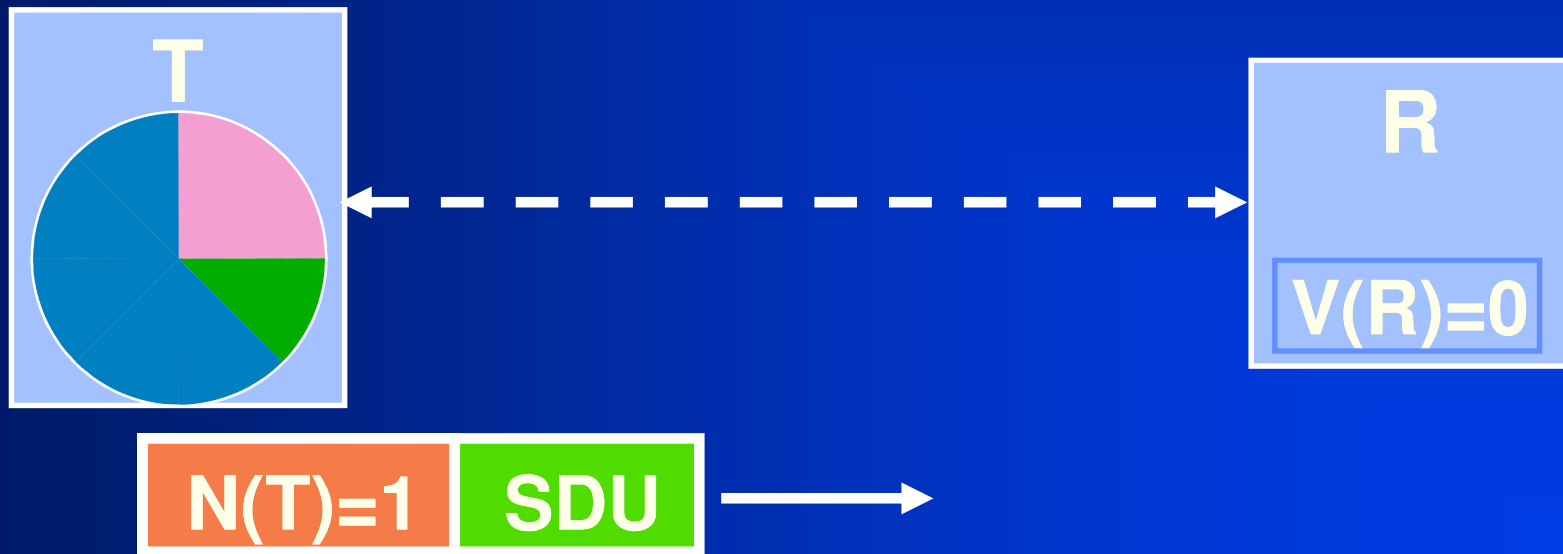


## Trasmissione PDU

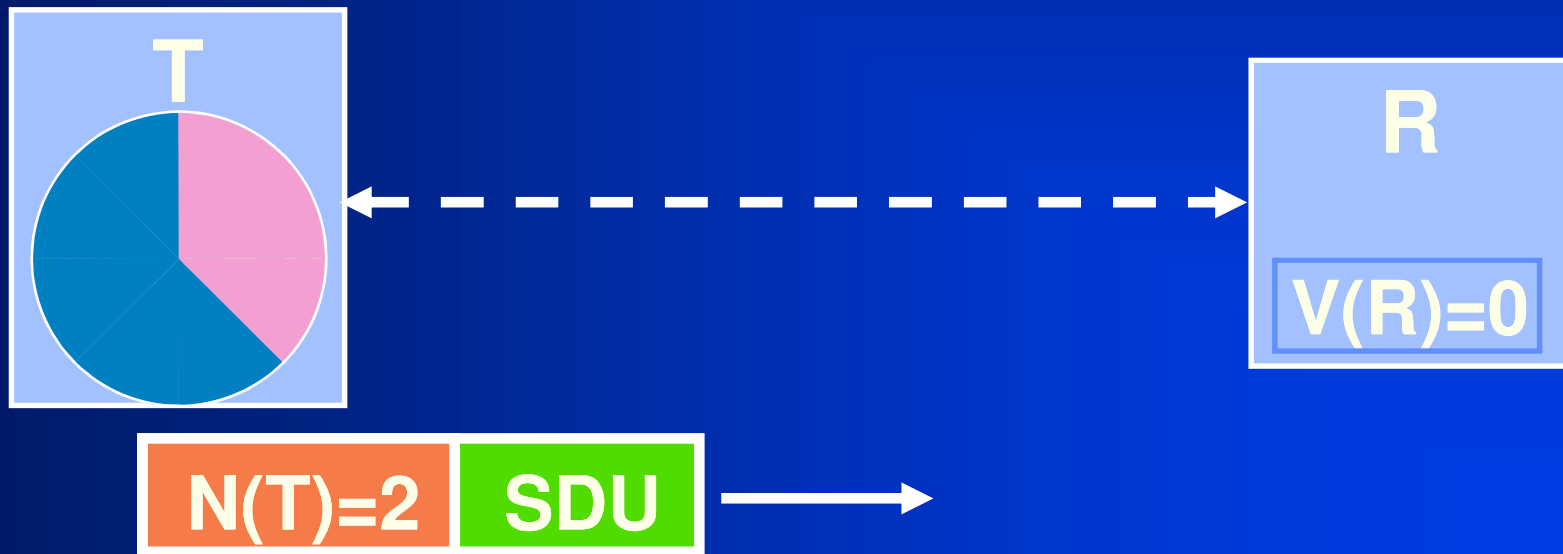




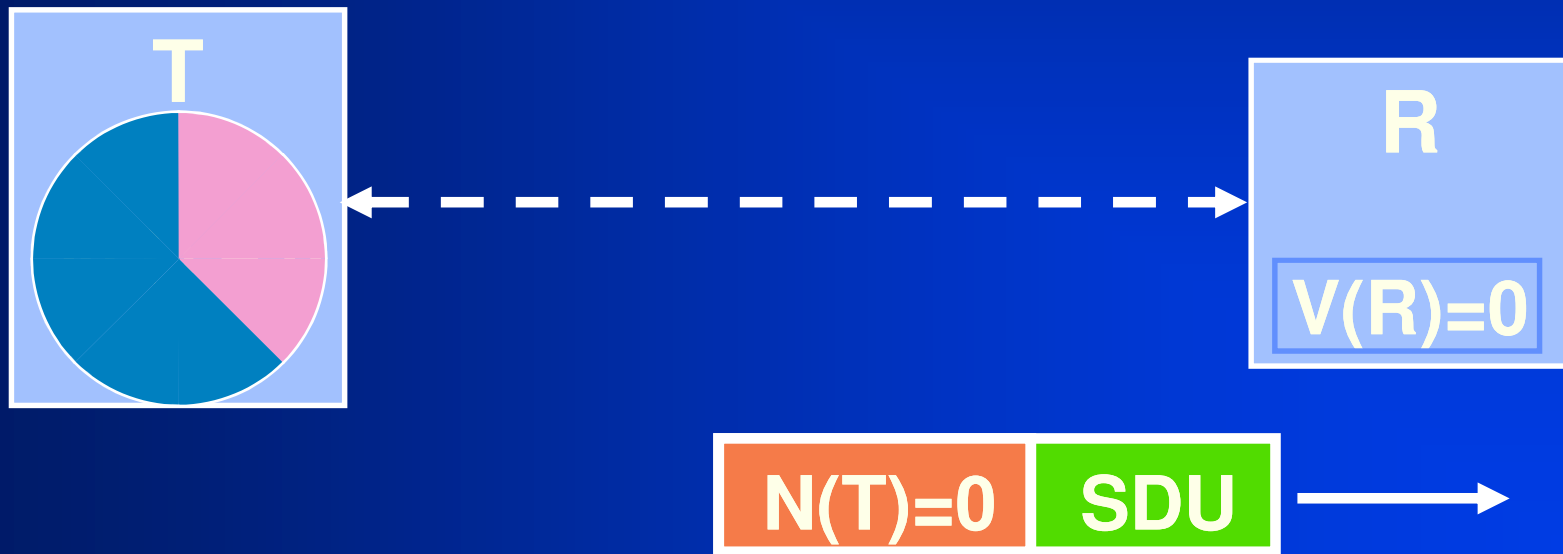
## Trasmissione PDU



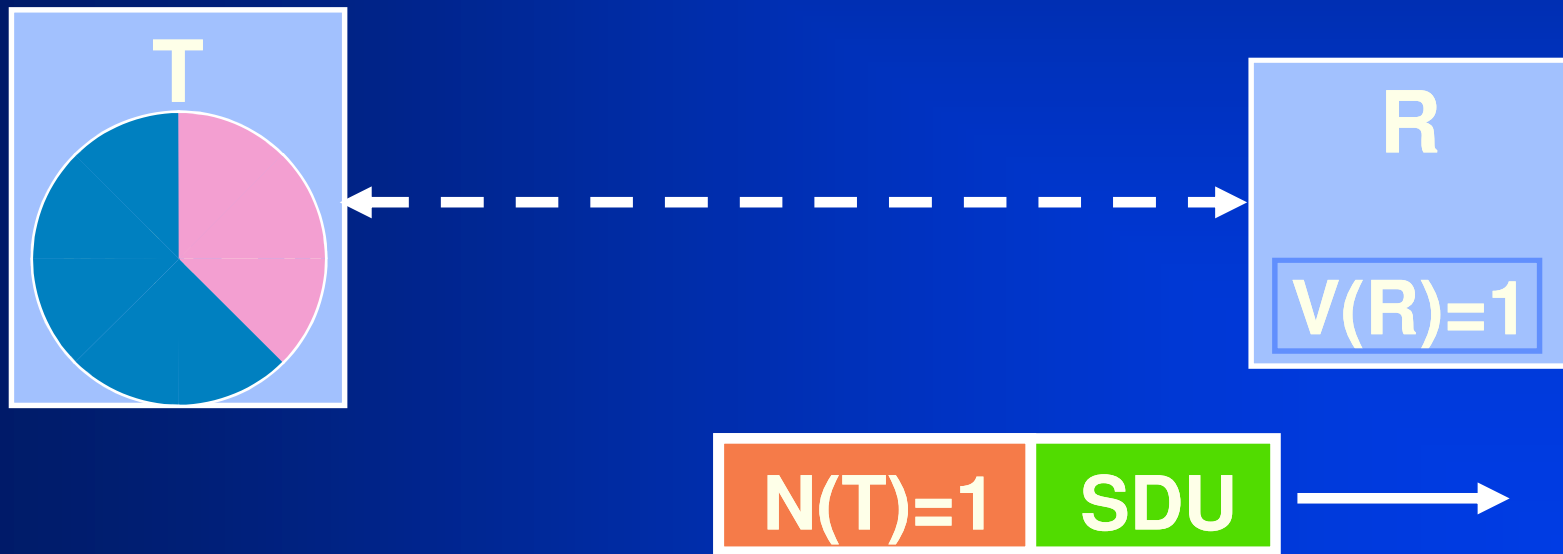
## Trasmissione PDU



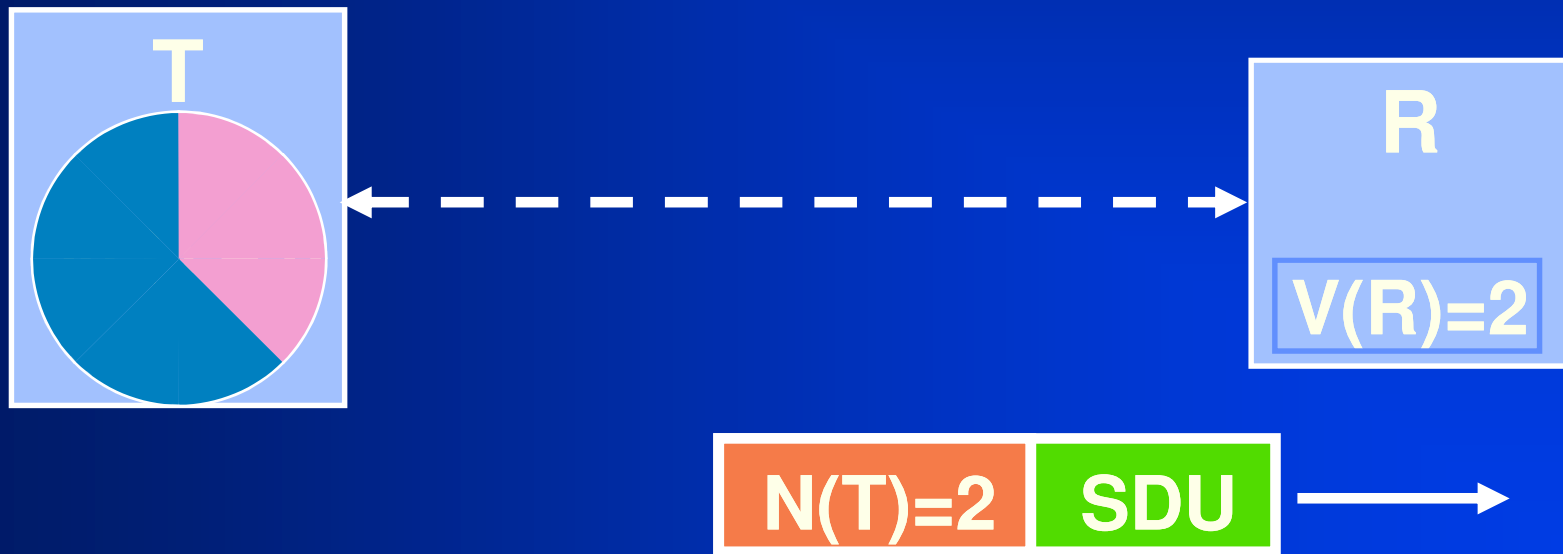
## Ricezione PDU



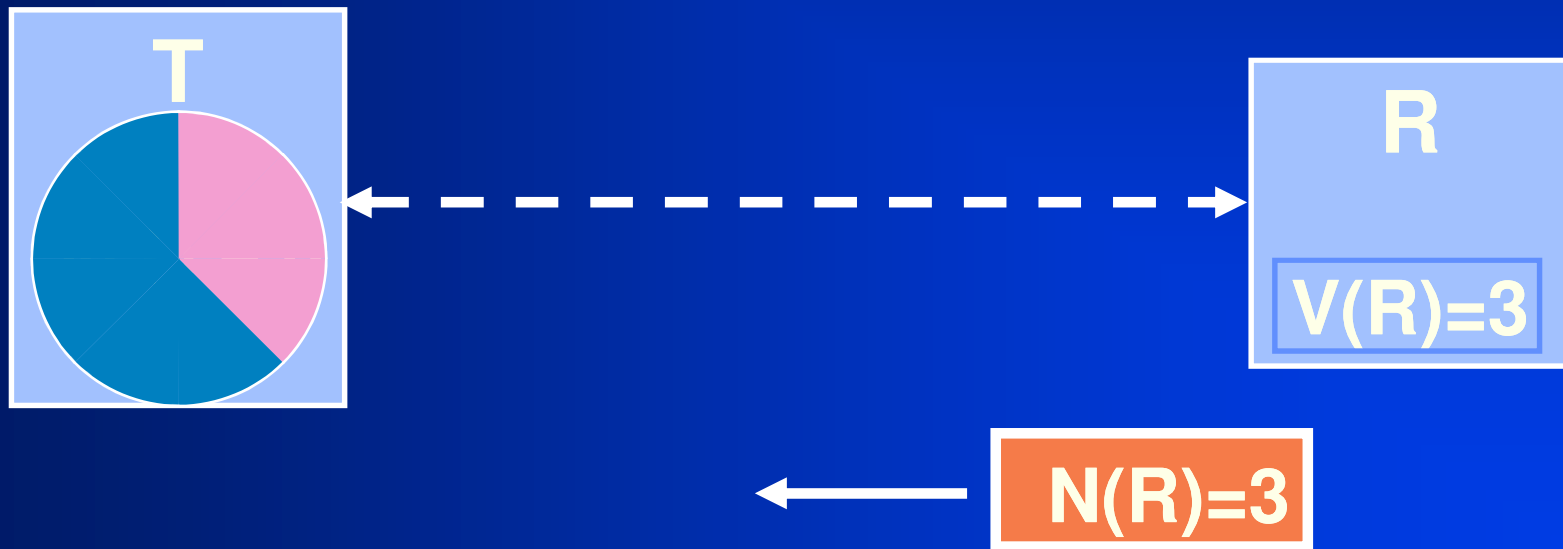
## Ricezione PDU



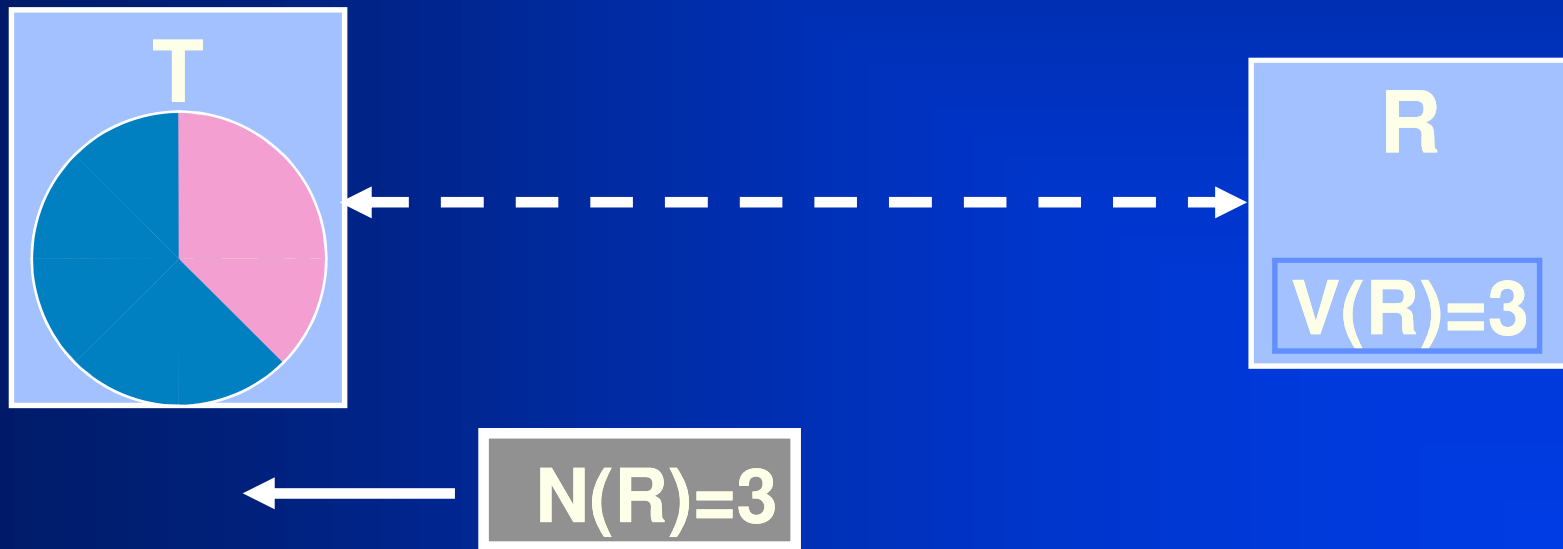
## Ricezione PDU



## Trasmissione ACK



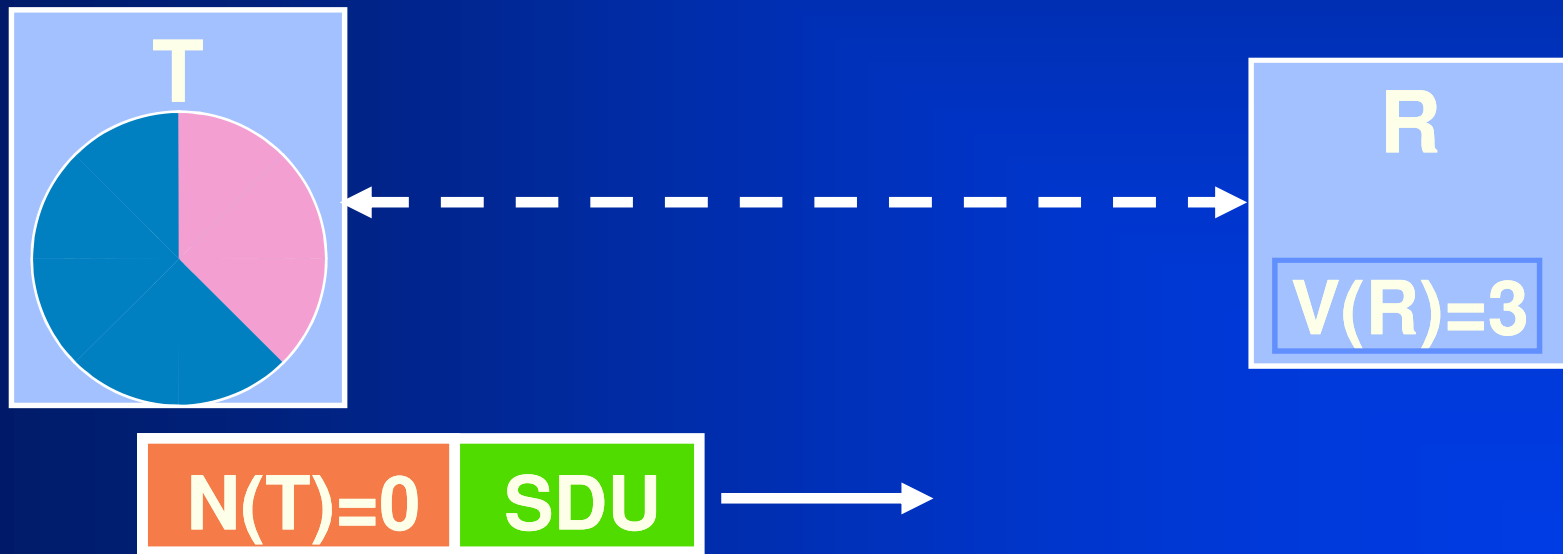
## Ricezione ACK errato



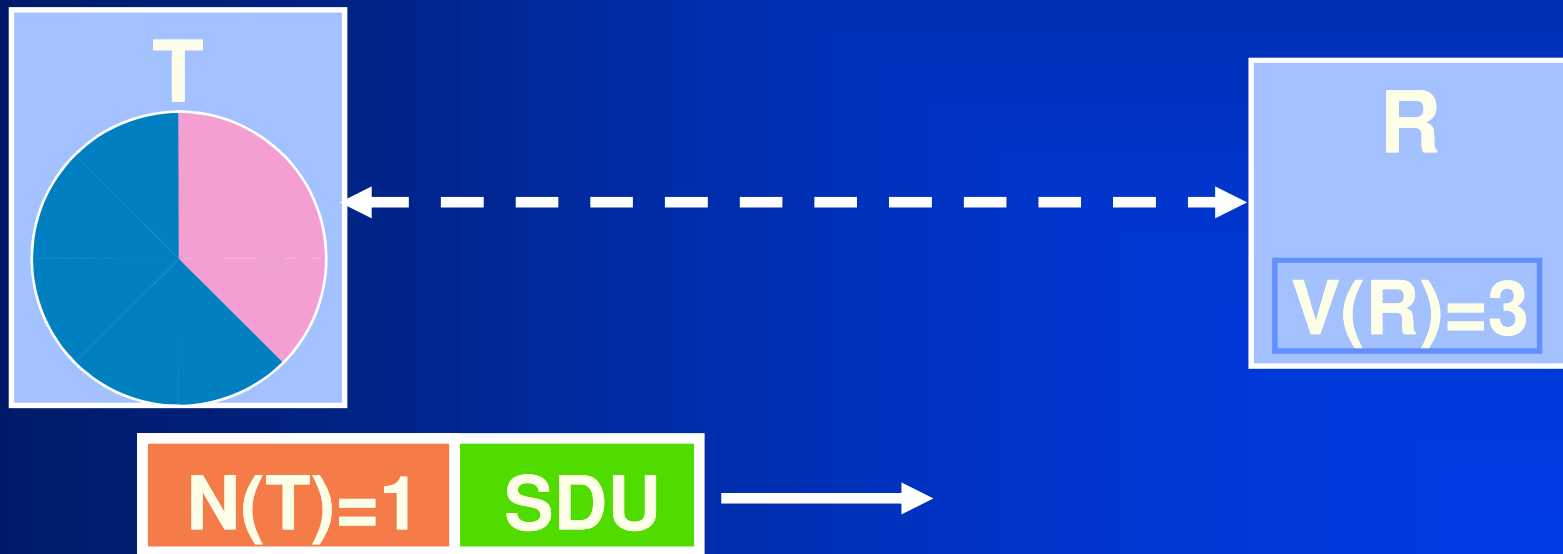
**Scade il timeout**



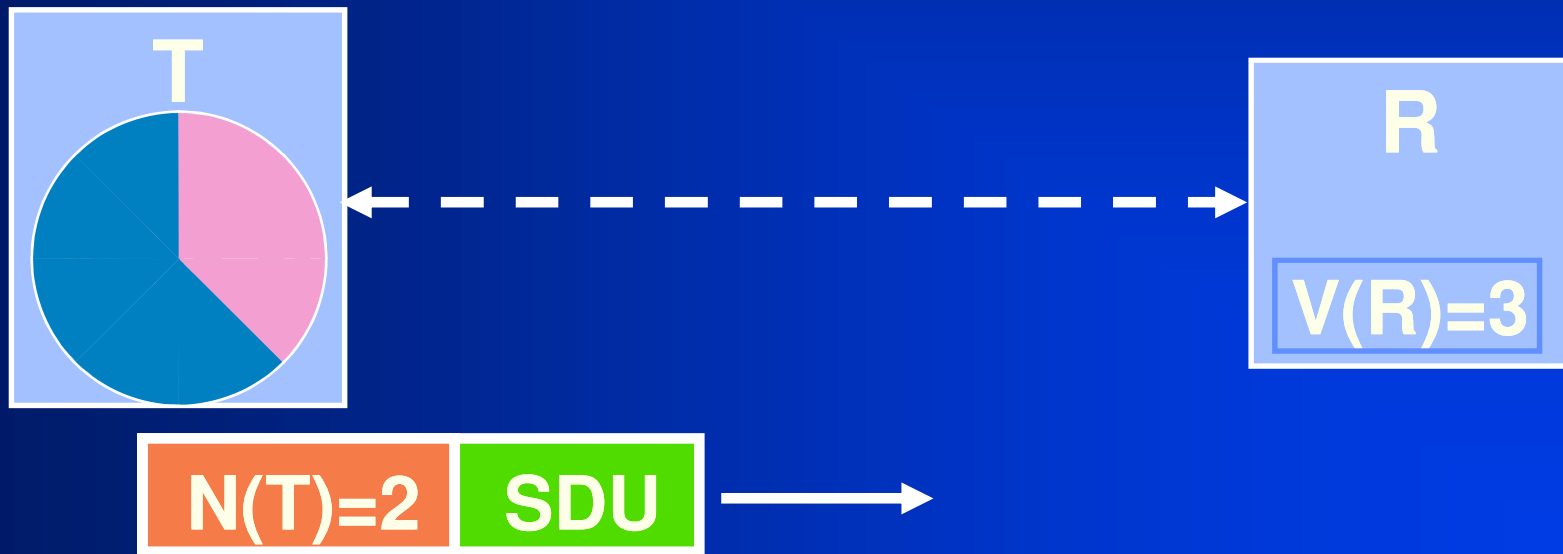
## Ritrasmissione PDU



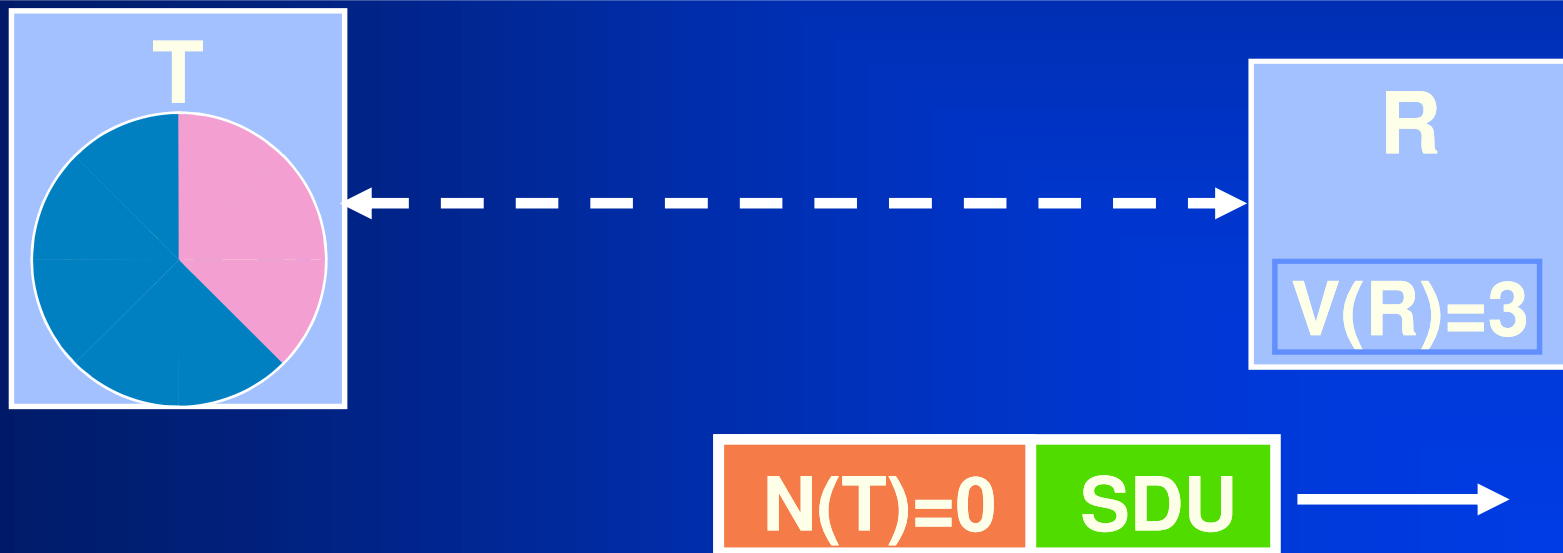
## Ritrasmissione PDU



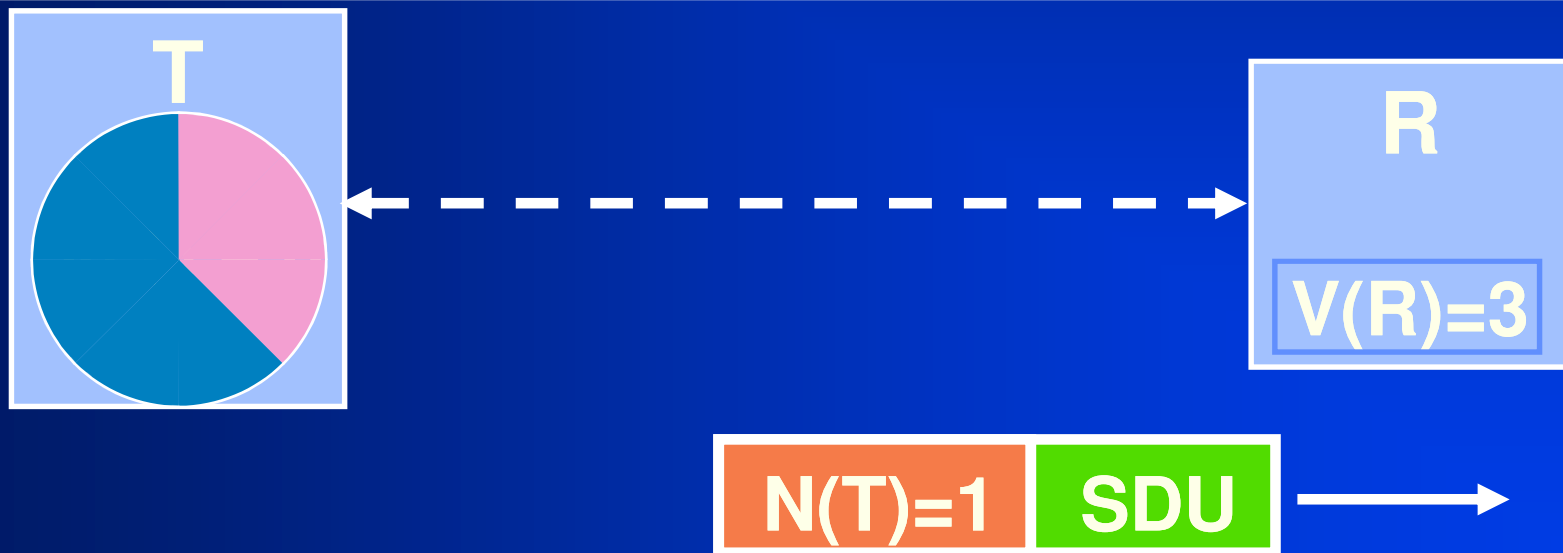
## Ritrasmissione PDU



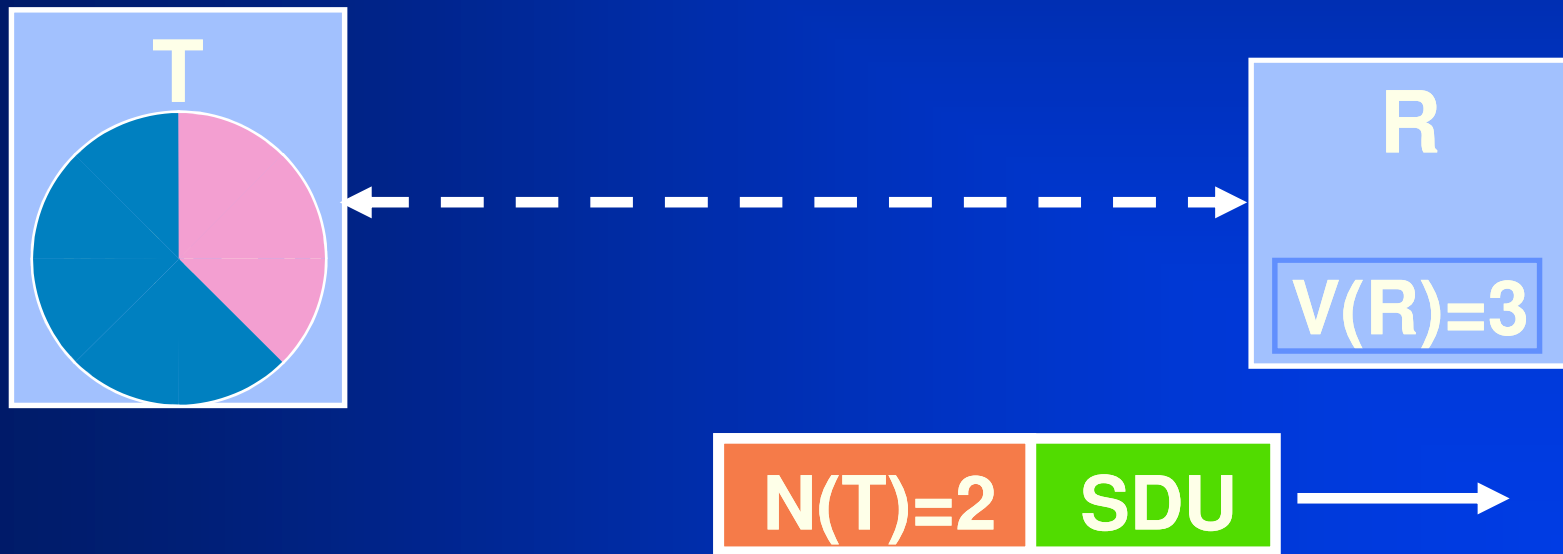
## Ricezione PDU fuori sequenza



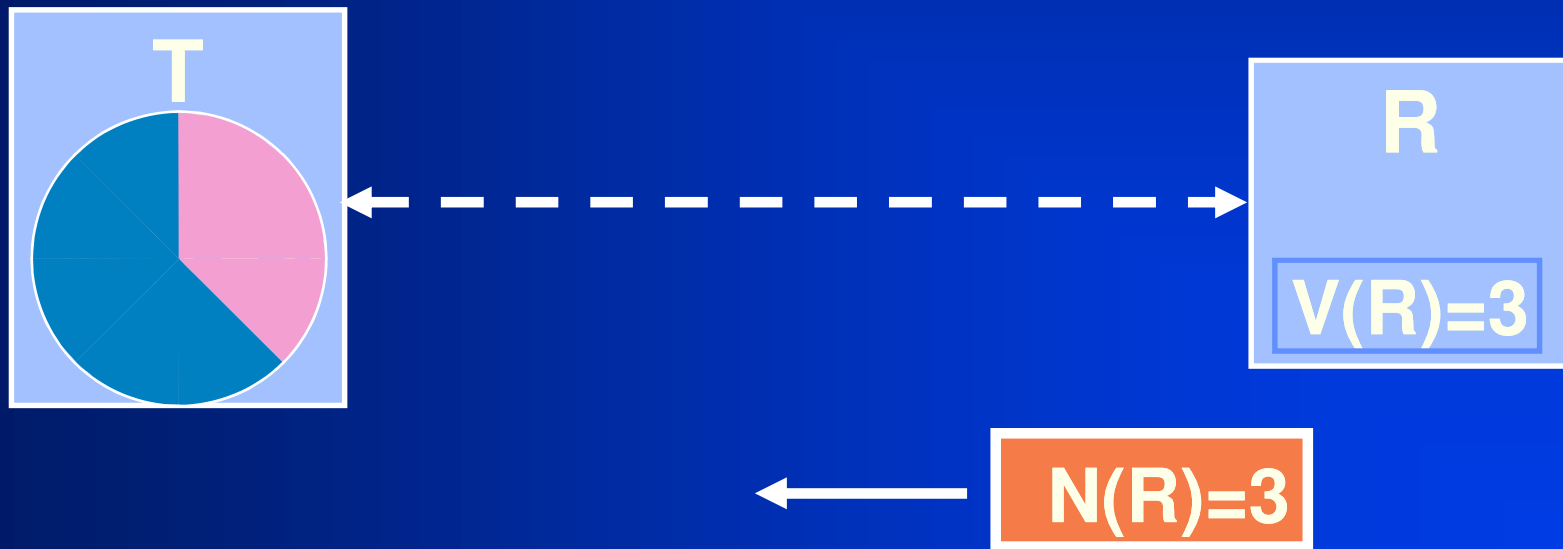
## Ricezione PDU fuori sequenza



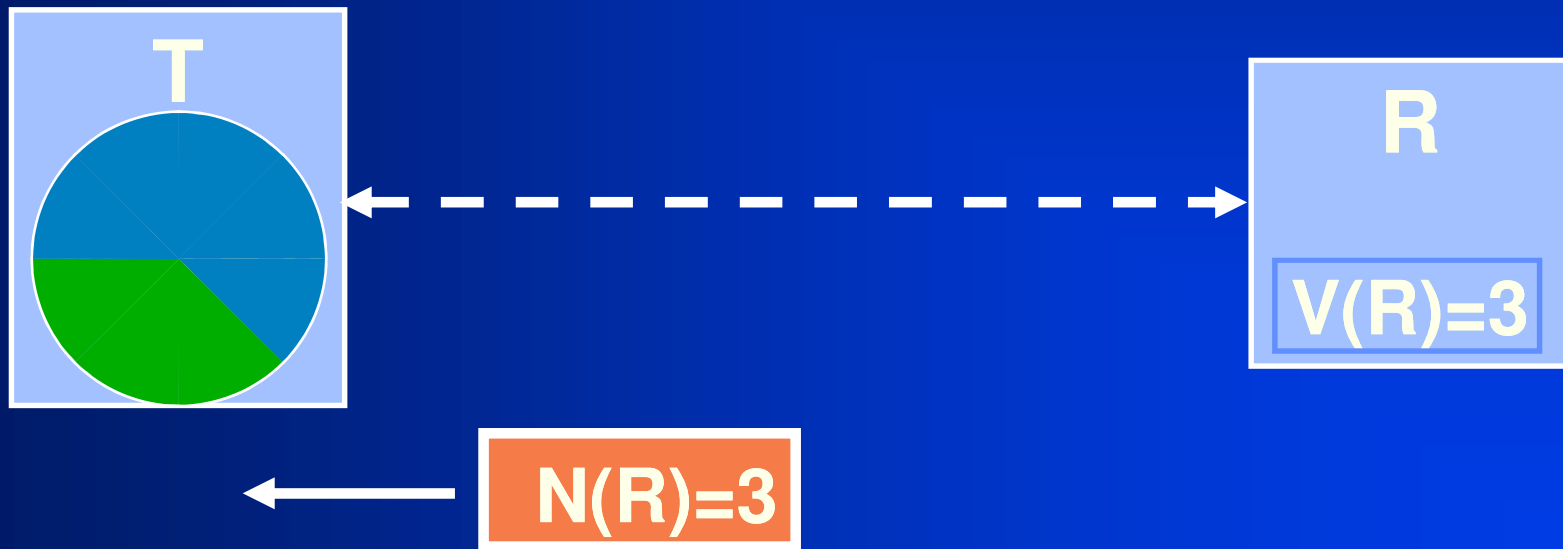
## Ricezione PDU fuori sequenza



## Trasmissione ACK



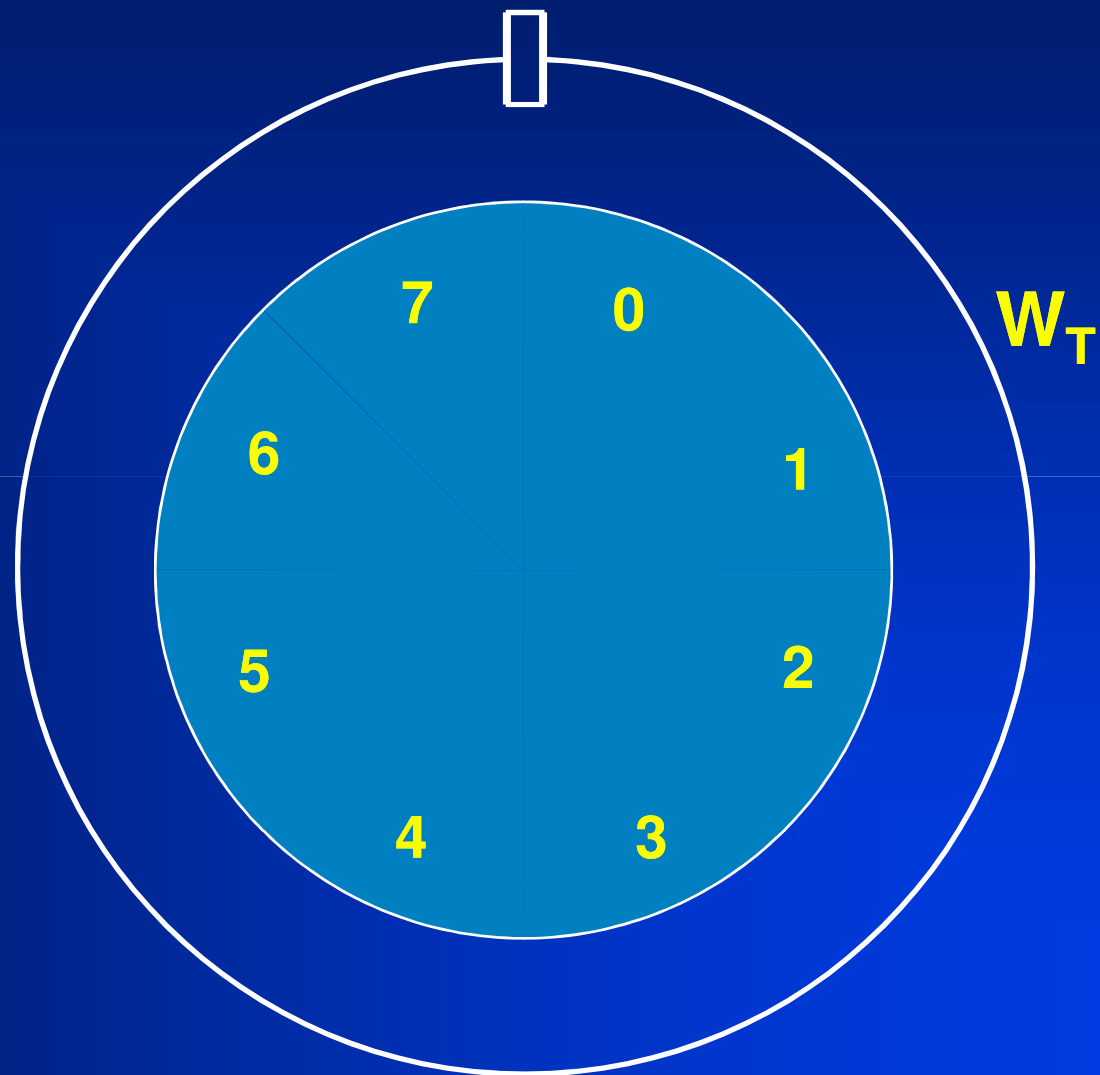
## Ricezione ACK



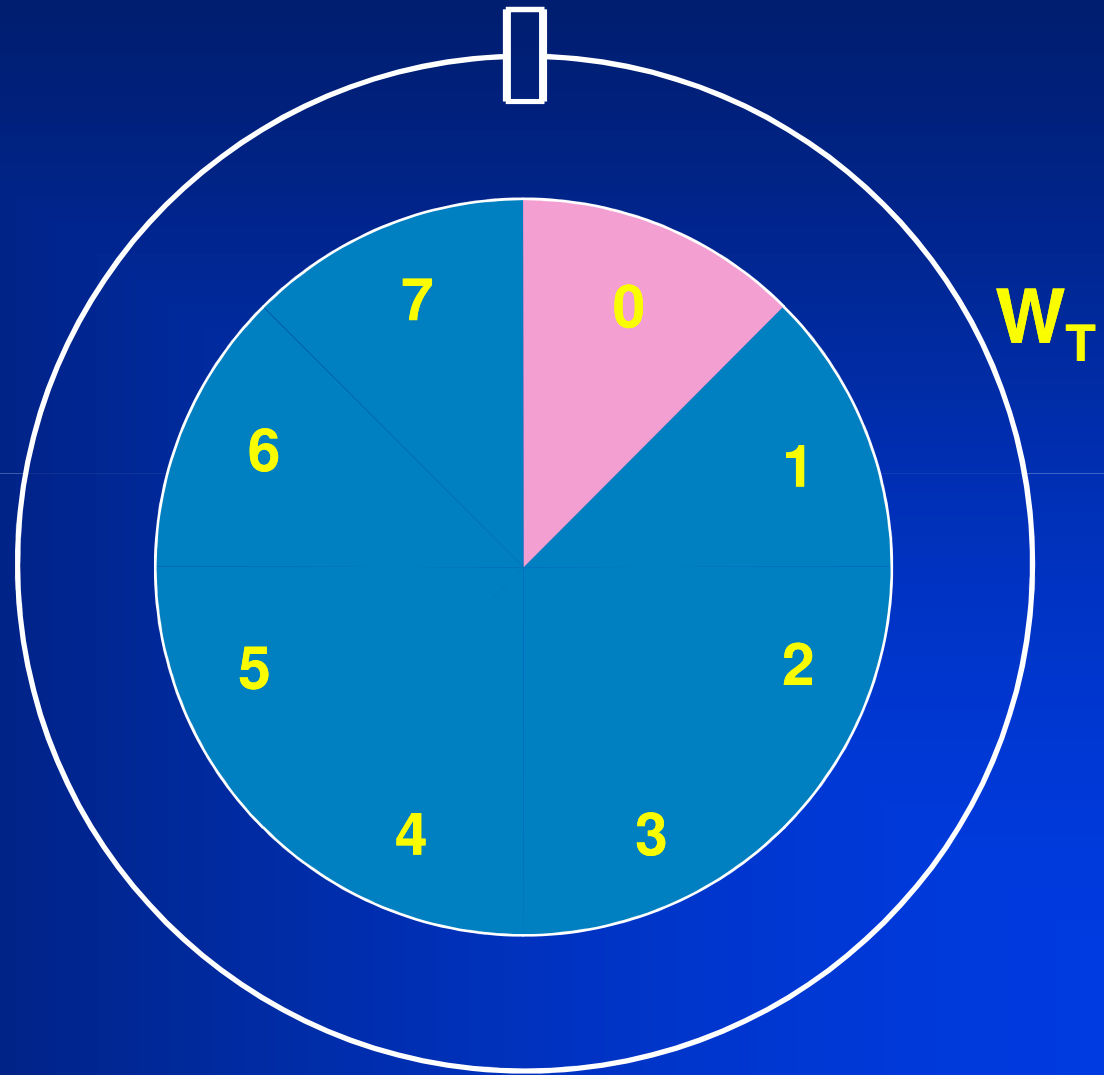


**La finestra di trasmissione non può avere dimensioni arbitrarie**

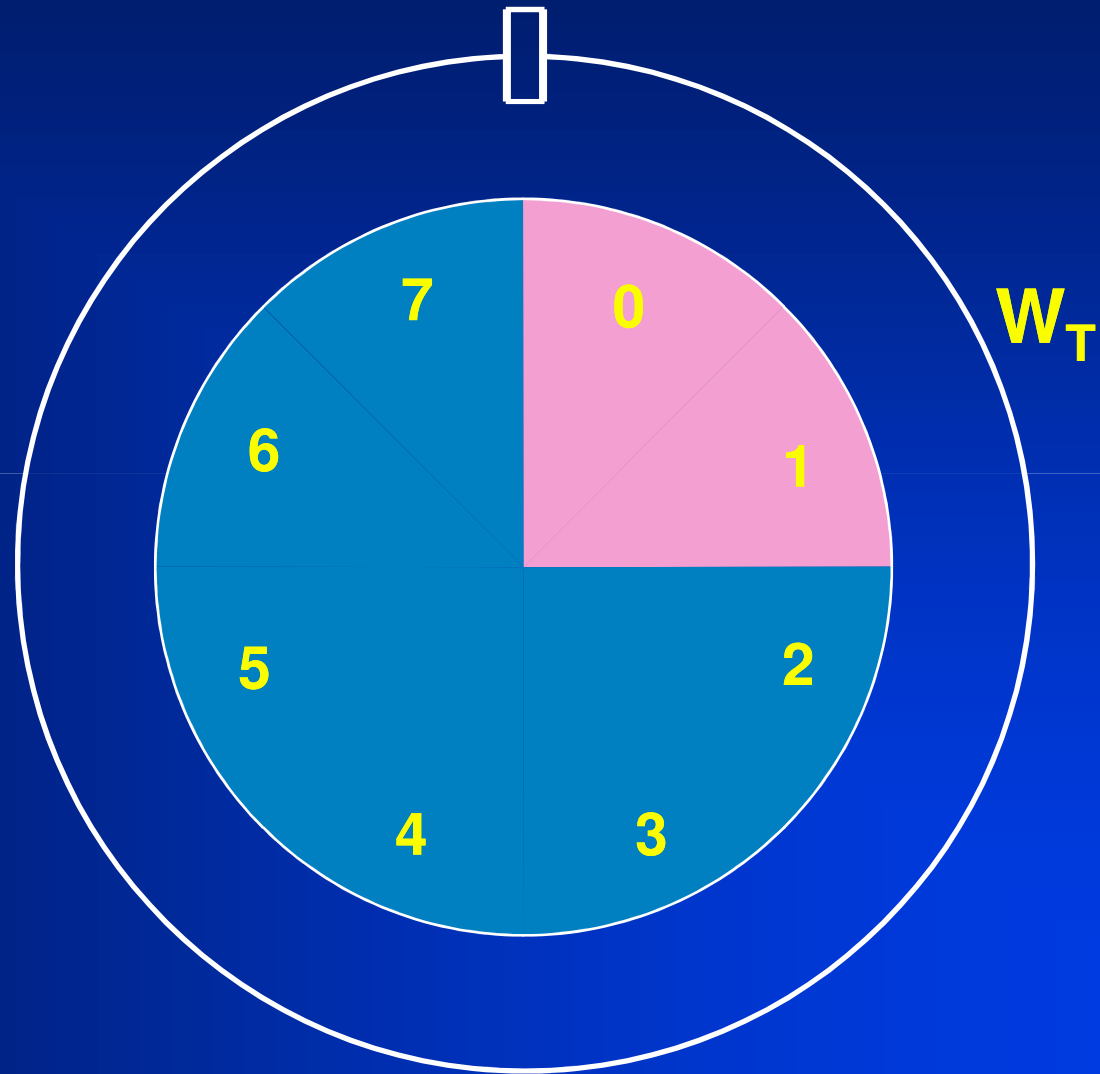
$$W_T < 2^k$$



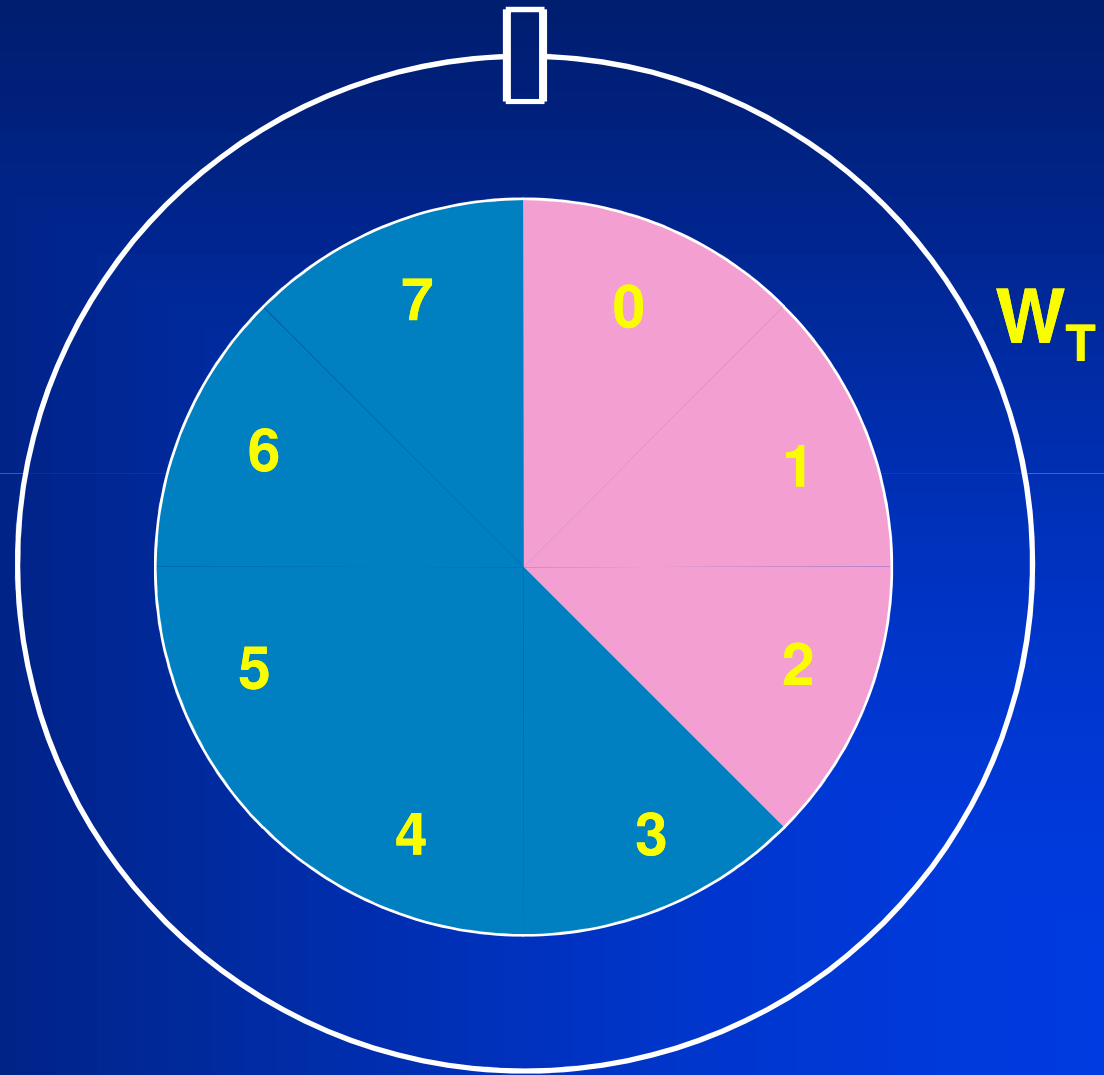
**PDU 0**



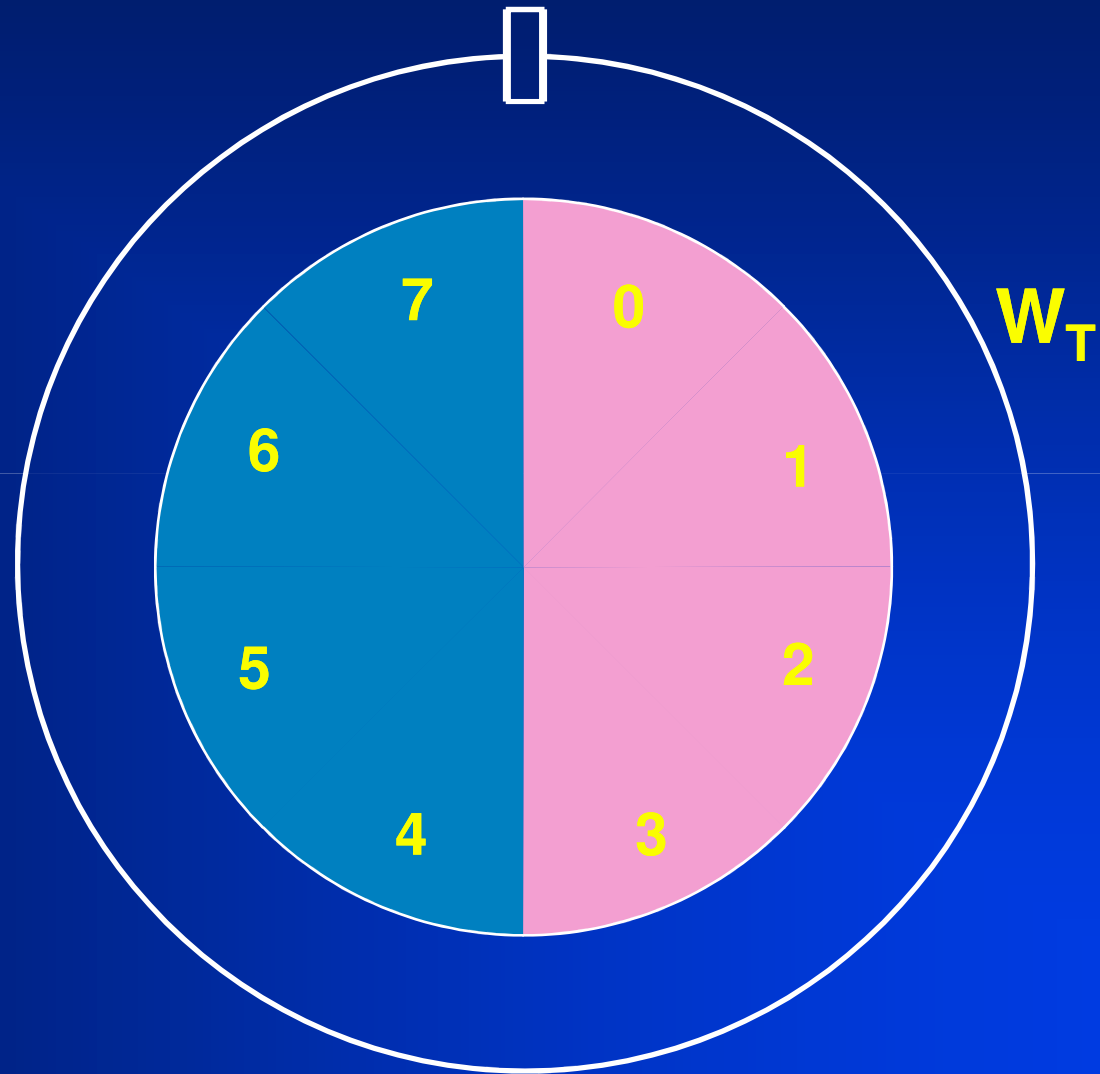
