



# coherence in industrial automation

Jean-Pierre Thomesse  
Institut National Polytechnique de Lorraine  
Nancy, France



# coherence in industrial automation

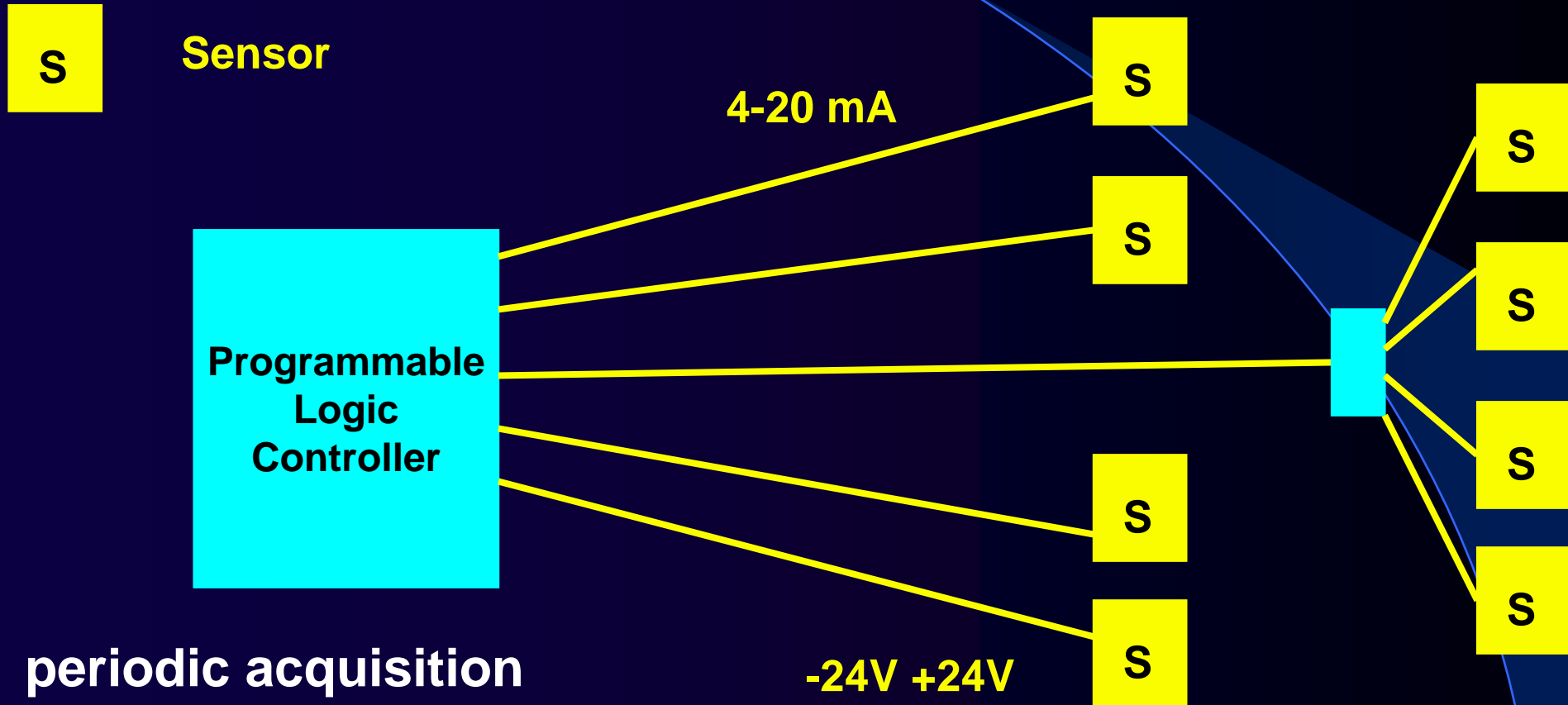
- coherence - intrinsic to applications
  - time coherence
- coherence - due to distribution
  - space coherence
- coherence vs. application architectures
- coherences of data and of actions
- to approach a “common global” state
- dynamic systems with their own evolution



# content

- **coherences - needs analysis**
- fieldbus requirements
- cooperation models - application relationships
- protocols and mechanisms
  - MAC and DLL
  - timeliness attributes
  - data list coherence - space coherence
  - coherence of actions
- application architecture
- conclusion

# coherences in centralized systems



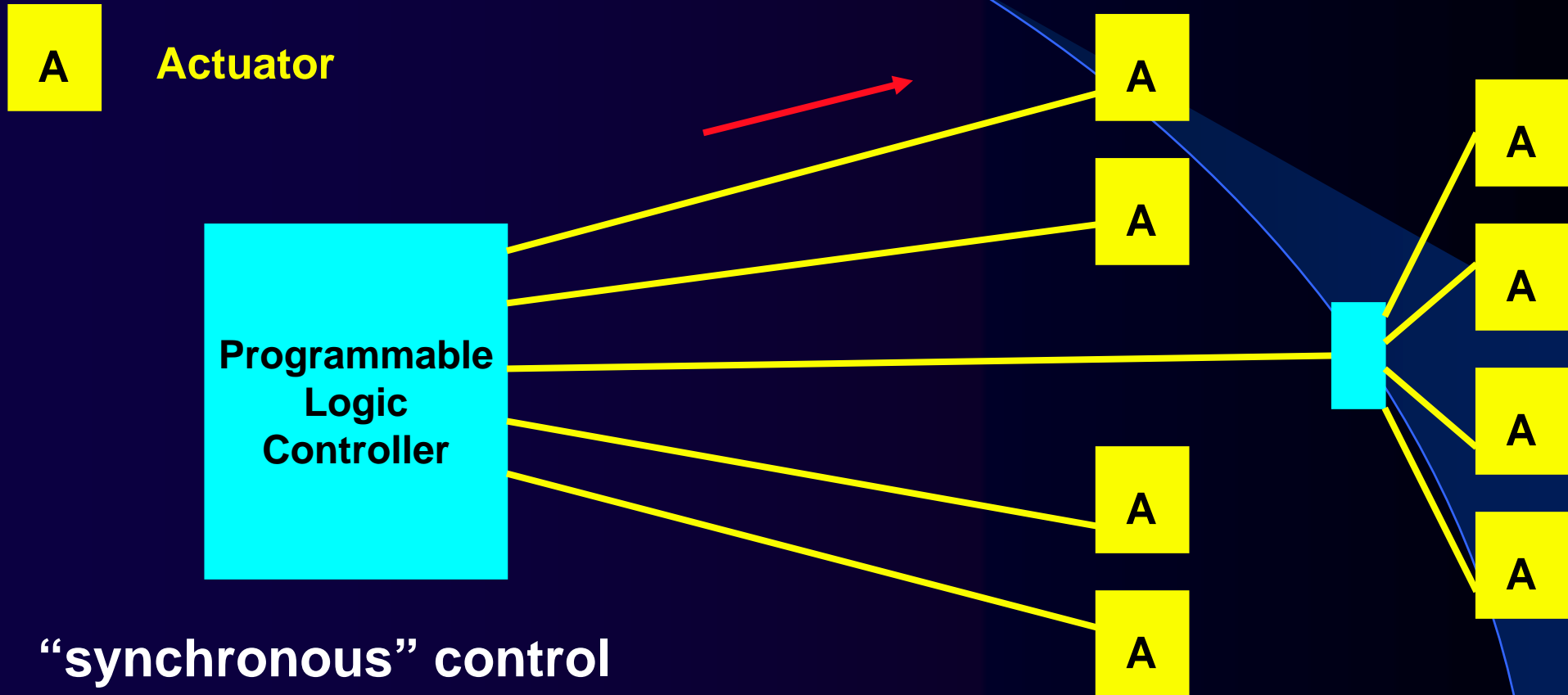
# coherences in centralized systems

- implicit temporal coherence
  - “simultaneous” acquisitions
  - all the physical laws suppose a common time
  - $pV=Cste$  :  $p(t)*V(t)= Cste$
- hypothesis
  - acquisition speed  $\implies$  simultaneity
  - no time stamping, no specific control

# coherences in centralized systems

- data acquisition
  - analog sensors: 4-20 mA standard
  - digital sensors: 0-5V, -24V+24V, 0-24V...
- polling at programmable sampling frequency
- interrupt (hardware sampling at high frequency)
- time stamping (centralized, at reception)

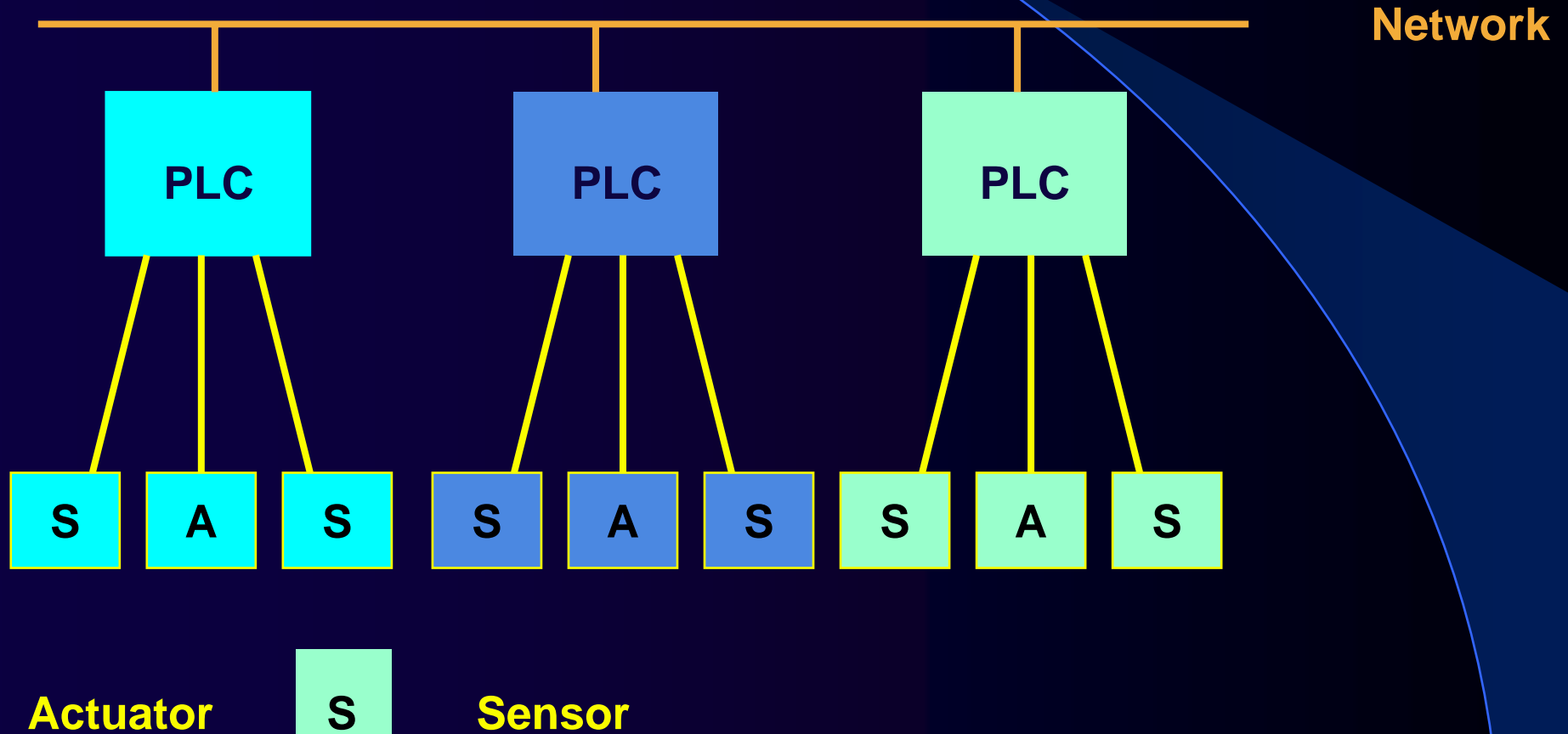
# coherences in centralized systems



# coherences in centralized systems

- implicit temporal coherence
  - “synchronous” control
  - actions done at the same time
  - according to the system dynamics
- hypothesis
  - speed of actions control  $\implies$  “simultaneity”
  - no time stamping, no specific control

