DQMAN Overview

Jesús Alonso-Zárate, PhD.
Research Associate
CTTC

Weekly Seminar, CTTC.

DQMAN: Background

1992
W. Xu and G. Campbell
(*Illinois Institute of Technology*) → DQRAP
(Cable Television Distribution)

2001
Luis Alonso (*UPC*) → DQRAP/CDMA
(3G Mobile Communications)

Hybrid protocol based on distributed queues with near-optimum performance

2005 **DQMAN**
(Distributed Queuing MAC Protocol for Mobile Ad Hoc Networks)

2005 **DQCA for Wireless LAN**
(Distributed Queuing Collision Avoidance Protocol)

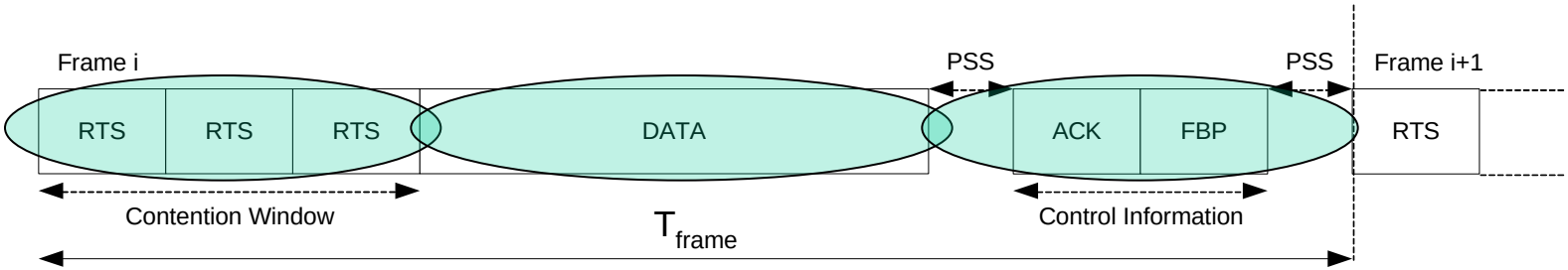
CTTC
Centre Tecnològic de Telecomunicacions de Catalunya

DQMAN: Overview

- **Self-organized** MAC protocol for Mobile Ad Hoc NETworkS.
- It implements a MASTER & SLAVE architecture
- Dynamic **self-constructing** algorithm.
- Based on two distributed queues that manage both the **data transmission** and the **resolution of collisions.**
- It does not suffer from instability under all traffic conditions
- Eliminates back-off intervals

CTTC
Centre Tecnològic de Telecomunicacions de Catalunya

DQMAN: Frame structure



- Special Request to Send Packets (RTS)
  - Known **sequence of chips** that a node with data to transmits sends → minislot selected at random with equal probability
- Data packet
  - Packet with user payload → ideally free of collisions
- Control Information Exchange
  - ACK → from **destination** to **source**
  - FBP → Feed Back Packet **broadcasted by master** node containing the state of the contention minislots, being IDLE, SUCCESS or COLLISION.

