

*Wakeup & Startup.*



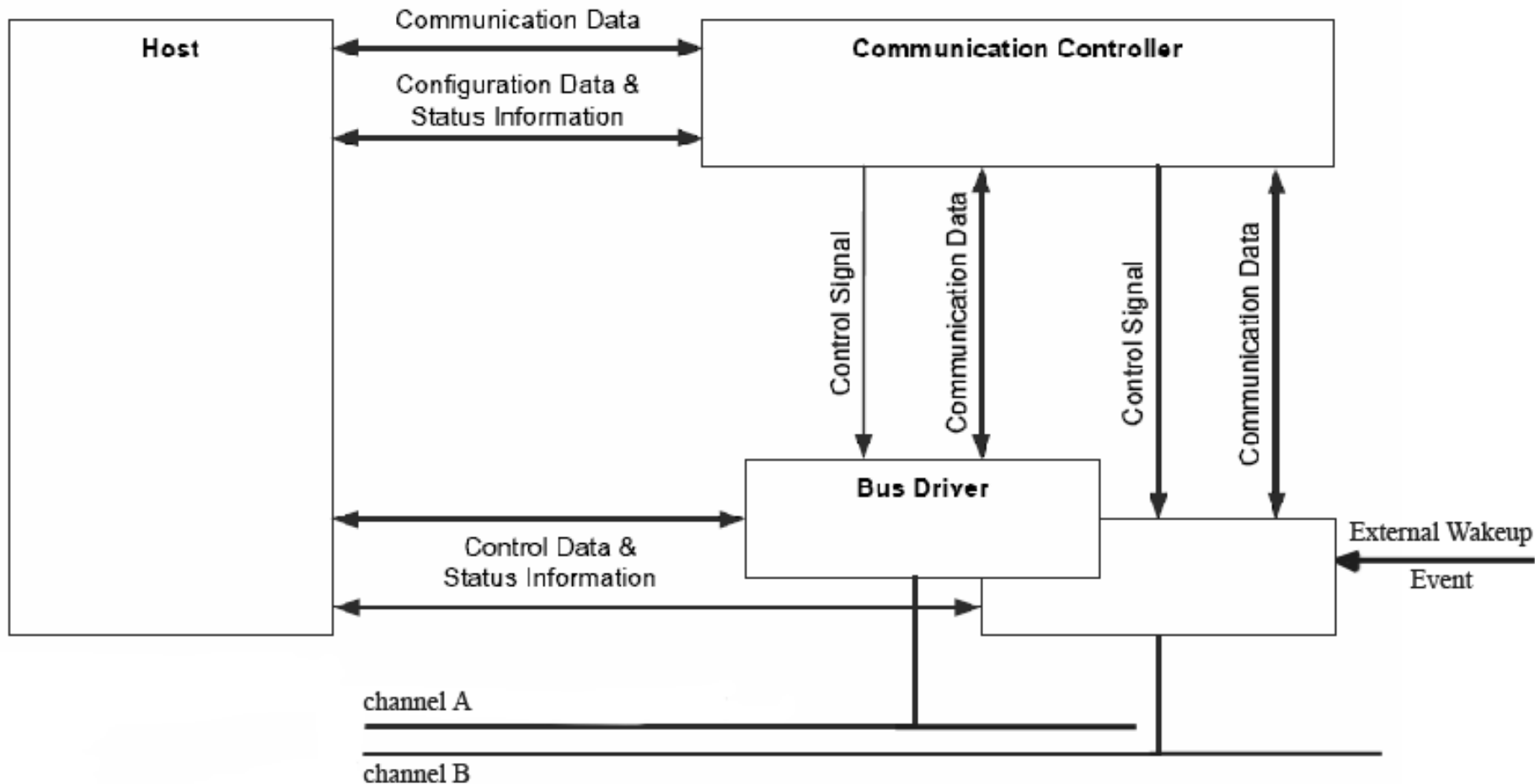
# Contents.

- General Notes
  - Node architecture
  - Wakeup & startup Diagram
- Wakeup
  - Wakeup state diagram
- Startup
  - Coldstart node status
  - Non-coldstart node status
  - Cluster startup
  - Theorem

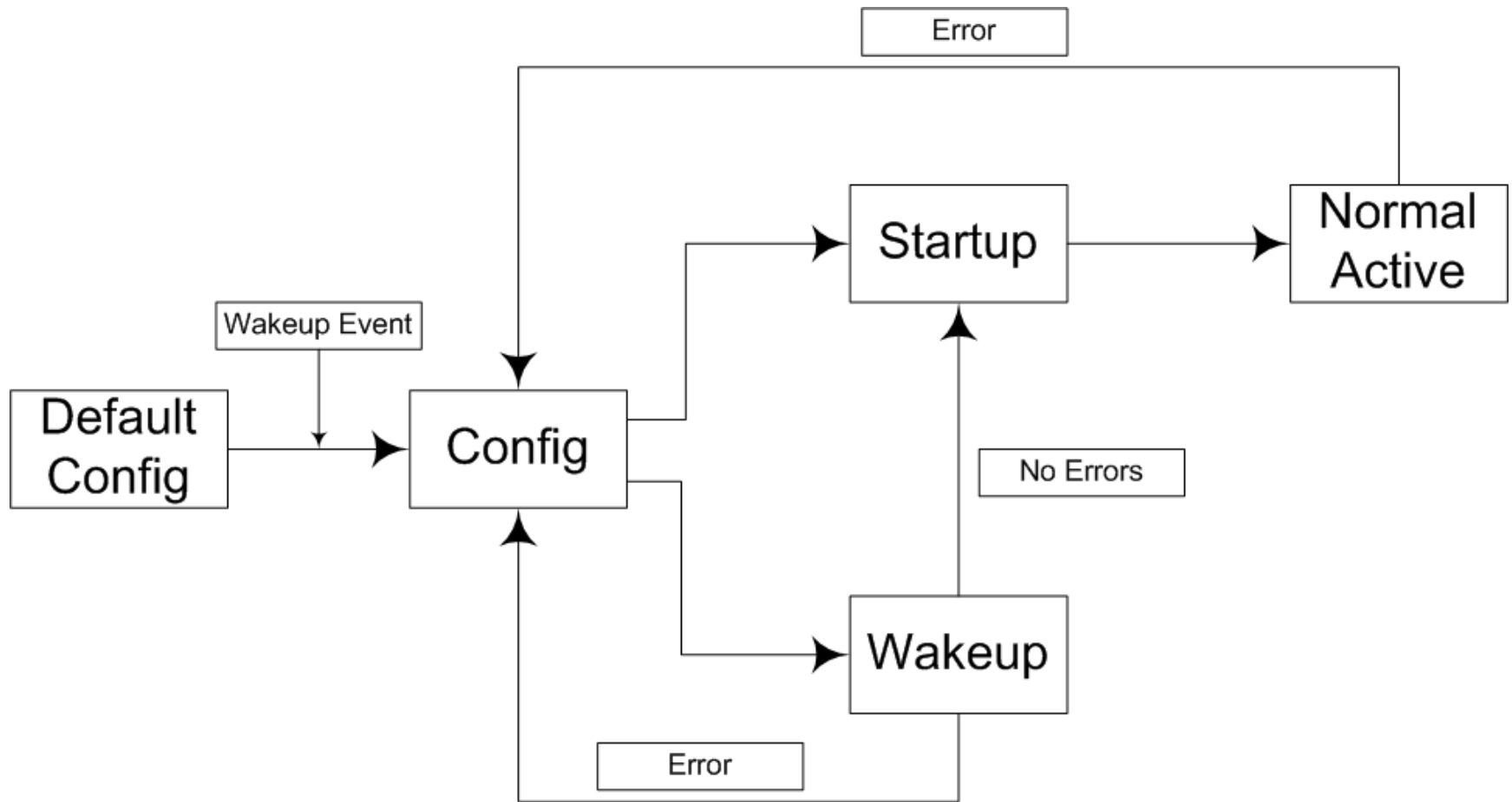
# General Notes.

- FlexRay uses TDMA (time-division multiple access)
- Clock in cluster must be synchronized.
- To bring cluster operational there are Wakeup and Startup states.