**PDU 1**



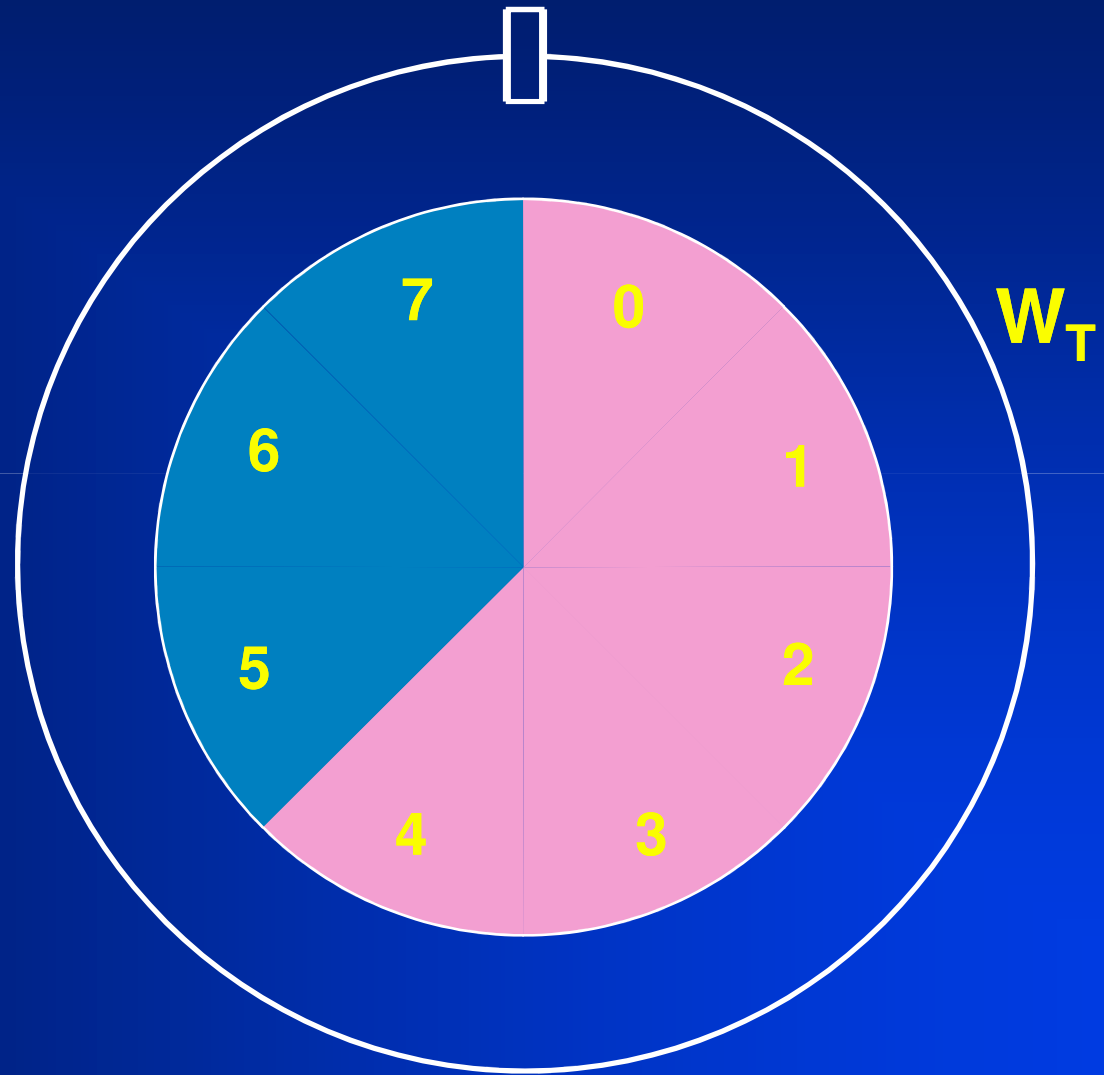
**PDU 2**



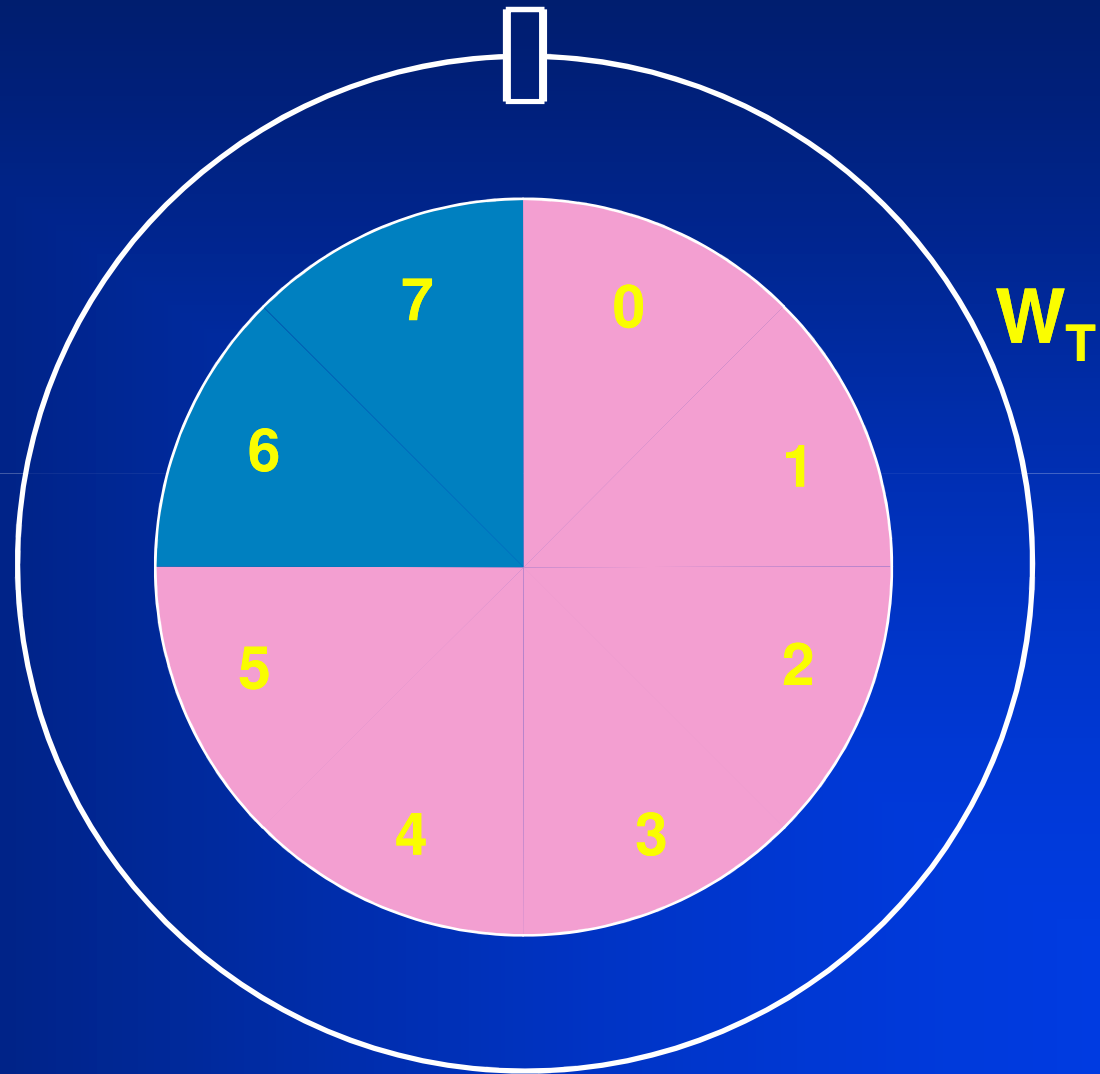
**PDU 3**



**PDU 4**

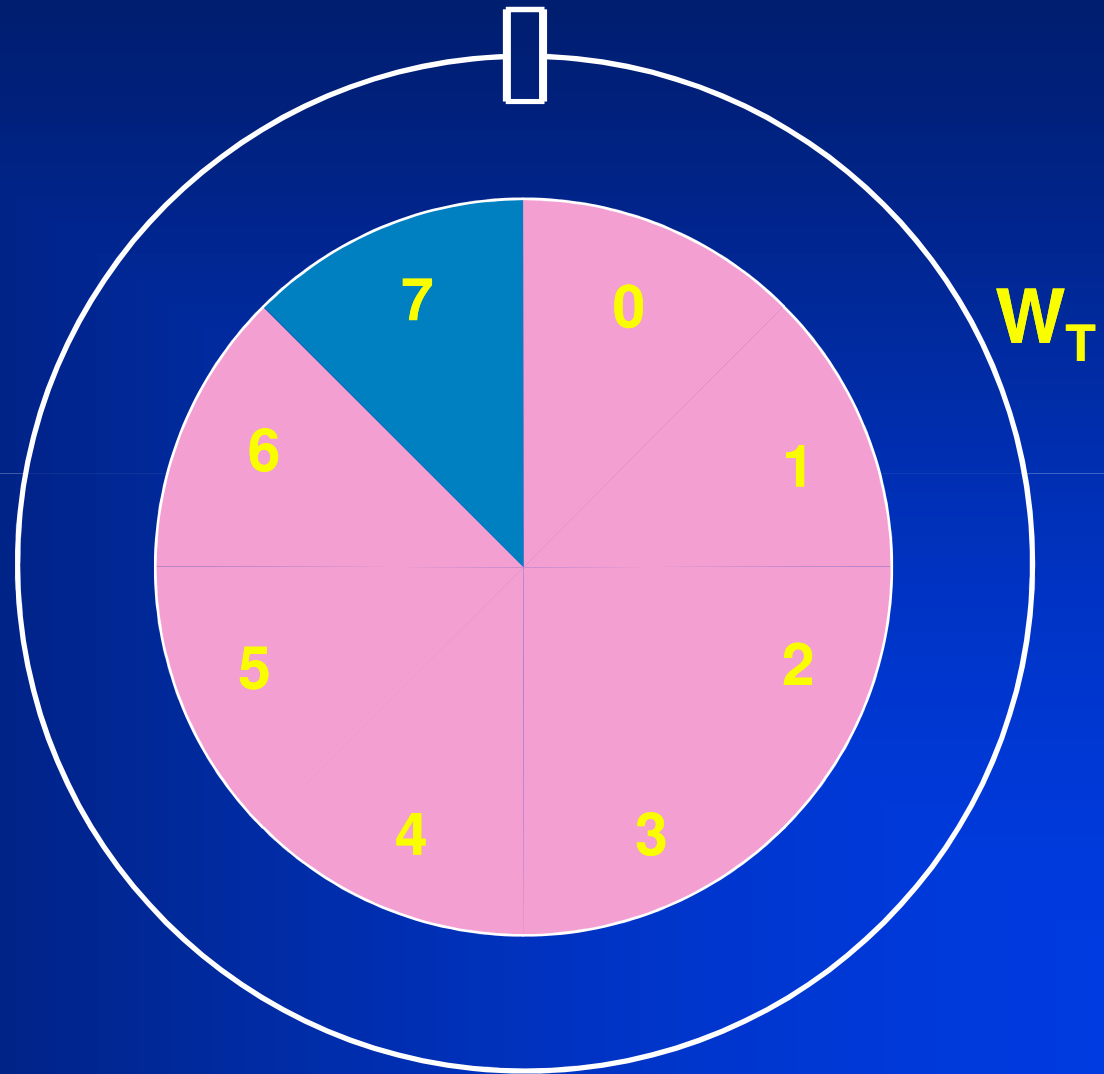


**PDU 5**

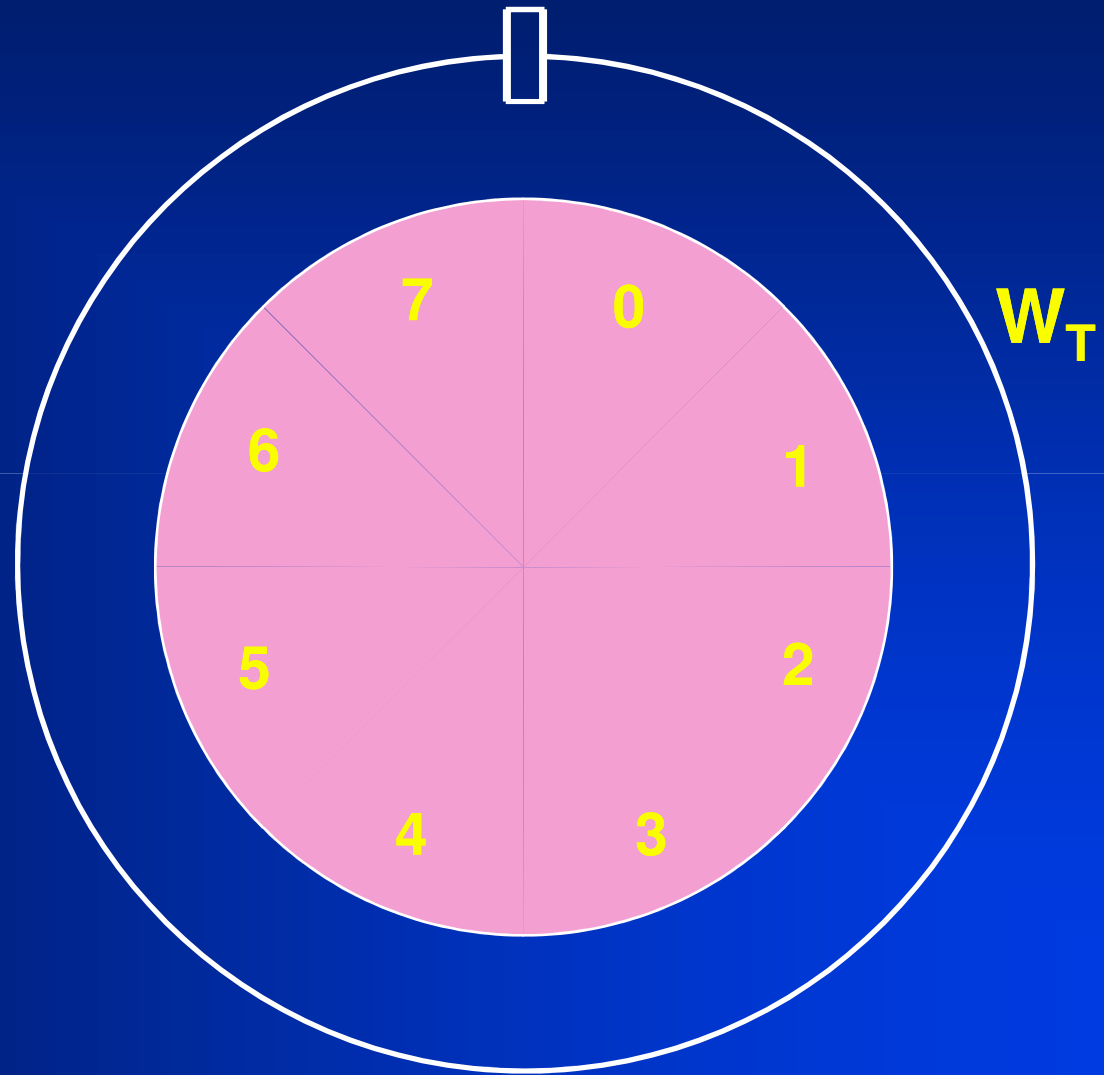




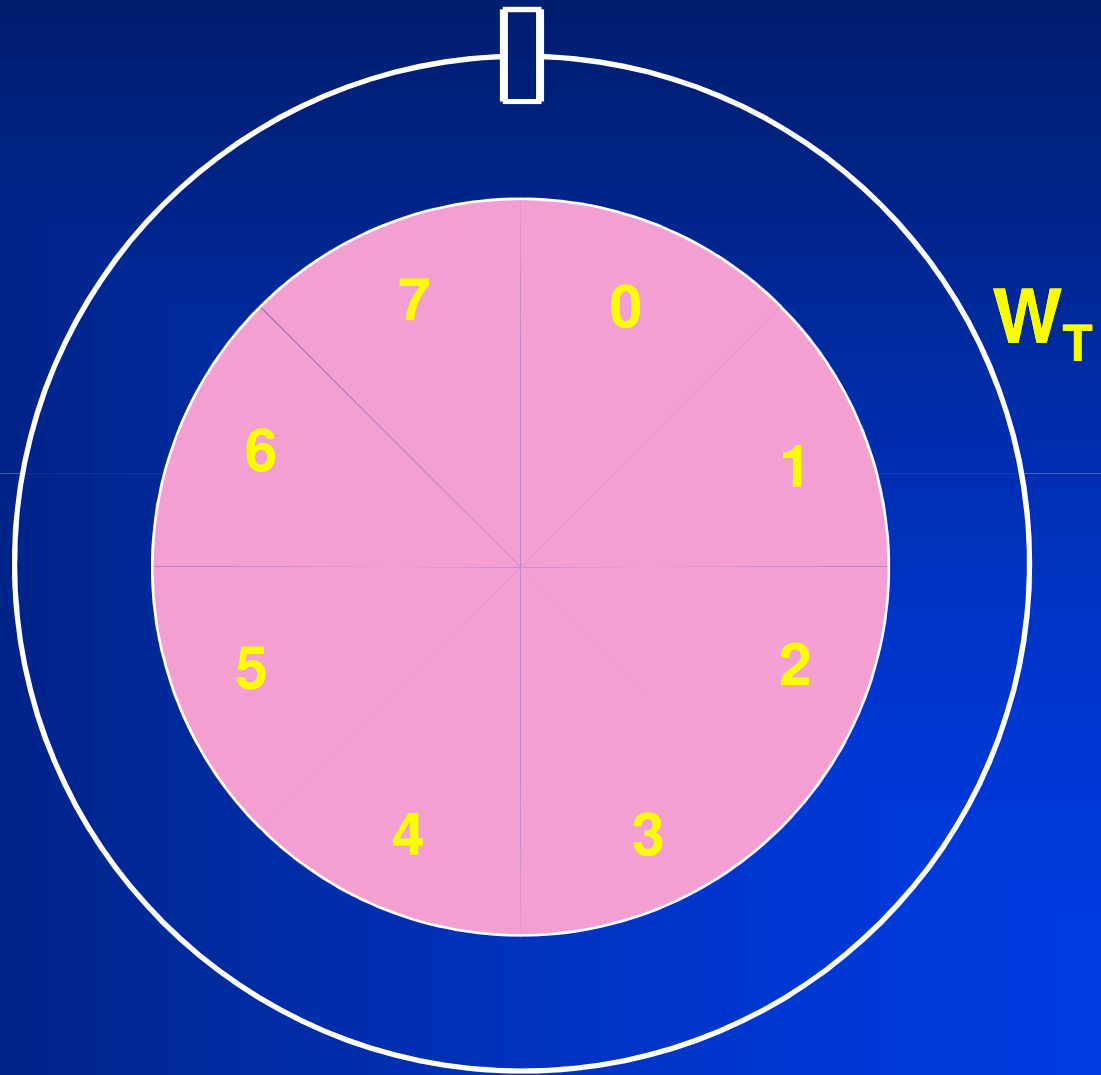
**PDU 6**



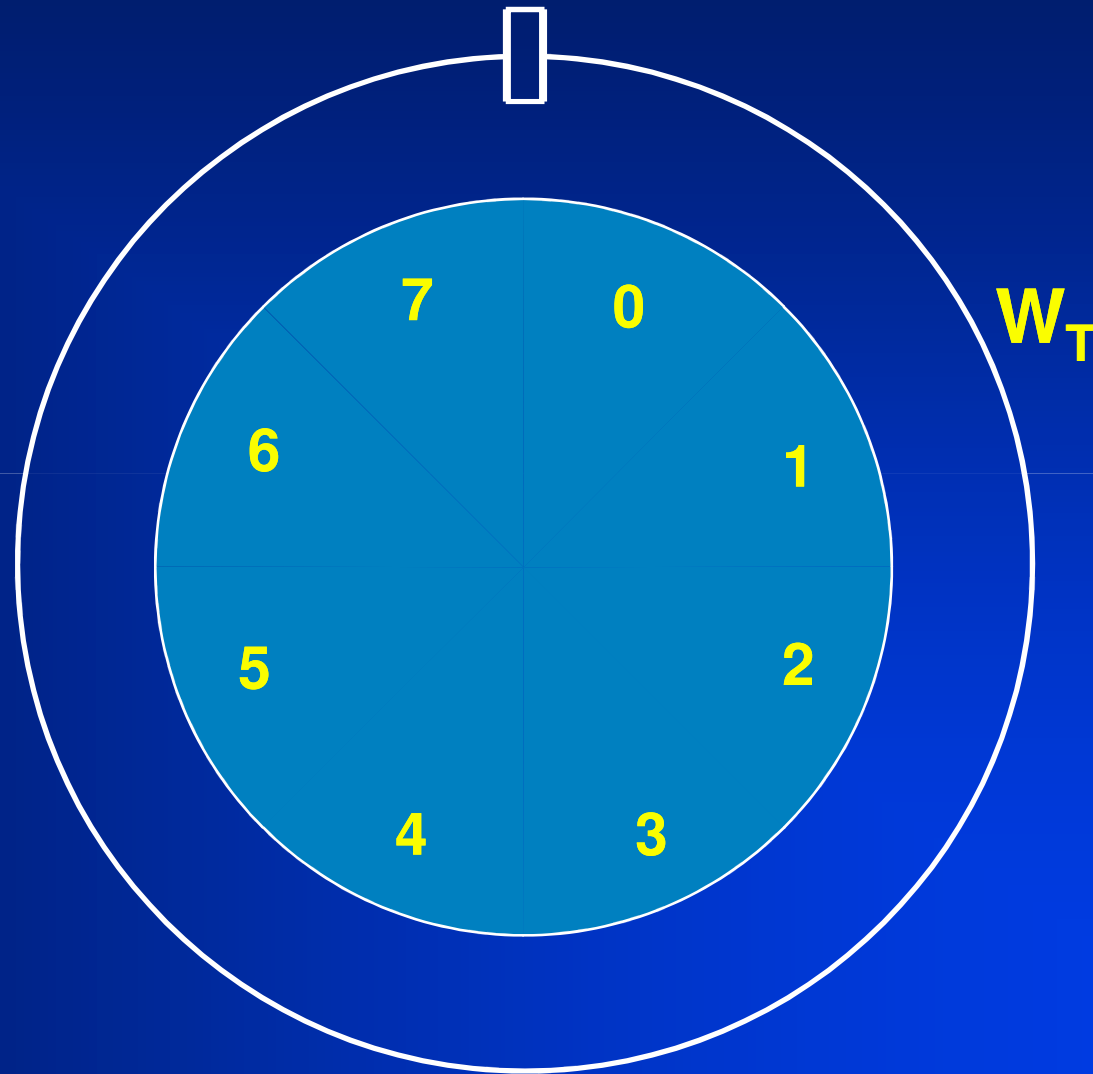
**PDU 7**



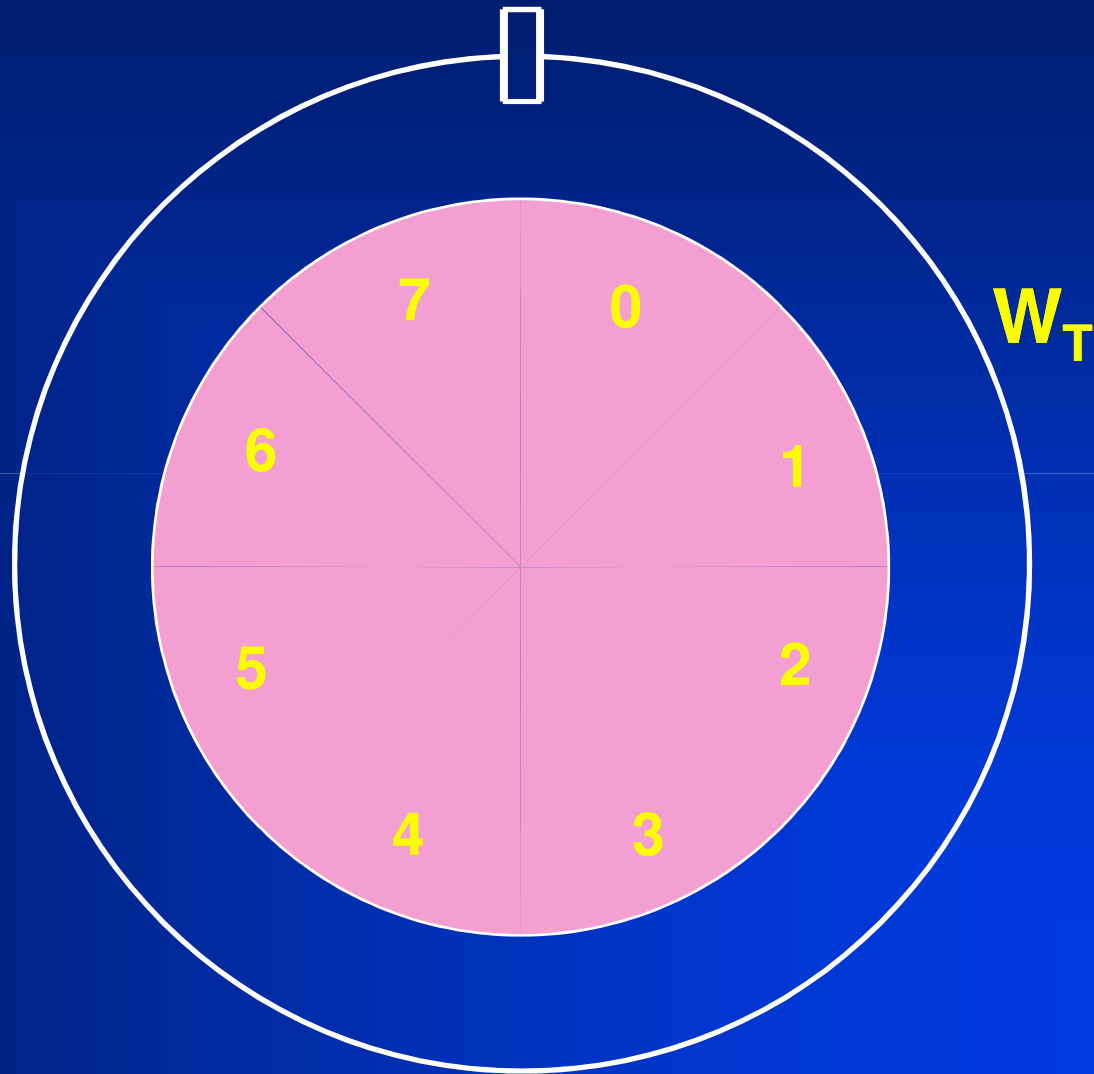
**ACK0**



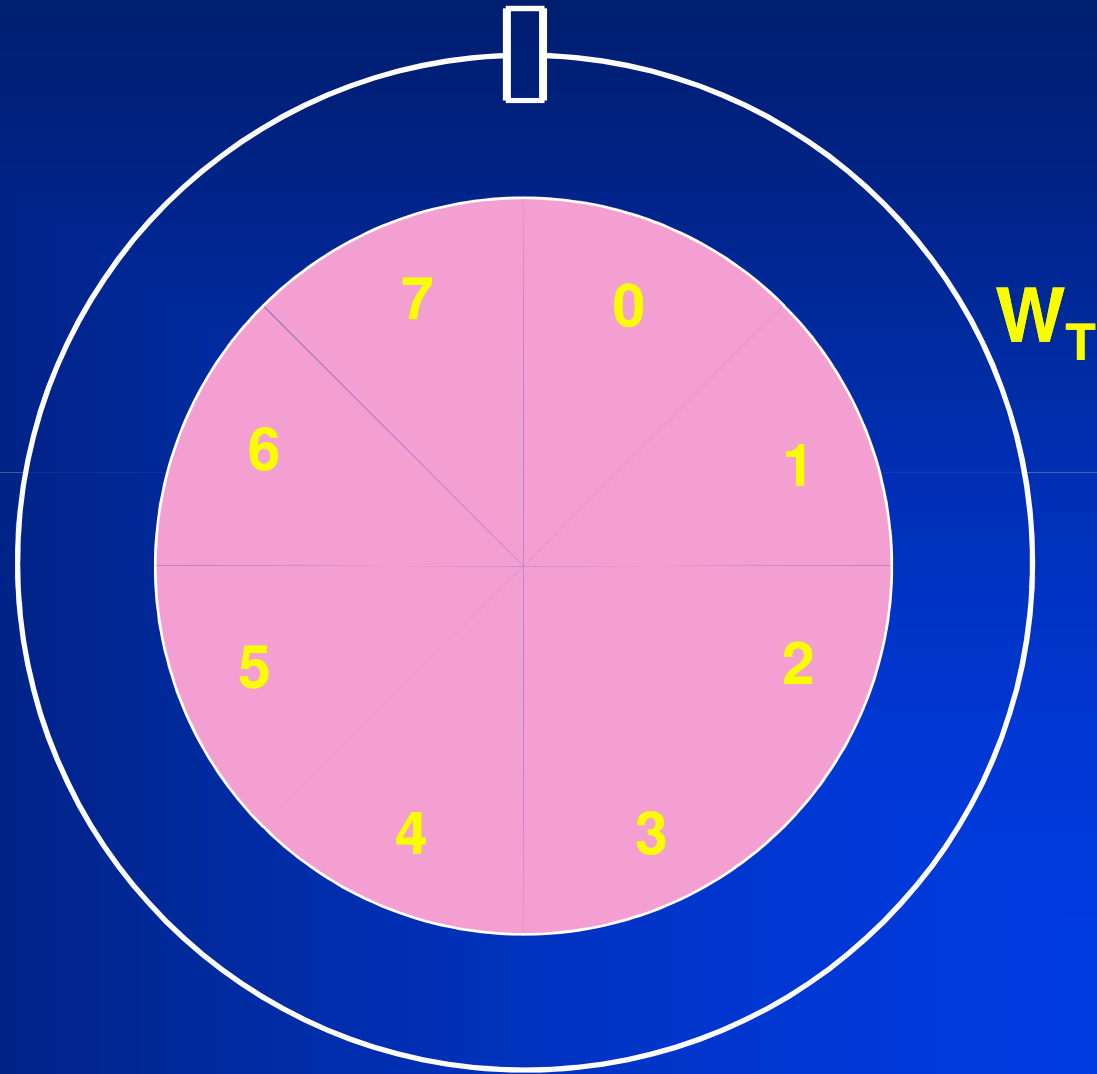
**il trasmettitore  
considera  
l'ack come un  
riscontro  
cumulativo per  
tutto il blocco  
e riapre la  
finestra**



**Trasmette un  
altro blocco  
di 8 PDU:  
PDU0-PDU7**



**ACK0: dilemma**  
- segnala la  
corretta  
ricezione di  
tutto il blocco  
- segnala la  
perdita di tutto  
il blocco?



**Il protocollo Go back N  
è più efficiente dello Stop and wait  
ma richiede**

- memoria**
- più orologi**
- algoritmi più complicati**

**al trasmettitore**

**Se non si verificano errori**

- **più è grande WT migliore  
è l'efficienza**



**La complessità del ricevitore  
rimane inalterata**

**La complessità del ricevitore  
rimane inalterata**

**La finestra di ricezione ha ancora  
dimensione 1**

**La complessità del ricevitore  
rimane inalterata**

**La finestra di ricezione ha ancora  
dimensione 1**

**Gli ACK cumulativi però richiedono  
un orologio**

**Quando il go back-N non  
funziona in modo efficiente ?**