# introduction of networks



# introduction of networks

- pros and cons
  - centralized control on each PLC
    - communication of states
    - each actuator controlled by a single system
  - communication between PLC
    - communication of sensors data through the network
    - transmission delays no more negligible
- problems of
  - time coherence and of space coherence (identical copies)



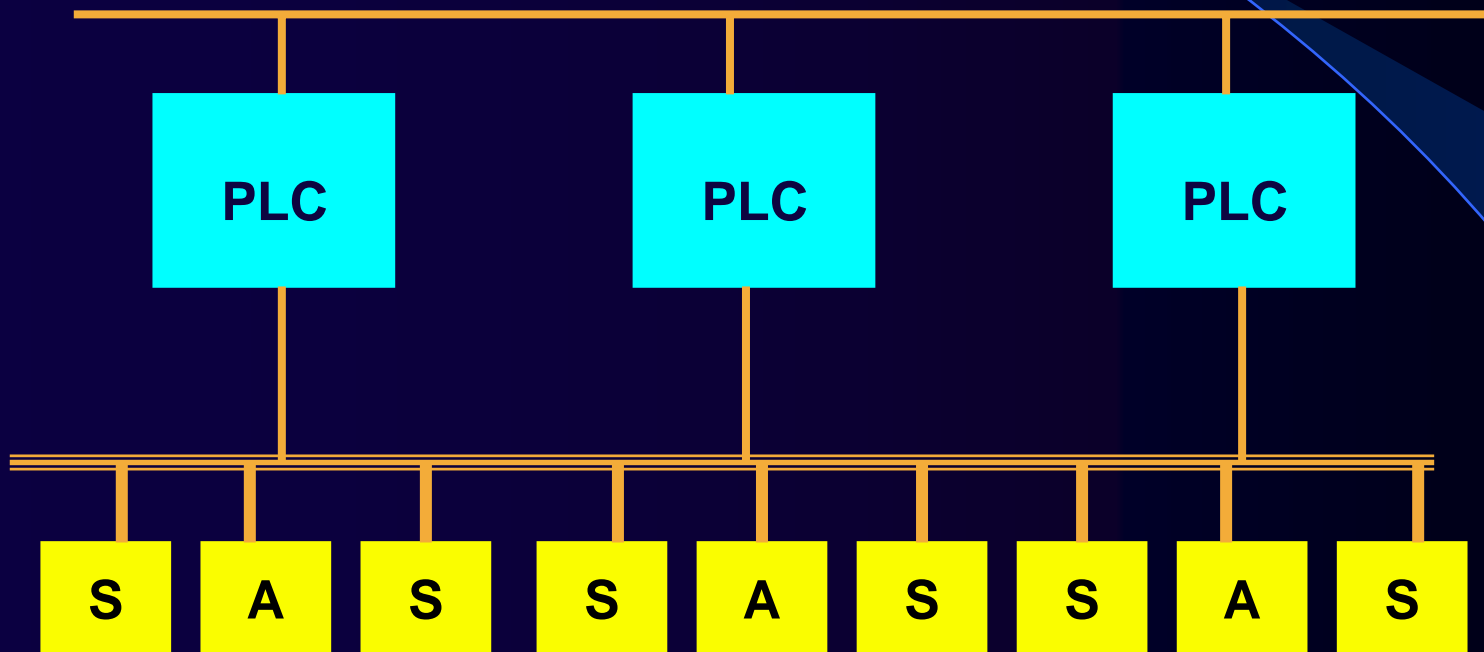
# introduction of networks

- idea of fieldbus
  - sensor and actuator networks
  - towards
    - sharing (multicasting) of sensors data
    - redundant control
    - common “global” state
    - smart instruments
    - really distributed control
    - with coherence properties

# introduction of networks

Network

Fieldbus



**A**

**Actuator**

**S**

**Sensor**

# introduction of networks

- two approaches
  - centralized control on each PLC
    - polling of own sensors
    - control of own actuators
  - distributed control
    - any data communication
    - transmission delays negligible
    - any distribution of application processes
- problems of
  - time coherence and of space coherence (identical copies)



# content

- coherences - needs analysis
- **fieldbus requirements**
- cooperation models - application relationships
- protocols and mechanisms
  - MAC and DLL
  - timeliness attributes
  - data list coherence - space coherence
  - coherence of actions
- application architecture
- conclusion



# questionnaire ISA-1985

- benefits of fieldbus
  - lowering cost, ease of adding devices, accuracy of information, enhancing the maintainability, remote access to data, advanced control strategies
- describing field devices
  - max response time and message frequency
- information flows
  - grouping of devices, topology, number of stations, redundancy
- application environment
  - power, wires type, insulation, flammable atmosphere

# requirements - application

- definition of traffic

- process (real time traffic)

- measurement
    - alarm setting
    - status

identified data or messages

name	value	status
------	-------	--------

- installation (non real time traffic)

- tag number
    - manufacturer's data
    - additional information for maintenance

identified data or messages

# requirements - application

- definition of other services
  - control of access - security
- types of data
- polled and unsolicited messages
- full logical connectivity
- application architectures (distributed or centralized)
- time coherences (of data, of actions)
- space consistency



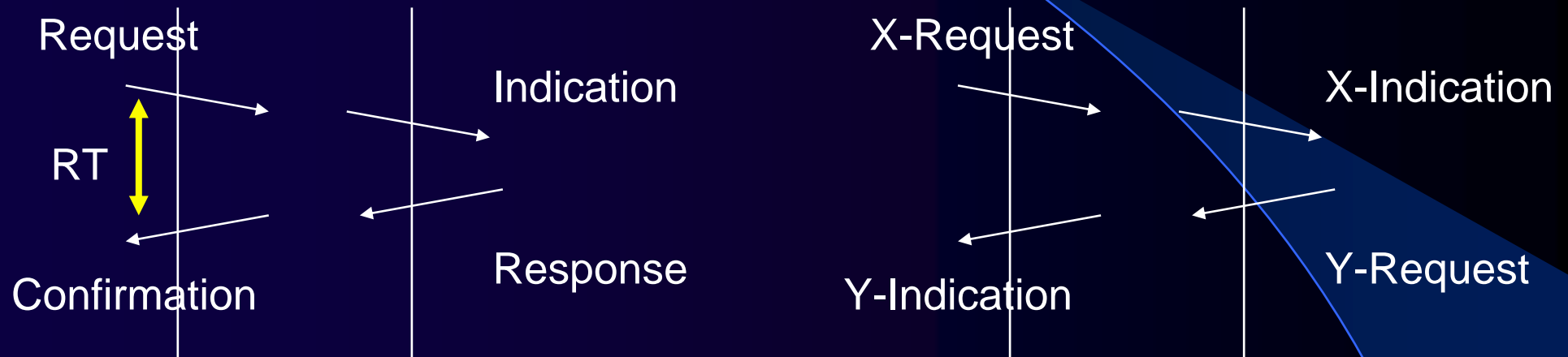
# content

- coherences - needs analysis
- fieldbus requirements
- **cooperation models - application relationships**
- protocols and mechanisms
  - MAC and DLL
  - timeliness attributes
  - data list coherence - space coherence
  - coherence of actions
- application architecture
- conclusion

# application layer

- two main classes of relationships
- client - server
  - for messages and data
    - and variants
- publisher - subscriber
  - for data
    - and variants

# client-server



quality of service:

reliability : confirmed service

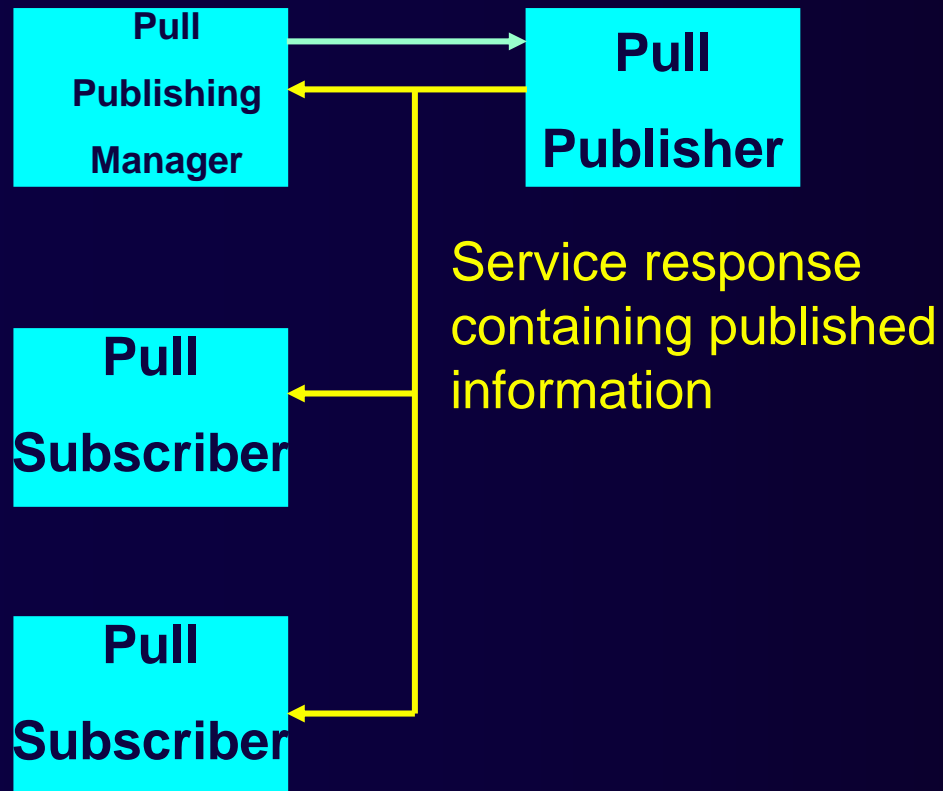
response time : transport delay + local server response time

# MMS example

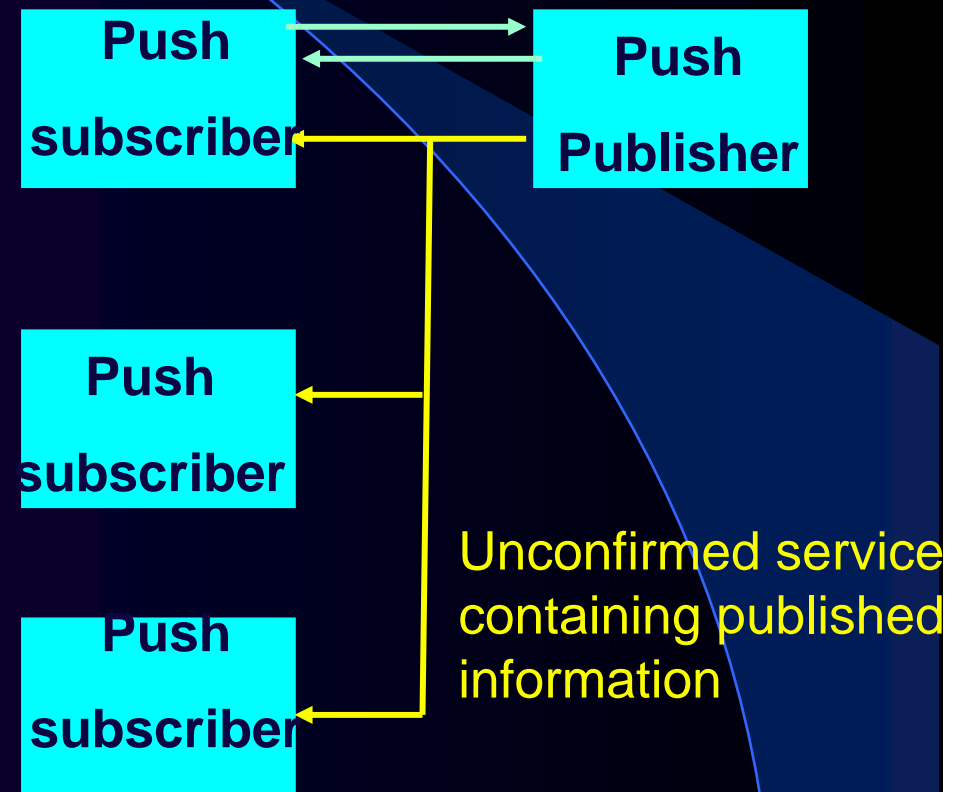
- VMD - Virtual Manufacturing Device
- objects such as
  - variables
  - processes
  - events
  - semaphores...
- coherence
  - server (if clients) and client (several servers) sides
  - no built-in coherence between clients
  - to be specified (multi-clients - server)