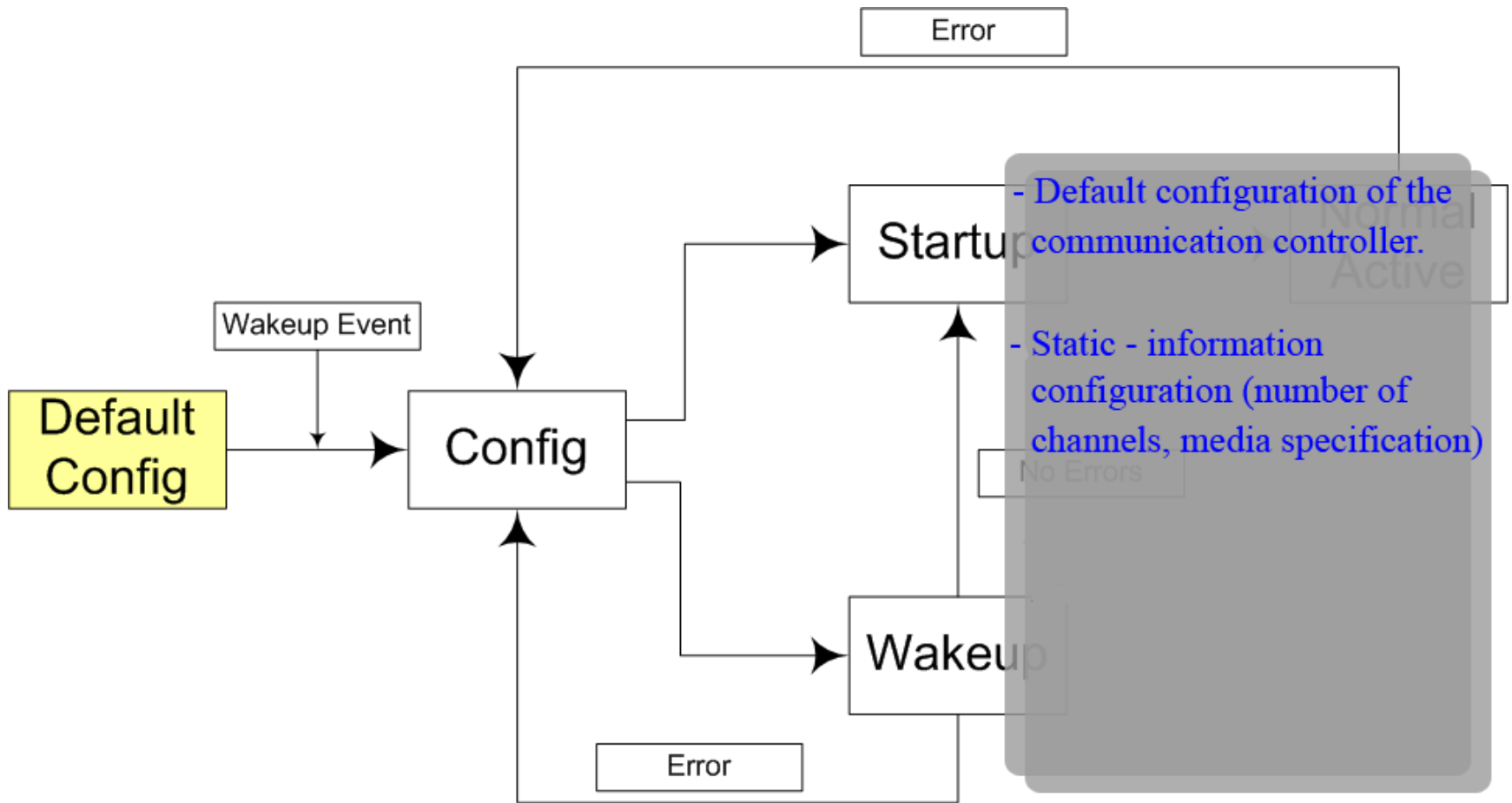
# Node Architecture.



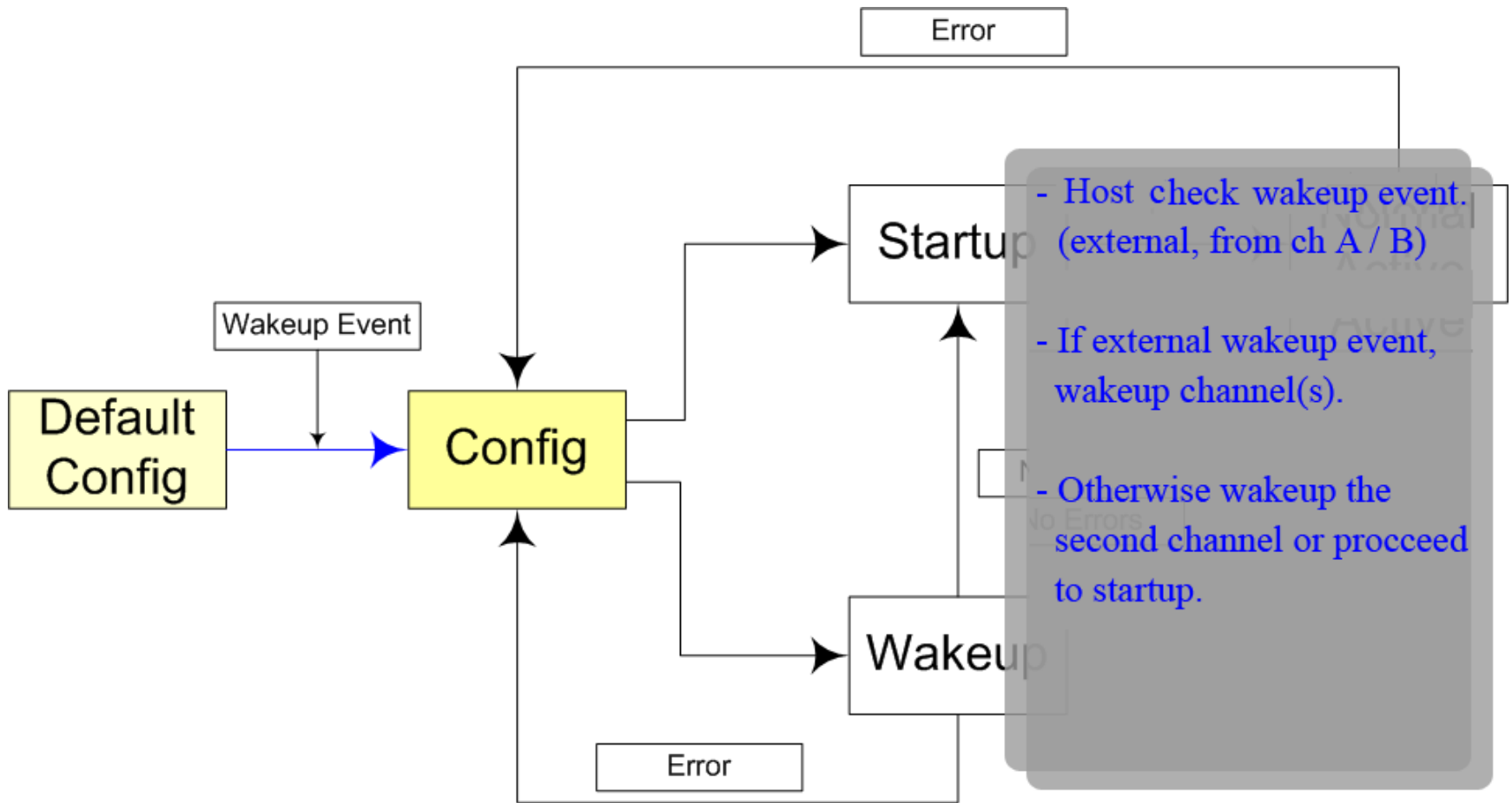
# Wakeup & Startup Diagram.