**Quando la dimensione della finestra e il ritardo end-to-end sono entrambi grandi.**

**In questo caso, possono trovarsi molti pacchetti in transito; un errore su un solo pacchetto può costringere il GBN a ritrasmettere un gran numero di pacchetti, molti dei quali potrebbero non essere necessari.**

**All'aumentare della probabilità di errore sul canale questo può diventare saturo di ritrasmissioni.**

**Nel protocollo Go back N il ricevitore può accettare solo PDU in sequenza**

**Accettare PDU corrette, ma fuori sequenza, migliora le prestazioni**

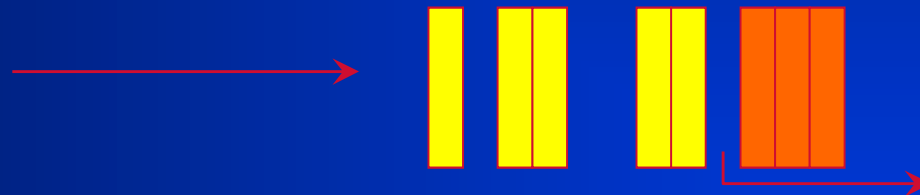
**Selective repeat**

**Il Selective repeat chiede la ritrasmissione di soli quei pacchetti arrivati errati o non arrivati.**

**Il receiver riscontra un pacchetto ricevuto correttamente, che sia o meno in ordine. I pacchetti fuori ordine sono archiviati in memoria finché non siano ricevuti tutti i pacchetti persi (con numeri di sequenza più bassi), quindi il gruppo di pacchetti è inviato in ordine allo strato superiore.**

# *Riordino delle trame*

- occorre mantenere un buffer in ricezione e inoltrare ai livelli superiori solo le sequenze corrette



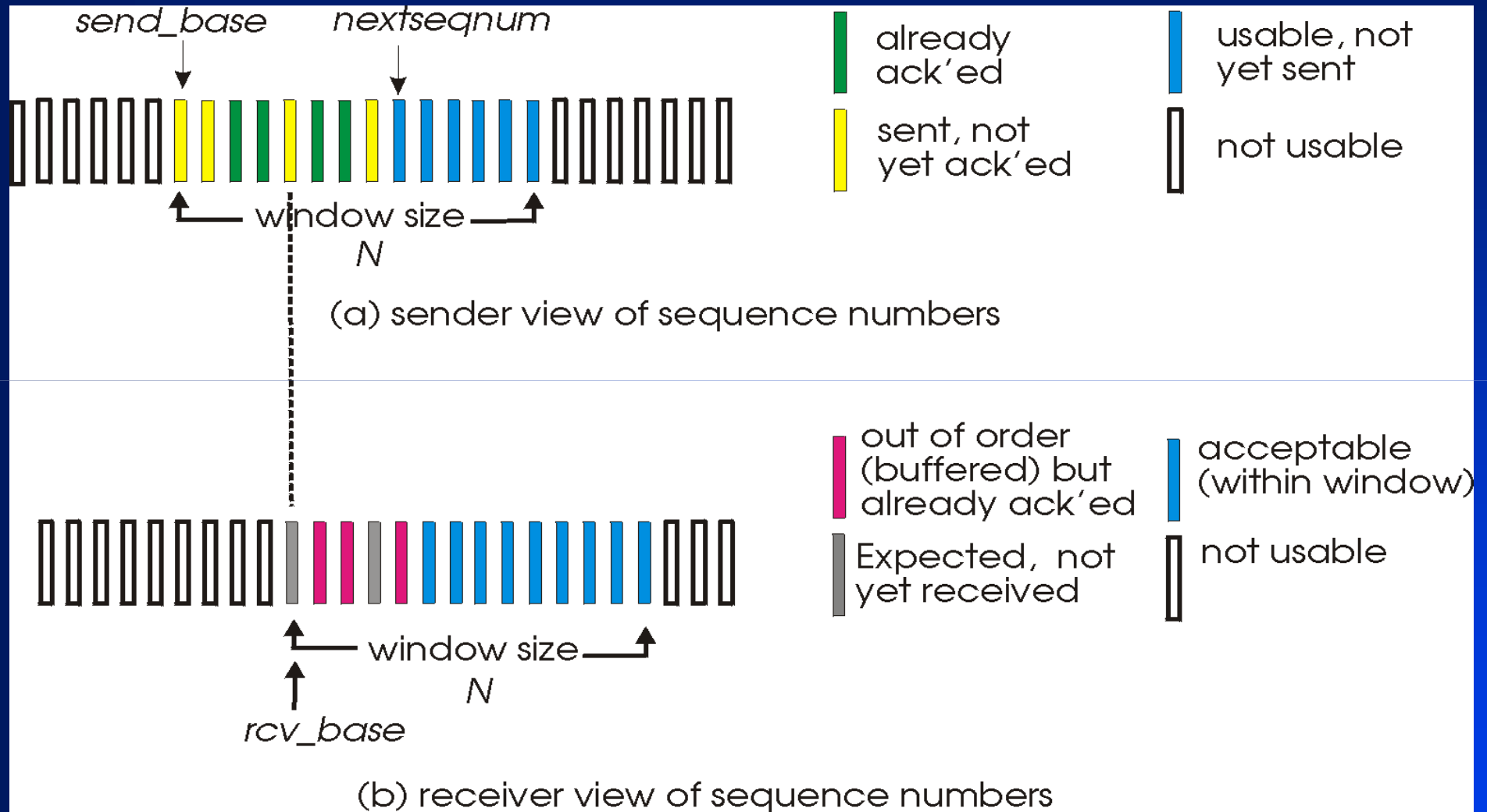


## **Il protocollo Selective repeat usa**

- finestra di trasmissione**
- finestra di ricezione**