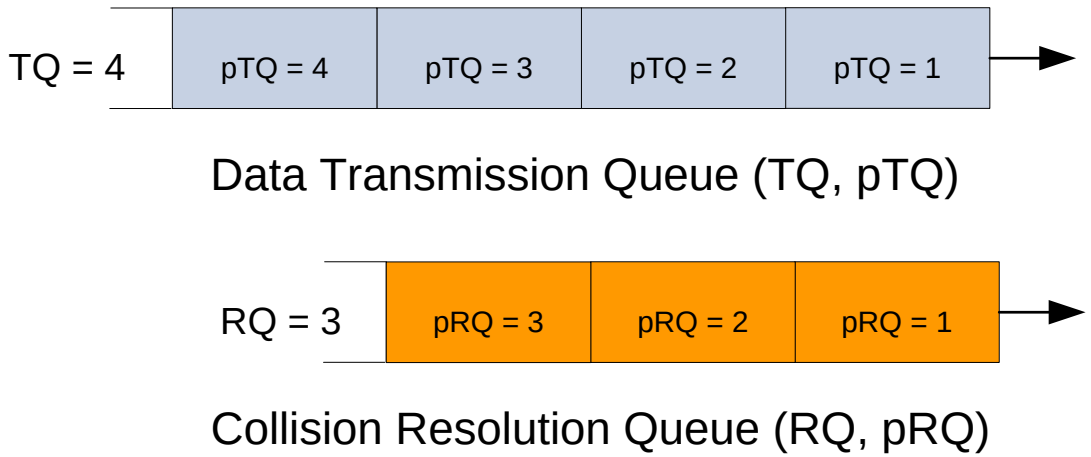
DQMAN: Distributed Queues

- What do we understand for a **distributed queue**?
- We have a logical queue → that was easy!
- Each node keeps track of the state of the queue
  - Total number of queuing nodes
  - Individual position within the queue
- With the proper control information, all nodes update the state of the queue in a synchronized way
- It is as if “all nodes **share** the same queue”
- **Again, all nodes know:**
  - **Total number of queuing nodes**
  - **Individual position within the queue**

DQMAN: Distributed Queues

- Represented by 4 integer numbers:

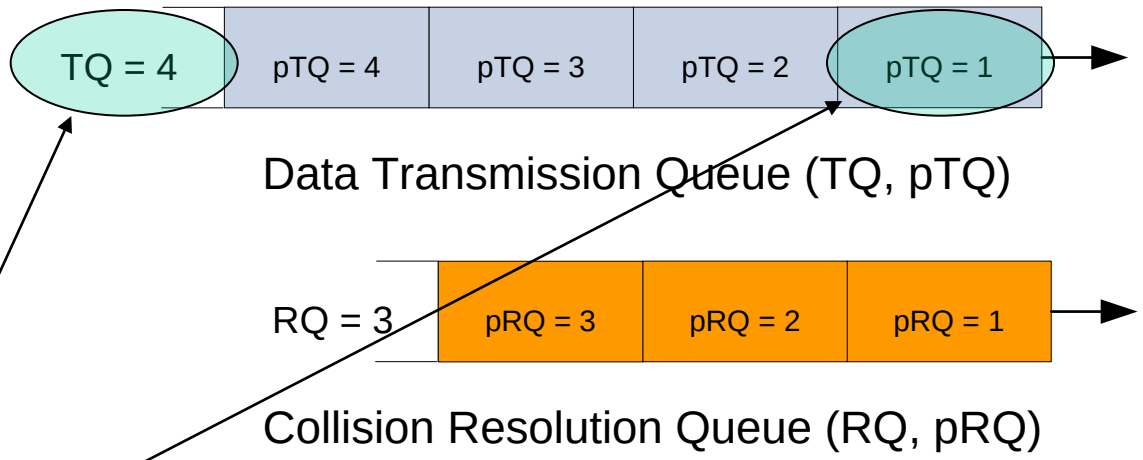
TQ // RQ // pTQ // pRQ



- FBP broadcasted by master node at the end of each frame contains the state of the access mini slots, which is the **<u>only needed information</u>** to update the queues

DQMAN: Distributed Queues

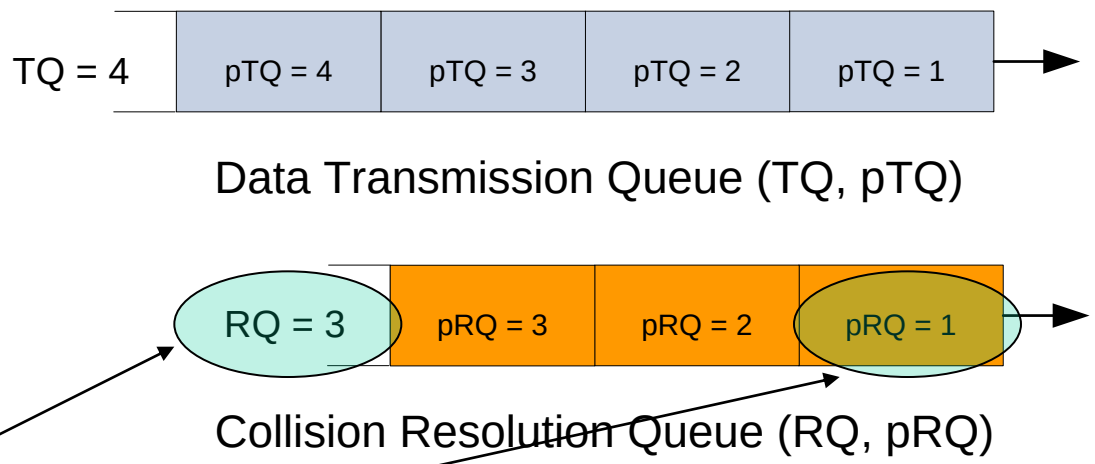
- **Data Transmission Queue (DTQ)**
  - Contains nodes waiting to transmit data because they succeeded in the transmission of the RTS
  - The node in the first position transmits within the next frame