# publisher - subscriber models

Confirmed service request

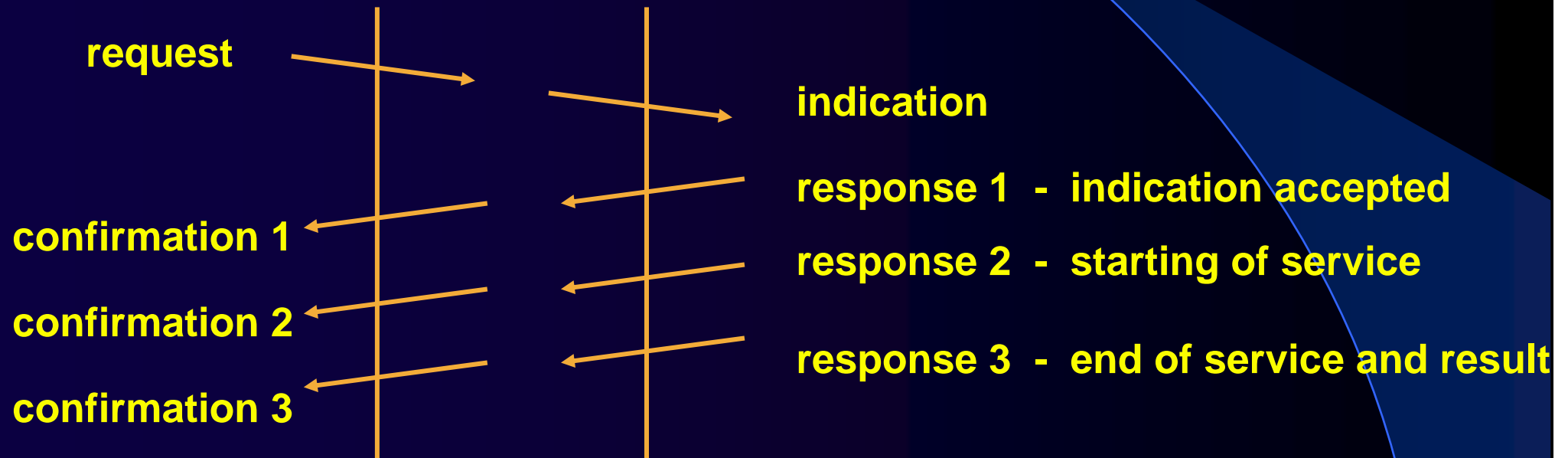


Confirmed service request/response



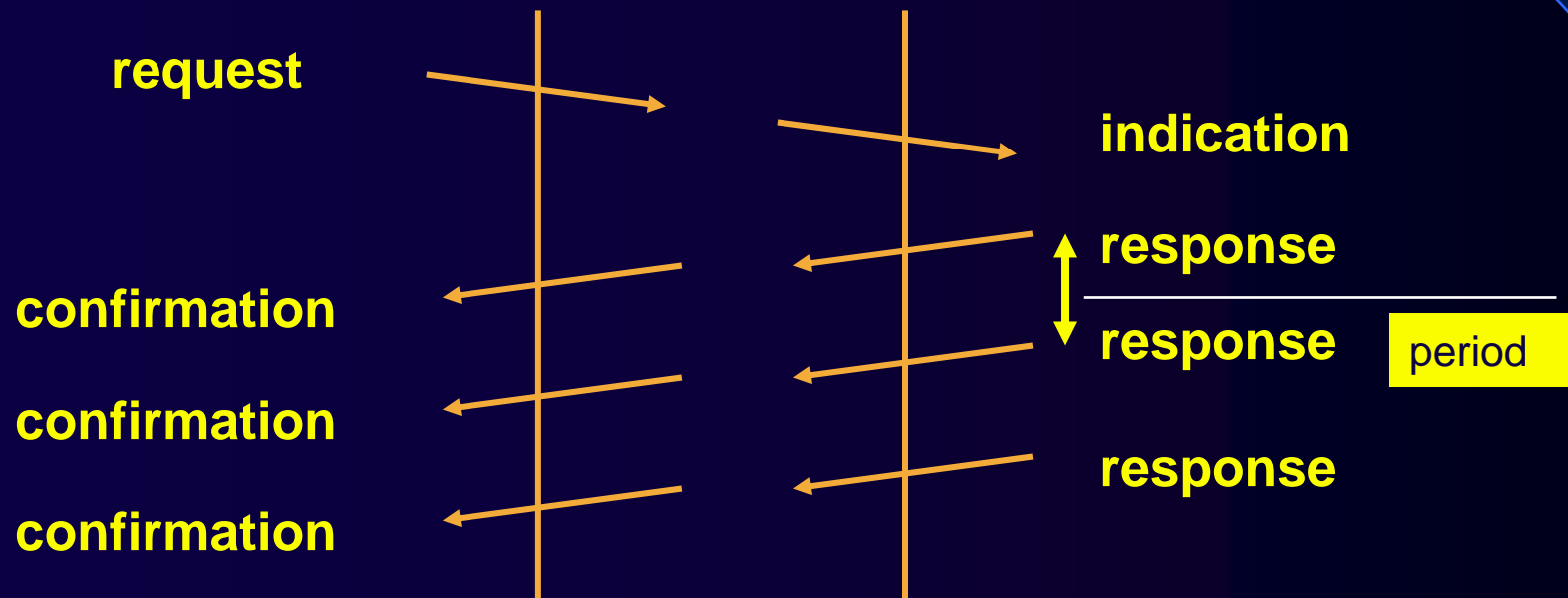
# other models

## multi confirmation client - server



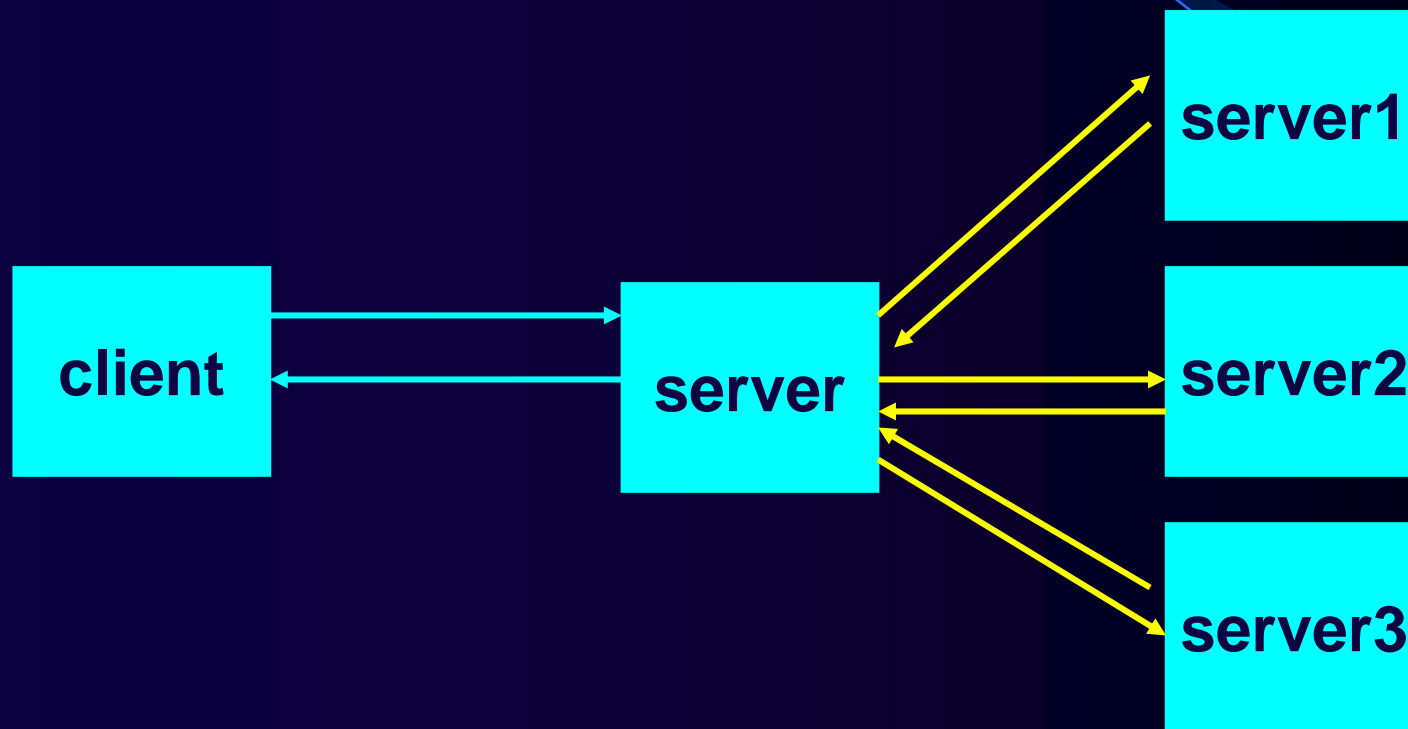
# other models

multi responses client - server - cf. Push Publisher



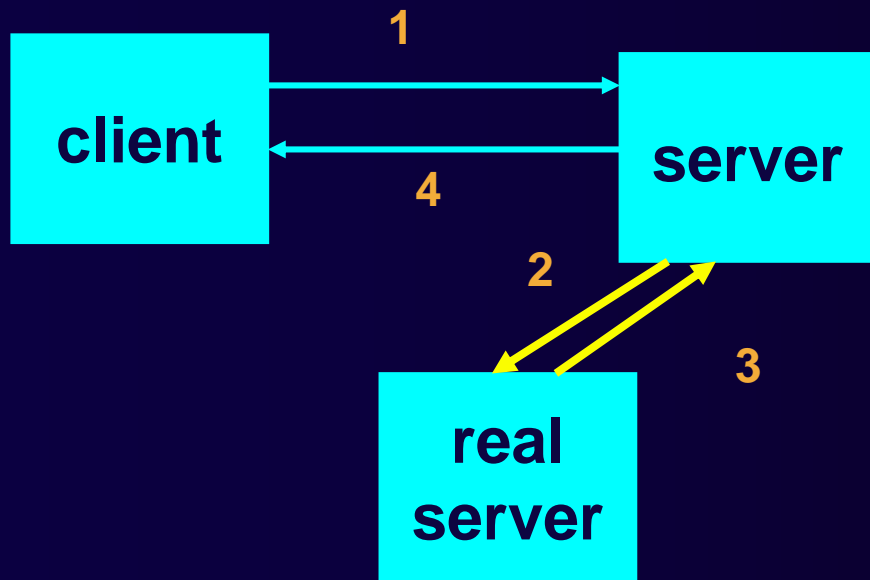
# other models

client - multi-server

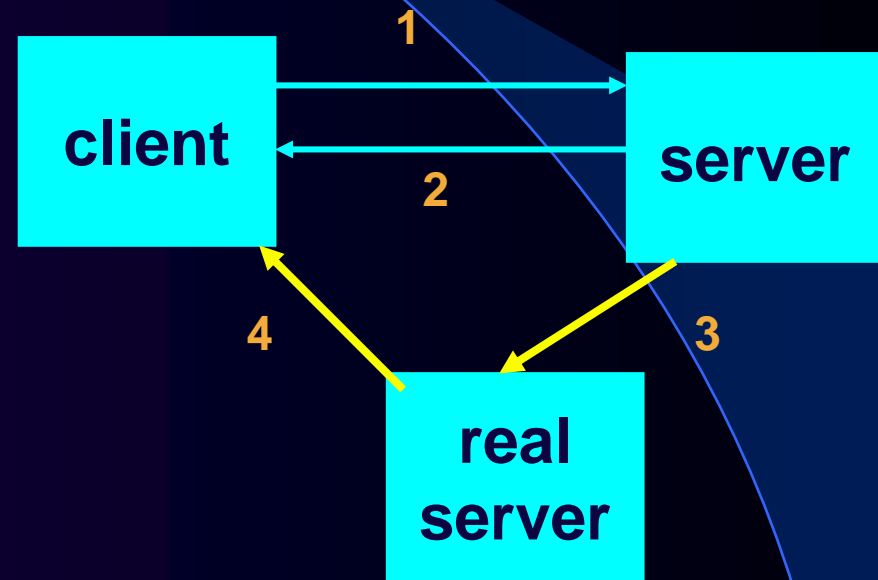


# other models

## third part model



case 1



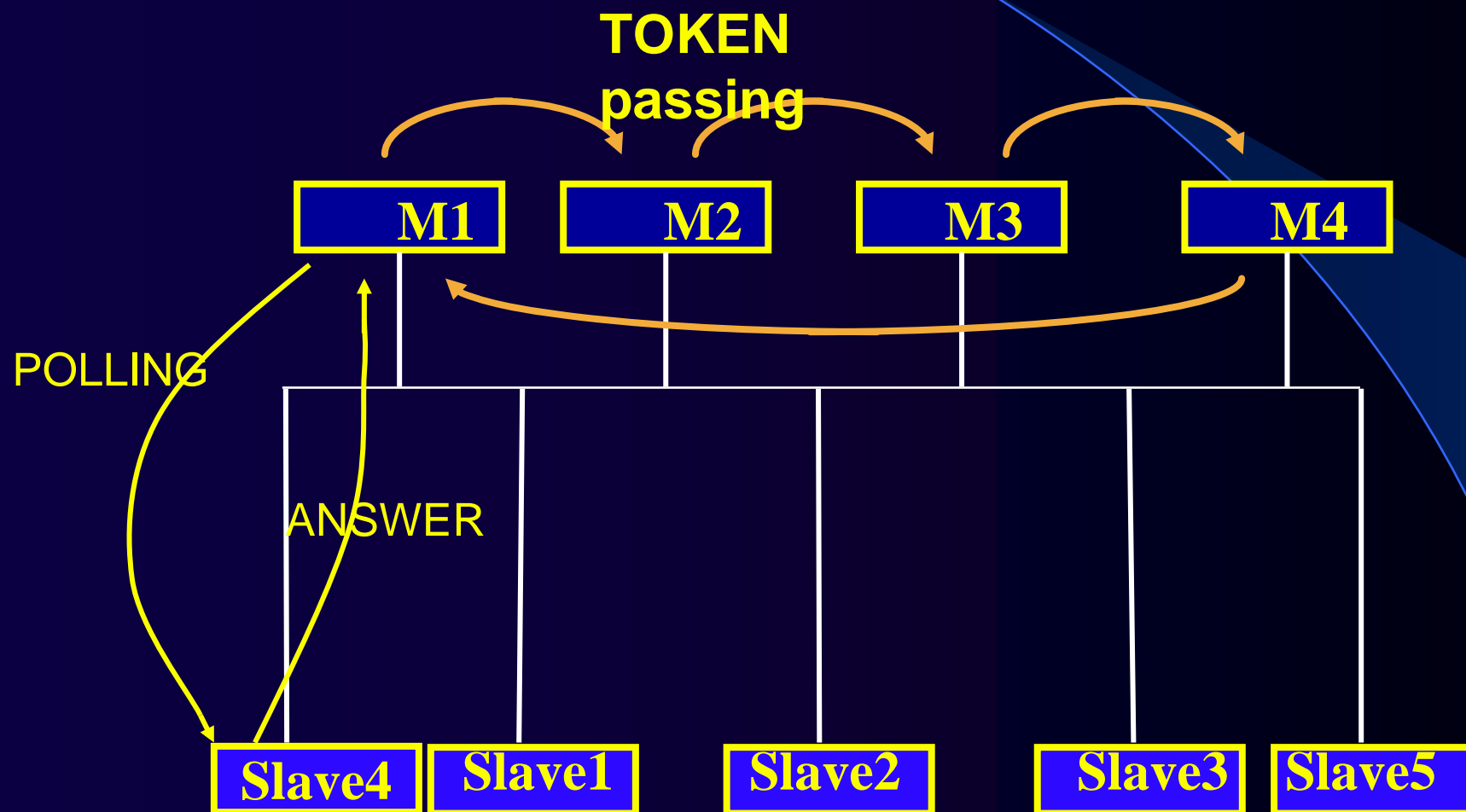
case 2



# content

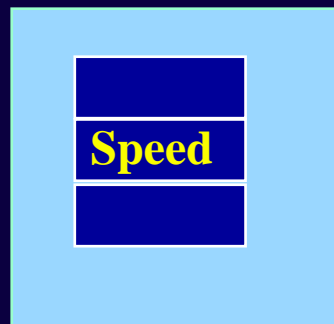
- coherences - needs analysis
- fieldbus requirements
- cooperation models - application relationships