# Wakeup & Startup Diagram.

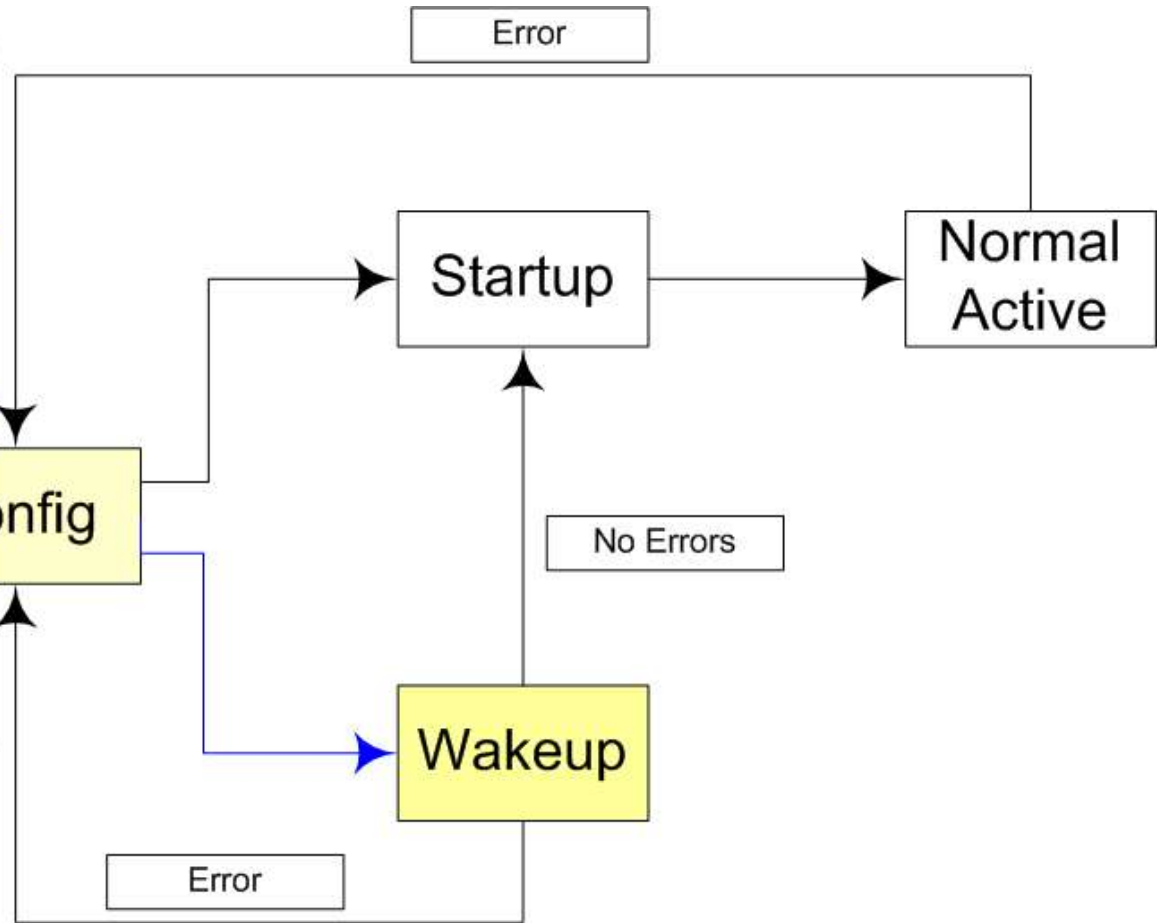


# Wakeup & Startup Diagram.



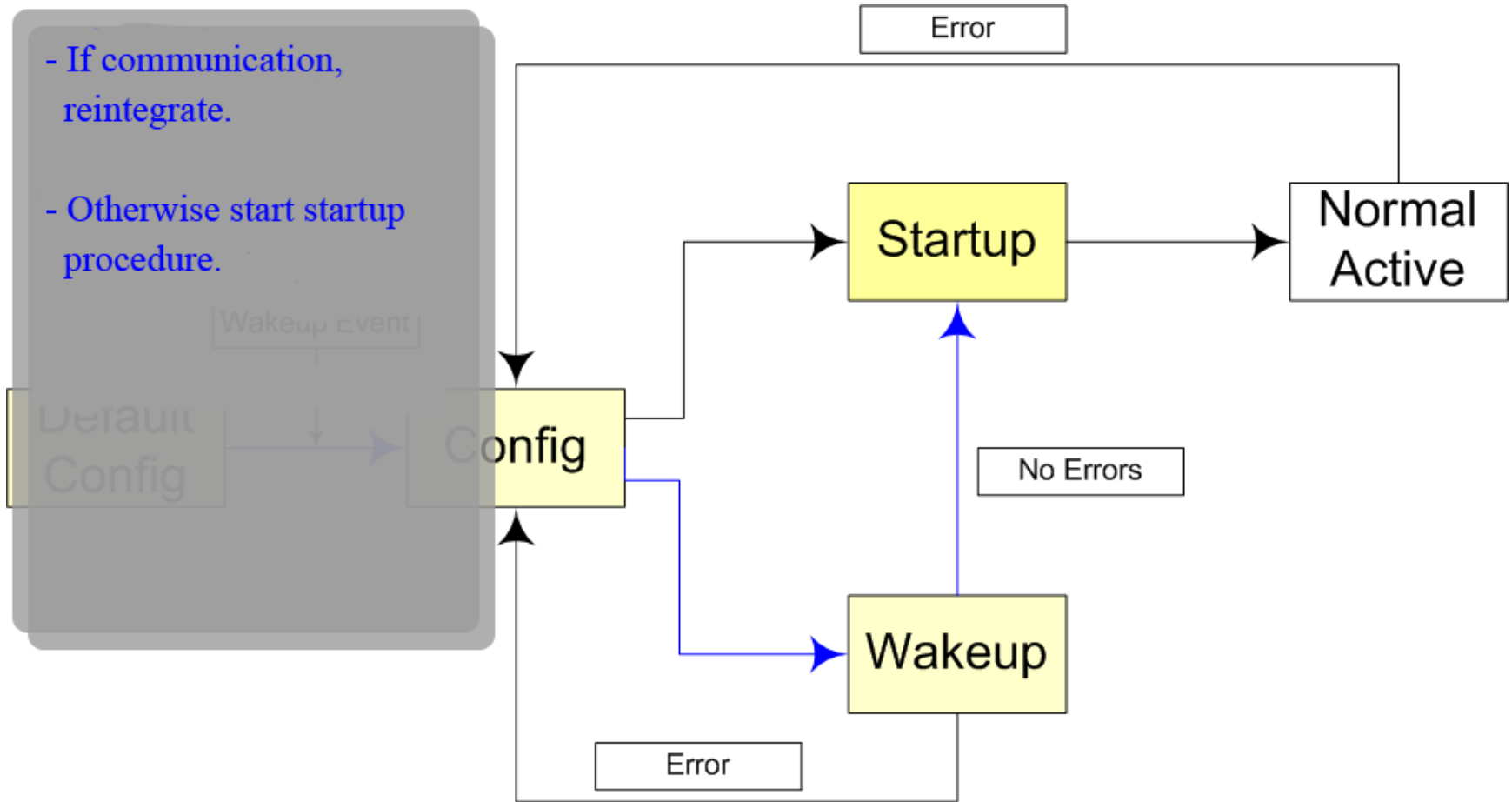
# Wakeup & Startup Diagram.

- Wakeup channel.
- If no errors, assume that channel is awake and proceed to startup.
- If error proceed to config and apply specific defined error-recovery procedure.





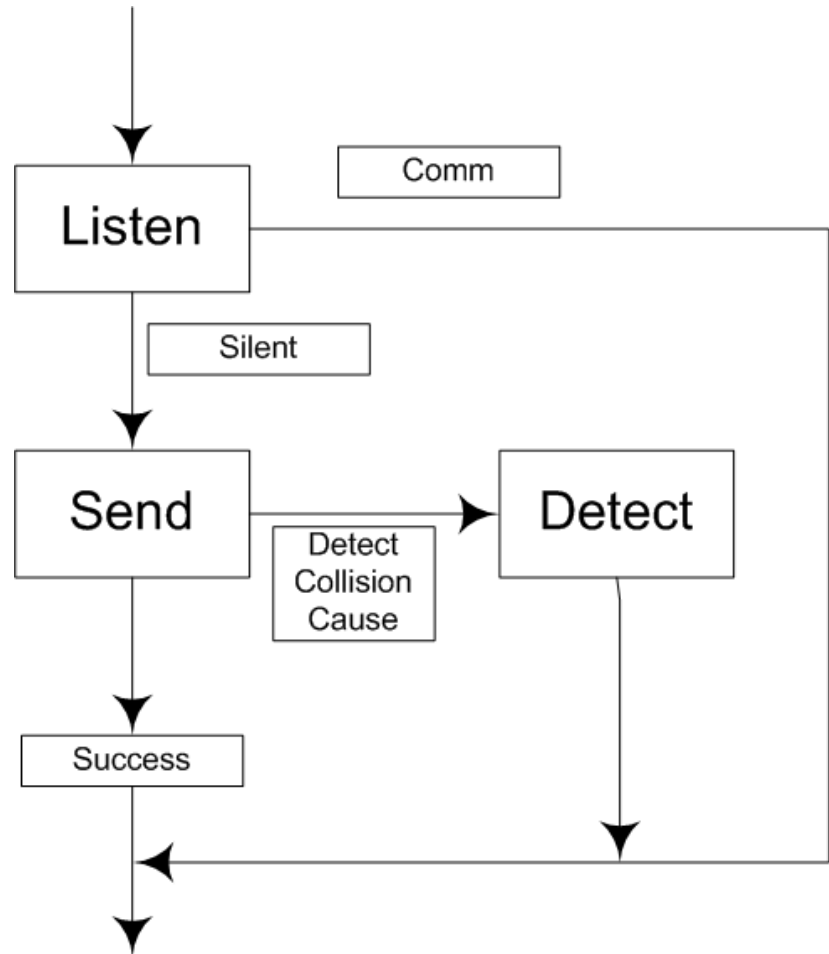
# Wakeup & Startup Diagram.



# Wakeup.

- Supports any number of nodes that wakeup simultaneously.
- At least one node in the cluster needs an external wakeup source.
- Node has the ability to transmit a special *wakeup pattern*.
- Wakeup pattern is collision resilient.
- Receiving nodes recognizes the wakeup pattern and triggers the node wakeup if it is still asleep.
- The node cannot check whether nodes are awake after the transmission of the wakeup pattern.

# Wakeup State Diagram.



# Wakeup State Diagram.

- Inhibit the transmission of the wakeup pattern if there is communication.
- When ongoing communication is detected the wakeup attempt is aborted.