- TQ → total number of nodes queuing at DTQ
- pTQ → individual position of each node at DTQ
  - **<u>pTQ has a different value for EACH node</u>**

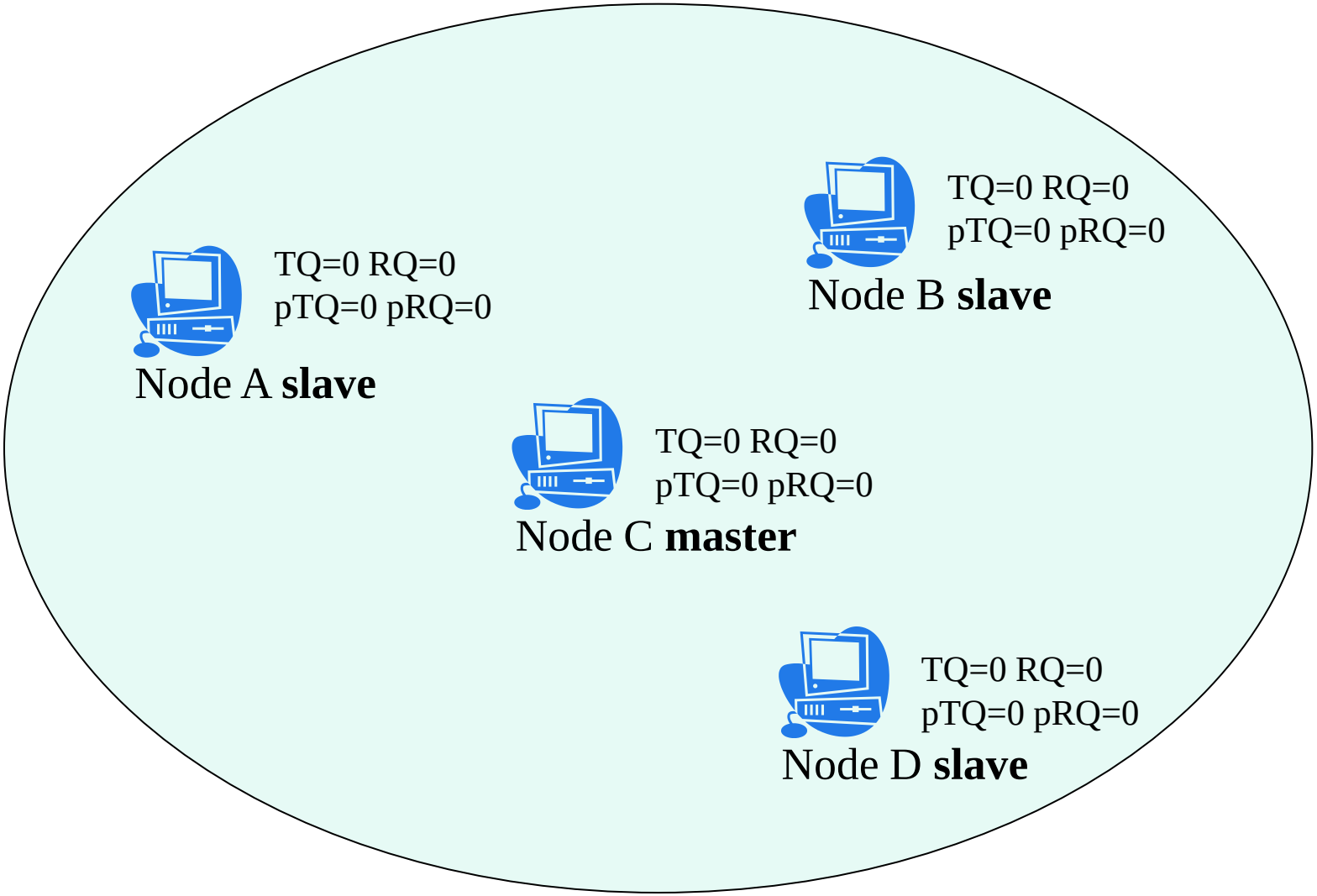
DQMAN: Distributed Queues

- **Collision Resolution Queue (CRQ)**
  - Contains nodes that suffered a RTS collision
  - The nodes in the first position try to resolve their collision within the next frame by sending a new RTS