- **protocols and mechanisms**
  - MAC and DLL
  - timeliness attributes
  - data list coherence - space coherence
- application architecture
- conclusion

# token and polling

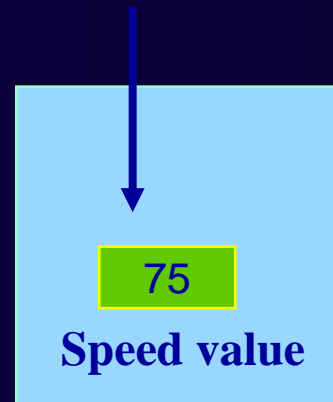


# polling of data - pull PS

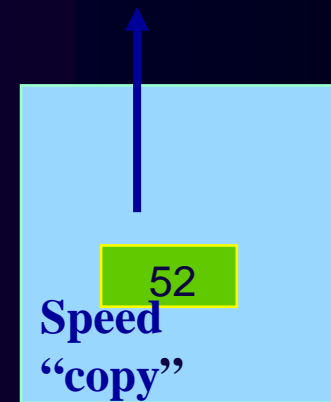
**Bus arbitrator  
polling table**



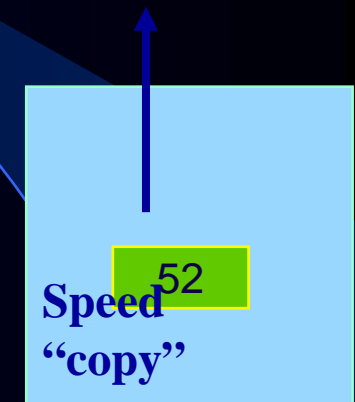
**Local Write**



**Local Read**

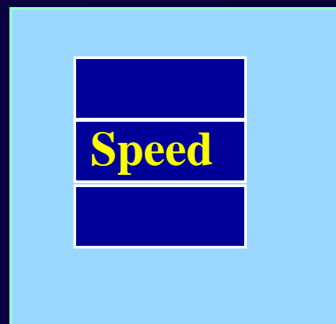


**Local Read**

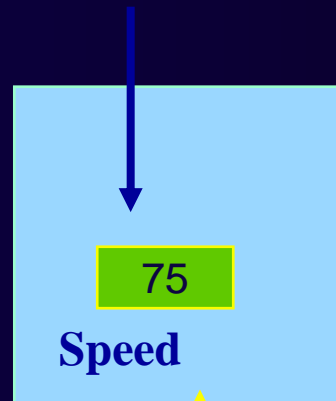


# polling of data

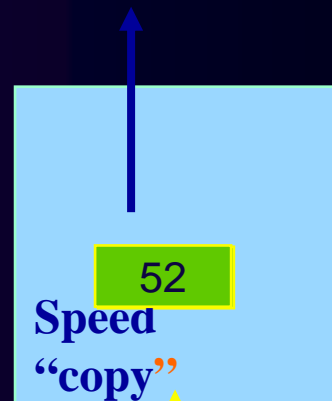
**Bus arbitrator**



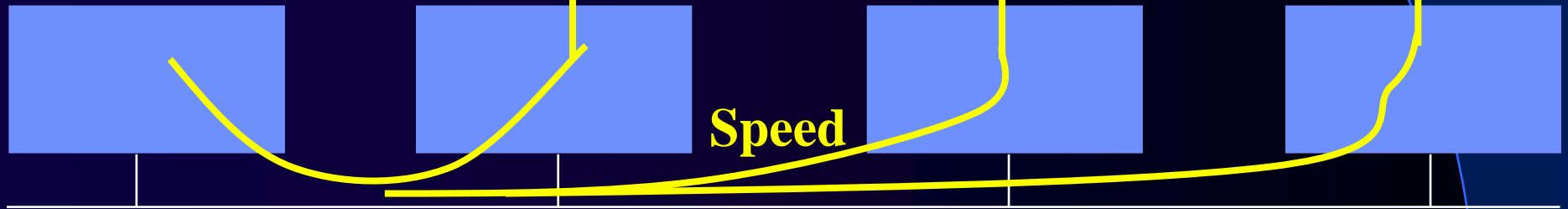
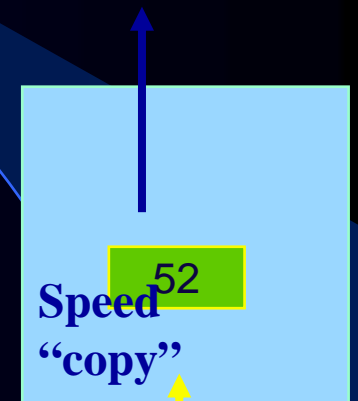
**Local Write**