## UNDEFINED

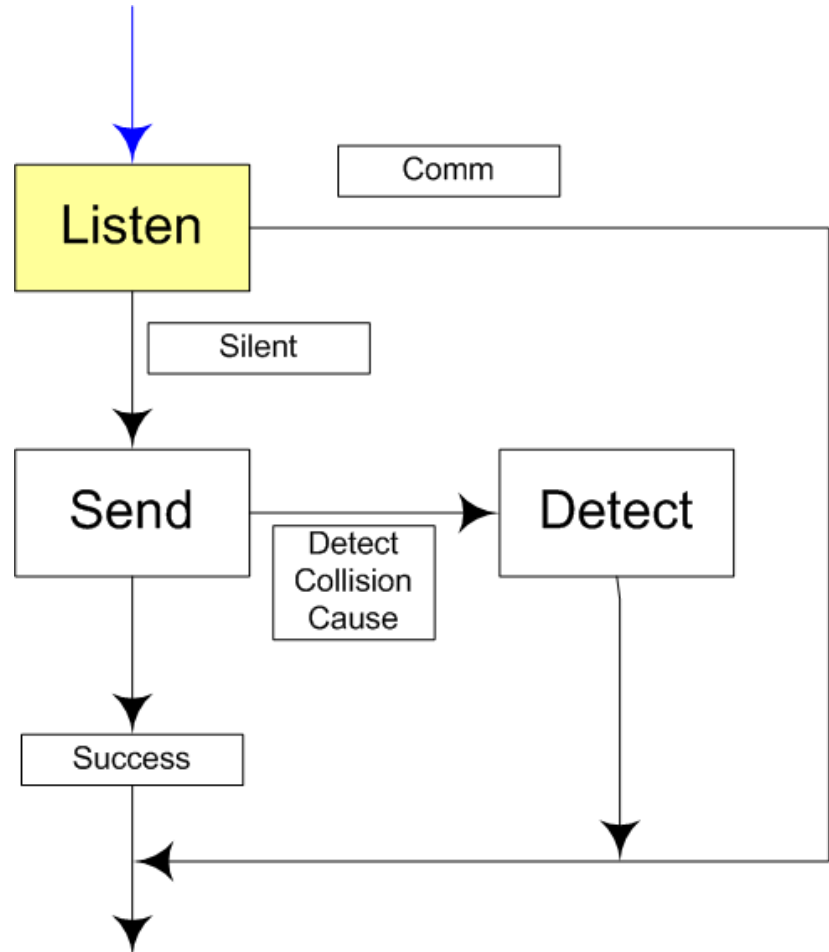
Wakeup mechanism was not been executed yet.

## RECEIVED HEADER

Communication detected.

## RECEIVED WUP (WakeUp Pattern)

Wakeup process is already in progress.

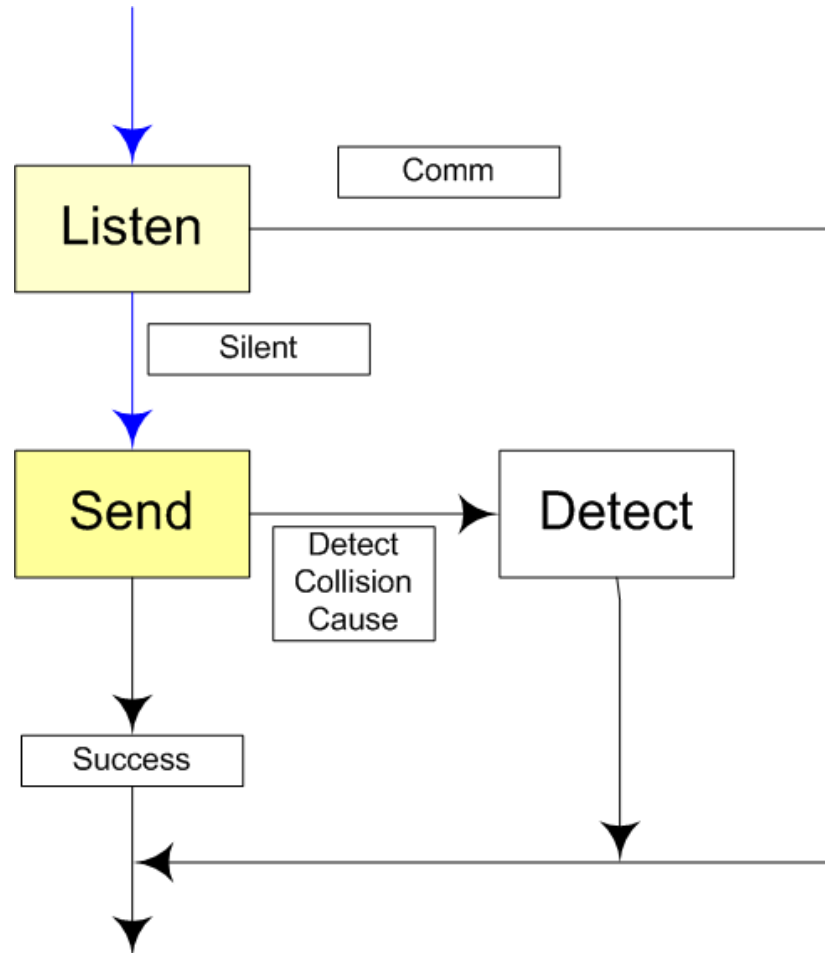


# Wakeup State Diagram.

- The node transmits the wakeup pattern on the configured channel and checks for collision.
- If activity is detected during of idle portions, communication controller enters a monitoring phase, to discover the cause of the collision.

## TRANSMITTED

Wakeup pattern was completely transmitted.



# Wakeup State Diagram.

- The communication controller attempts to discover the reason for the wakeup collision encountered in the previous state.
- Listening to channel during specified time.

## COLLISION HEADER

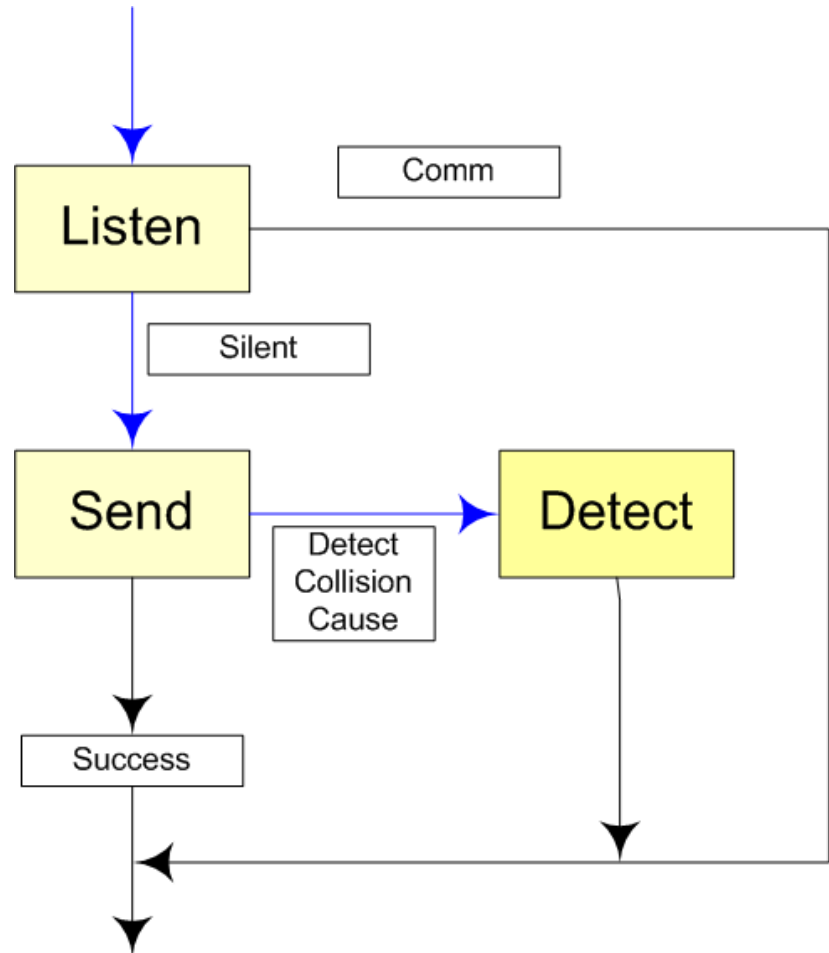
A collision detected by receiving a valid header.

## COLLISION WUP

A collision detected by receiving a valid wakeup pattern.

## COLLISION UNKNOWN

A collision without succeeding valid reception event detected.



# Startup.

- *Sync Frame's* header segment contains an indicator that the deviation measured between the is frame's arrival time and its expected arrival time should be used by the clock synchronization algorithm.
- *Startup frame's* header segment contains an indicator that integrating nodes may use timerelated information from this frame for initialization during the startup process.
- Cluster consists of coldstart and non-coldstart nodes.
- *Coldstart node* is a node capable of initiating the communication startup procedure on the cluster by sending startup frames.
- A non-coldstart node requires at least two startup frames from distinct nodes for integration.