- RQ → total number of nodes queuing at CRQ
- pRQ → individual position of each node at CRQ
  - **<u>Nodes that collided in the same access minislot have the same value for pRQ!!!</u>**

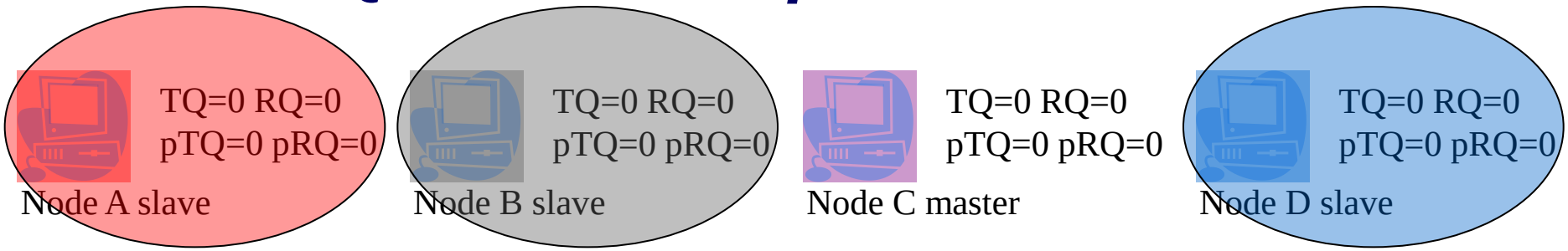
DQMAN: Example



DQMAN: Example

Ending of frame i-1

DQMAN: Example

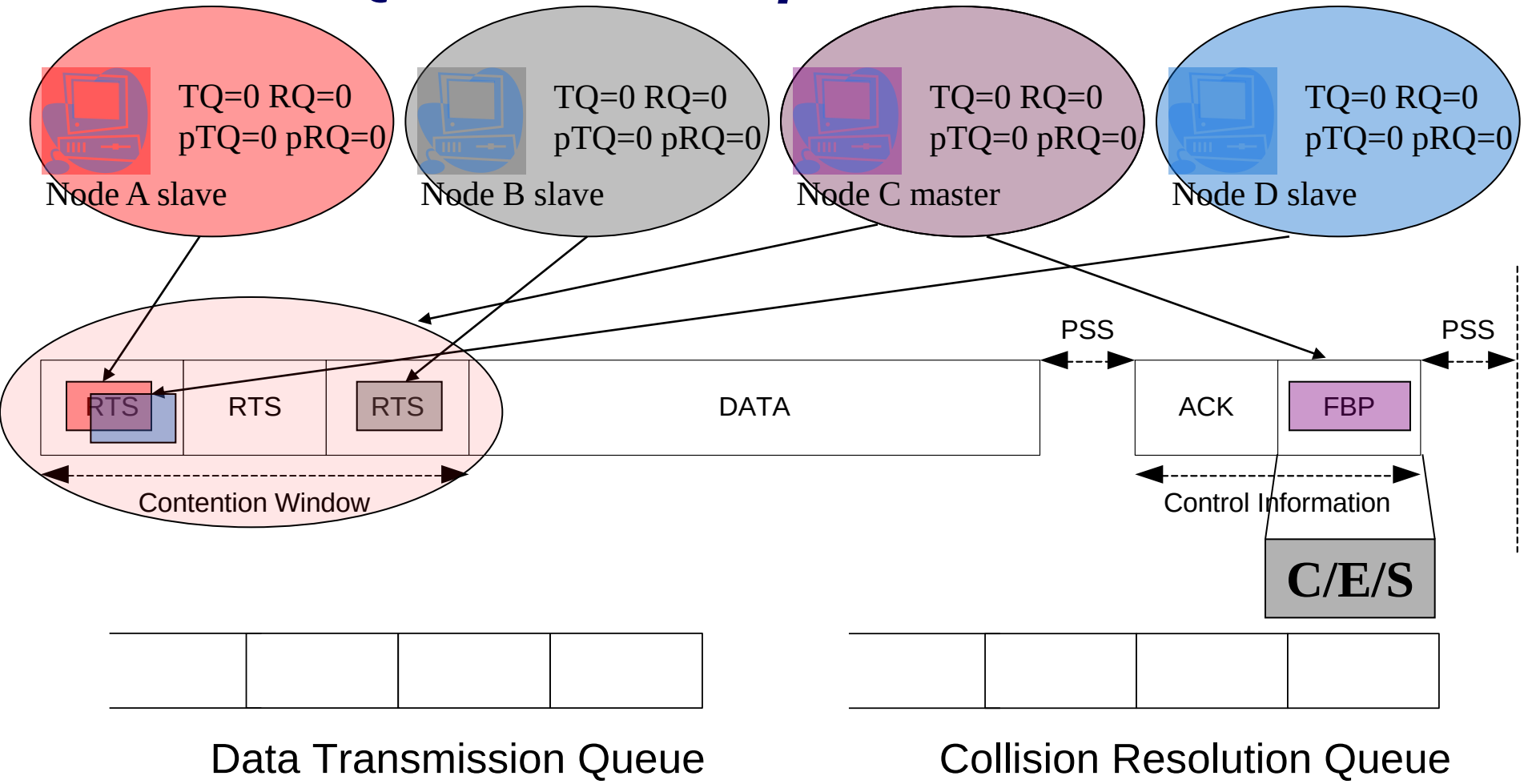


- **DTQ and CRQ are empty**
  - **TQ = 0 → no nodes waiting to transmit**
  - **RQ = 0 → no nodes waiting to solve a collision**
  - **pTQ = 0 and pRQ = 0 for all nodes**
- **Nodes A, B and D have to transmit**
- **They select AT RANDOM an access minislot to send and RTS**
  - **Node A selects minislot 1**
  - **Node B selects minislot 3**
  - **Node D selects minislot 1 → A and D WILL COLLIDE!**