**di dimensioni maggiori di 1**

## Selective repeat: finestre del sender e del receiver



**Le finestre di trasmissione e ricezione non devono necessariamente essere sincronizzate!**

# Selective repeat dilemma

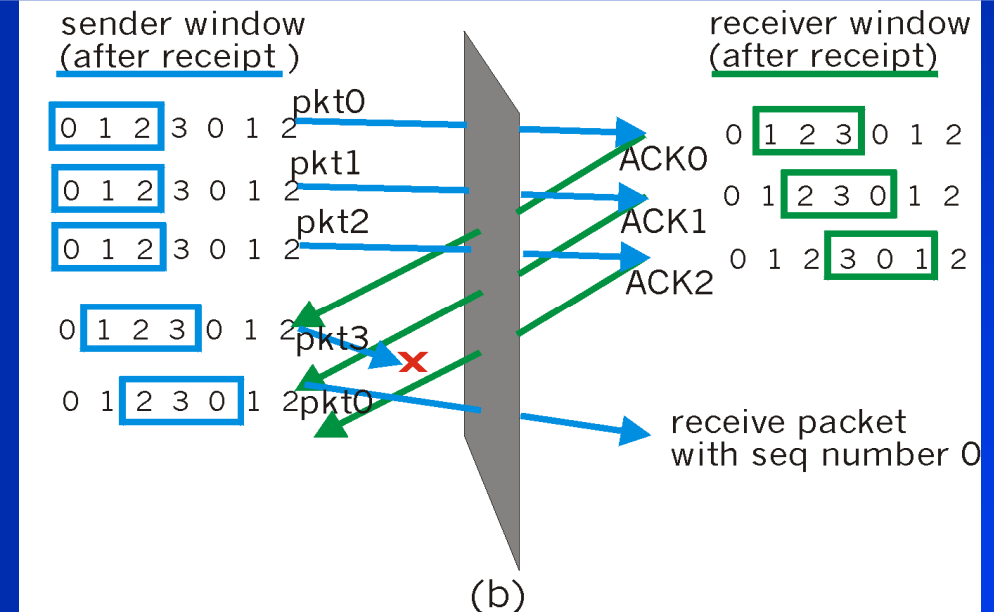
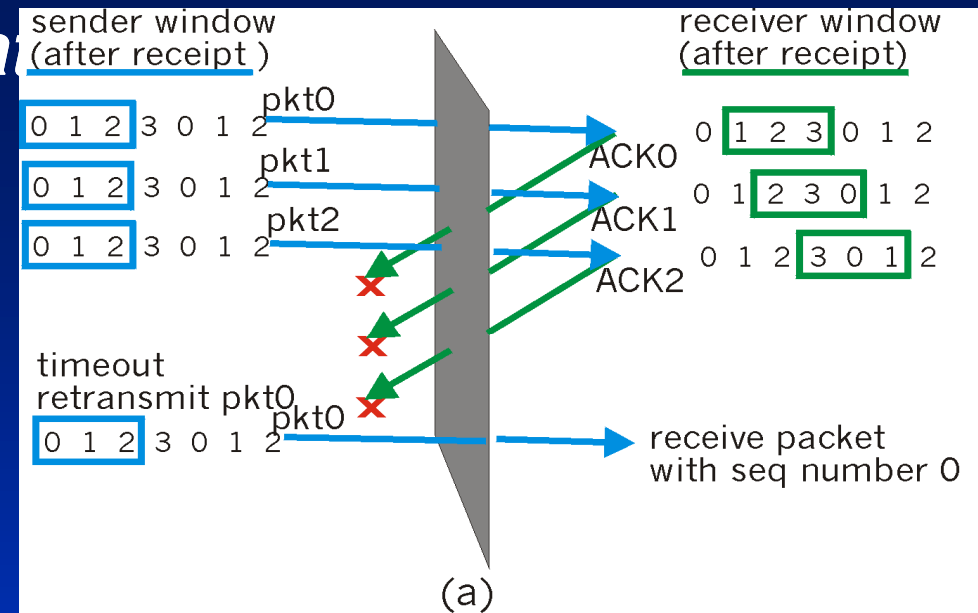
Esempio:

Num. Seq: 0, 1, 2, 3

WT=3

- Il receiver non vede differenza tra i due scenari!
- Erroneamente passa i dati duplicati come fossero nuovi (in (a))

Q: che relazione deve esserci tra il max numero di sequenza e la dimensione della finestra?



In questo caso deve essere

$$W_T + W_R \leq 2^k$$

## Selective repeat

### il trasmettitore

- invia fino ad N PDU
- attiva un orologio per ogni PDU
- si pone in attesa delle conferme di ricezione (ACK)
- se scade un timeout prima dell'arrivo della conferma, ripete la trasmissione della PDU

## Selective repeat

### il ricevitore

- riceve una PDU
- controlla la correttezza della PDU
- controlla il numero di sequenza

## Selective repeat

**il ricevitore**

- **se la PDU è corretta ed in sequenza  
invia un ACK**

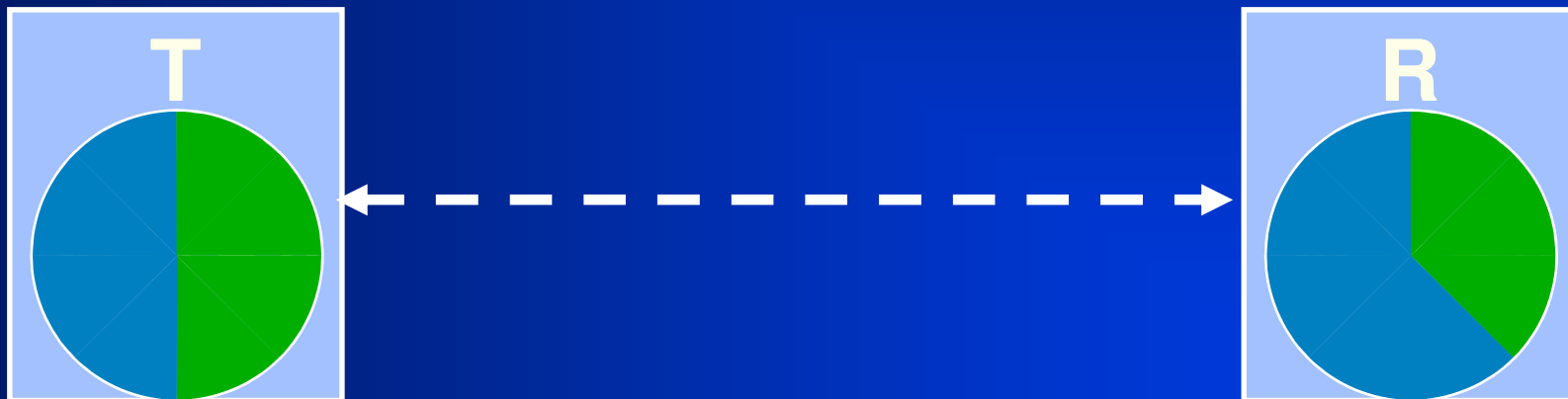
## Selective repeat

### **il ricevitore**

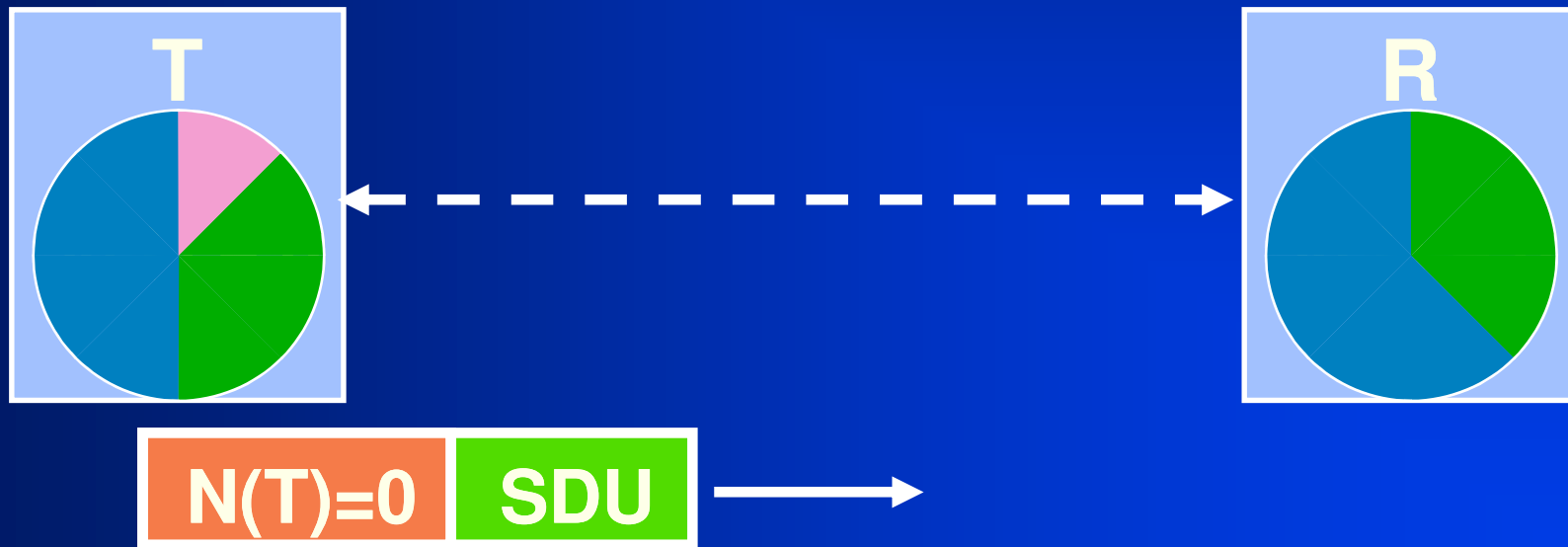
- **se la PDU è corretta ma non in sequenza**
  - **se è entro la finestra di ricezione la memorizza ed invia un ACK relativo all'ultima PDU ricevuta in sequenza**
  - **se è fuori dalla finestra di ricezione la scarta ed invia un ACK relativo all'ultima PDU ricevuta in sequenza**