**Local Read**



**Local Read**



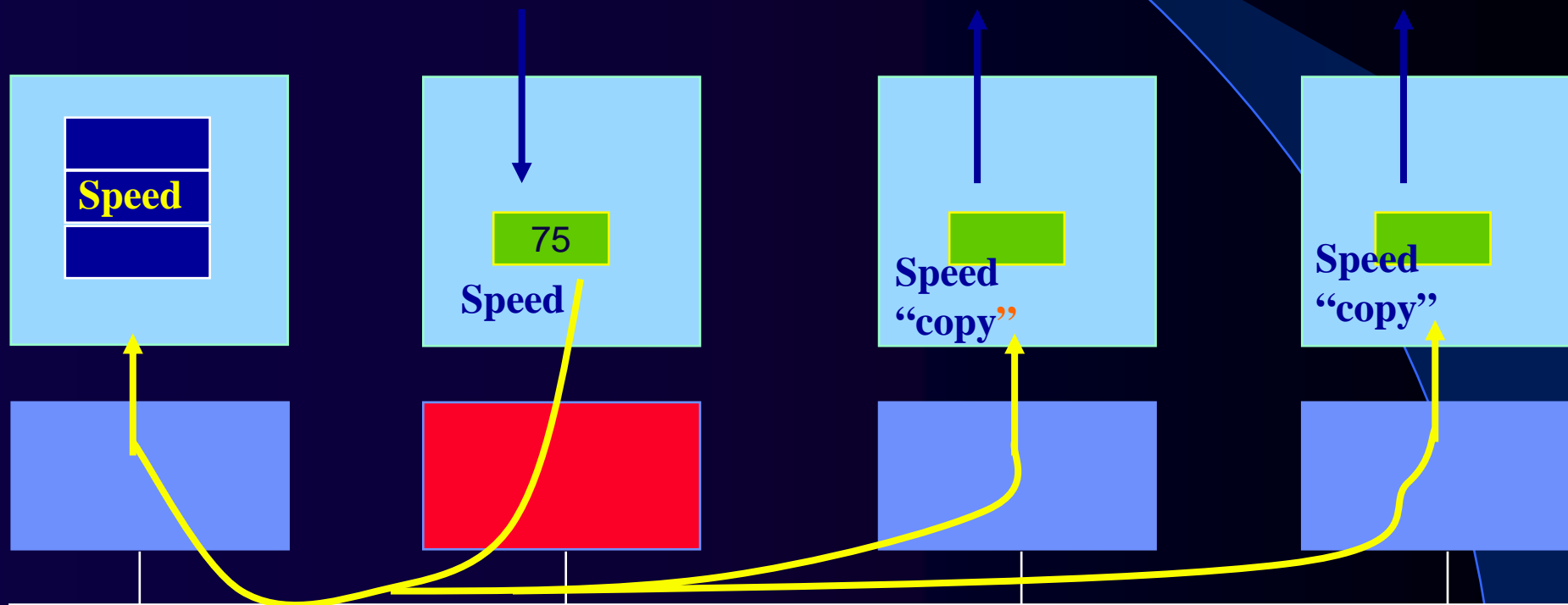
# polling of data

Bus arbitrator

Local Write

Local Read

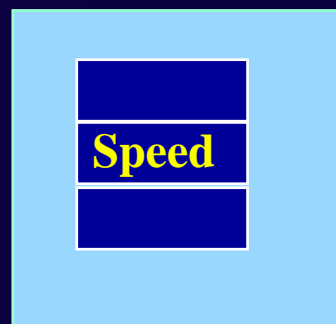
Local Read



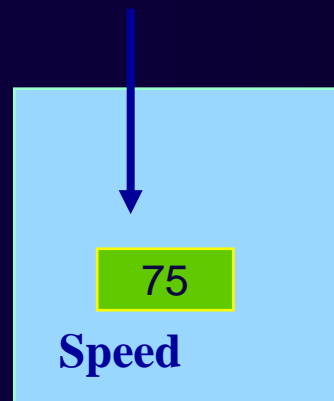
$$v(\text{Speed})=75$$

# polling of data

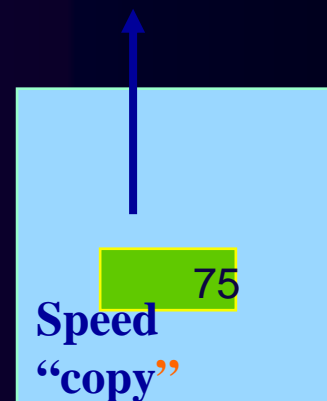
**Bus arbitrator**



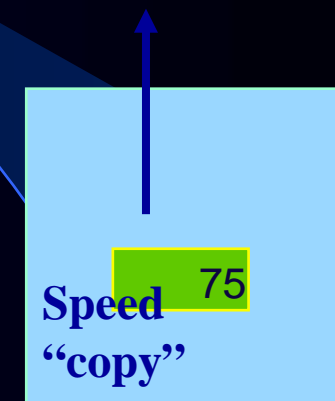
**Local Write**



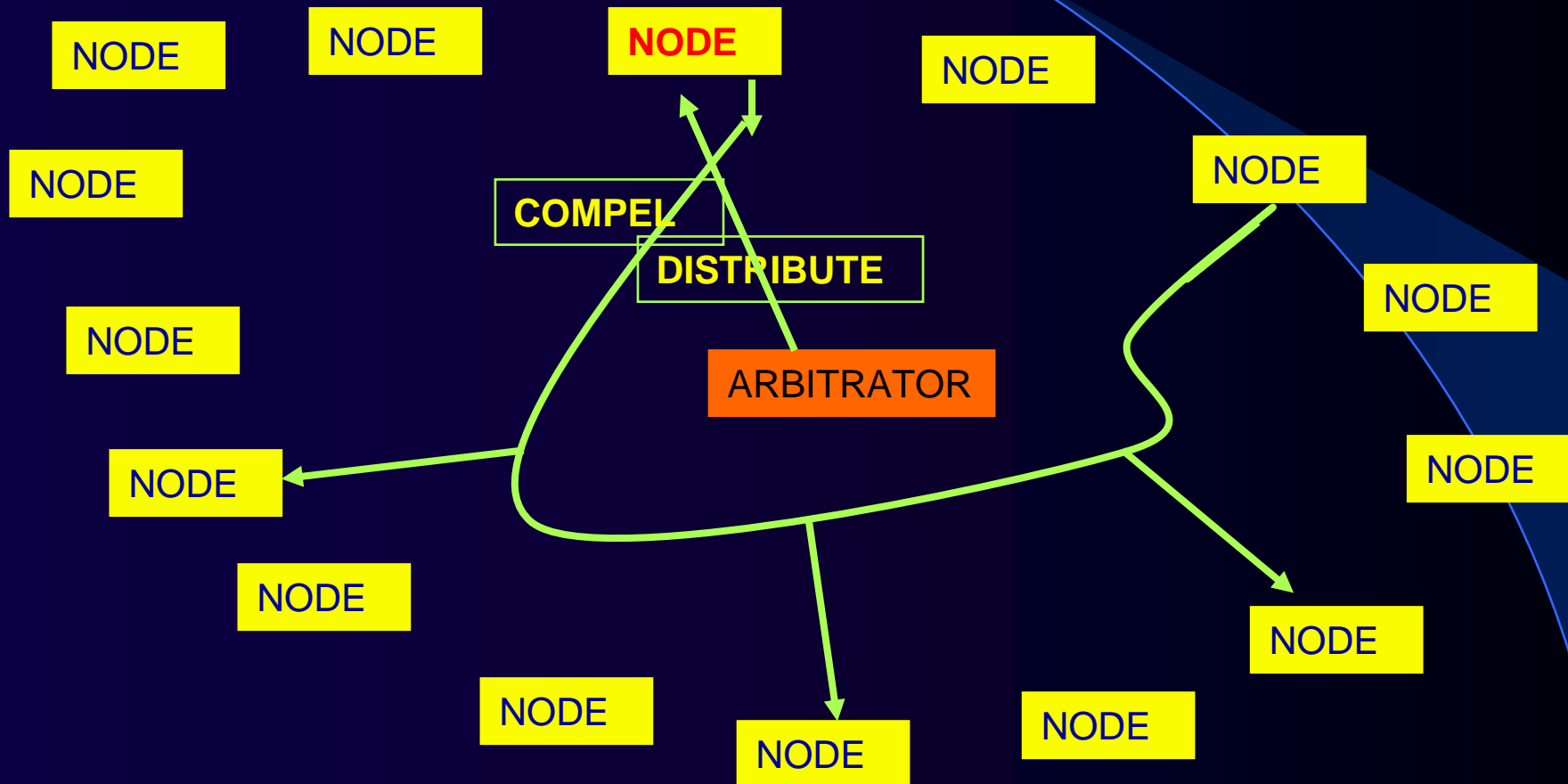
**Local Read**



**Local Read**

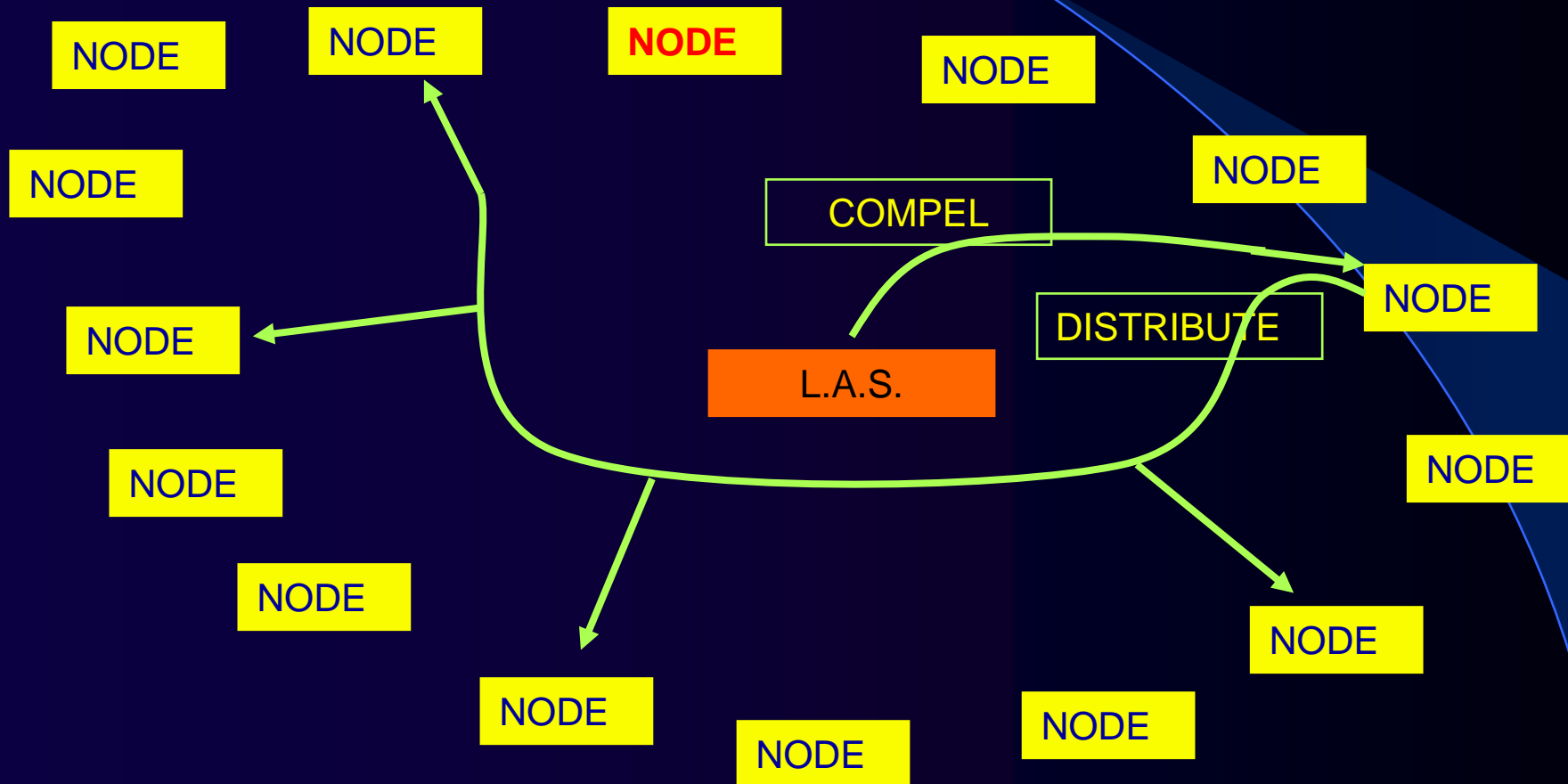