DQMAN: Example

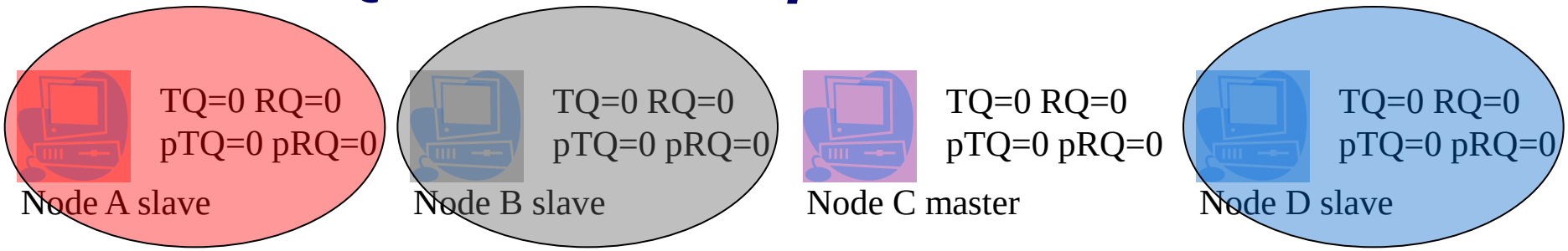
Beginning of frame i

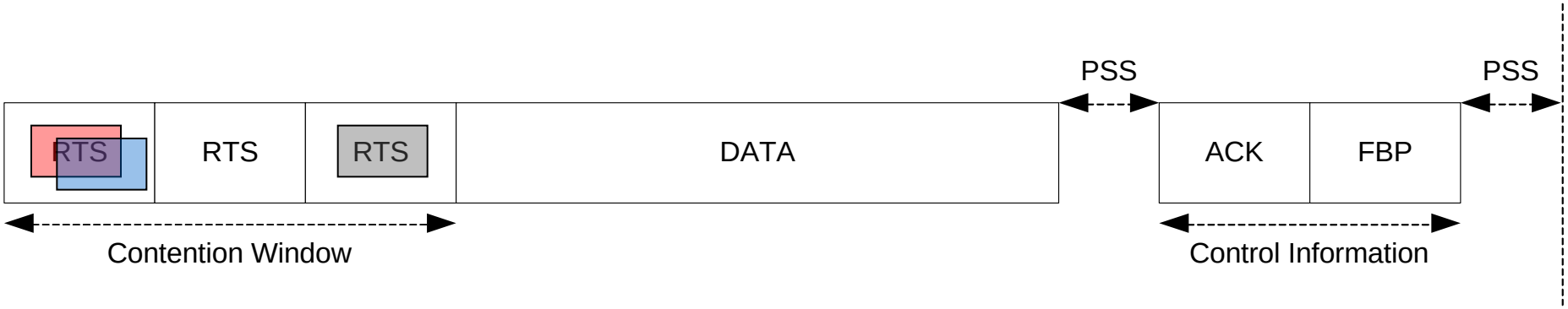
DQMAN: Example



Current Frame = i

DQMAN: Example





Data Transmission Queue

Collision Resolution Queue

Ending of Current Frame = i → Nodes execute protocol rules with FBP information

DQMAN: Example

Ending of frame i

DQMAN: Example

TQ=1 RQ=1
pTQ=0 pRQ=1

Node A slave

TQ=1 RQ=1
pTQ=1 pRQ=0

Node B slave

TQ=1 RQ=1
pTQ=0 pRQ=0

Node C master

TQ=1 RQ=1
pTQ=0 pRQ=1