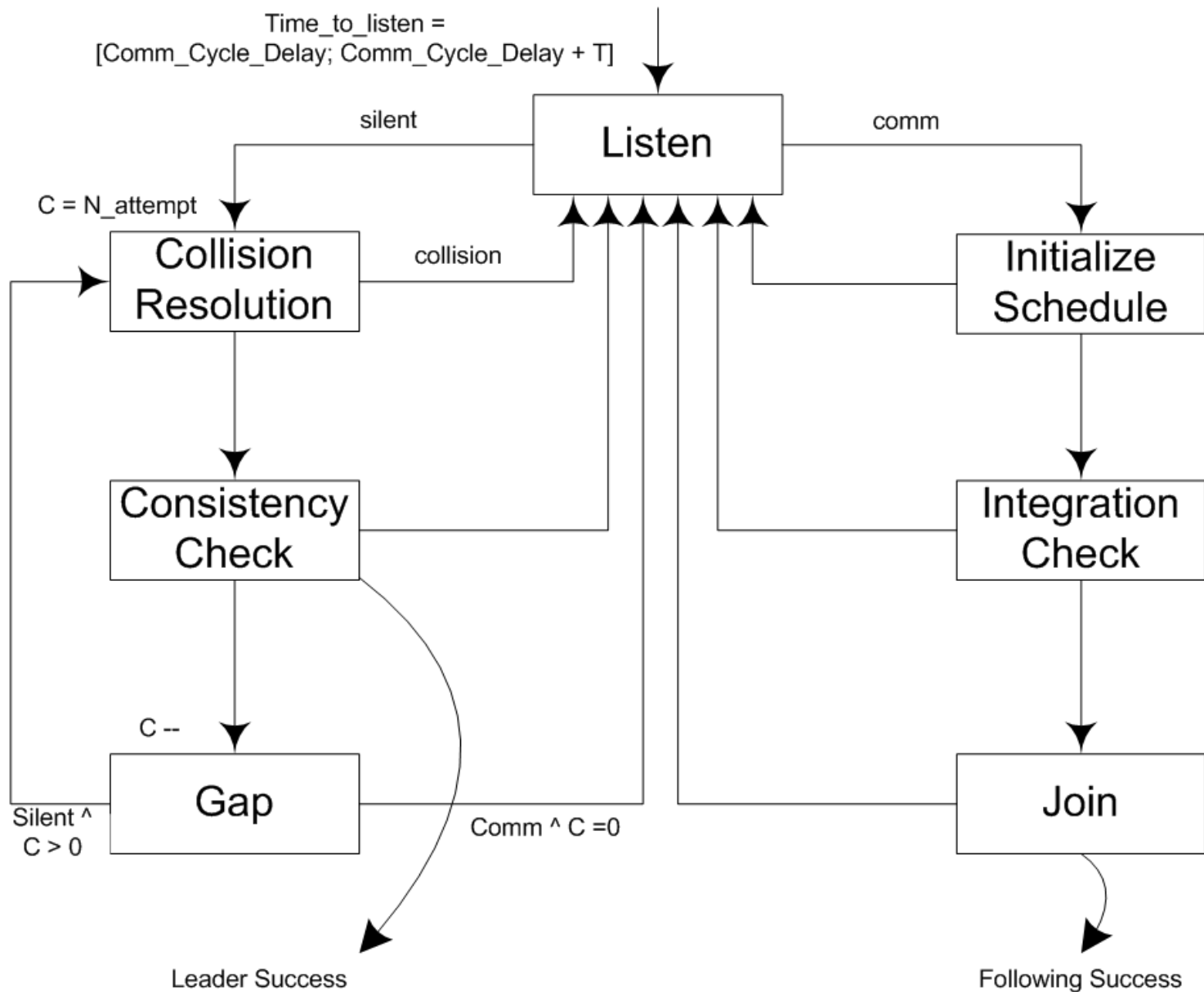
# Startup.

- *Sync Frame*'s header segment contains an indicator that the deviation measured between the is frame's arrival time and its expected arrival time should be used by the clock synchronization algorithm.
- *Startup frame*'s header segment contains an indicator that integrating nodes may use timerelated information from this frame for initialization during the startup process.
- Cluster consists of coldstart and non-coldstart nodes.
- *Coldstart node* is a node capable of initiating the communication startup procedure on the cluster by sending startup frames.
- A non-coldstart node requires at least two startup frames from distinct nodes for integration.
- 

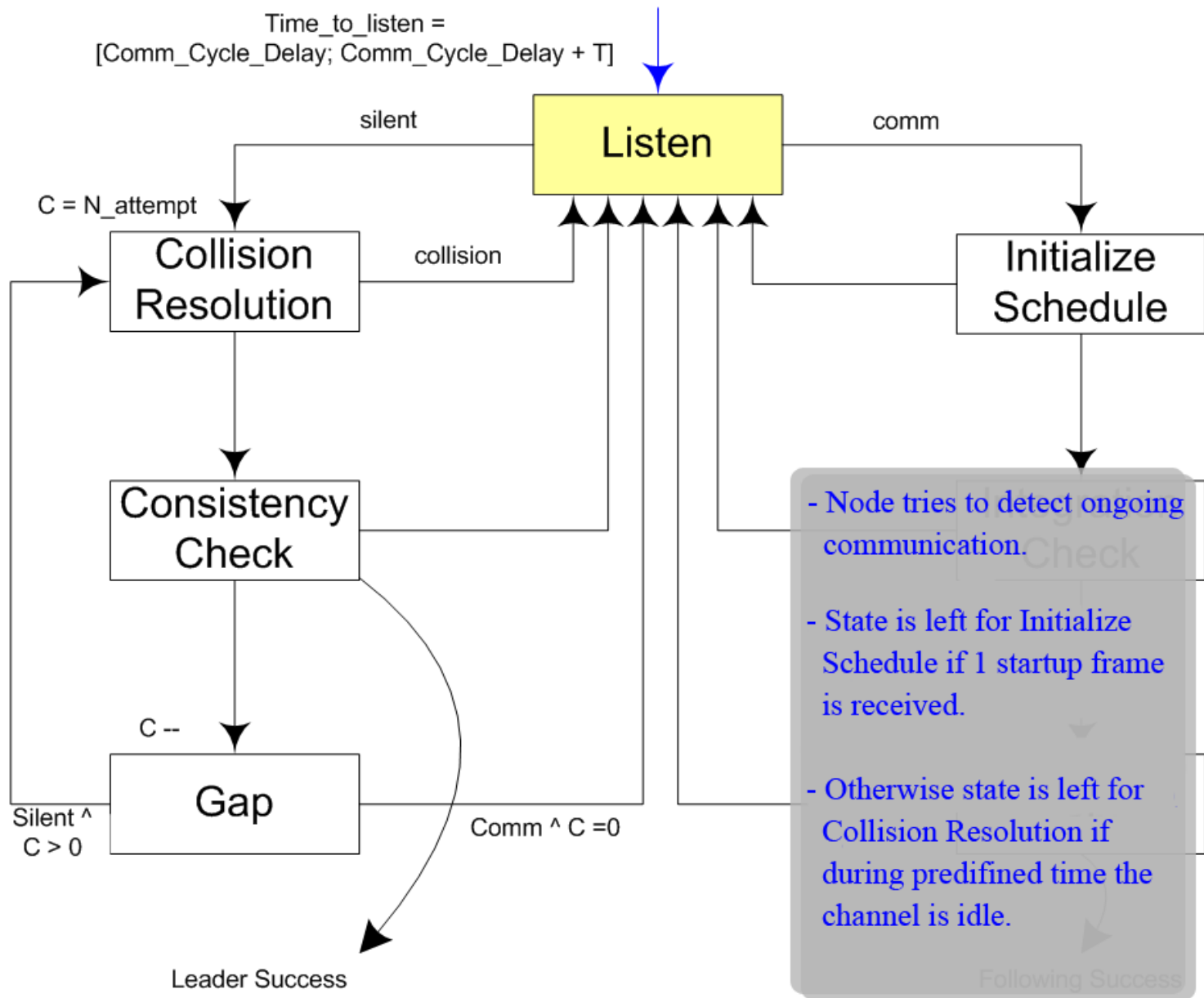




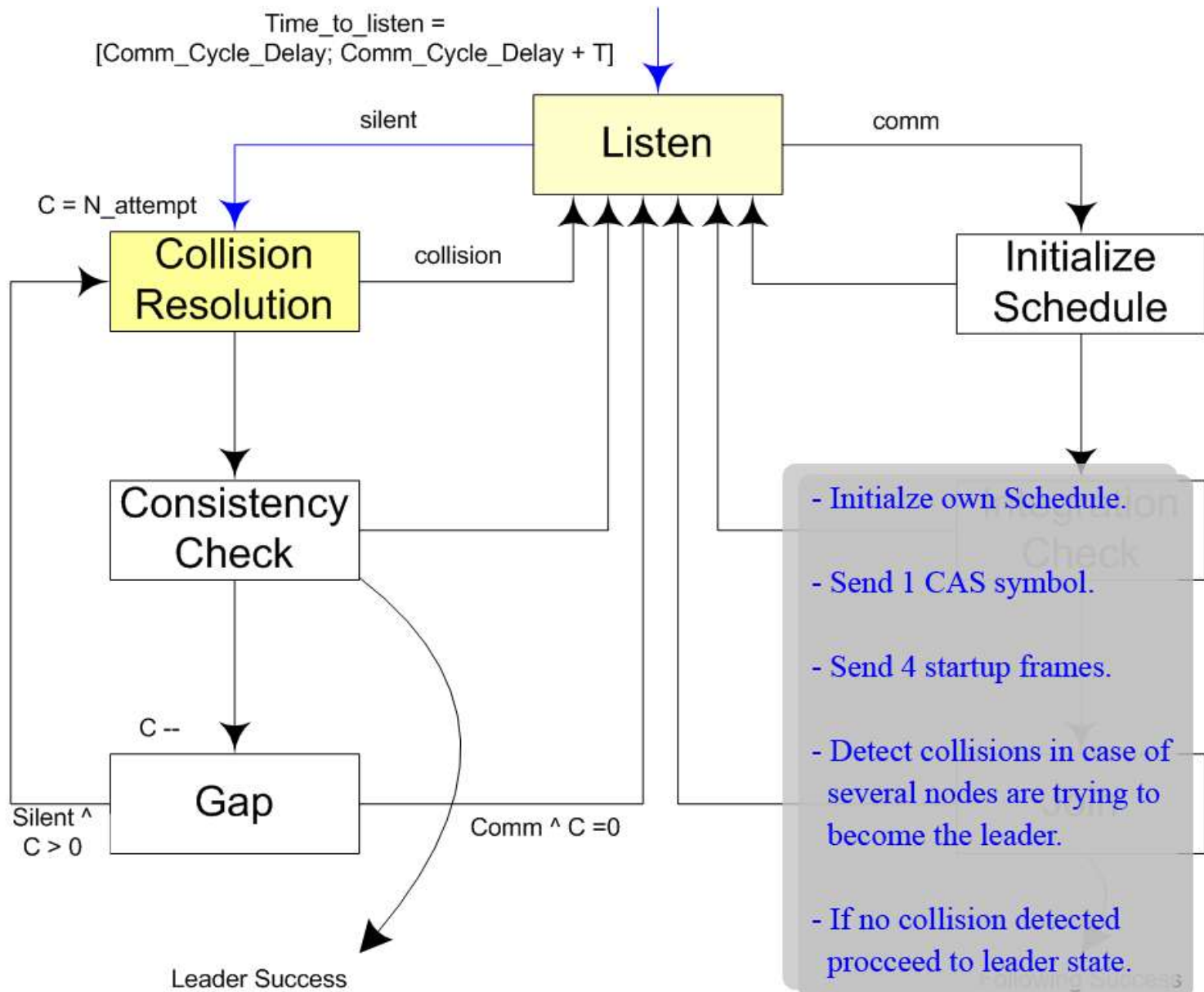
# Coldstart Nodes Status.