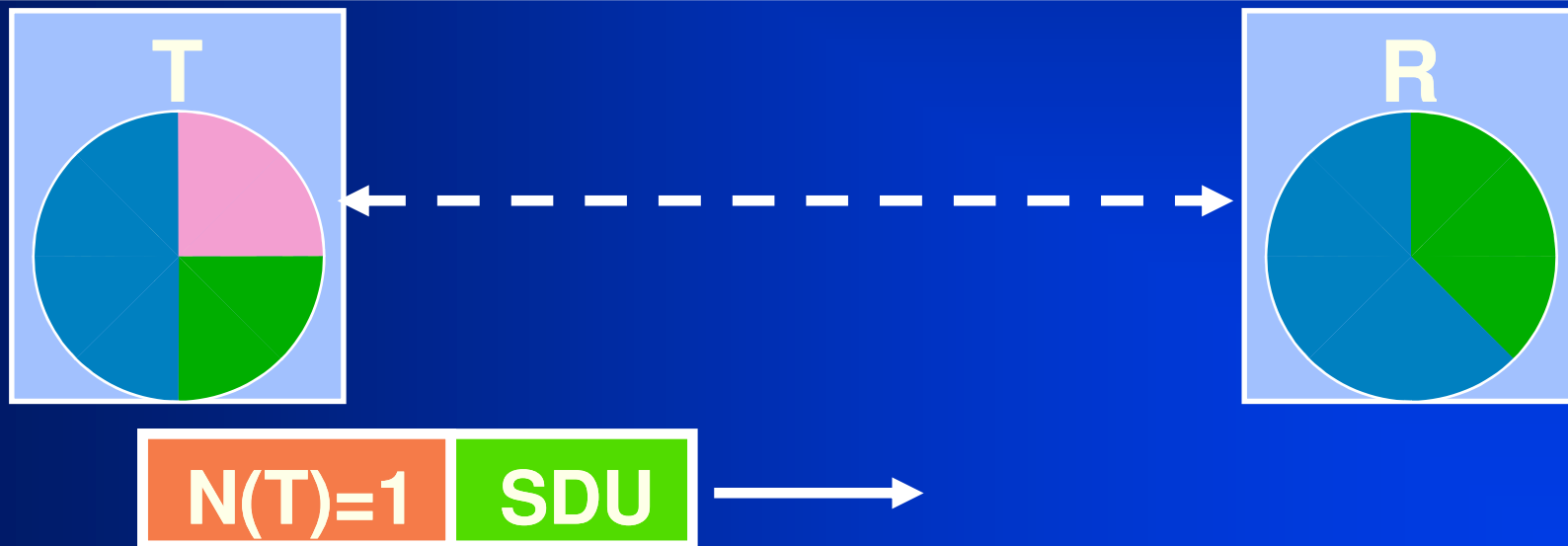
# Inizializzazione



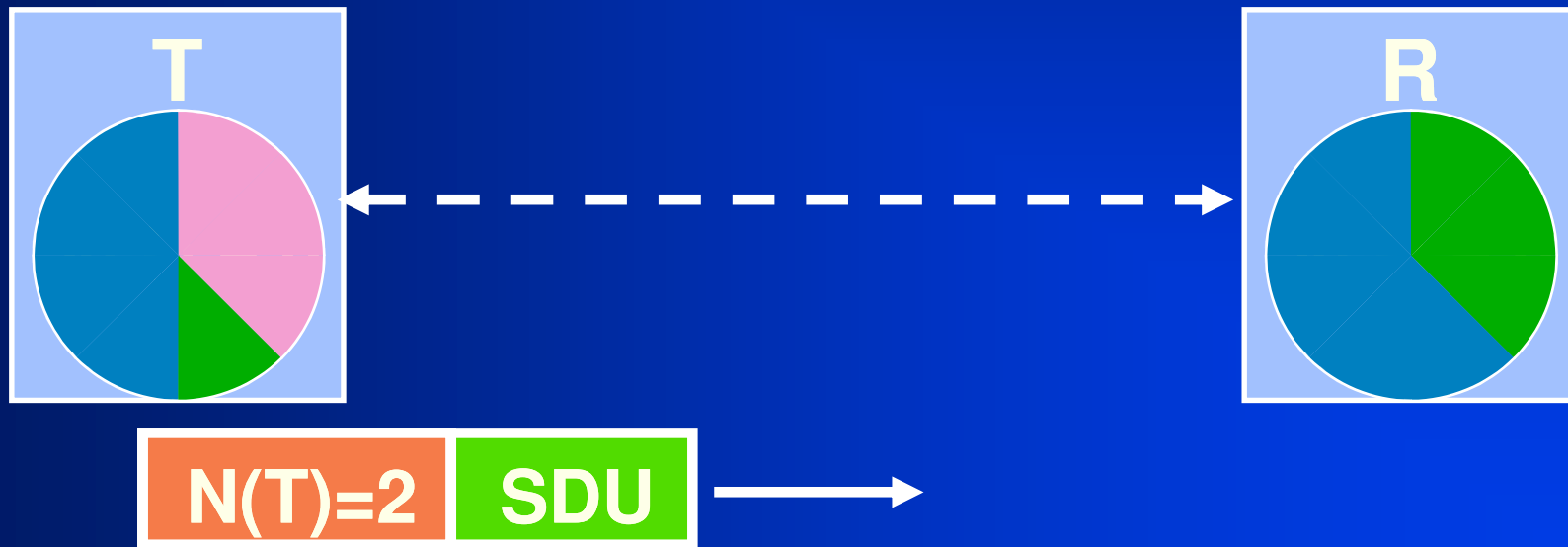
## Trasmissione PDU



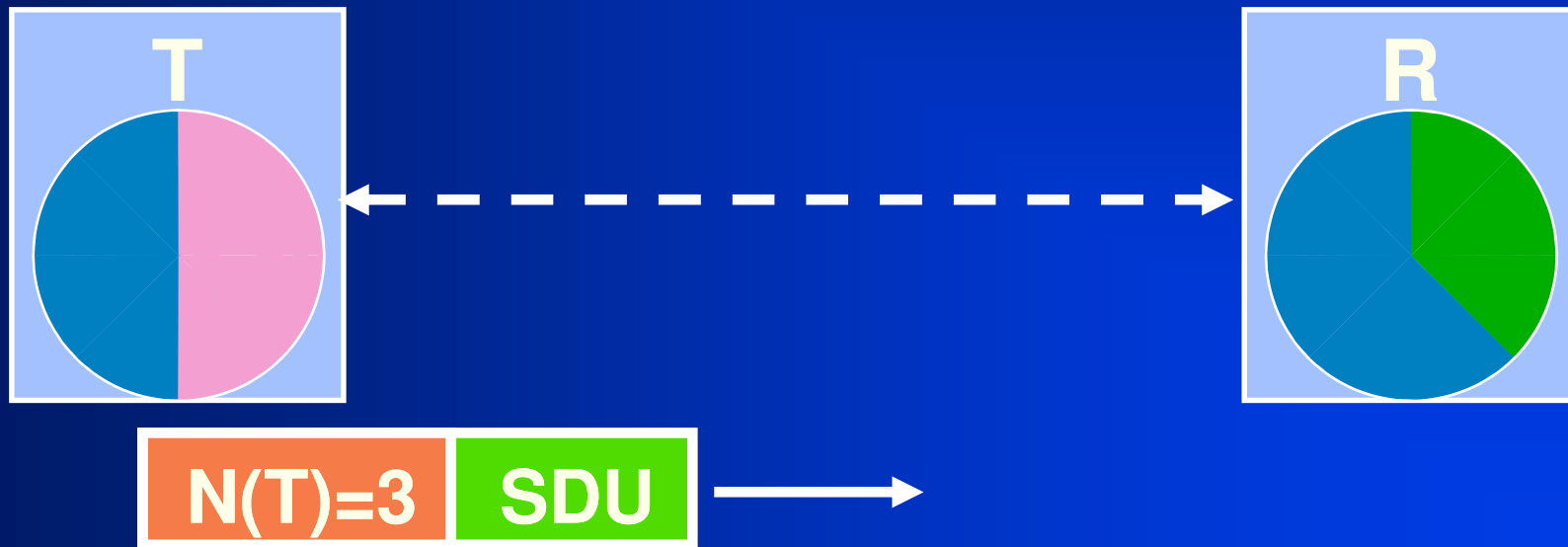
## Trasmissione PDU



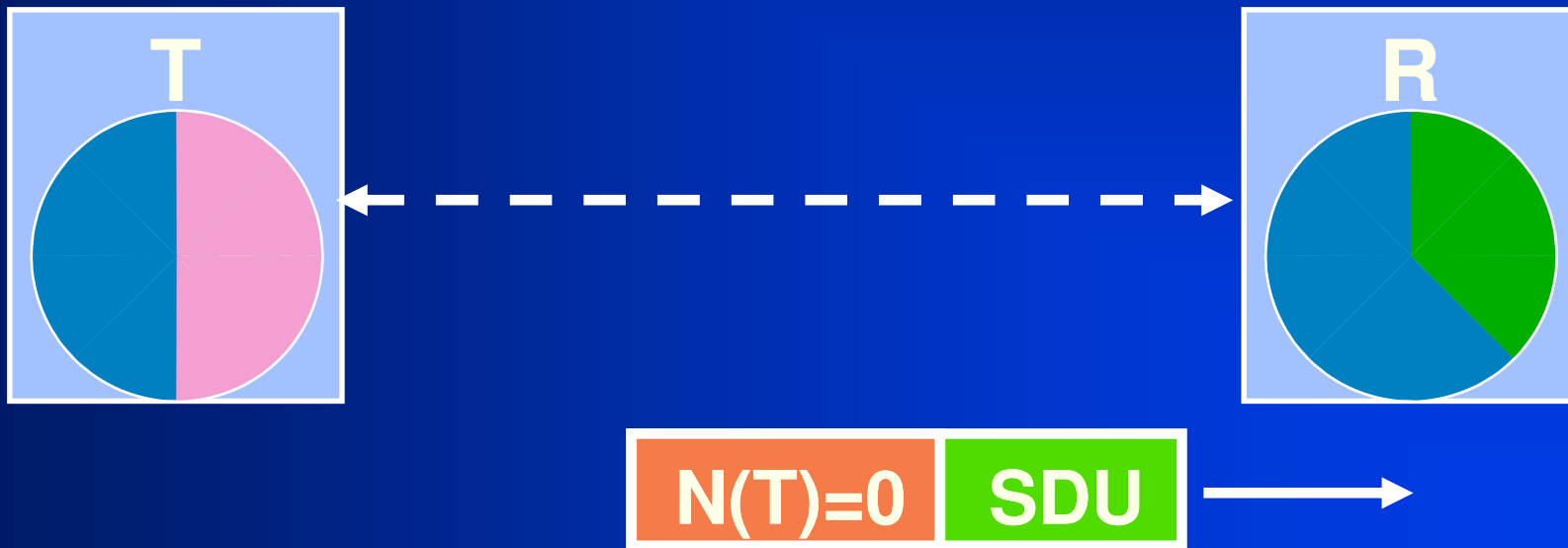
## Trasmissione PDU



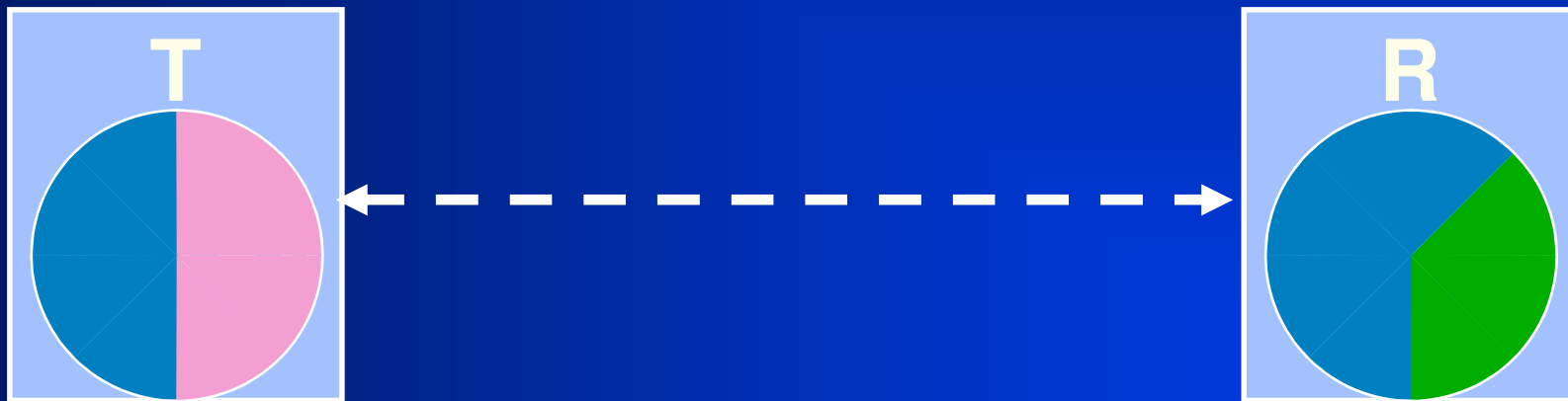
## Trasmissione PDU



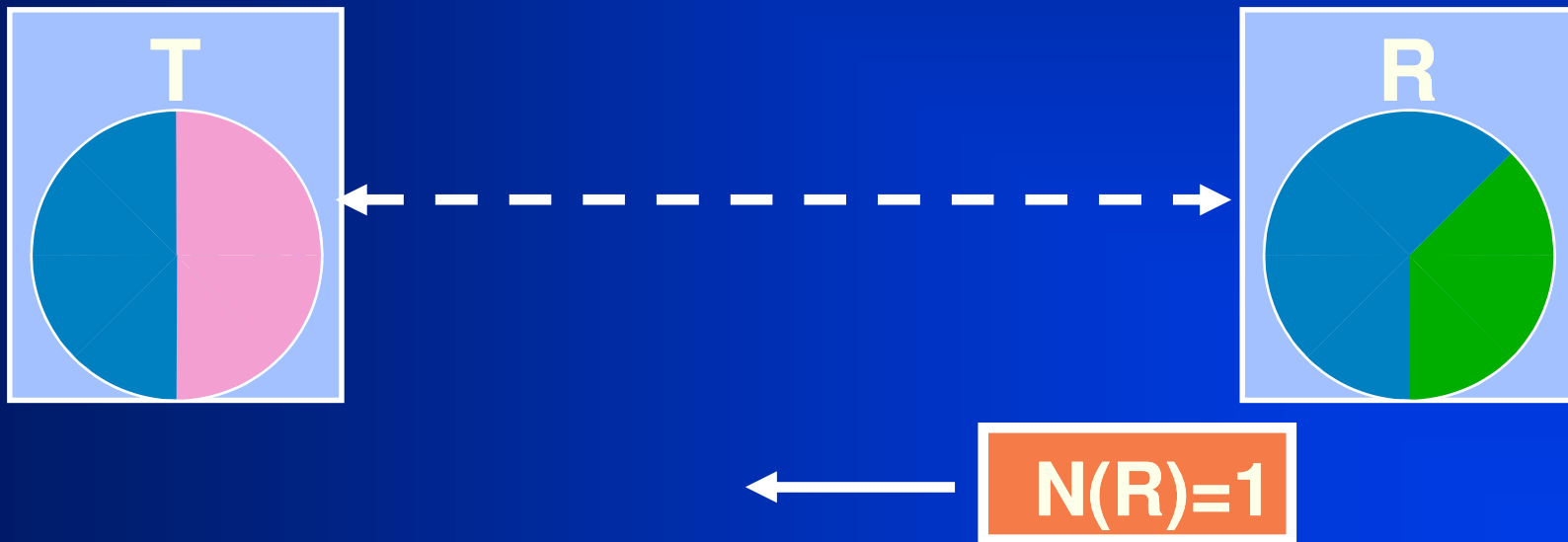
## Ricezione PDU corretta in sequenza



## Ricezione PDU corretta in sequenza

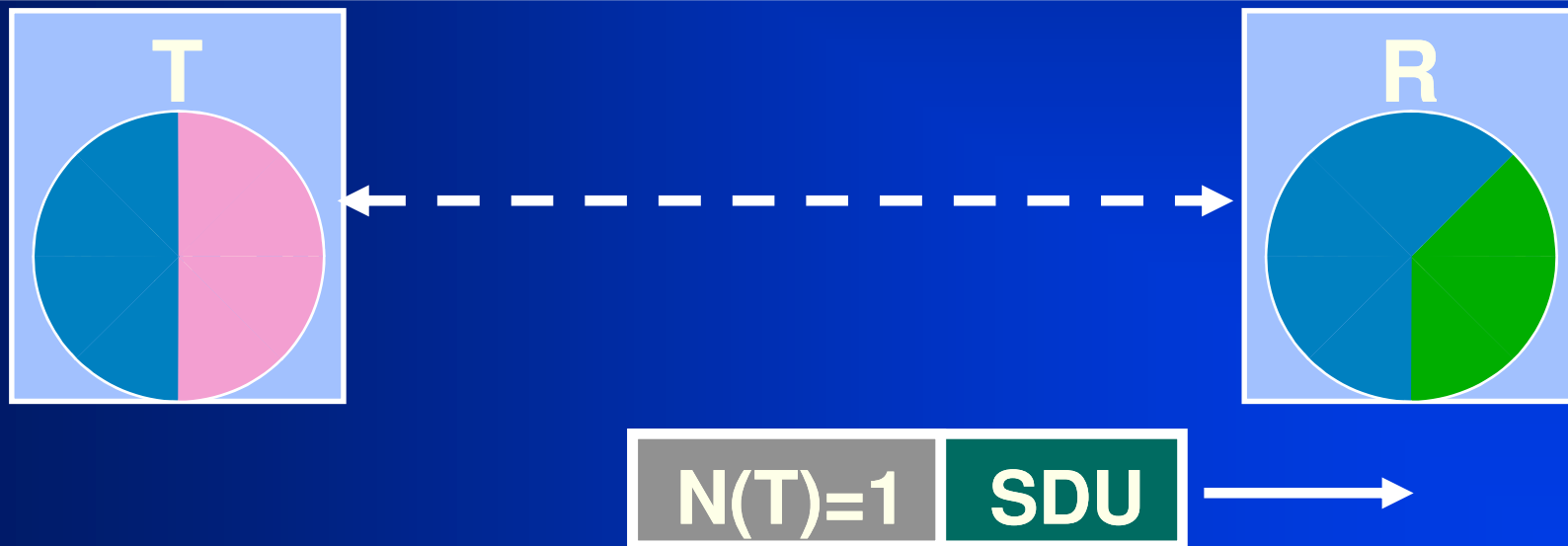


## Trasmissione ACK

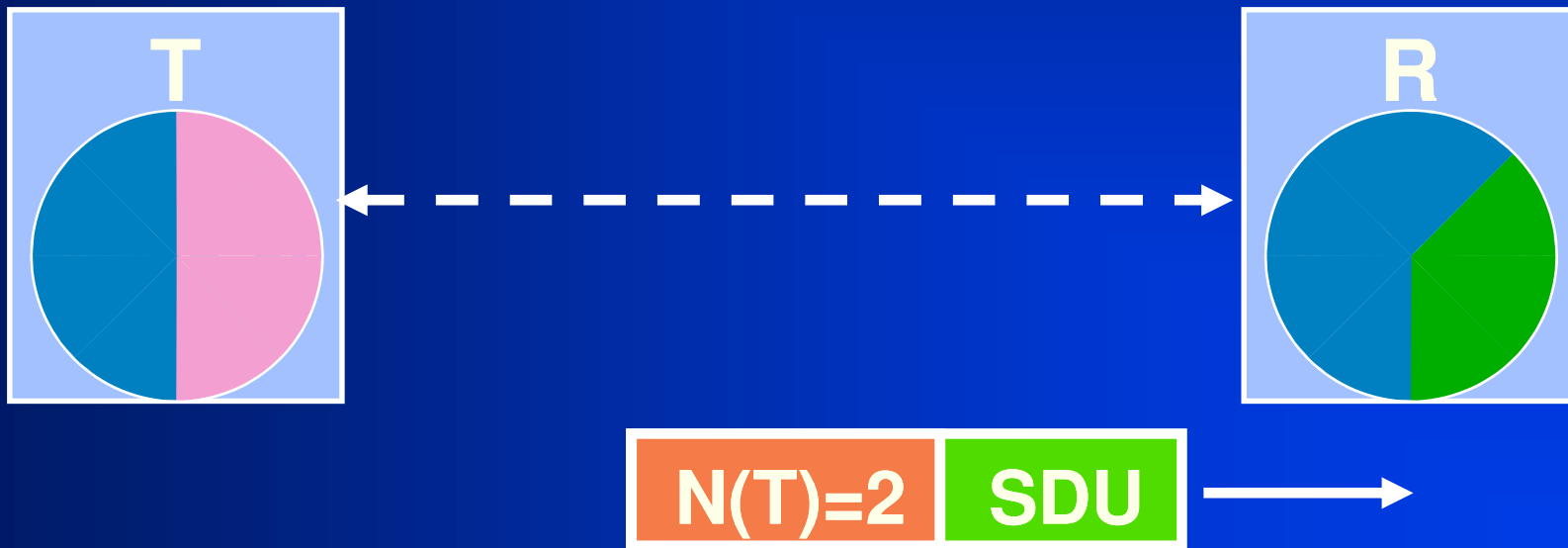




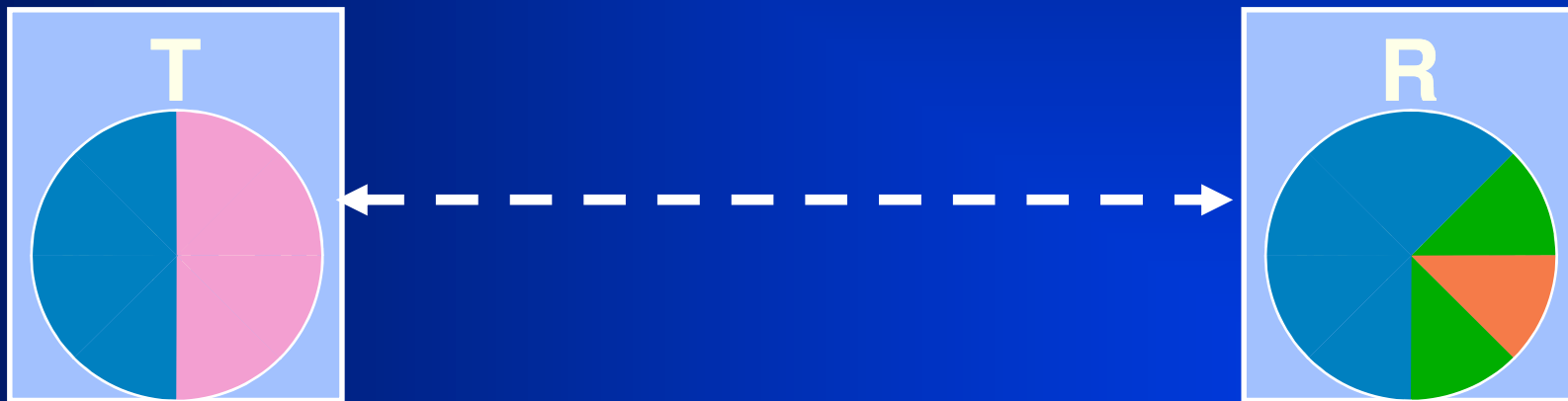
## Ricezione PDU errata



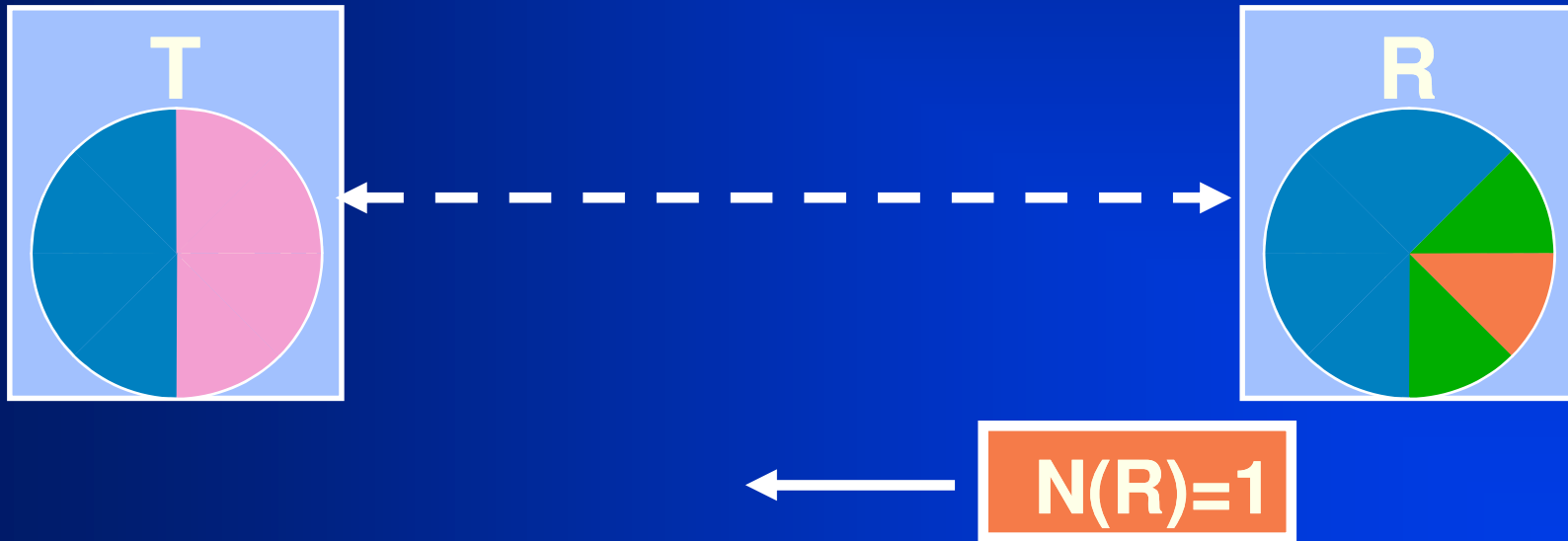
## Ricezione PDU corretta fuori sequenza



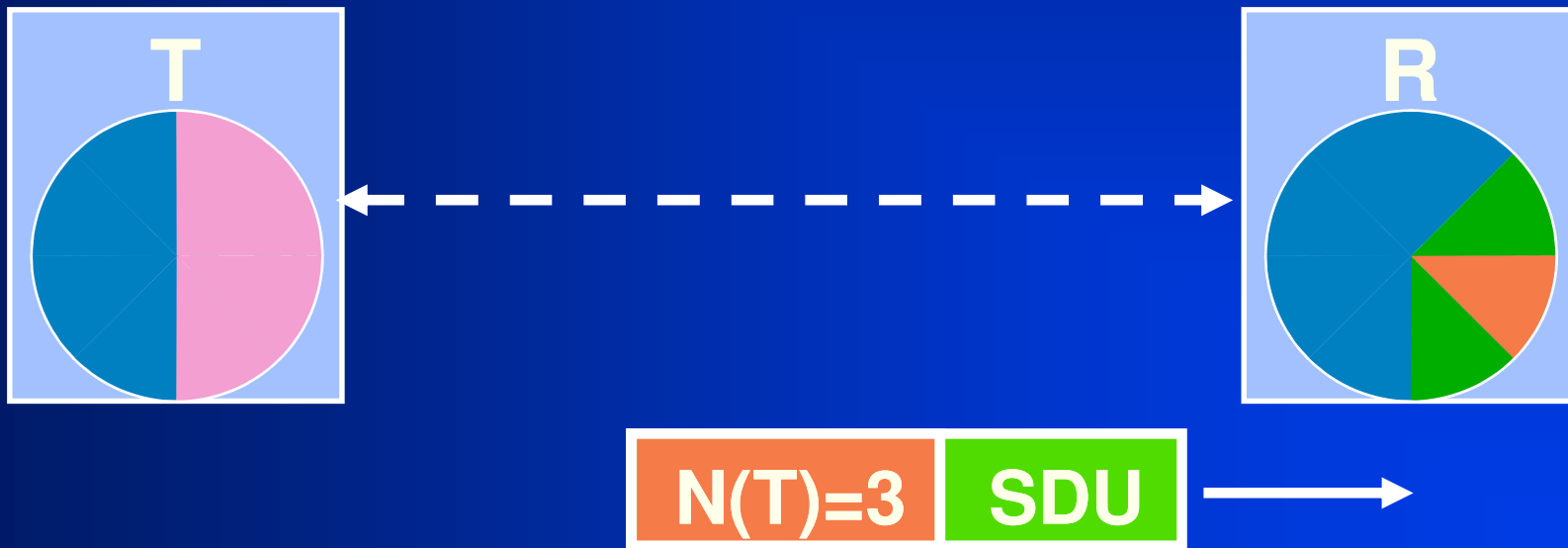
## Ricezione PDU corretta fuori sequenza



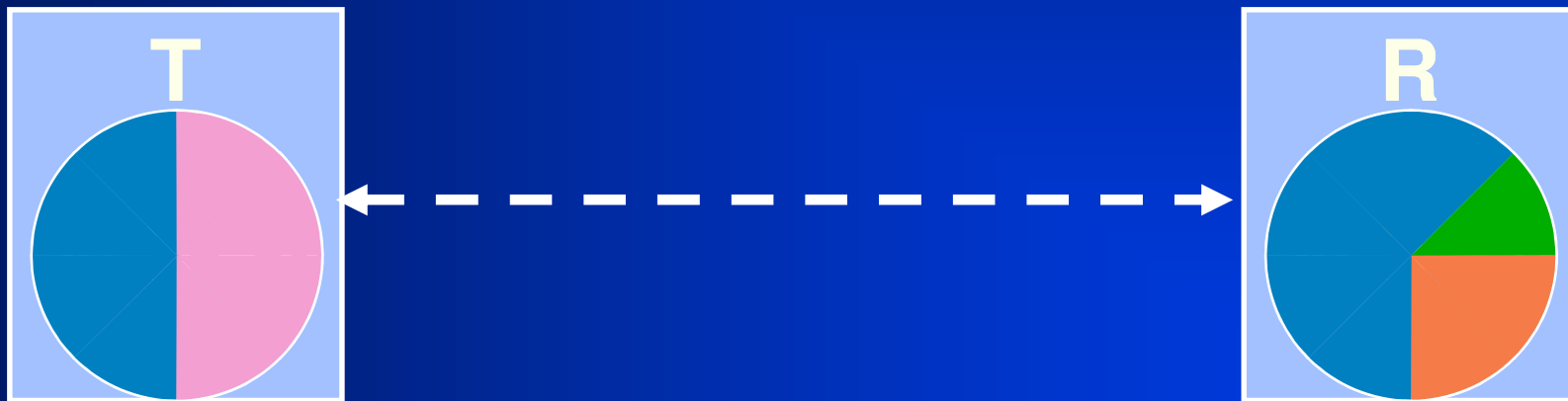
## Trasmissione ACK



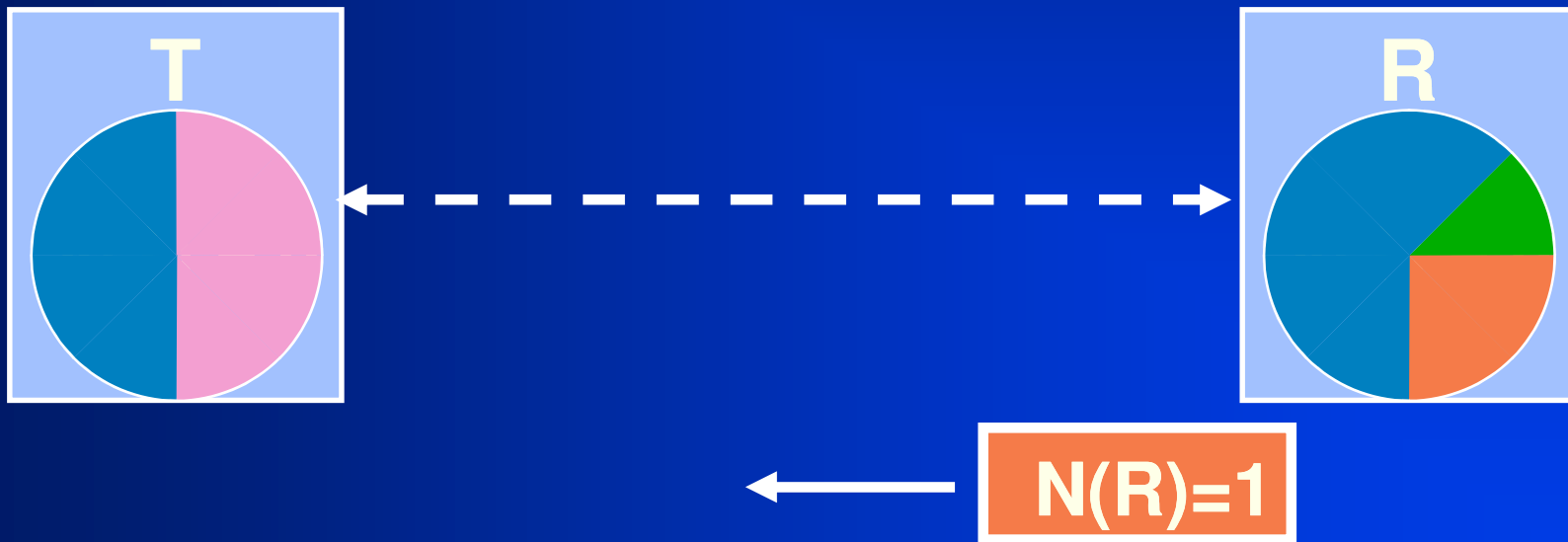
## Ricezione PDU corretta fuori sequenza



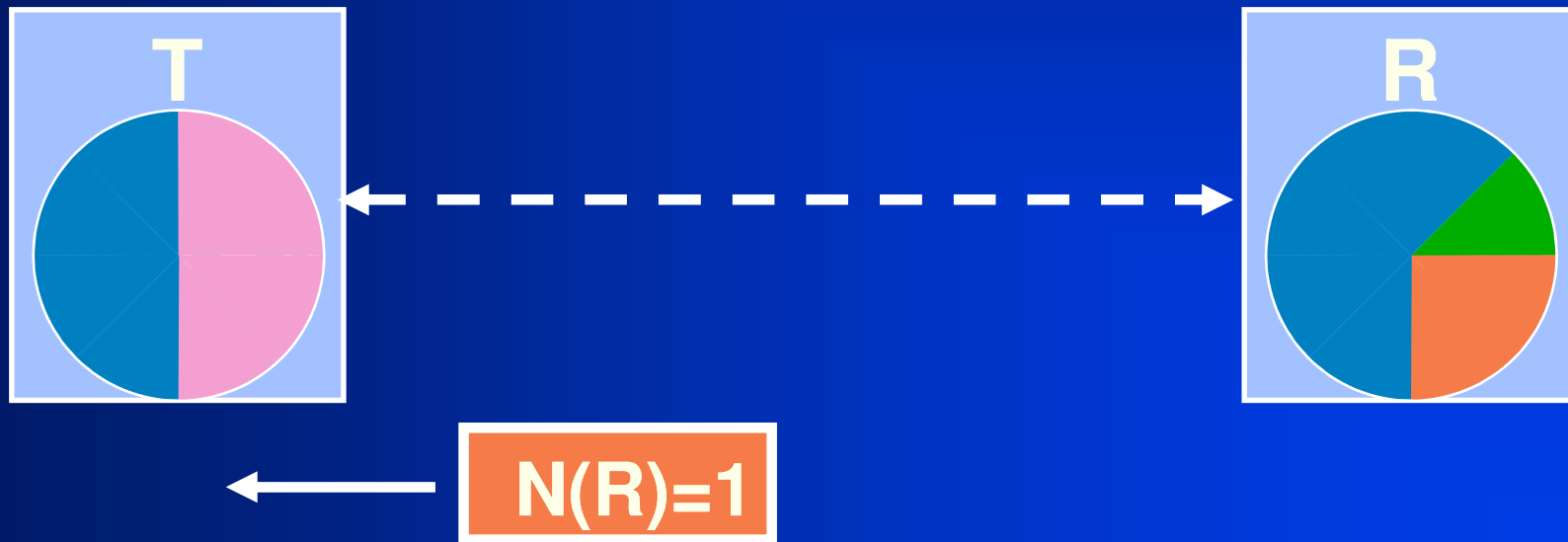
## Ricezione PDU corretta fuori sequenza