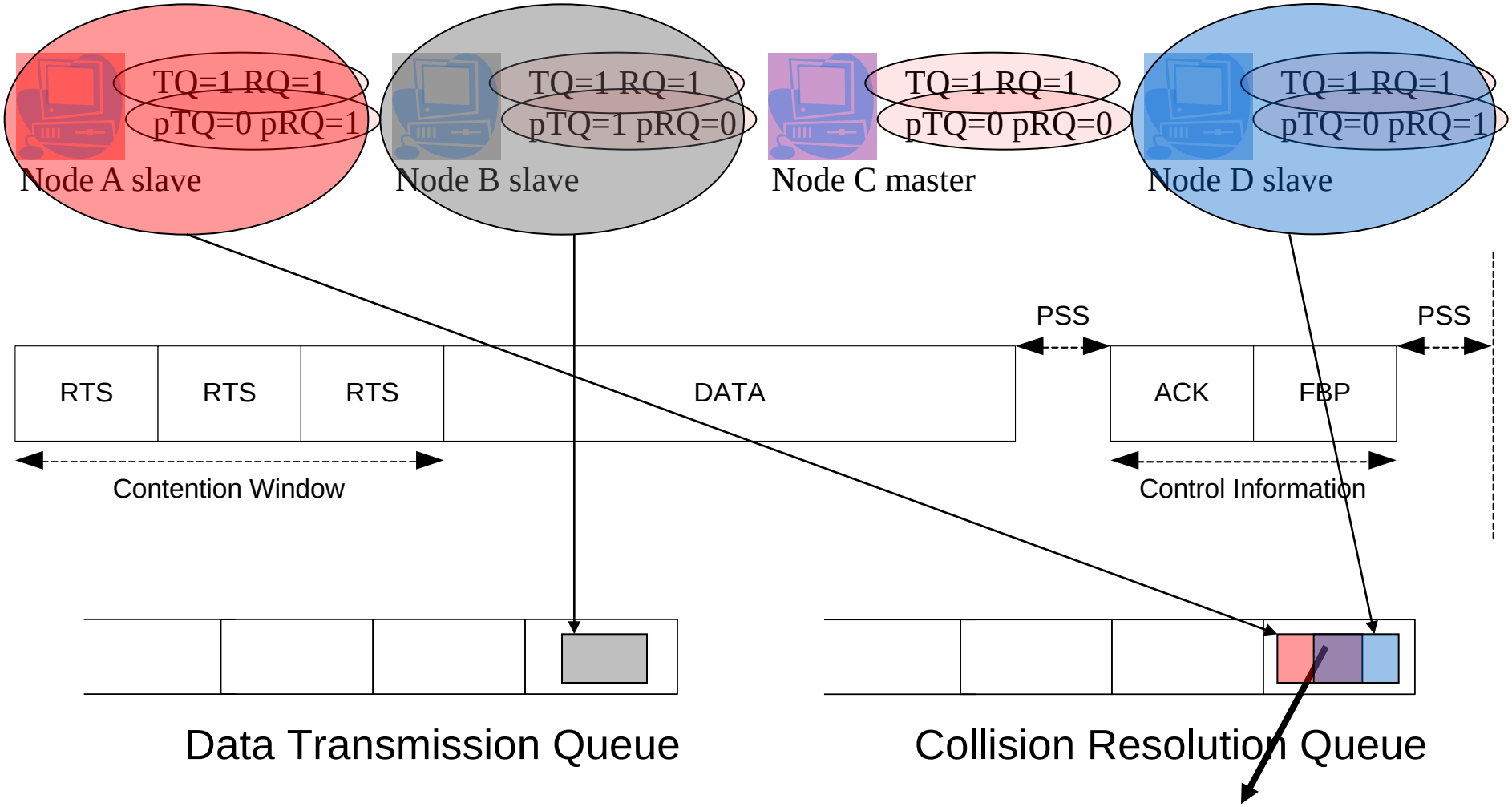
Node D slave

- **Node B succeeded → pTQ =1**
  - **Node B will transmit data in the next frame**
- **Nodes A and D collided in their access request**
  - **They enter CRQ**
  - **They are in the first position, i.e. pRQ=1 for both**
  - **They try to solve the collision in the following frame**
  - **They reselect AT RANDOM an access minislot**
    - **Node A selects minislot 2**
    - **Node D selects minislot 1**
    - **THEY WILL SUCCEED!**