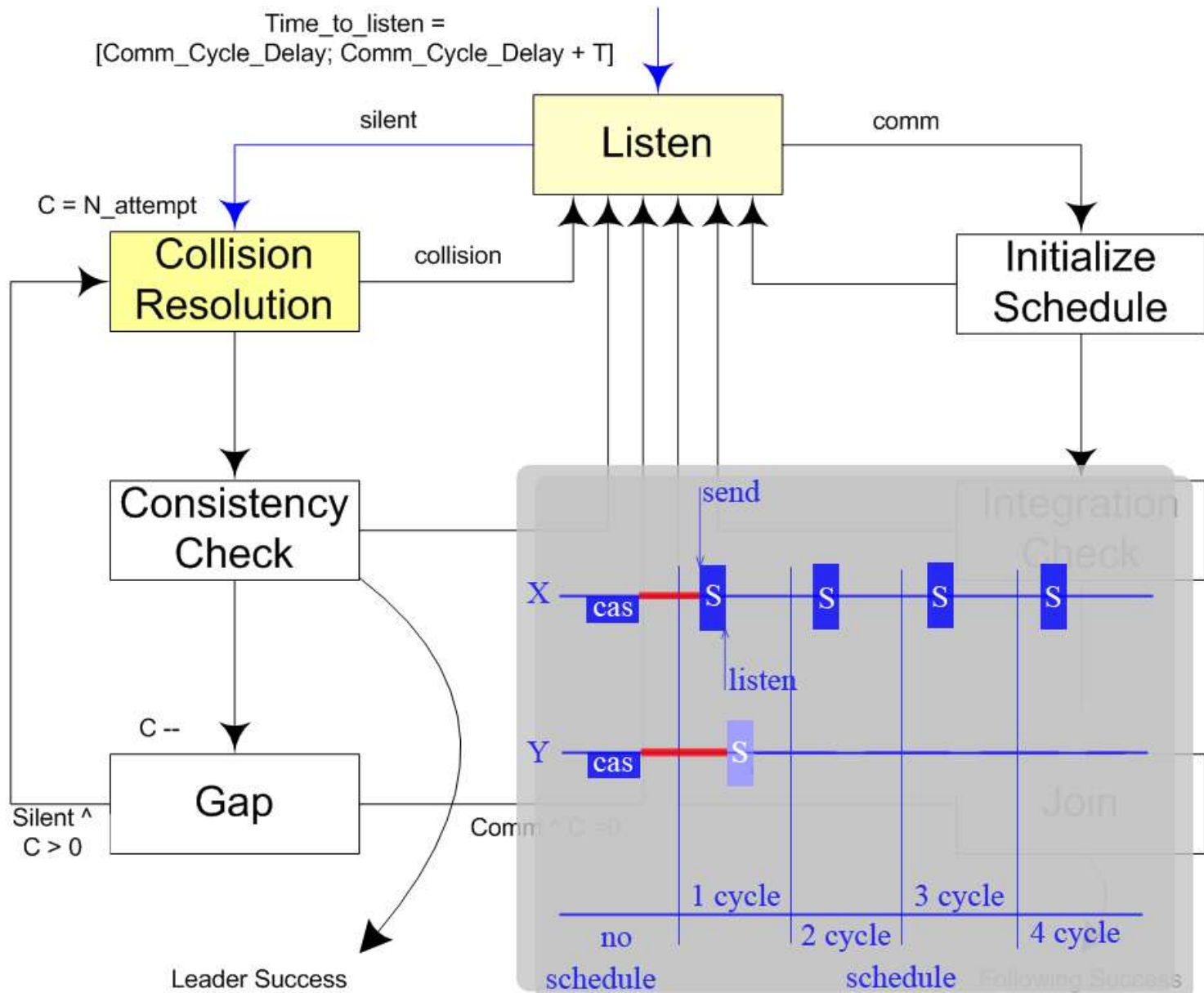
# Coldstart Nodes Status.



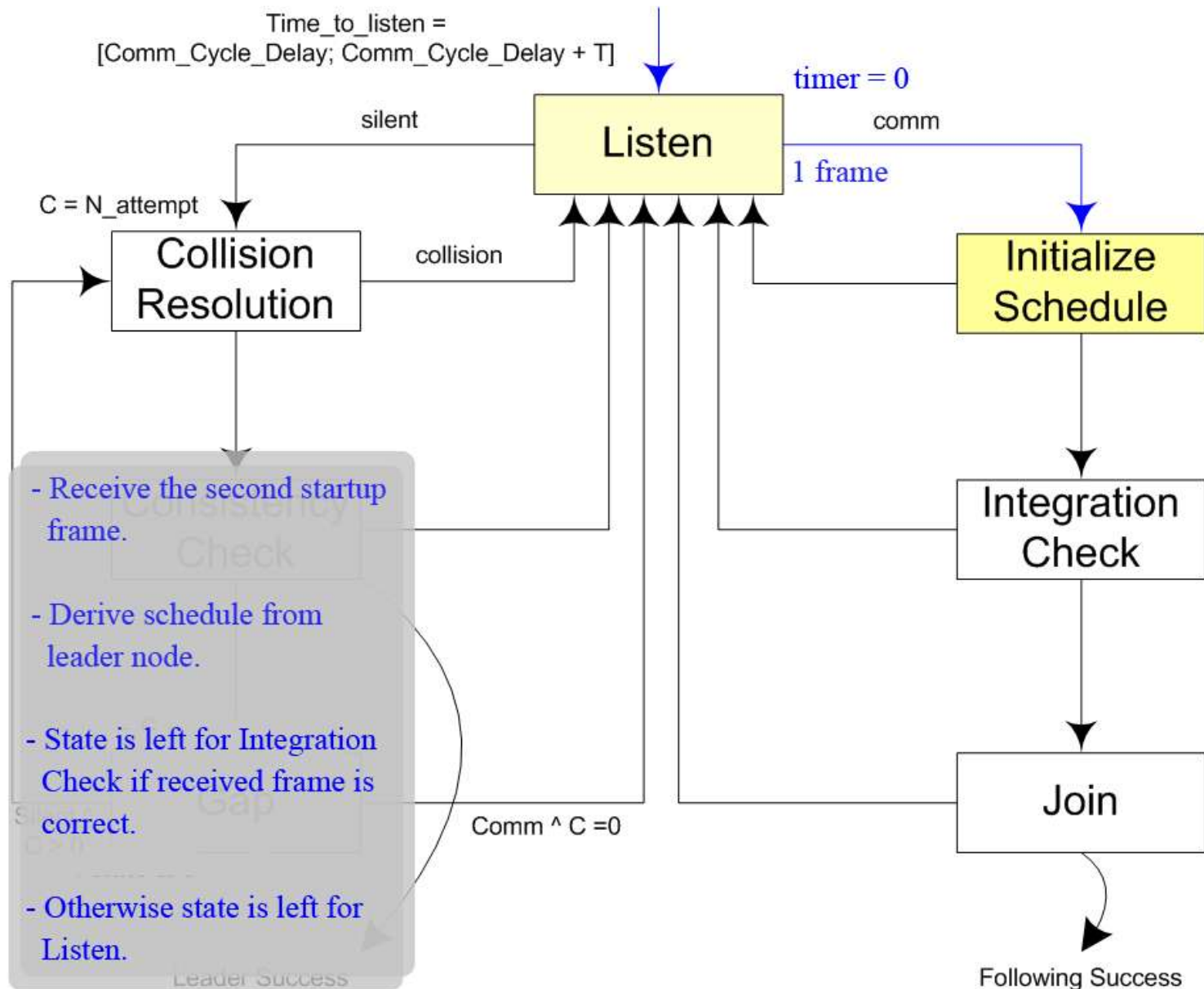
# Coldstart Nodes Status.