# arbitrator

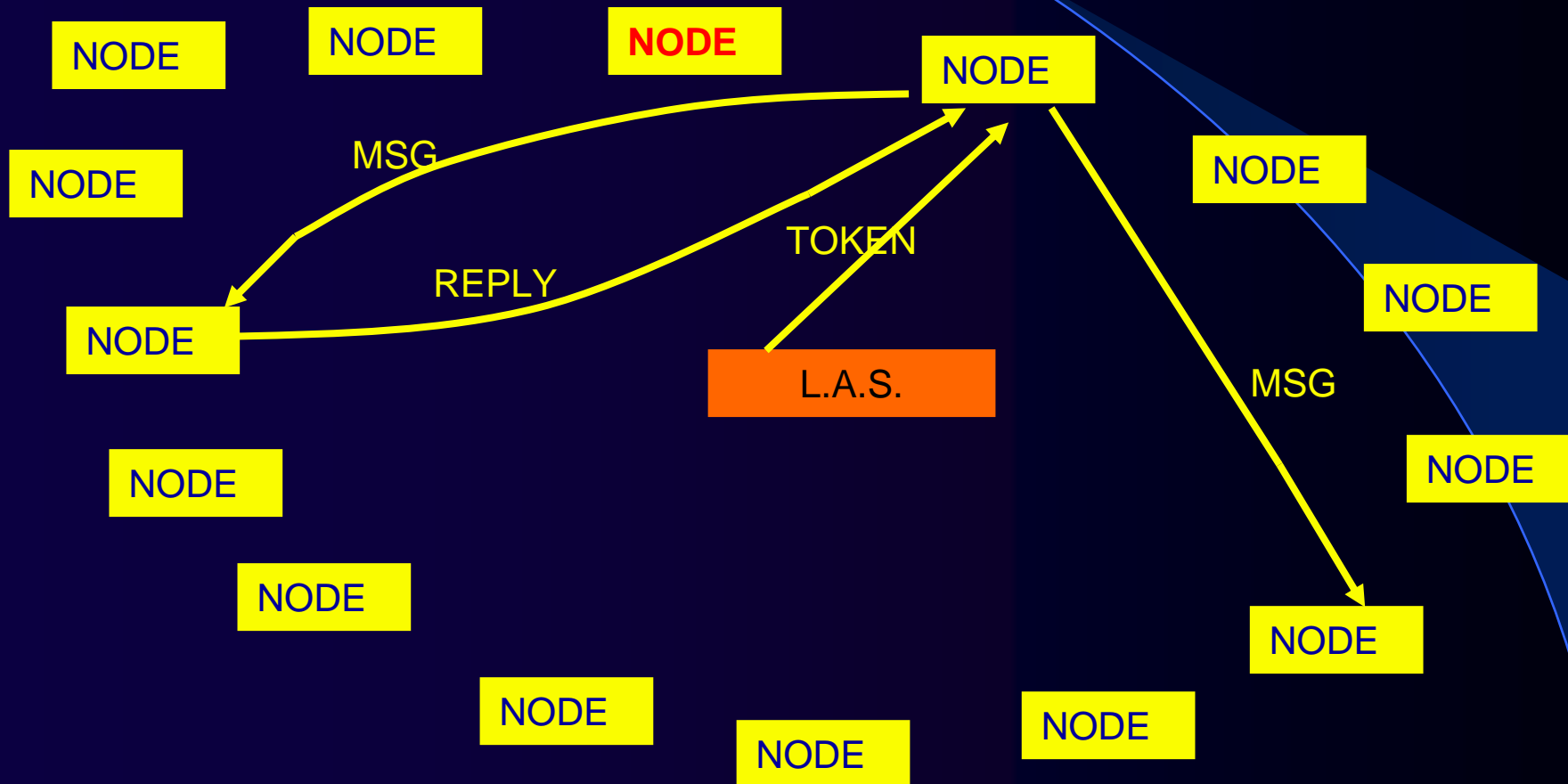




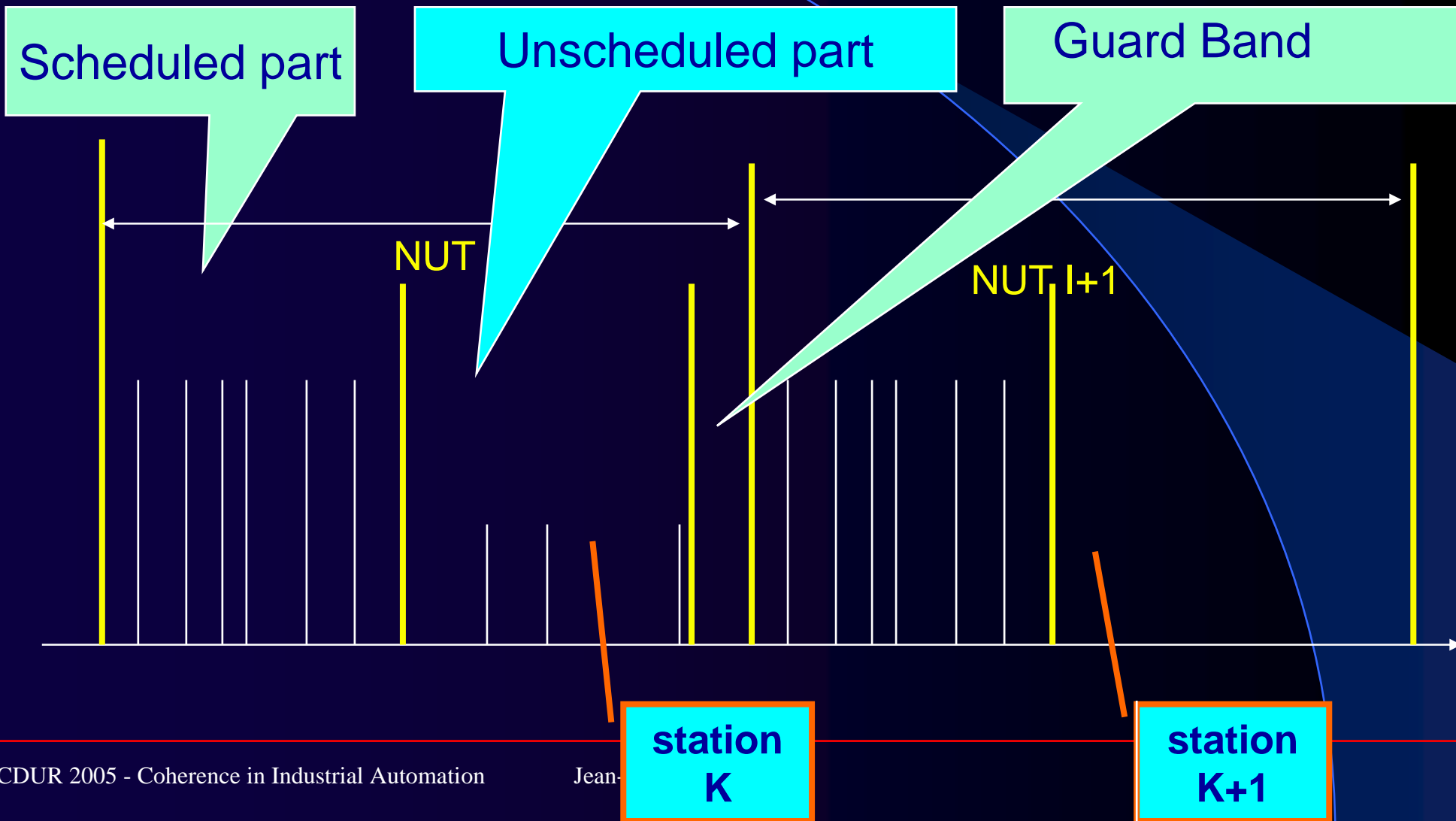
# L.A.S.



# L.A.S.



# ControlNet - 1

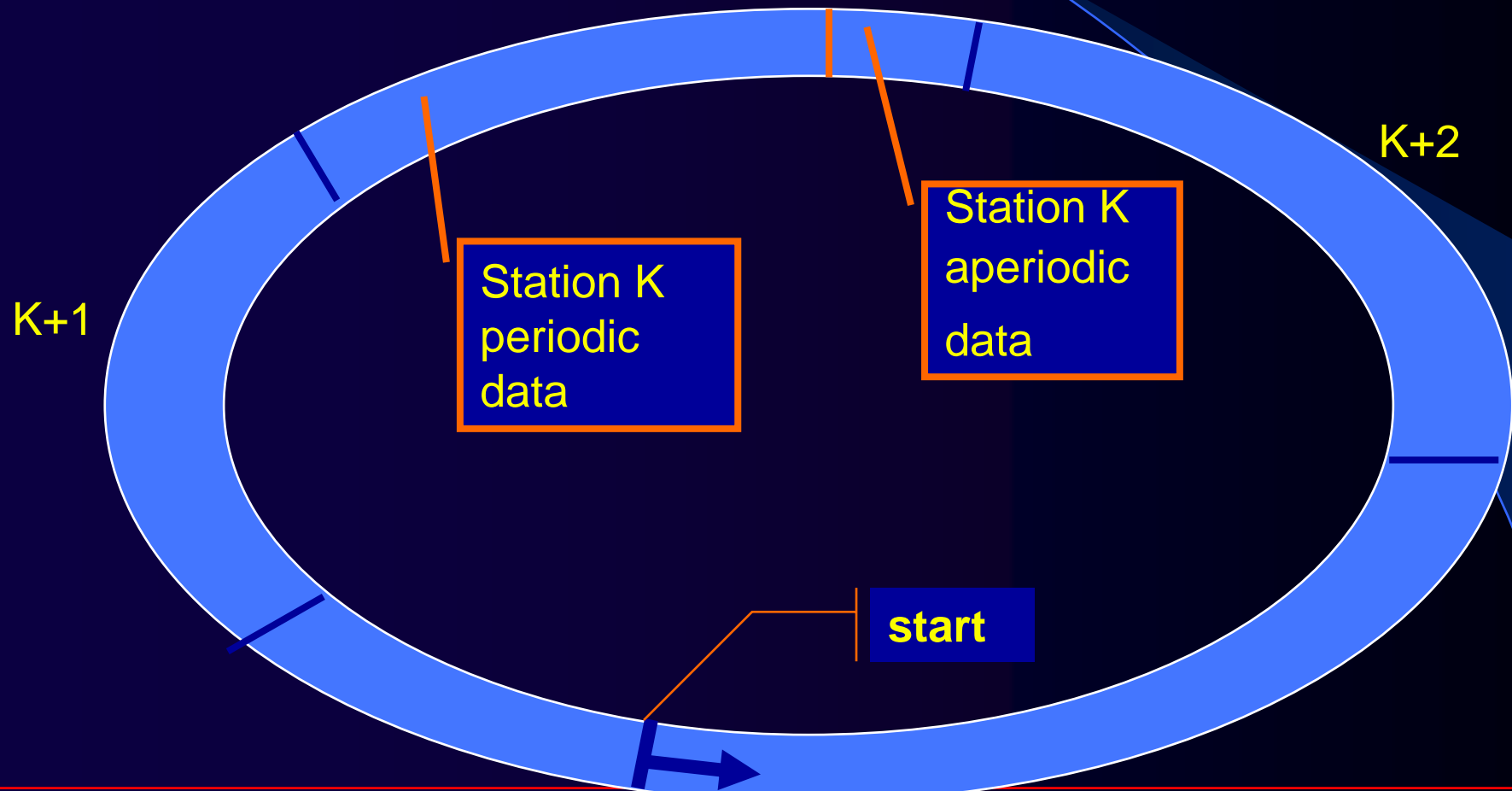




## ControlNet - 2

- in a NUT, three time windows
  - scheduled
  - unscheduled
  - Guard Band
- one MAC Frame by node in scheduled part
- predictable and deterministic manner
- Round Robin in the unscheduled part

# Interbus

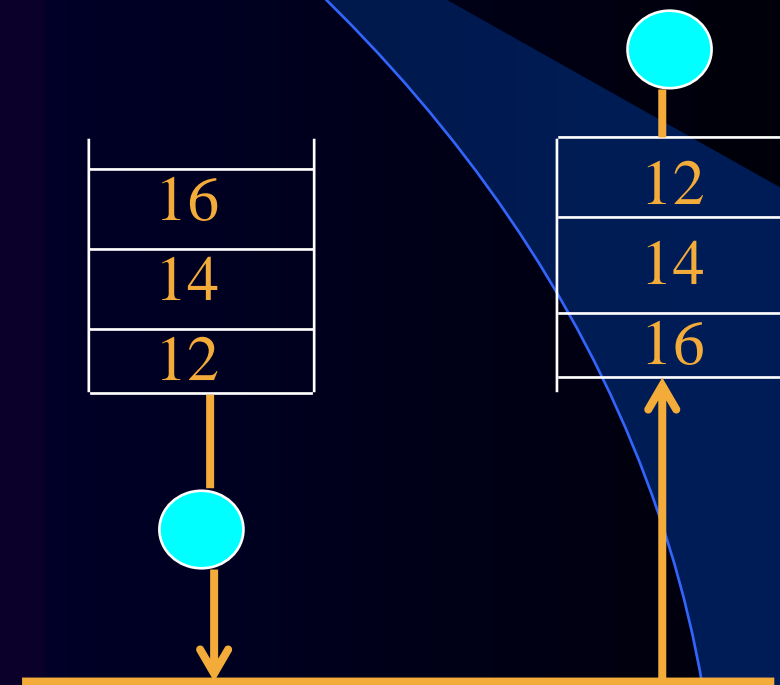
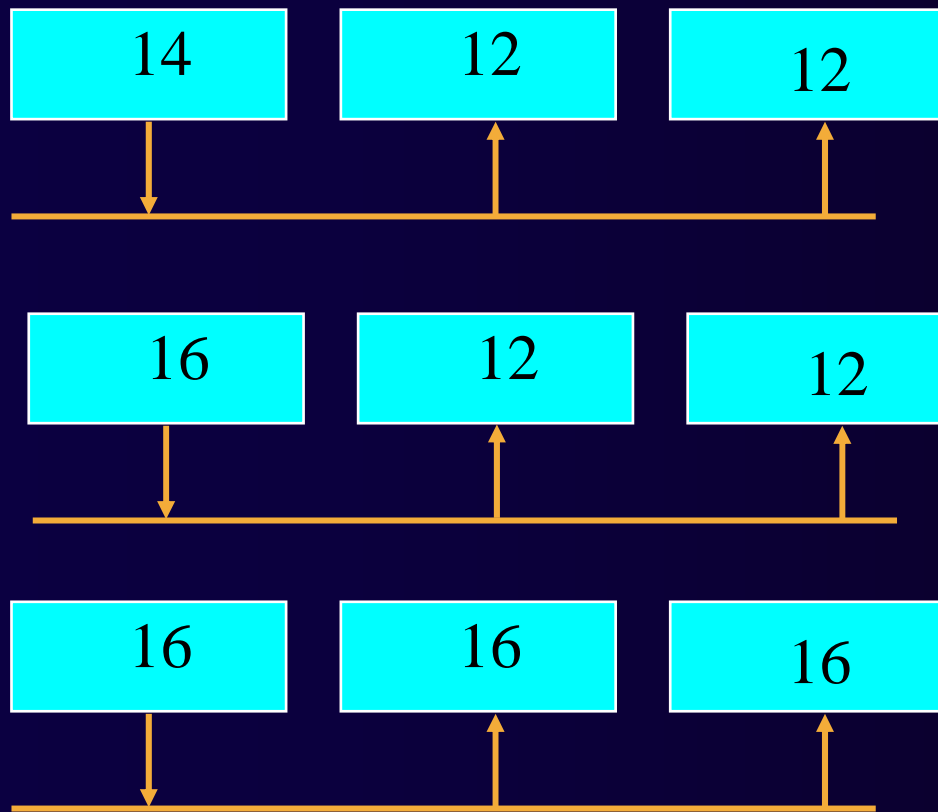