## Trasmissione ACK

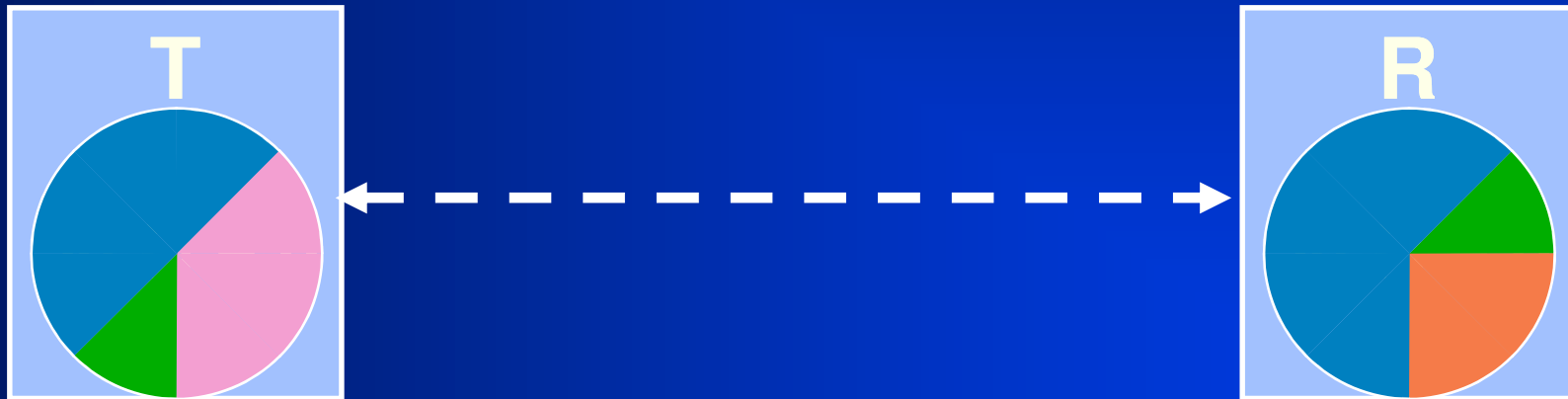


## Ricezione ACK



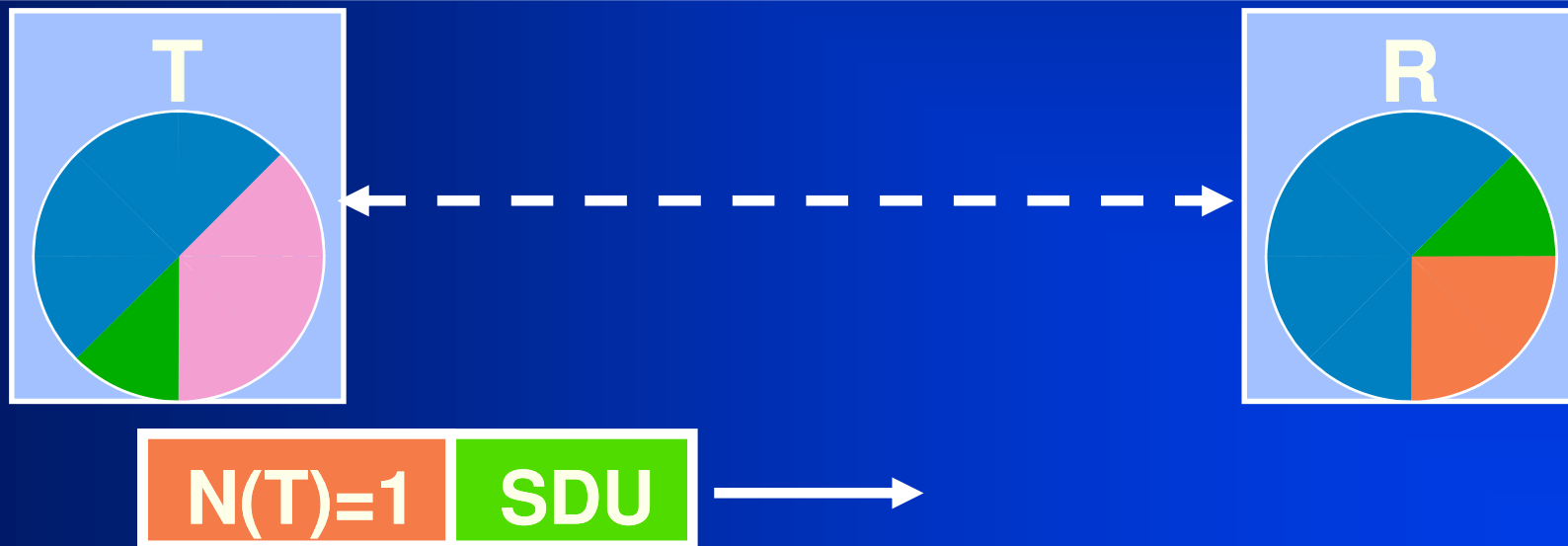


## Ricezione ACK

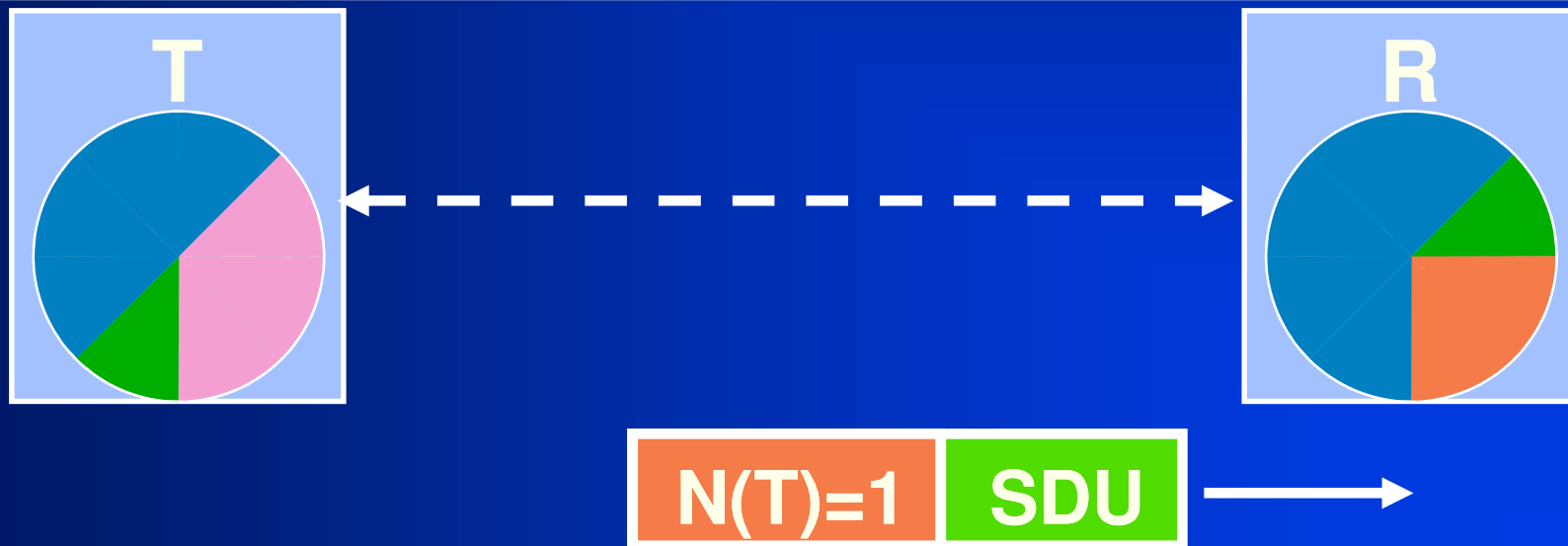


**Scade il timeout**

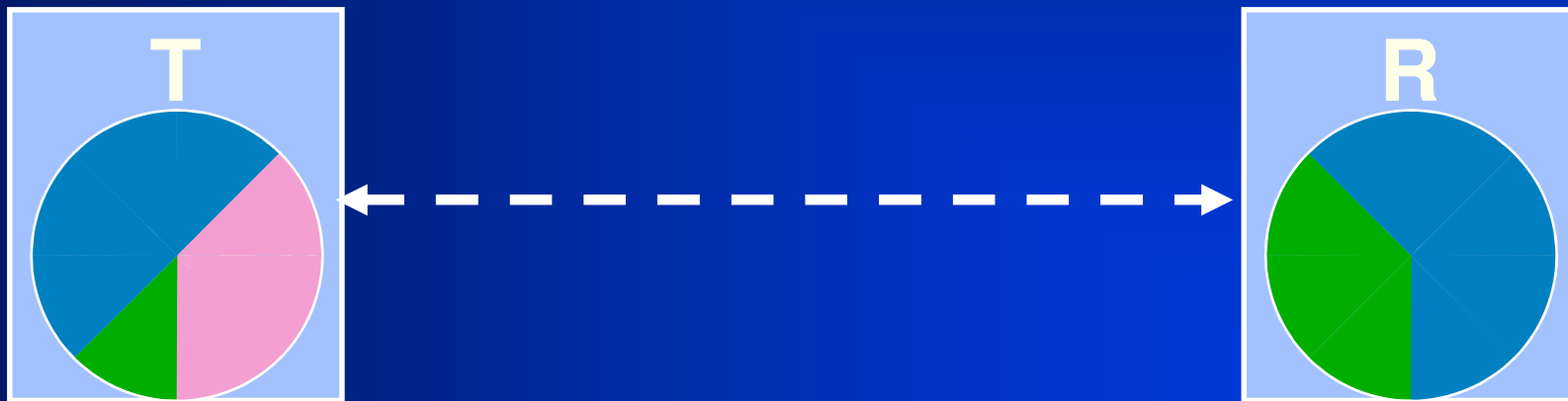
## Ritrasmissione PDU



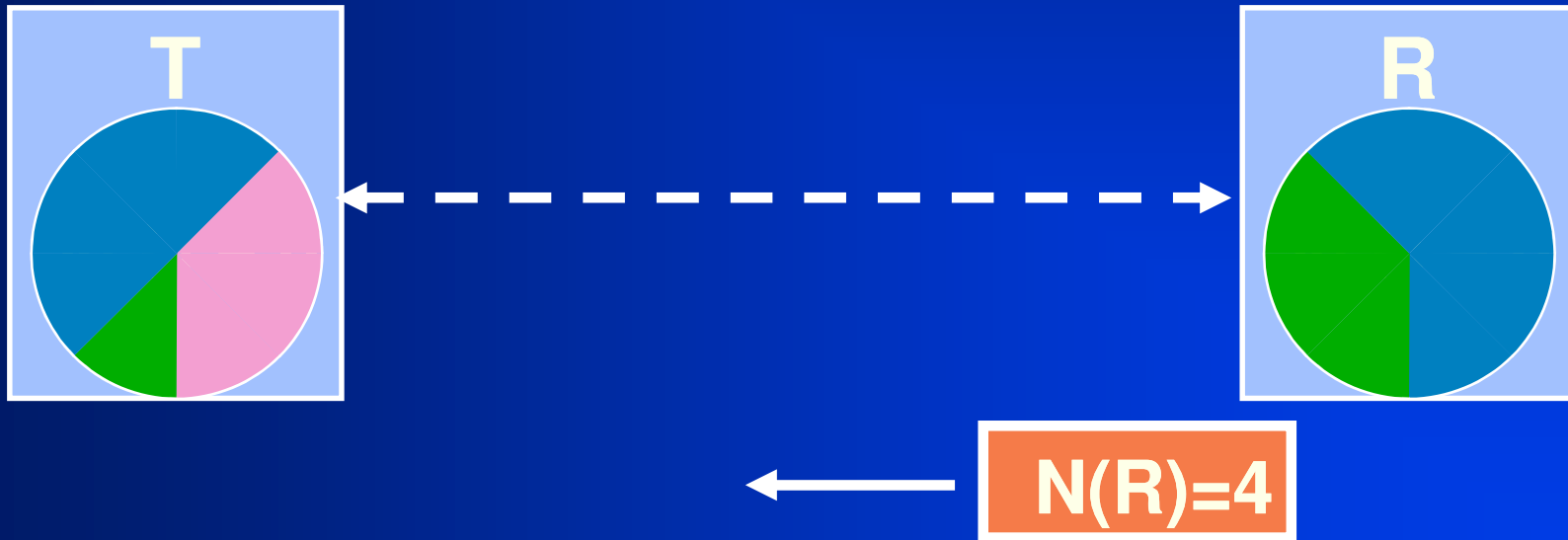
## Ricezione PDU corretta in sequenza



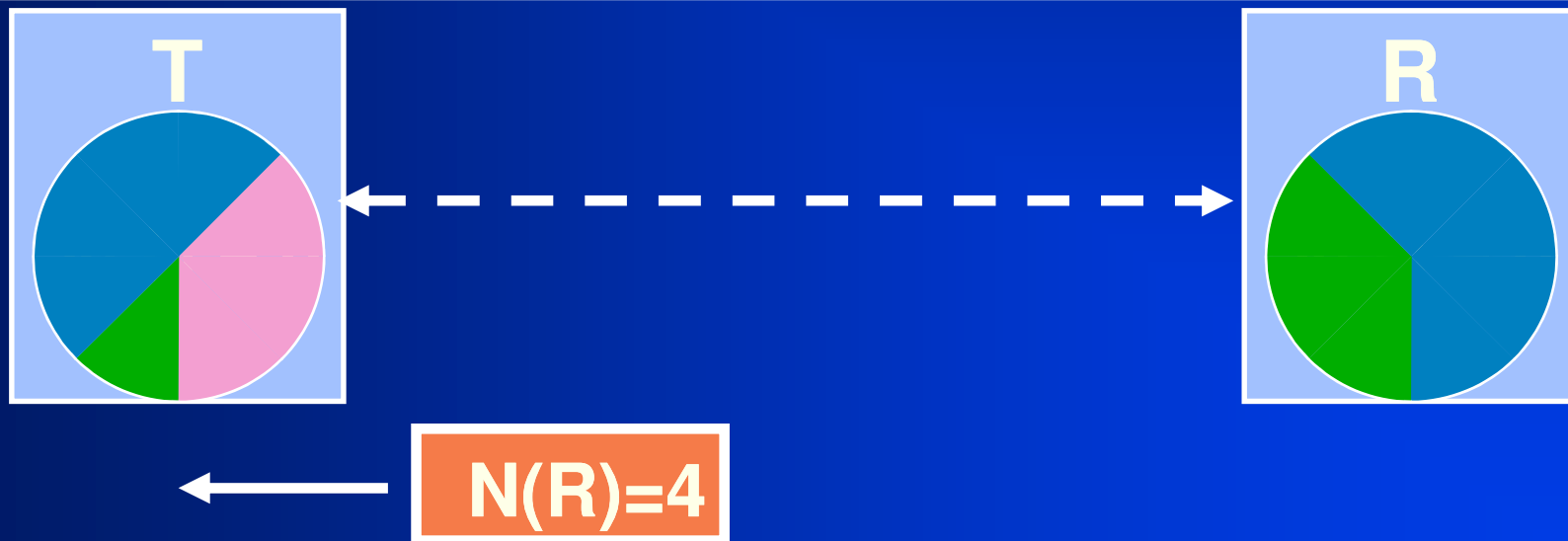
## Ricezione PDU corretta in sequenza



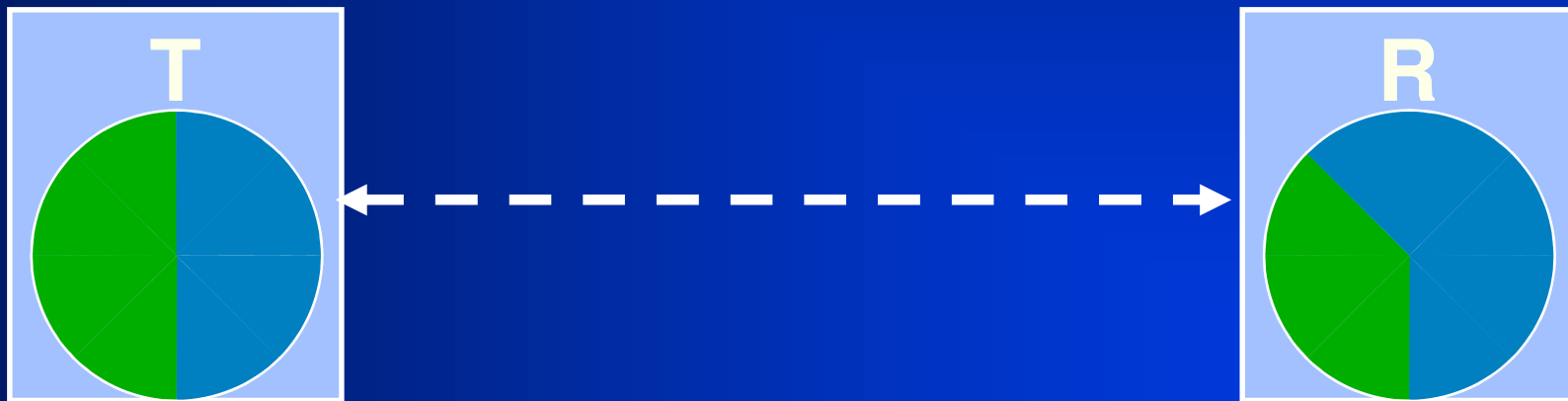
## Trasmissione ACK



## Ricezione ACK



## Ricezione ACK





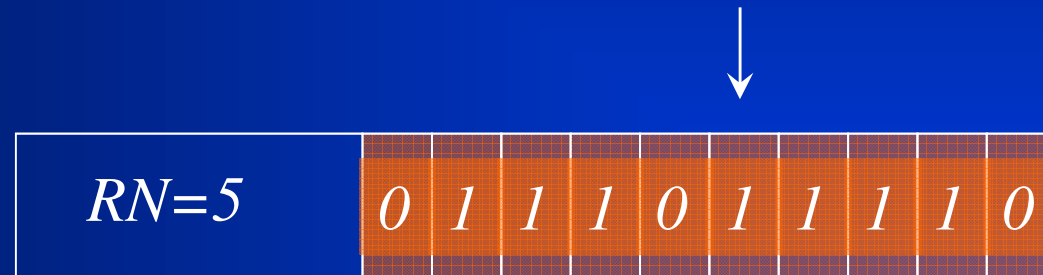
# Selective Repeat

**Il vantaggio del Selective repeat sul Go-back N è significativo se WR è sufficientemente grande.**

**Per avere vantaggi rispetto al go-back-N occorre utilizzare ack individuali oppure avere timer associati alla finestra.**

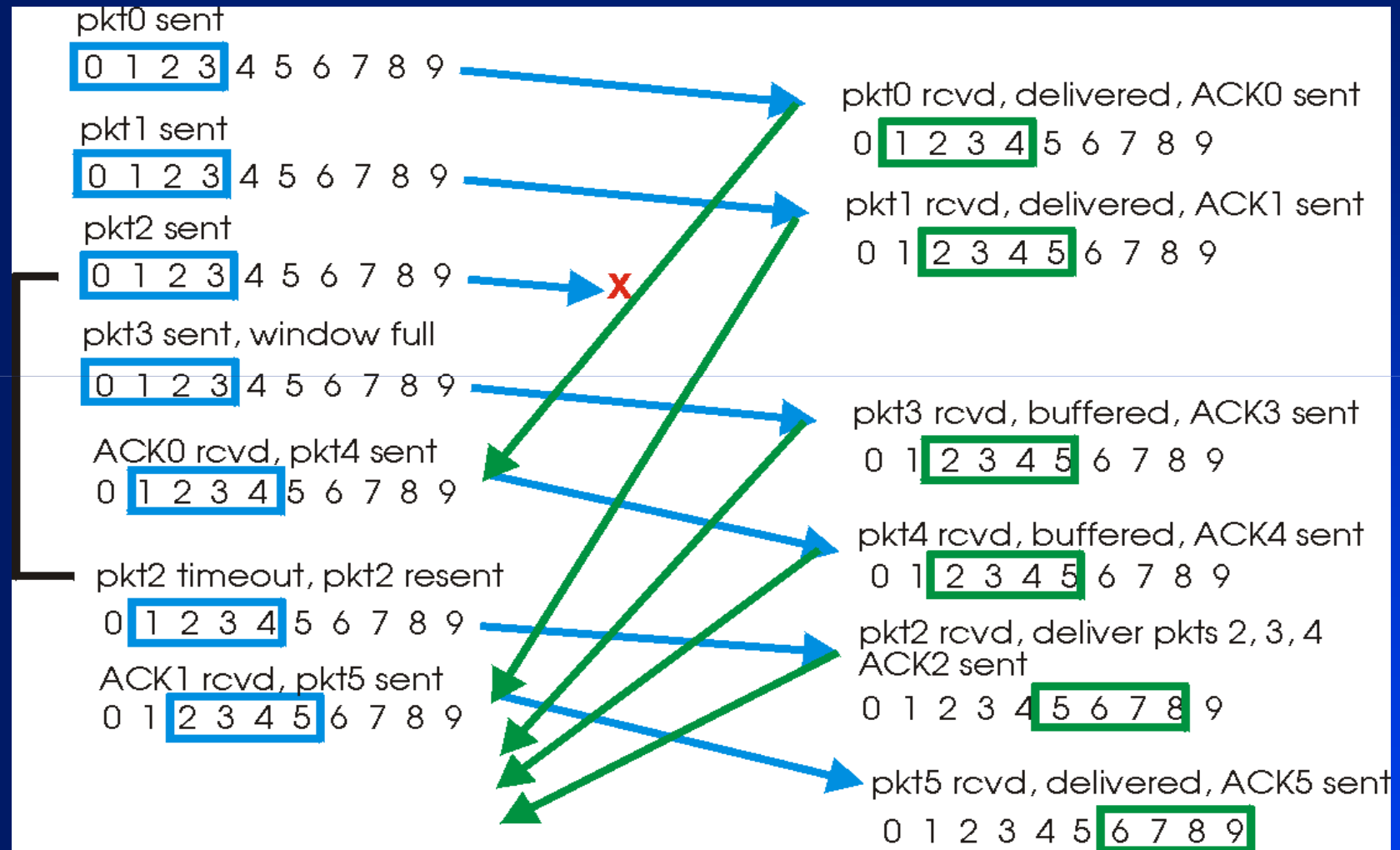
# Trasmissione degli Ack

- ACK singoli per ogni pacchetto
  - ritrasmissione per i pacchetti senza ACK dopo un time-out
- ACK cumulativi: bitmap