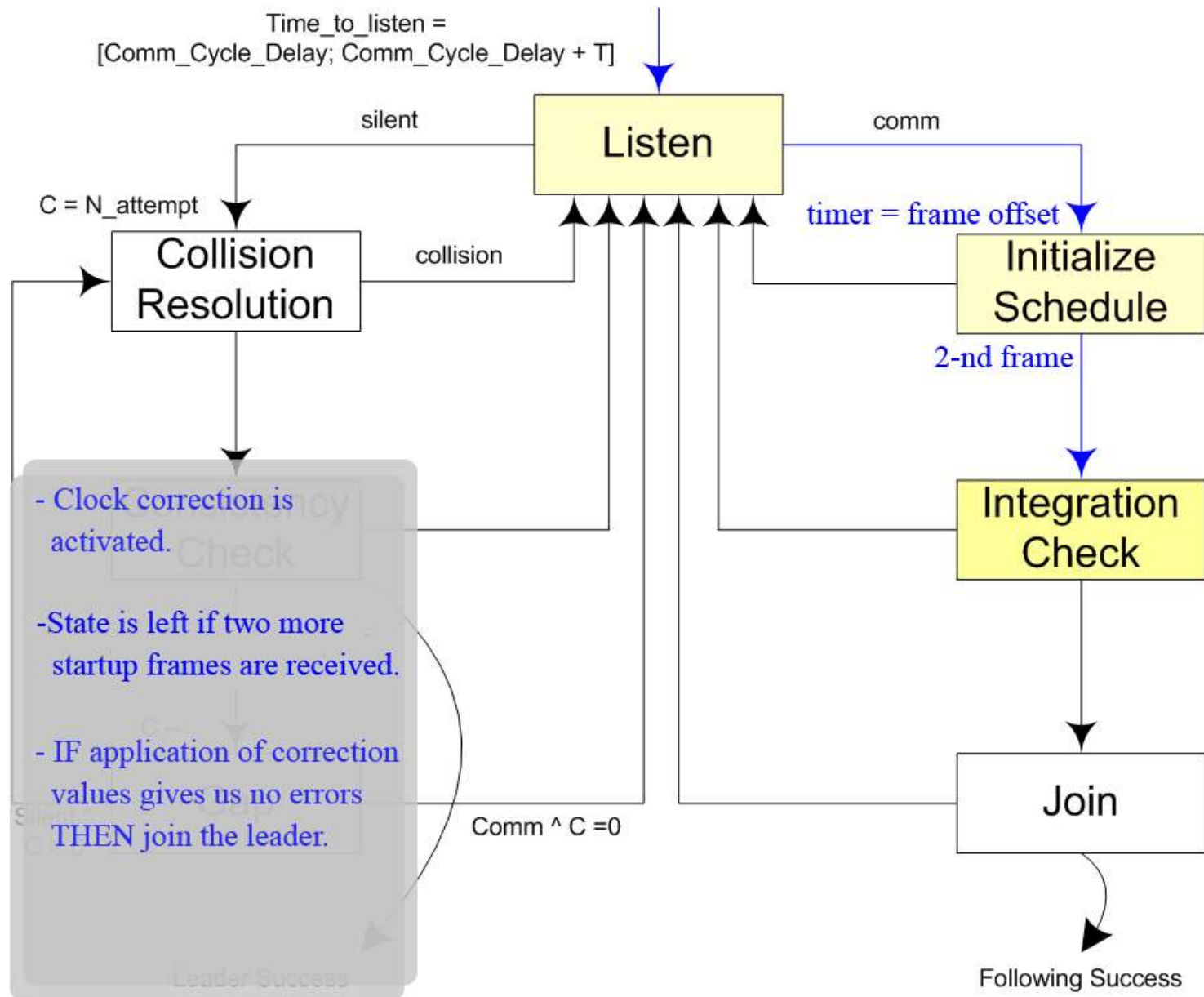
# Coldstart Nodes Status.



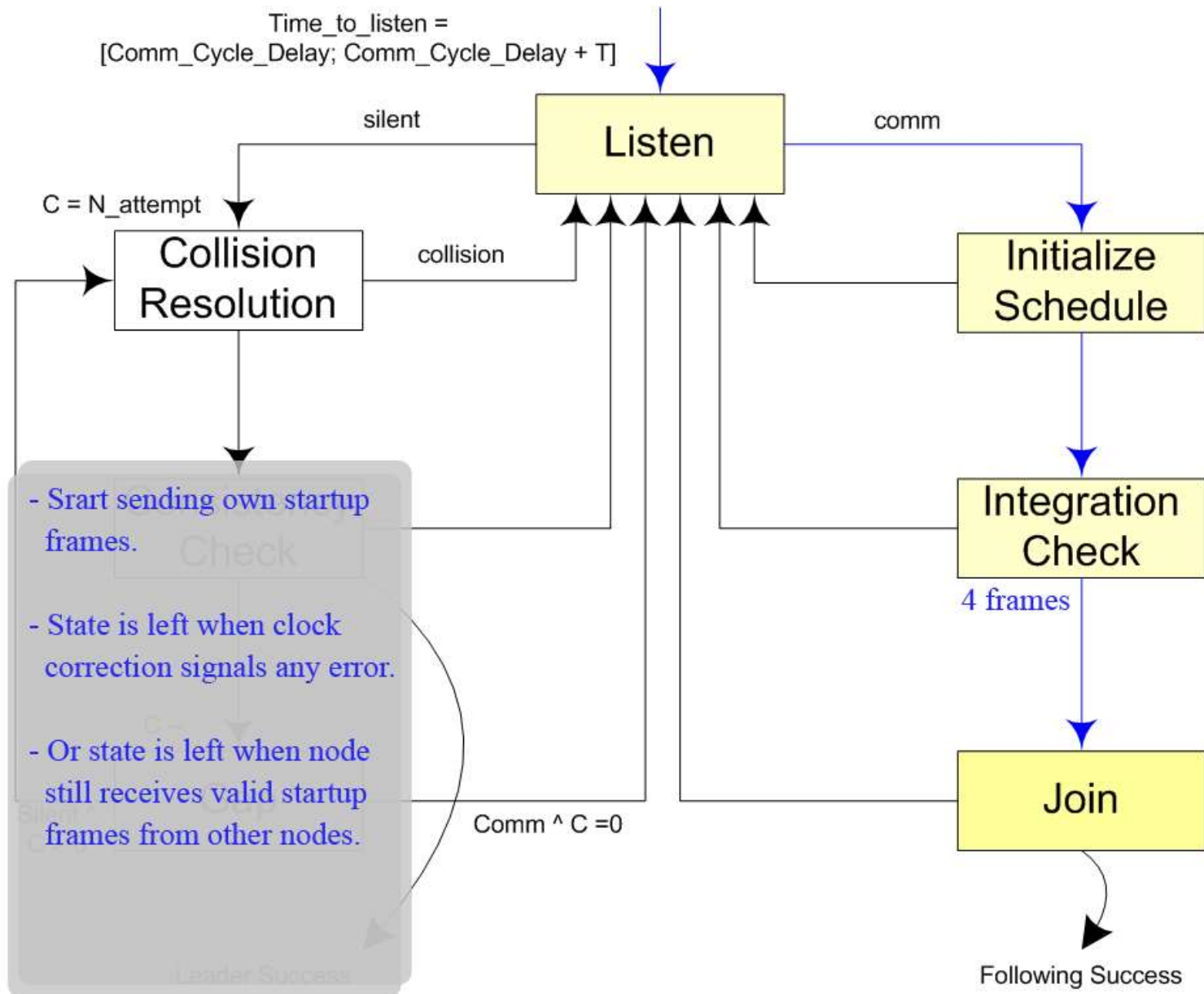
# Coldstart Nodes Status.