DQMAN: Example

Beginning of frame i+1

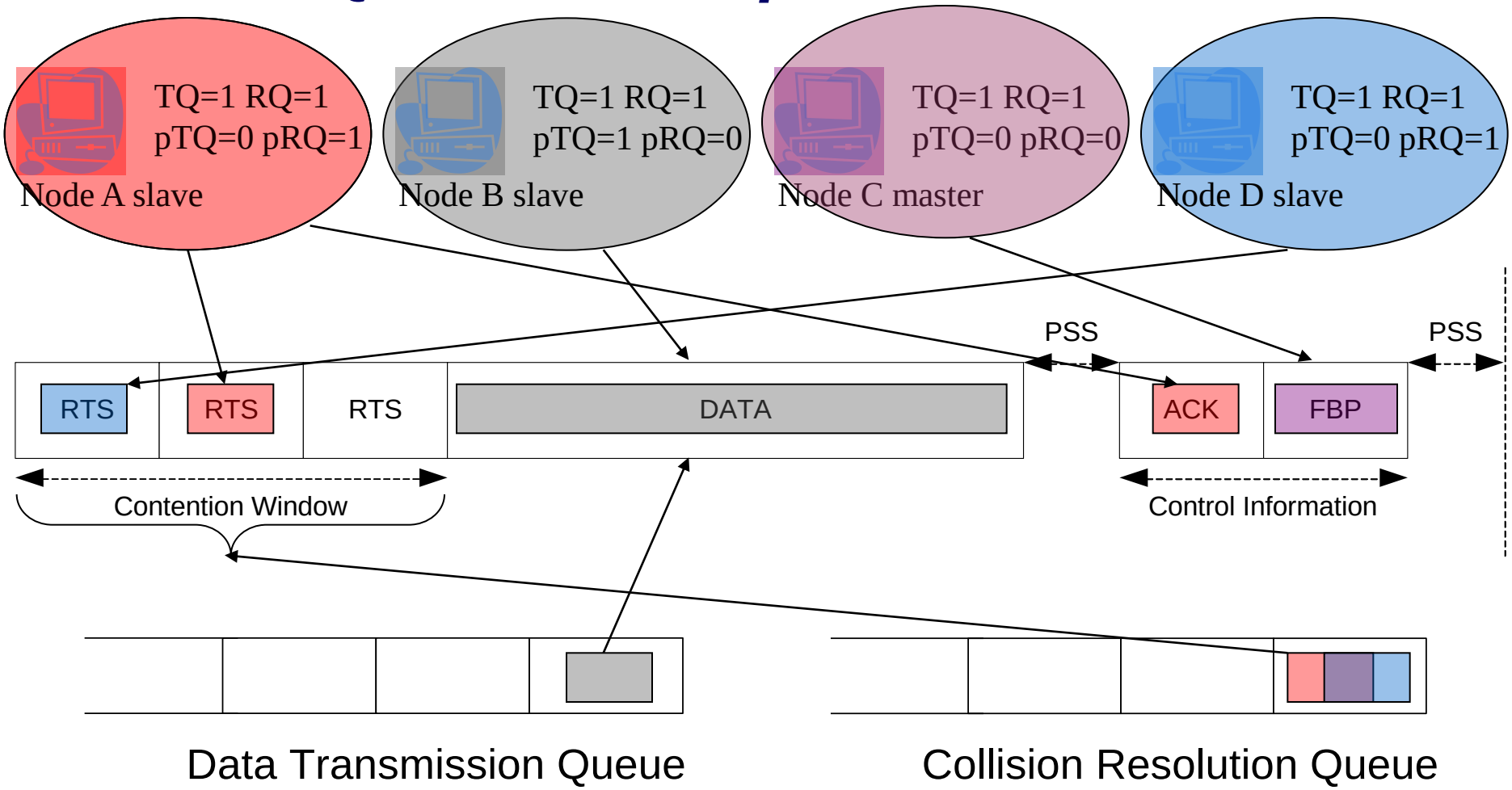
DQMAN: Example



New Access Channel Requests Forbidden

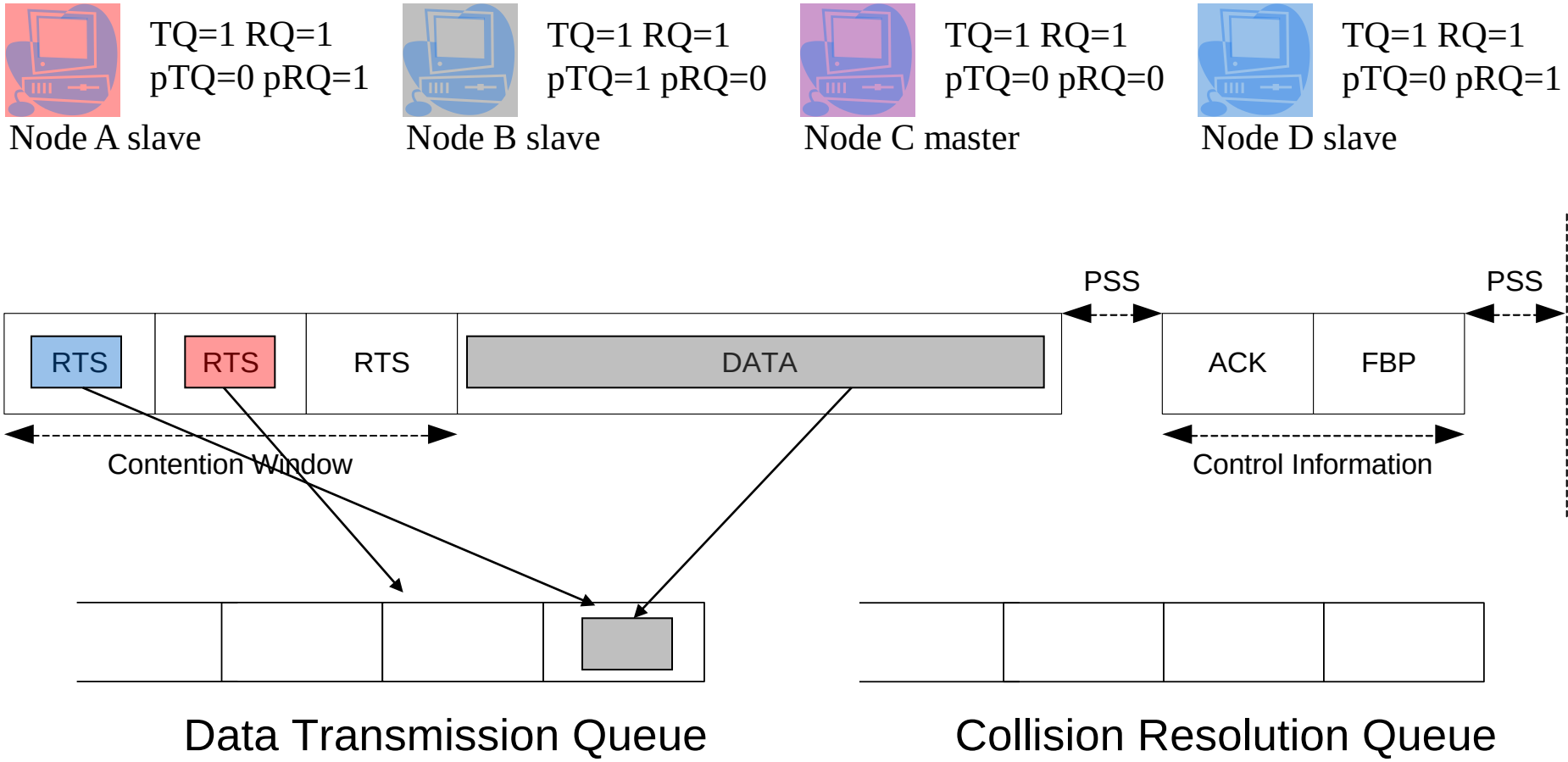
Current Frame = i +1

DQMAN: Example



Current Frame = i +1

DQMAN: Example



Current Frame = i +1

Thanks for your kind attention!

- Questions?

Jesús Alonso-Zárate, PhD.

Research Associate

CTTC

jesus.alonso@cttc.es