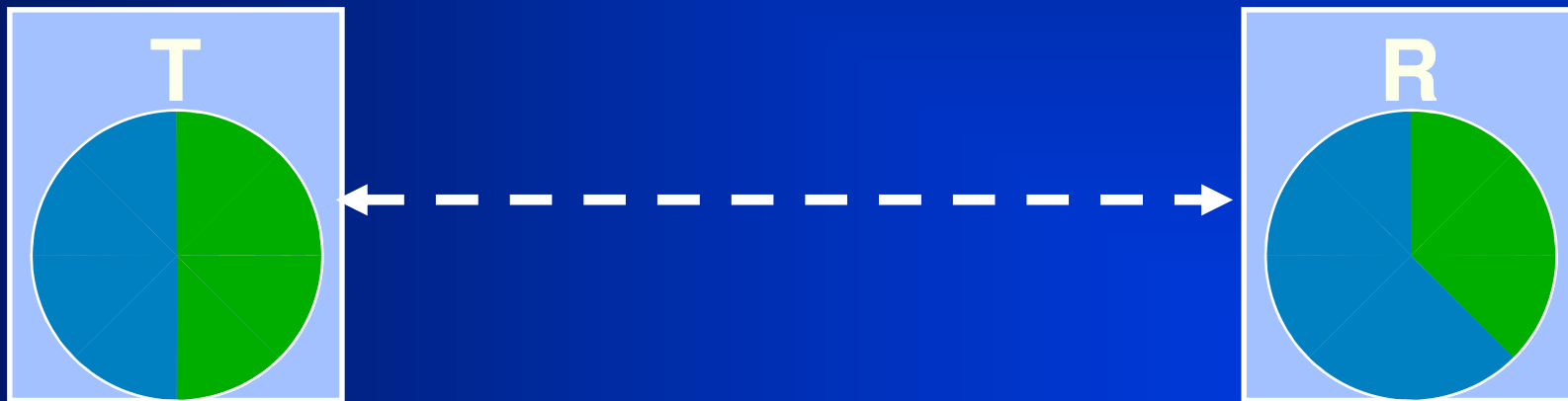
- l'informazione necessaria è maggiore

# Selective repeat

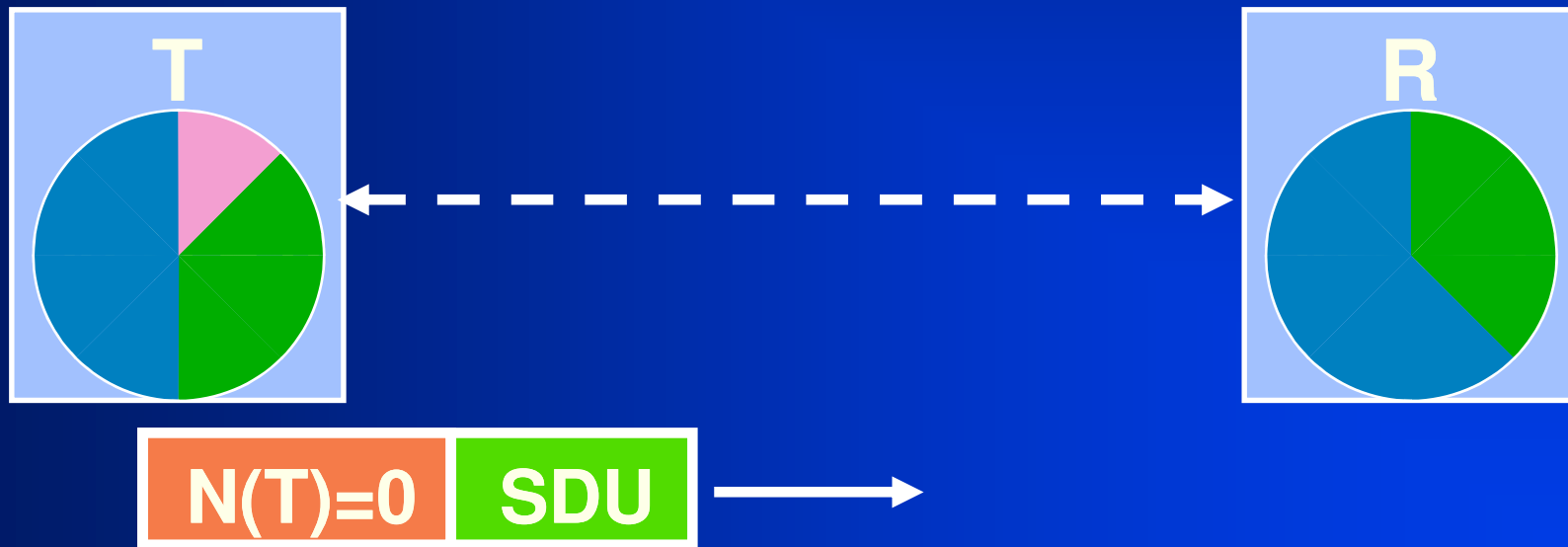


**Si possono usare  
conferme negative (NAK)**

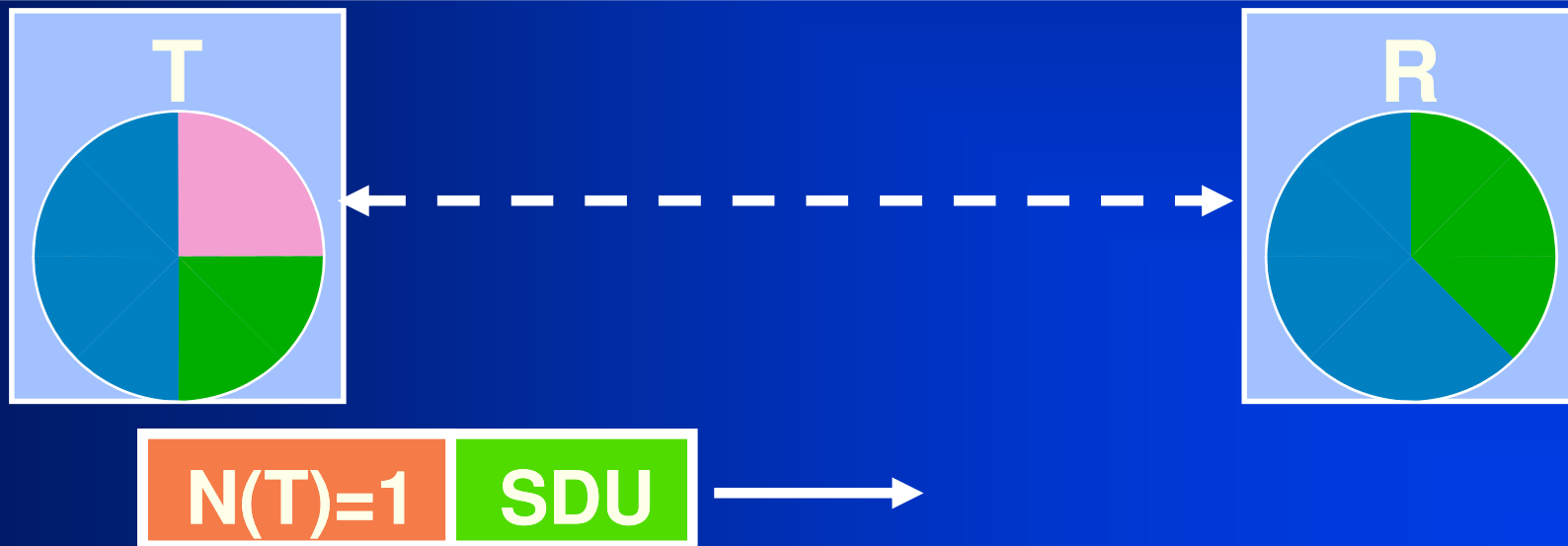
# Inizializzazione



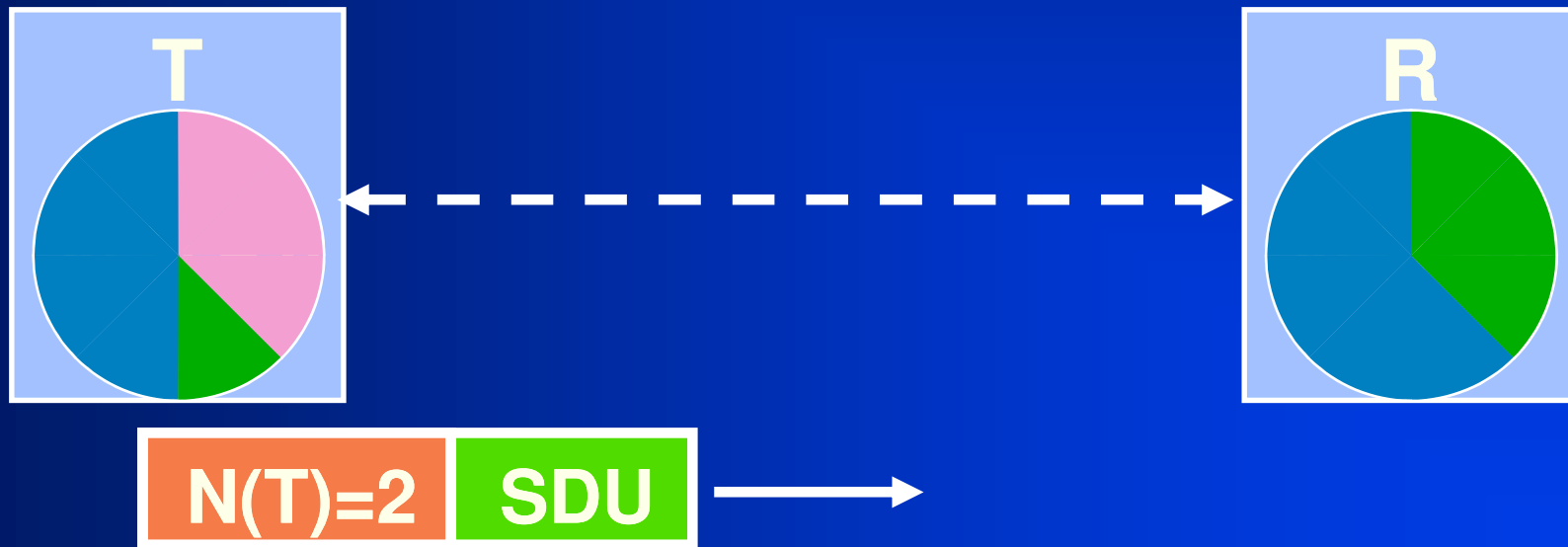
## Trasmissione PDU



## Trasmissione PDU

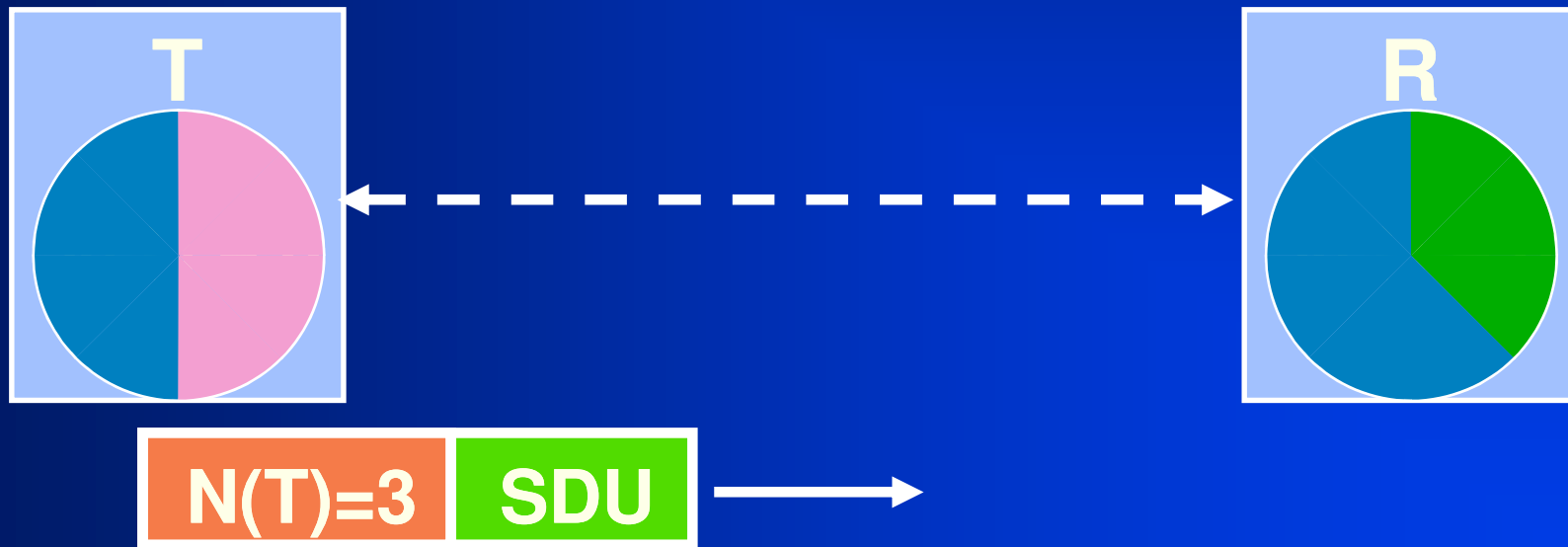


## Trasmissione PDU

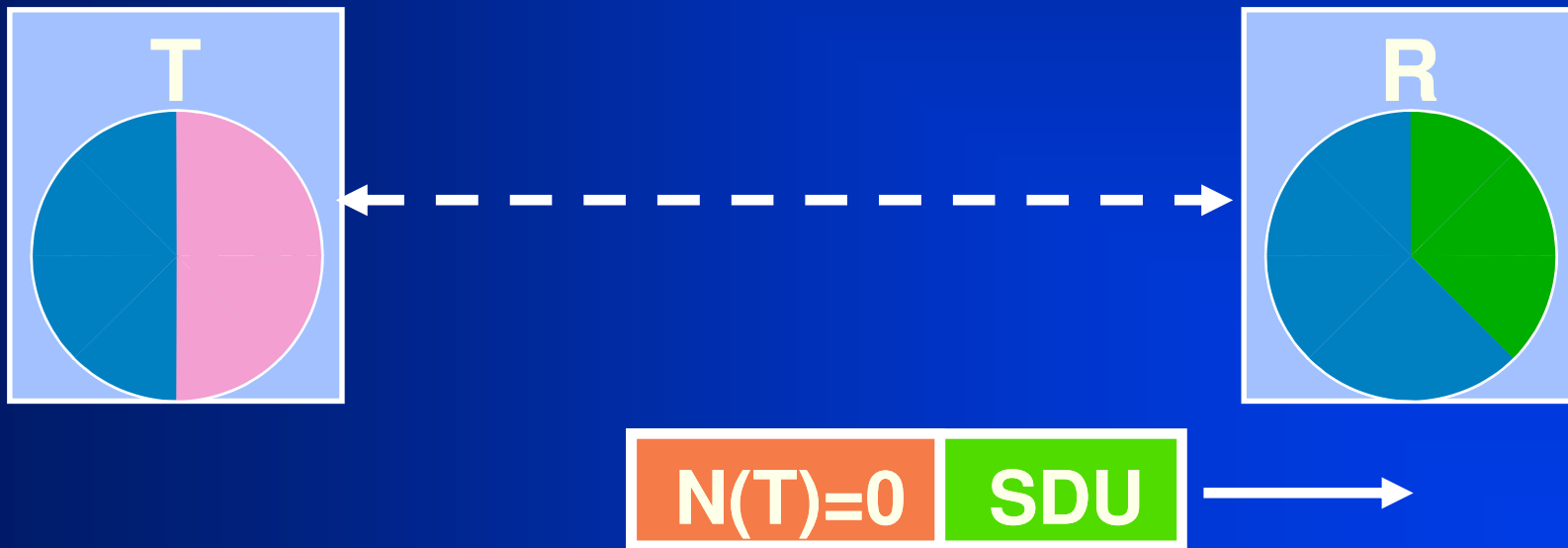




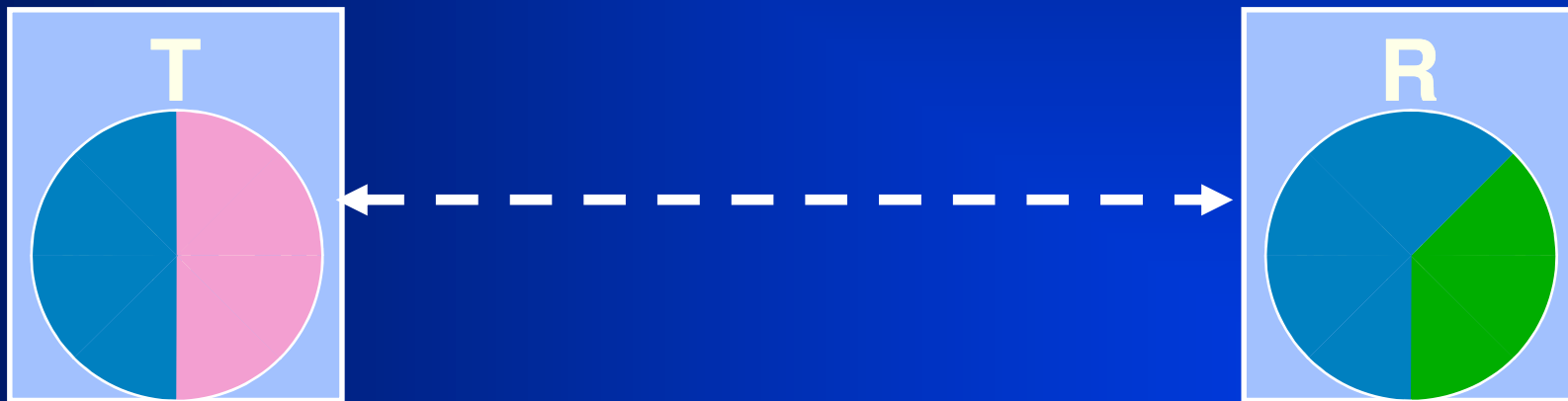
## Trasmissione PDU



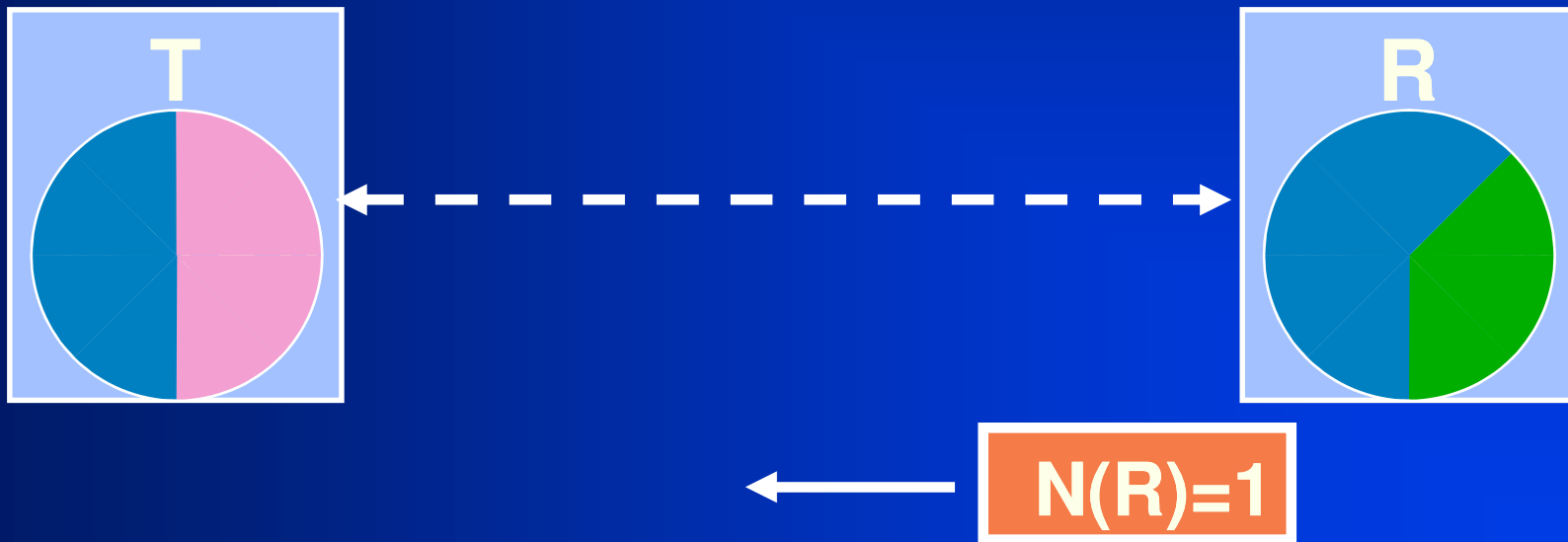
## Ricezione PDU corretta in sequenza



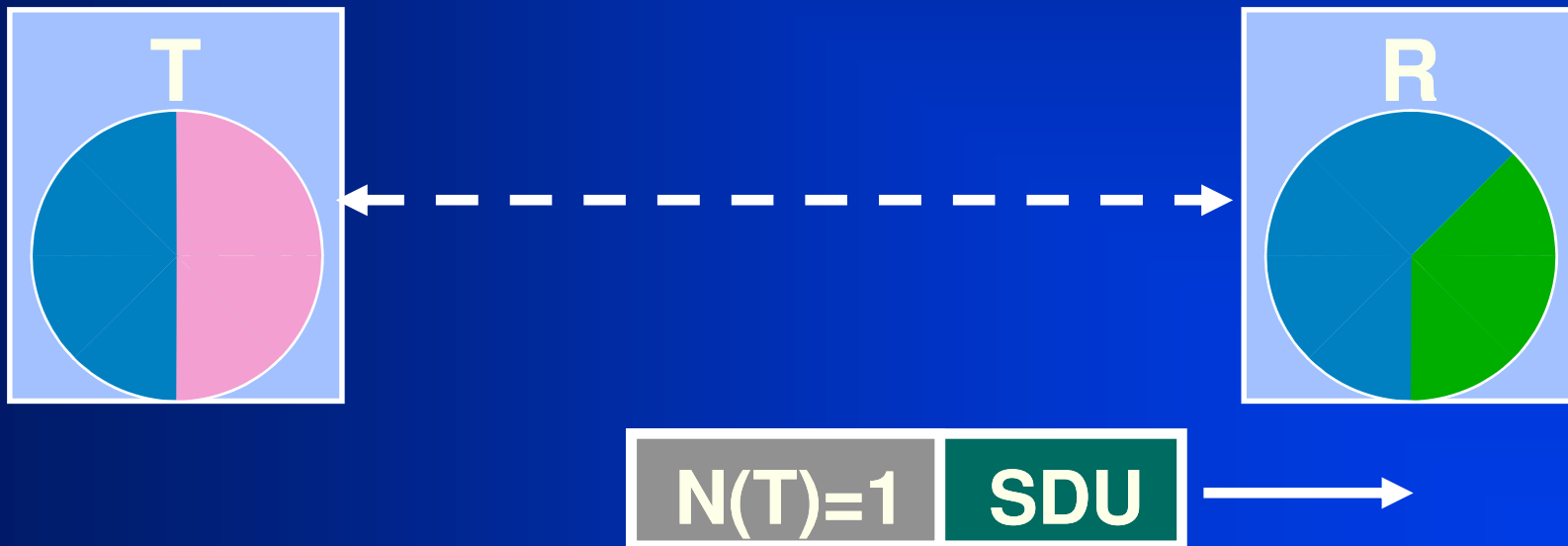
## Ricezione PDU corretta in sequenza



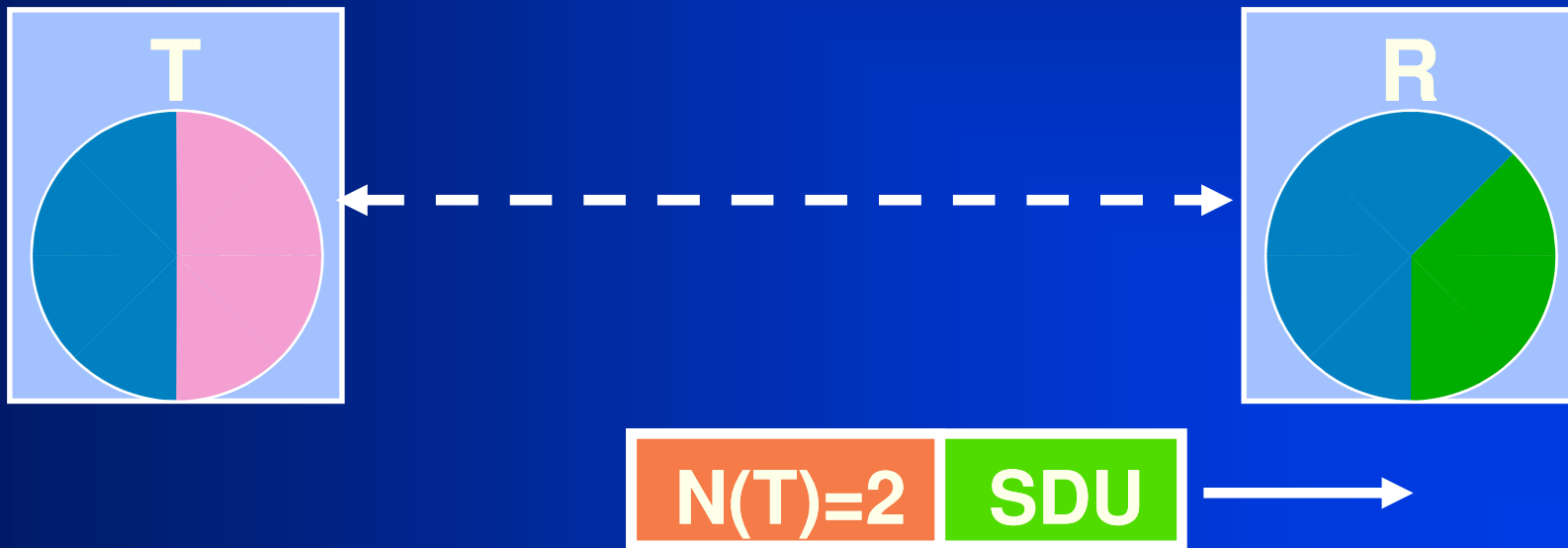
## Trasmissione ACK



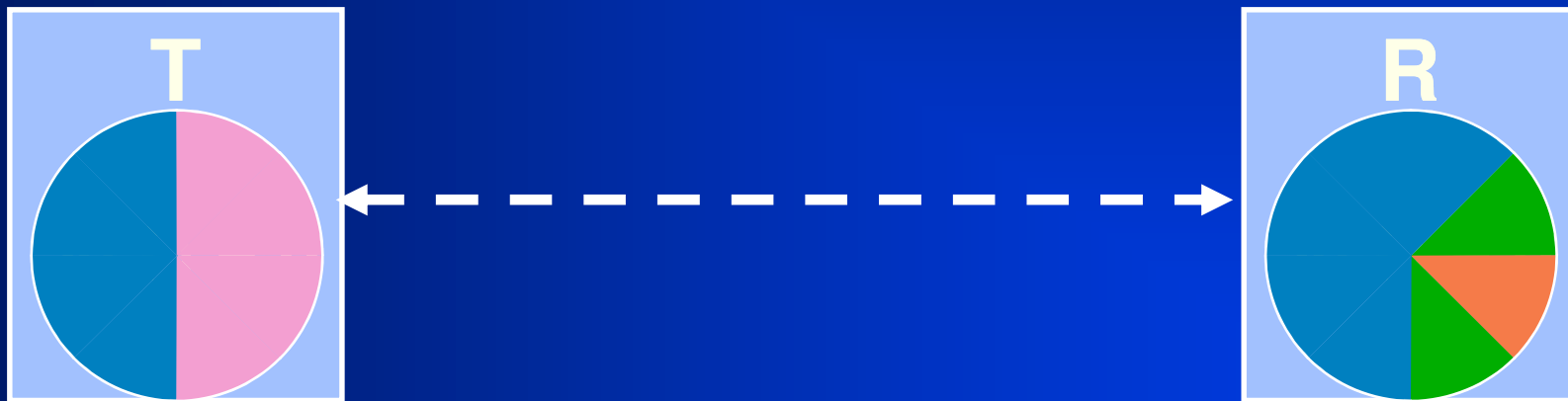
## Ricezione PDU errata



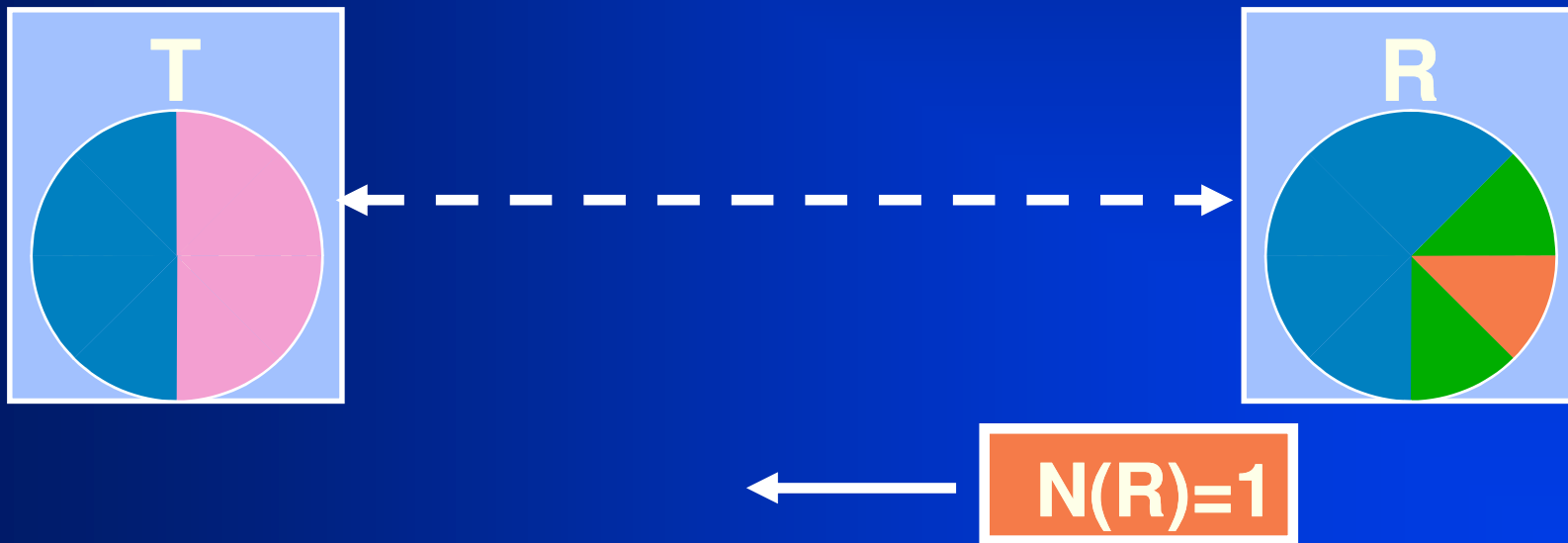
## Ricezione PDU corretta fuori sequenza