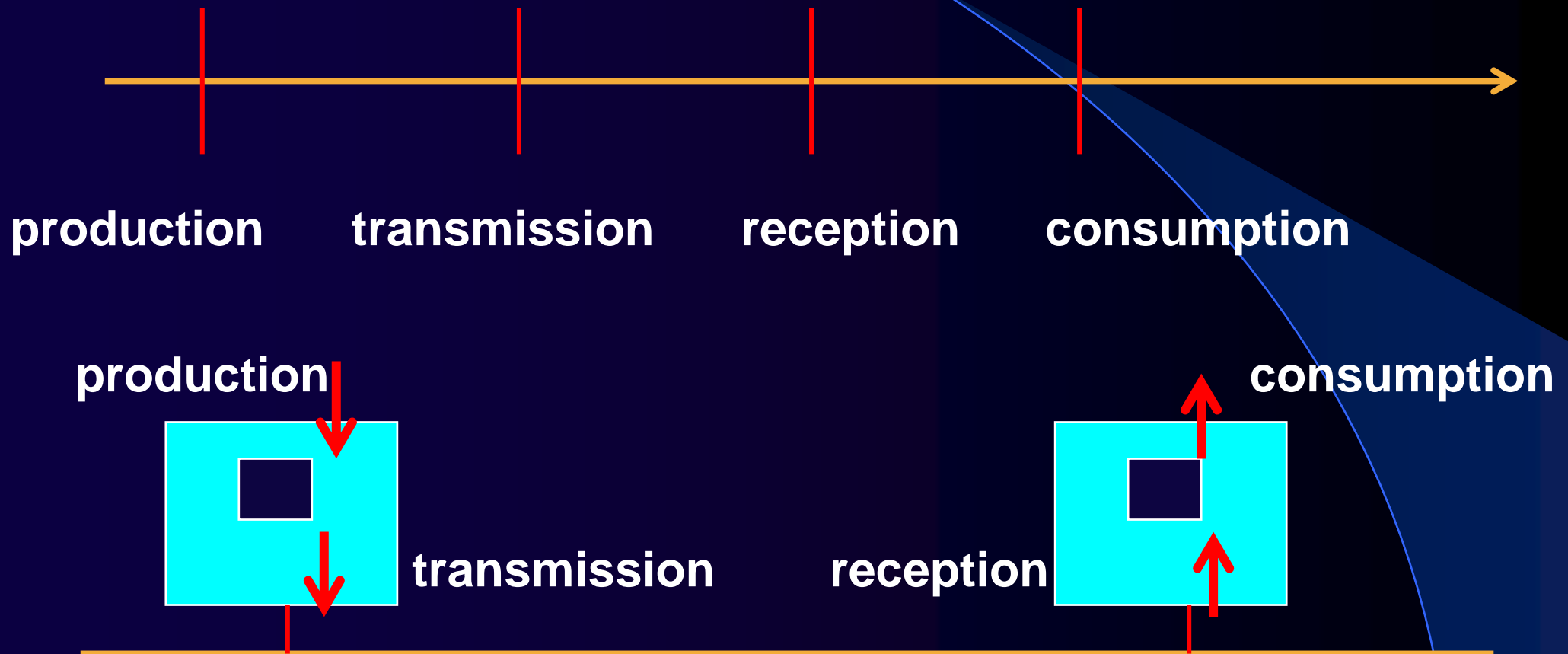
# content

- coherences - needs analysis
- fieldbus requirements
- cooperation models - application relationships
- **protocols and mechanisms**
  - MAC and DLL
  - **timeliness attributes**
  - data list coherence - space coherence
- application architecture
- conclusion

# buffers and queues

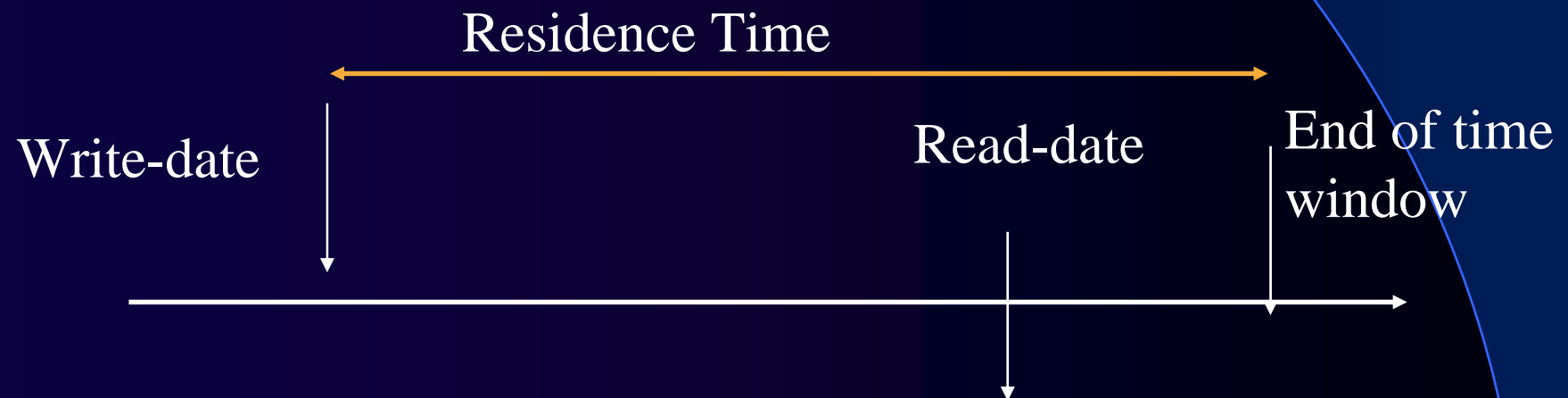


# timeliness



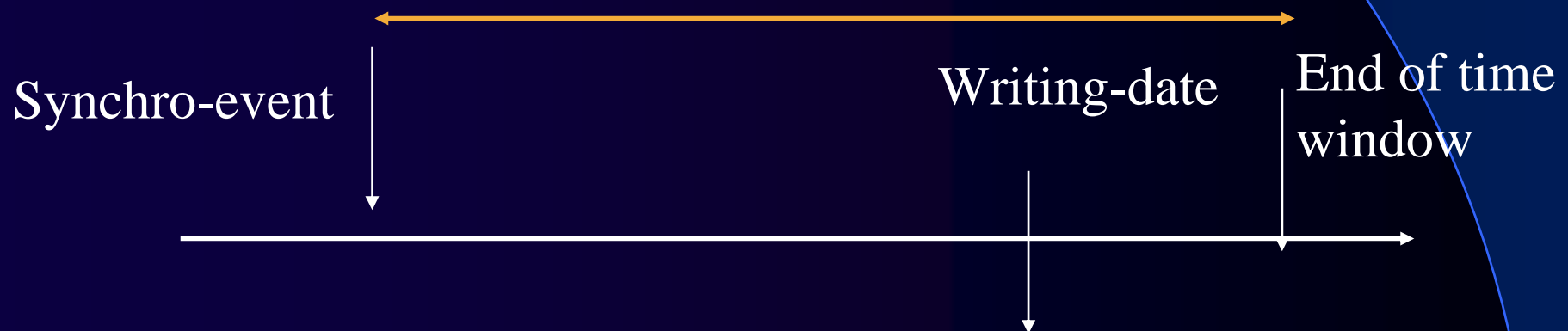
# residence attribute

- assessment based upon the time that a data unit has been resident in a buffer.



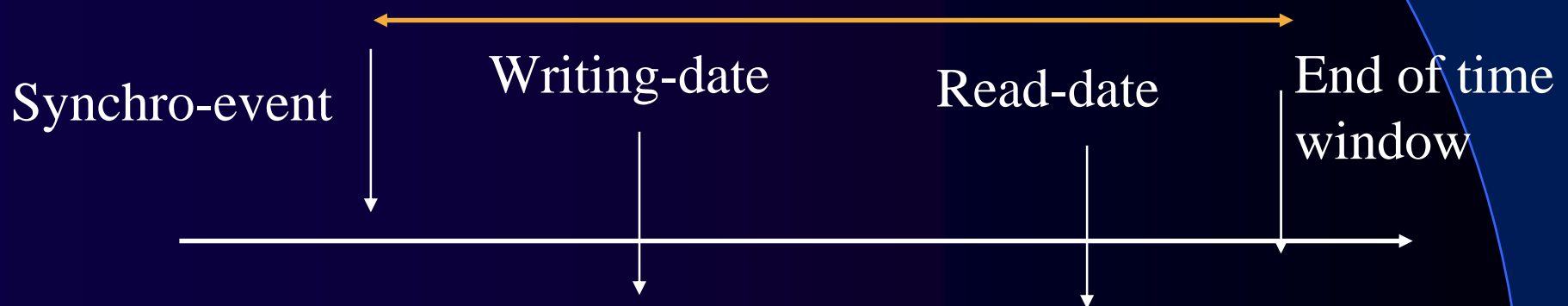
# update attribute

- assessment based upon the time interval between a synchronising event and the moment the buffer is written  
Update-Time



# synchronous attribute

- assessment based upon the time intervals and timing relationships between
  - a synchronising event
  - the moment when the buffer is written
  - the moment the buffer is read



# time coherence

- time coherence of data, of actions, of events
- simultaneity of events
- occurrences in a given time window
- time coherence of
  - productions
  - consumptions
  - other actions or events

# time coherence of data

- subscriber of data 1 and data 2
  - value of data1 and data2 with possibly
    - production attribute
    - transmission attribute
    - reception attribute
    - consumption attribute
- data1 and data2 are time coherent when the attributes are all correct



# coherence of actions

- synchronization of actions
- synchronization variables - multicasting
- local interpretation of
  - variable received indications



# time coherence

- control of time coherence
  - data received indication
  - allows, in multi peer connections, the synchronization of subscribers
- usable to control any actions simultaneity
- verification of time coherence
  - by timeliness attributes