## Ricezione PDU corretta fuori sequenza

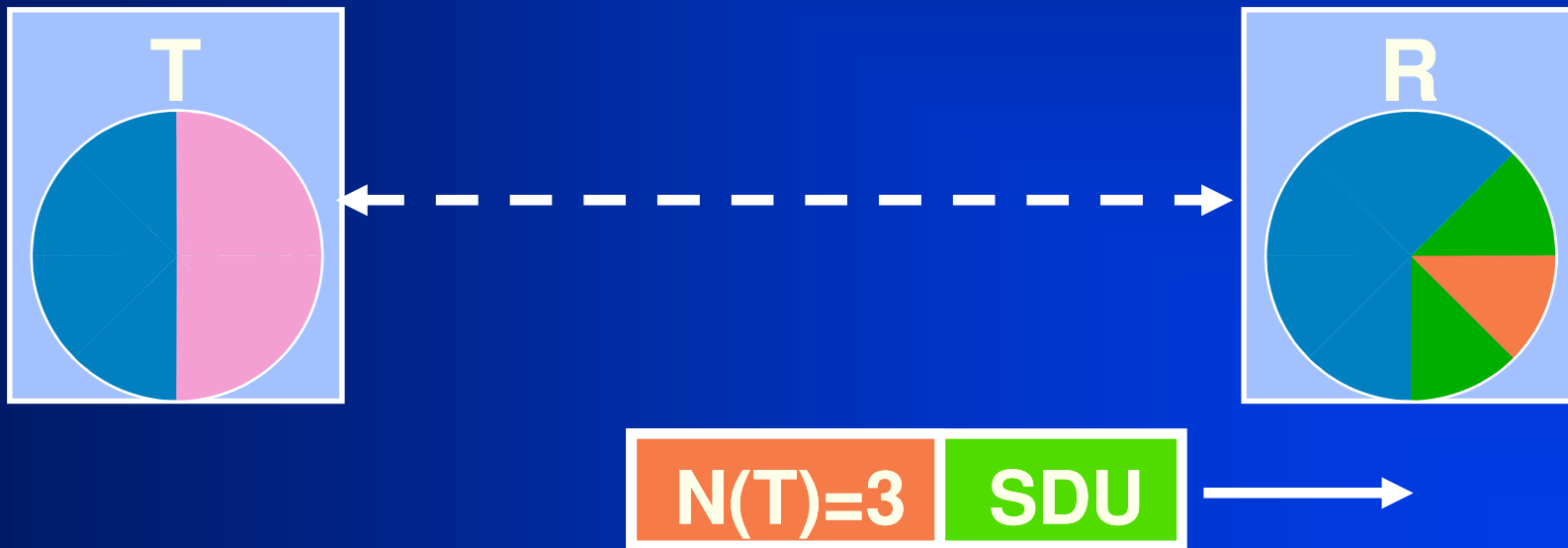


## Trasmissione NAK

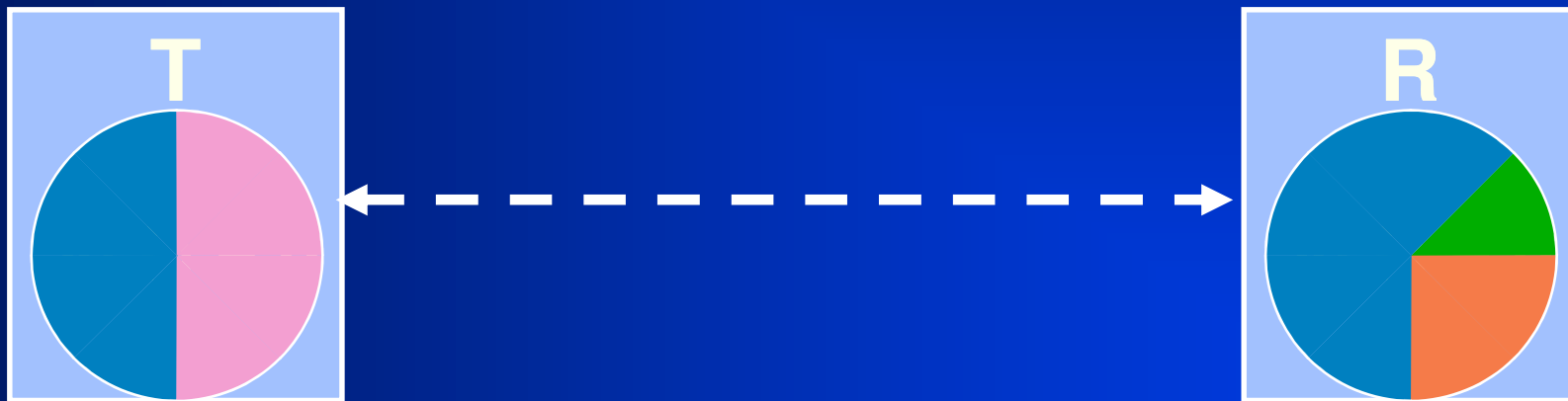




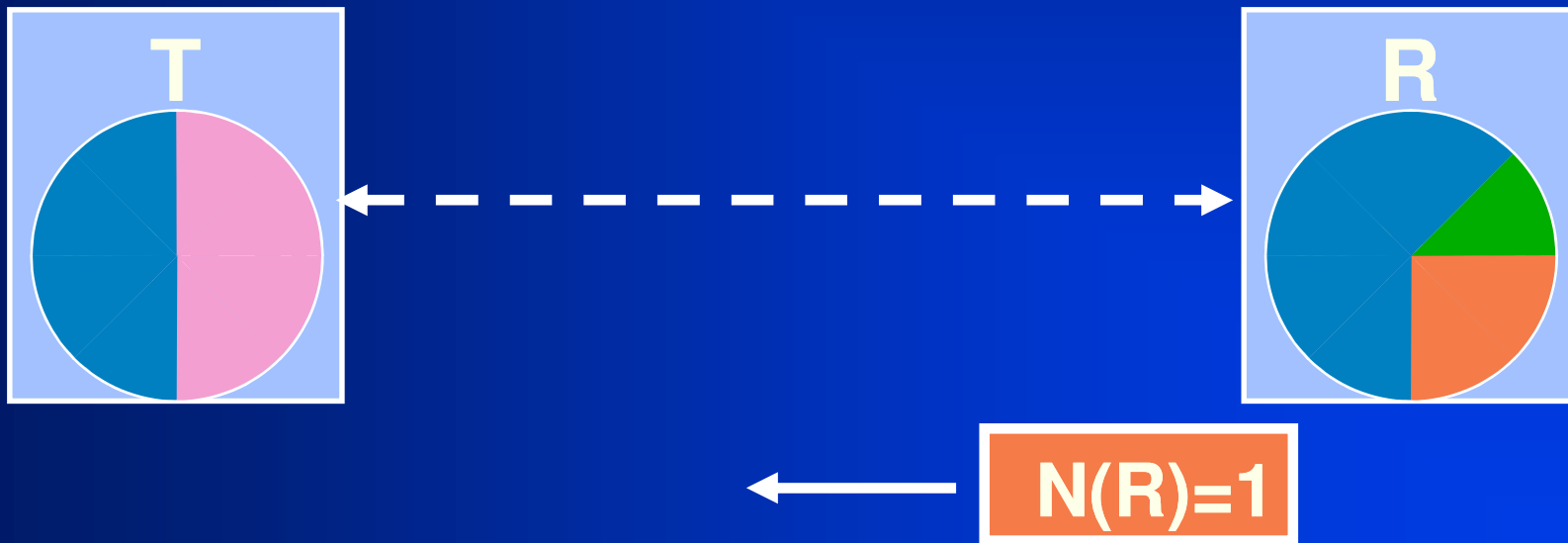
## Ricezione PDU corretta fuori sequenza



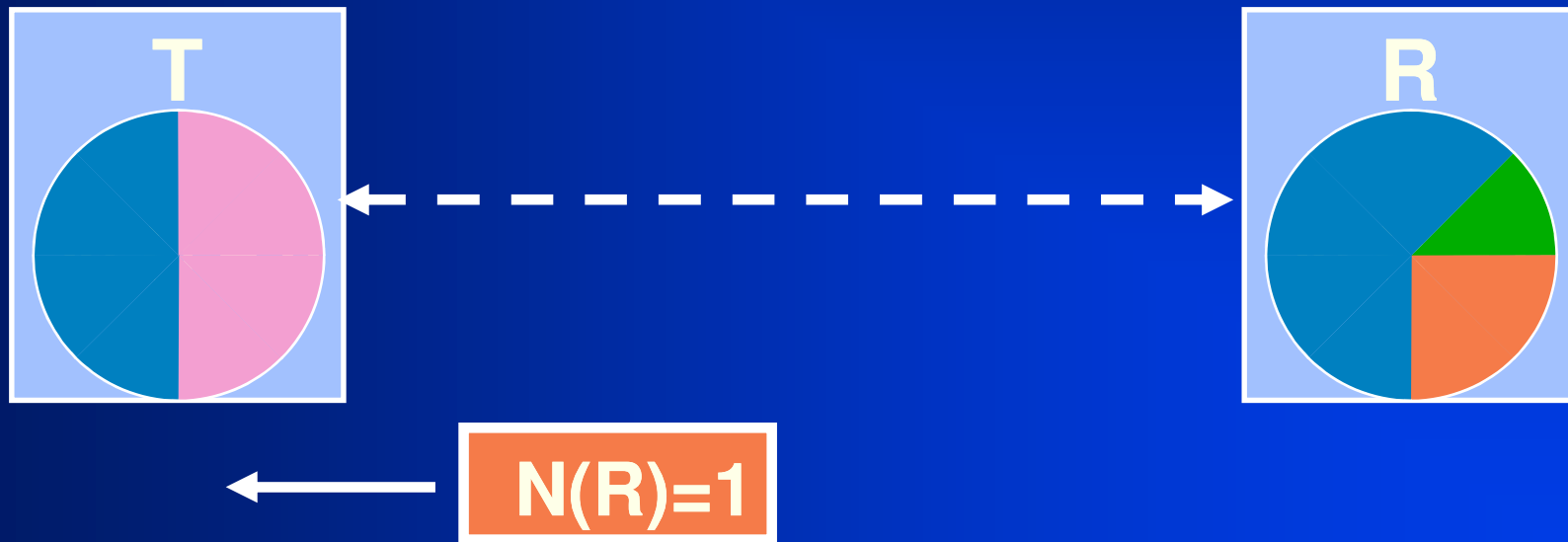
## Ricezione PDU corretta fuori sequenza



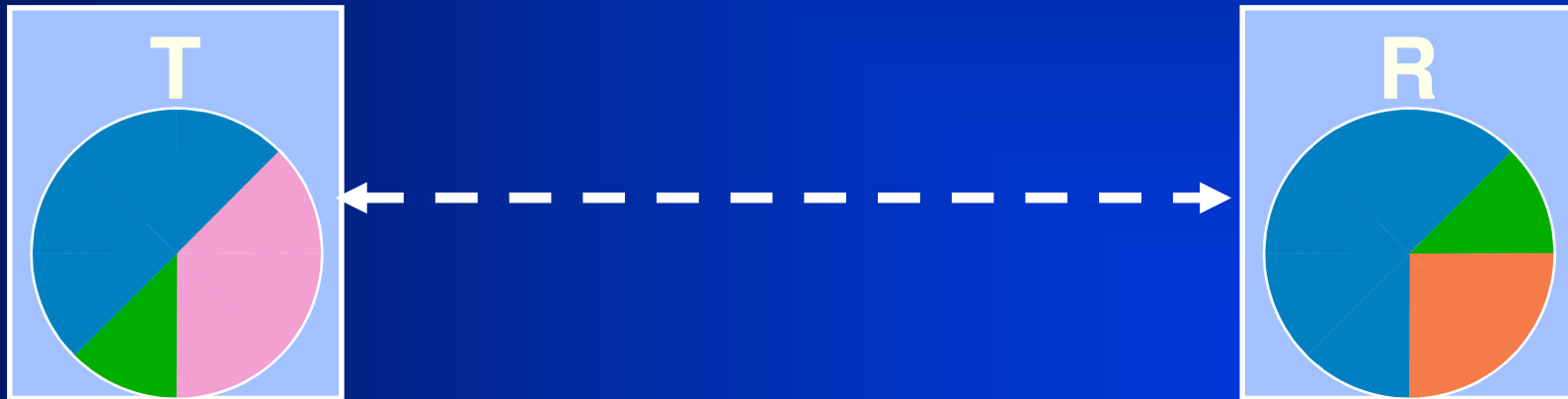
## Trasmissione NAK



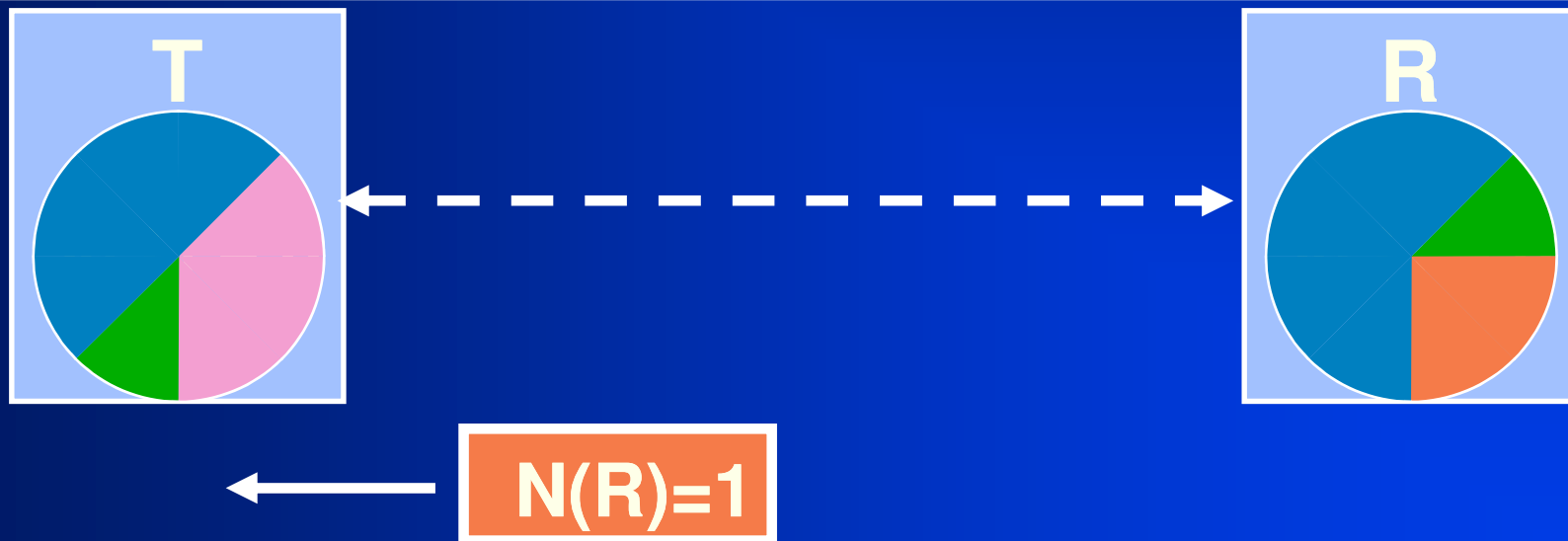
## Ricezione ACK



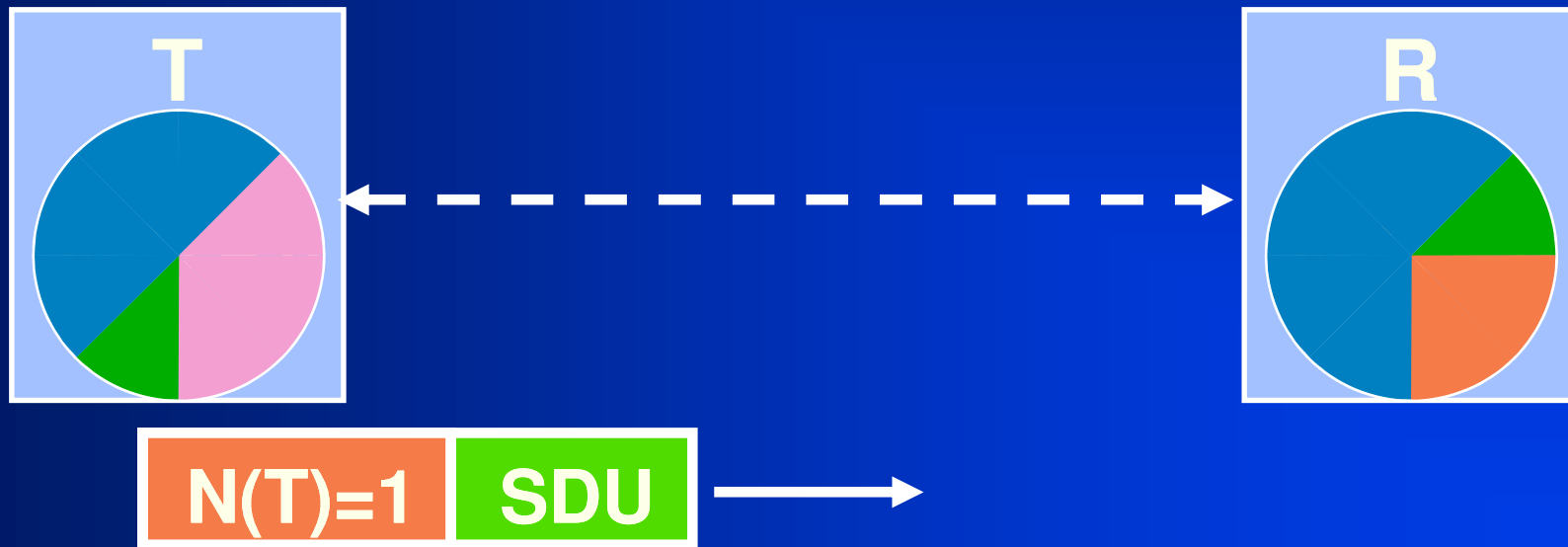
## Ricezione ACK



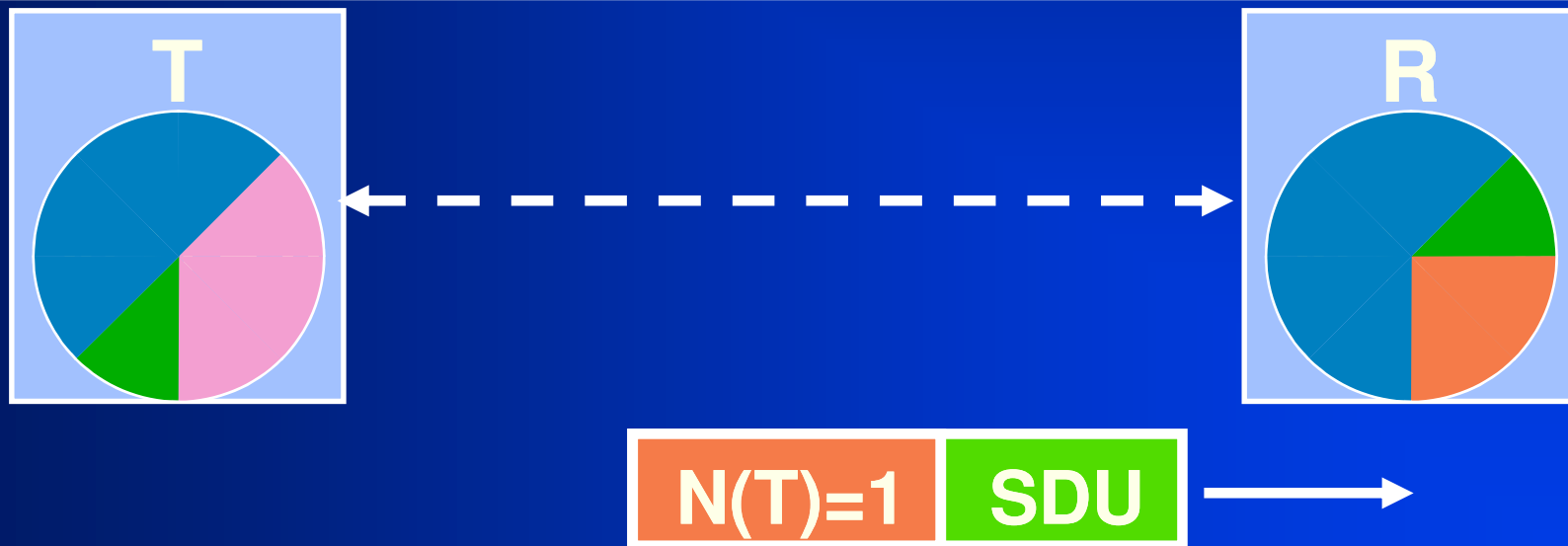
## Ricezione NAK



## Ritrasmissione PDU

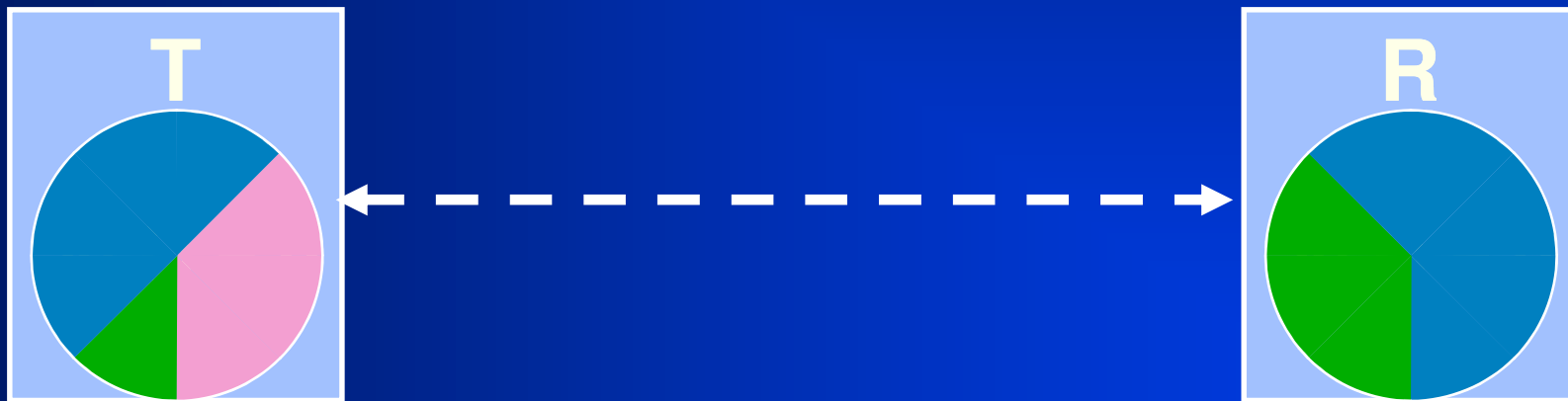


## Ricezione PDU corretta in sequenza

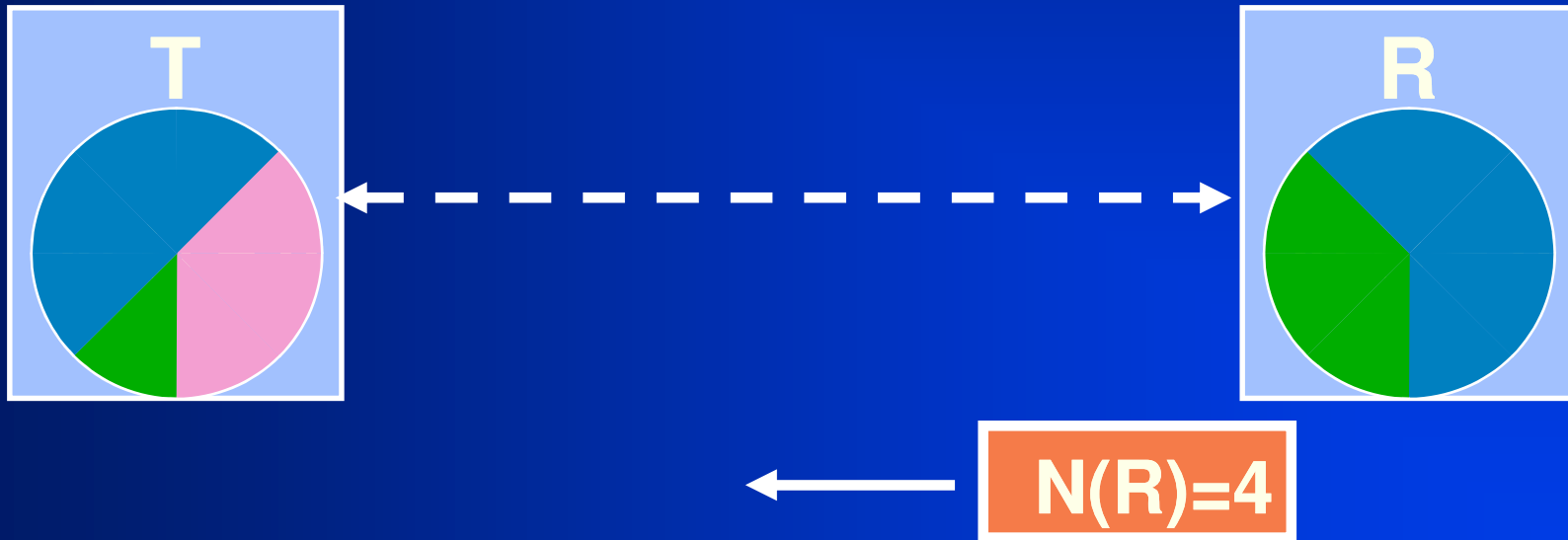




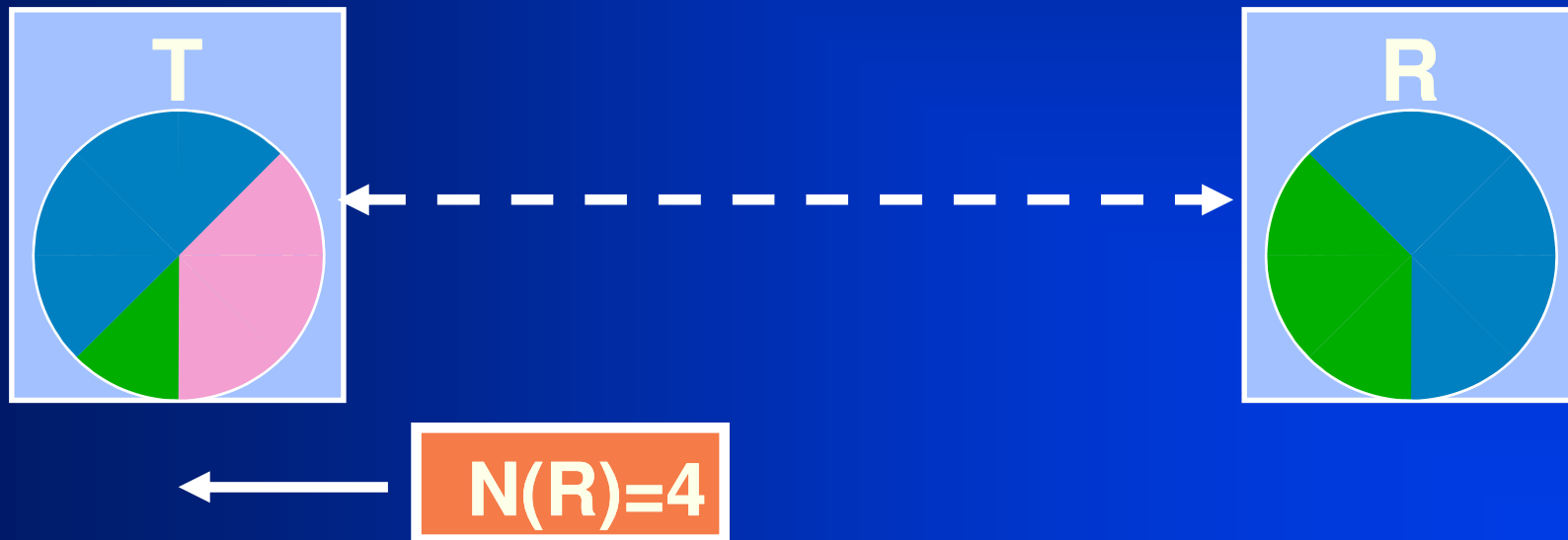
## Ricezione PDU corretta in sequenza



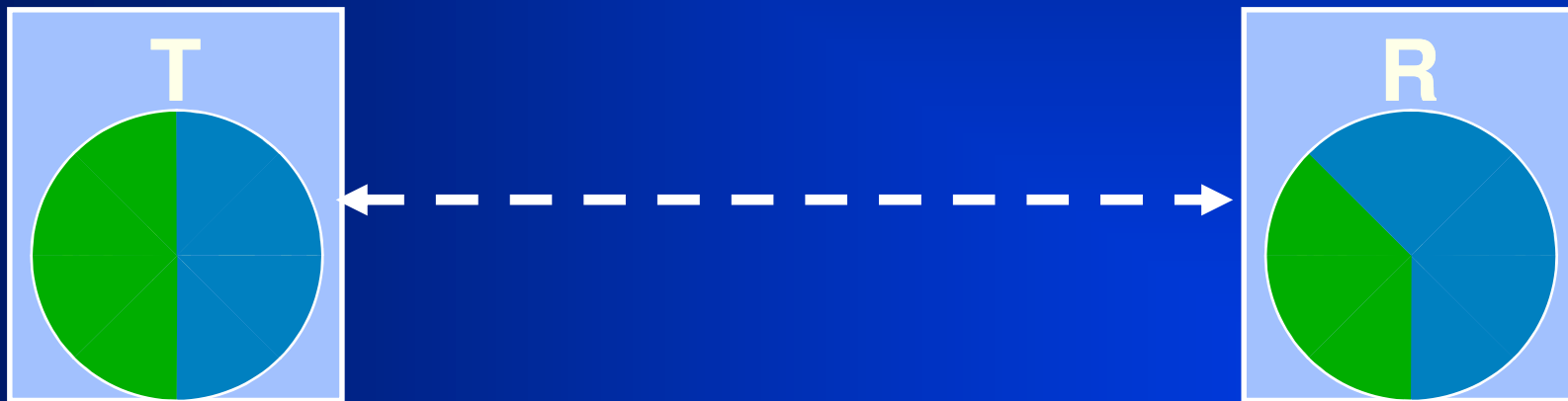
## Trasmissione ACK



## Ricezione ACK



## Ricezione ACK



# *Osservazioni sull'uso del NACK*

- come inviare un NACK dopo la ricezione di una trama errata?
- occorre sapere il SN della trama errata, ma se e' errata ...
- se arriva la trama successiva si può ipotizzare che sia andata persa la trama precedente, ma nei casi in cui non e' garantita la consegna in ordine questo non e' possibile

**L'uso del NAK può avvantaggiare  
anche**

- **Stop and Wait**
- **Go back N**

# *Considerazioni*

**Il *go-back N* è molto utilizzato perché ha un'efficienza di gran lunga maggiore rispetto allo stop&wait e richiede un'elaborazione supplementare minima.**

**Il *selective repeat* offre un'efficienza moderatamente maggiore rispetto al go-back N ma i costi in termini di memoria e elaborazione supplementare sono tali che spesso si preferisce usare il go-back N**