# content

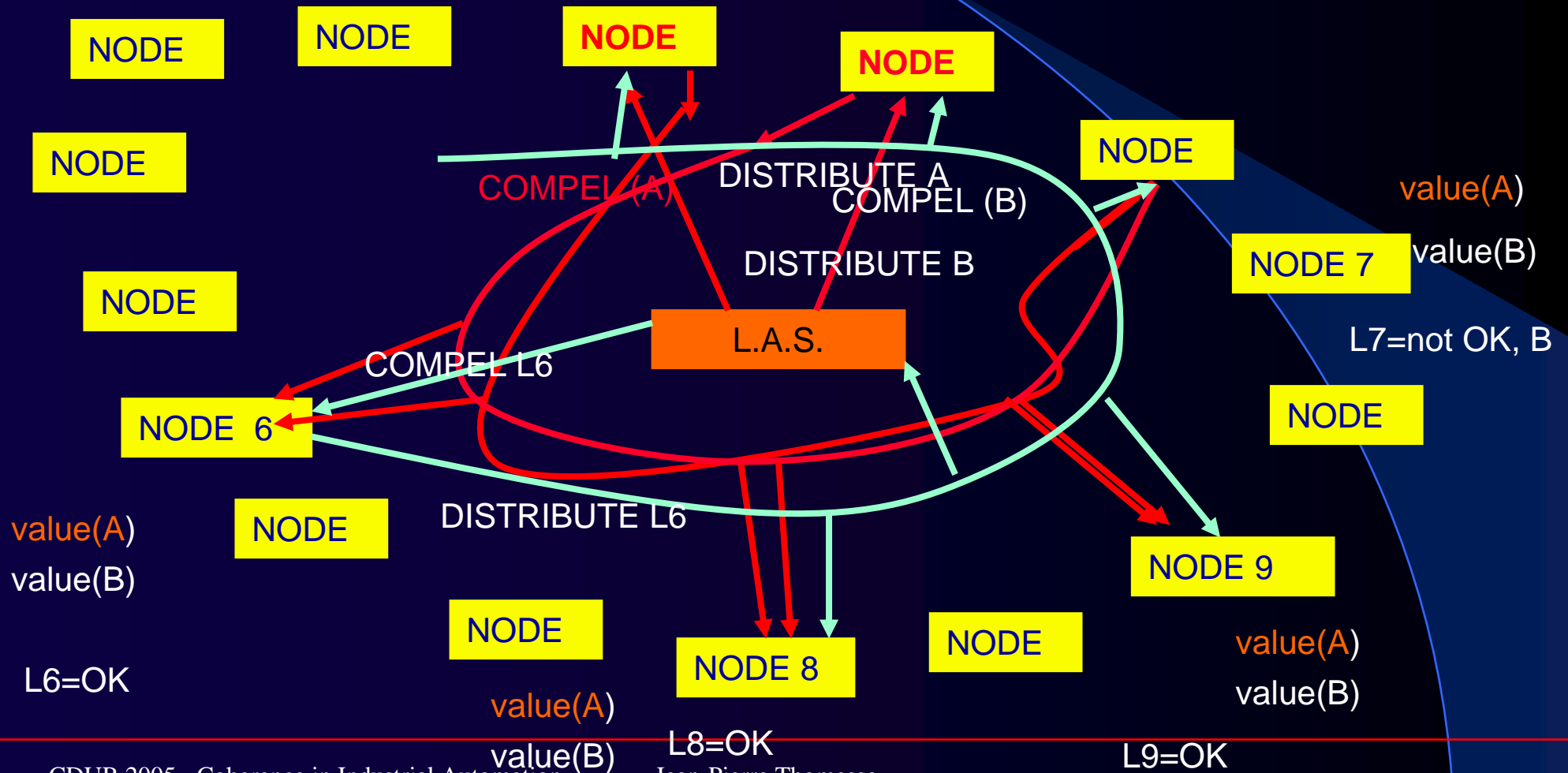
- coherences - needs analysis
- fieldbus requirements
- cooperation models - application relationships
- **protocols and mechanisms**
  - MAC and DLL
  - timeliness attributes
  - **data list coherence - space coherence**
- application architecture
- conclusion



# space - time consistency

- need “reliable multicasting”
- management of lists of variables (copies)
  - produced by different publishers
  - consumed by several subscribers
- verification and correction to obtain identical lists by the subscribers
- kind of global acknowledgement for different transmitters
- hypothesis:
  - two remote copies are considered identical if they are received without error and correct timeliness attributes

# space - time consistency





# content

- coherences - needs analysis
- fieldbus requirements
- cooperation models - application relationships
- protocols and mechanisms
  - MAC and DLL
  - timeliness attributes
  - data list coherence - space coherence
- **application architectures**
- conclusion

# application architectures

- functional architecture
  - functions, control flow, data flow
- physical - material architecture
  - processors and operating systems
  - networks and protocol stack
  - middleware
- operational architecture
  - mapping of functional architecture on physical architecture



# operational architecture design

- location assignment of functions to physical hosts
- implementation of
  - data flow
  - control flow
- on communication stacks

# operational architecture

- requirements

- cooperation models
- control flow
- data flow



- coherence requirements

- mechanisms

- application relationships

- coherence mechanisms in communication stack



# content

- coherences - needs analysis
- fieldbus requirements
- cooperation models - application relationships
- protocols and mechanisms
  - MAC and DLL
  - timeliness attributes
  - data list coherence - space coherence
- application architectures
- **conclusion**



# conclusion

- time and space coherences
- coherence of data, of actions, of events
- approximation of a “global common” state
- applications synchronized by the network
- challenge for the application distribution
- challenge for dependability

Reference: Fieldbus technology in industrial automation, Proc. of IEEE, June 2005



# IEC 61158

- 8 types for data link layer
  - Type 1 : proposed compromise (Technical Report)
  - Type 2 : ControlNet
  - Type 3 : Profibus
  - Type 4 : P-Net
  - Type 5 : Foundation fieldbus
  - Type 6 : Swiftnet
  - Type 7 : WorldFIP
  - Type 8 : Interbus



# IEC 61158

- 10 types for application layer
  - Type 1 : compromise (Technical Report)
  - Type 2 : ControlNet
  - Type 3 : Profibus
  - Type 4 : P-Net
  - Type 5 : Foundation fieldbus
  - Type 6 : Swiftnet
  - Type 7 : WorldFIP
  - Type 8 : Interbus
  - Type 9 : Foundation fieldbus H1
  - Type 10 : Profinet

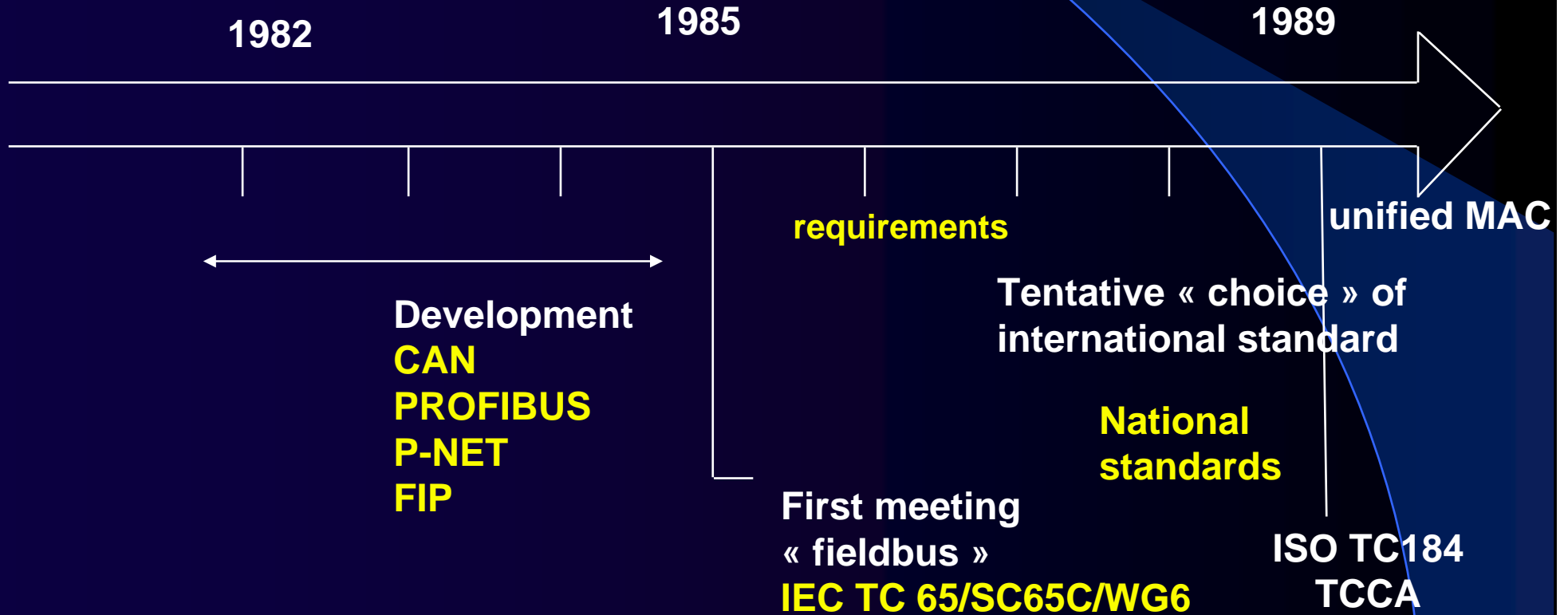


# IEC 61 784

- Communication Profile Family - CPF
- IEC 61 784-1
  - 16 profiles
- IEC 61 784 -2 under specification (based on Ethernet)
  - CPF 10: VNET/IP,
  - CPF 11: TCNet,
  - CPF 12: EtherCAT,
  - CPF 13: EtherNet PowerLink,
  - CPF 14: EPA : Ethernet for Plant Automation
  - CPF 15: ModBus RTPS : Real Time Publish - Subscribe

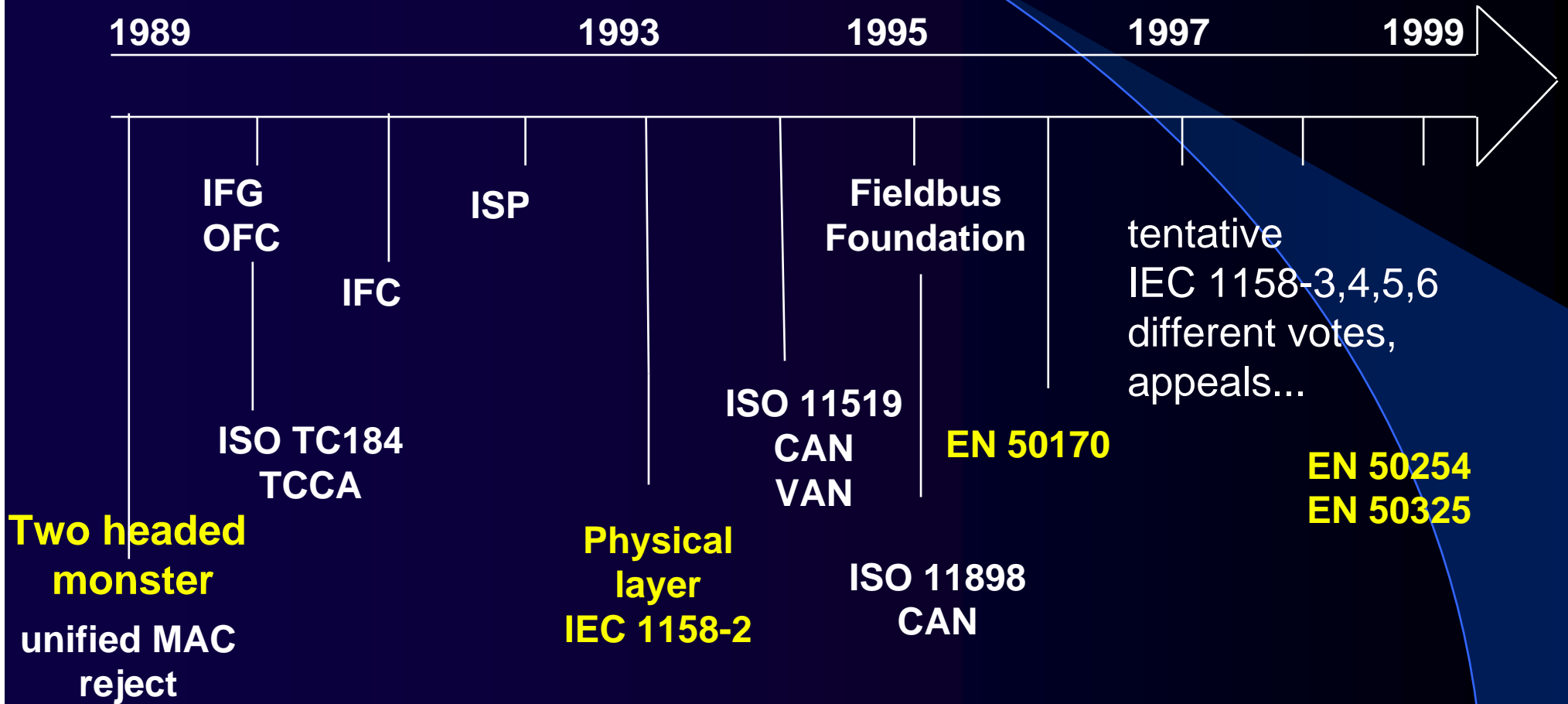


# history - 1





# history - 2





# history - 3

1999

2001

2004

2005