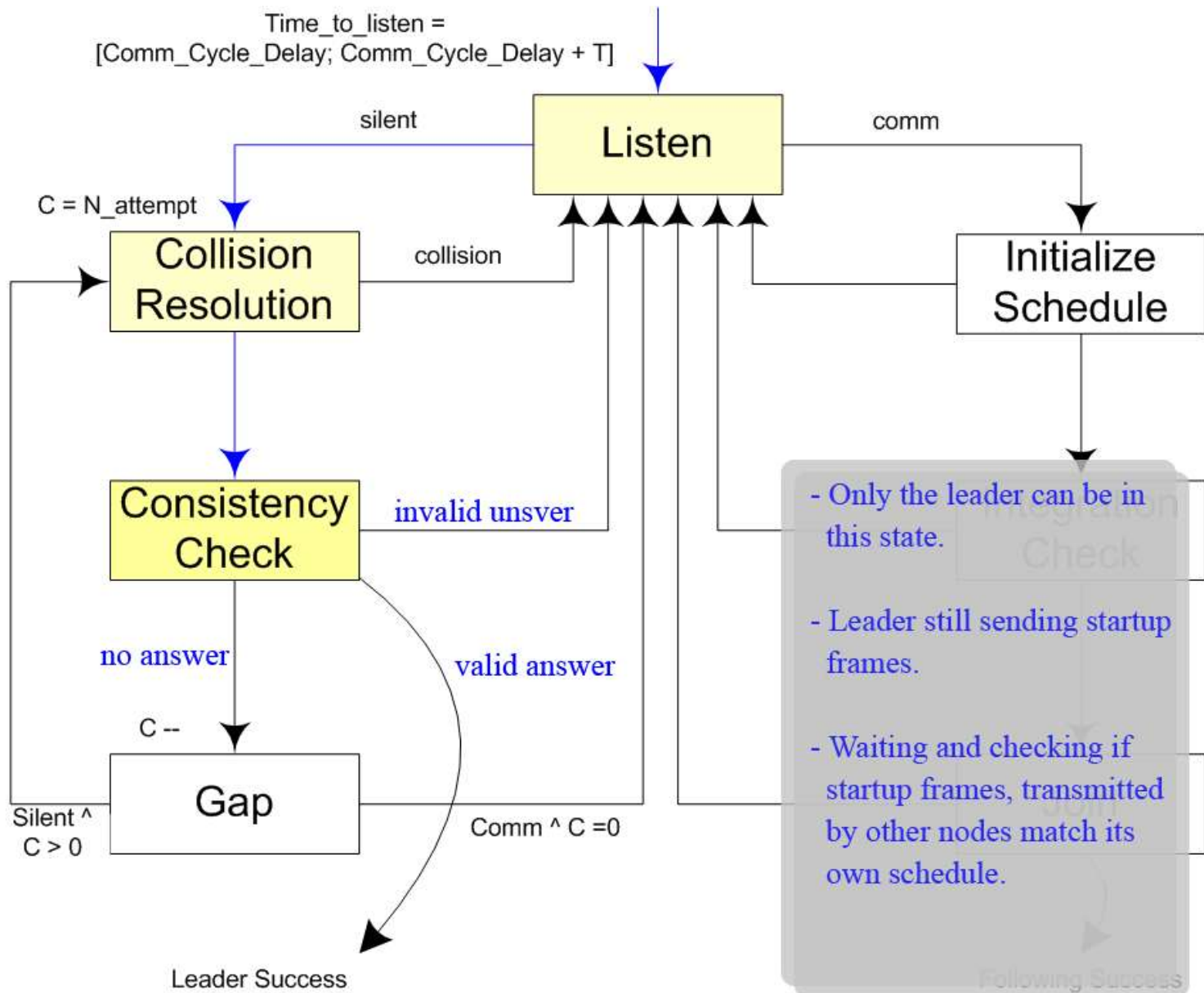
# Coldstart Nodes Status.



# Coldstart Nodes Status.

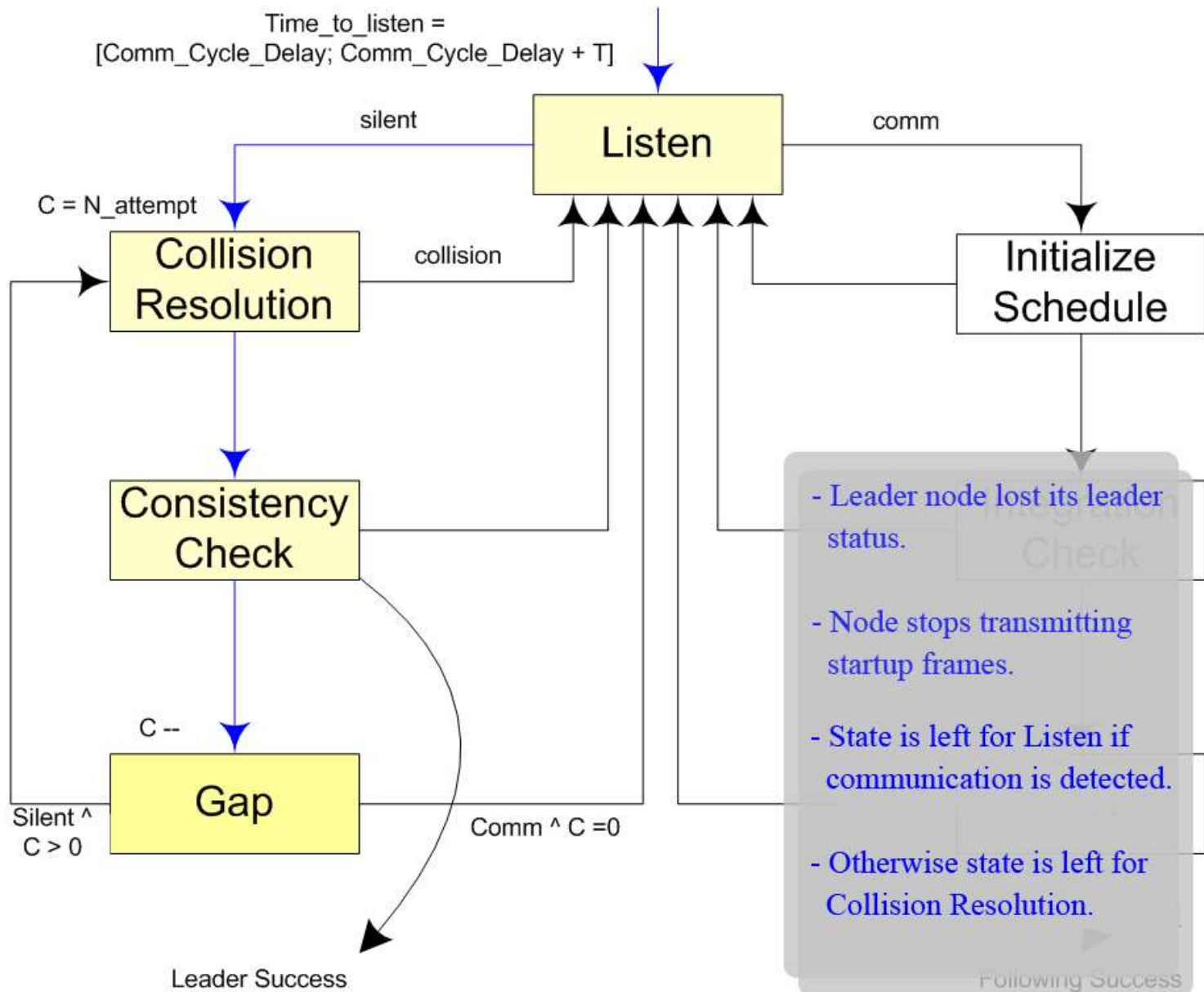


# Coldstart Nodes Status.





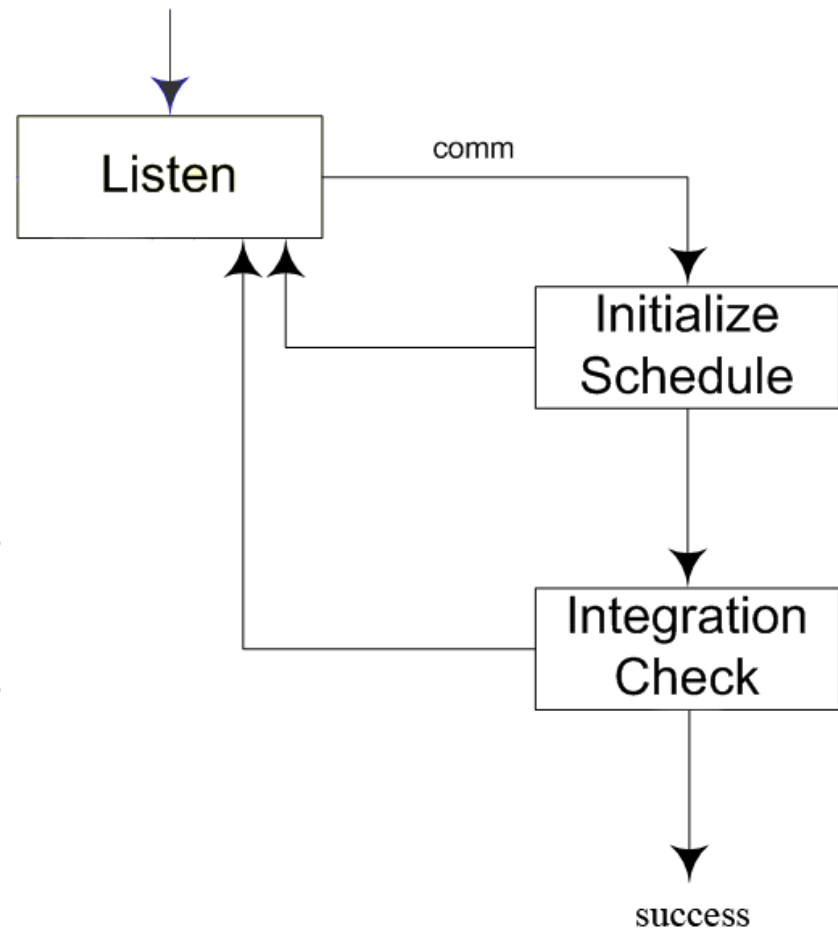
# Coldstart Nodes Status.



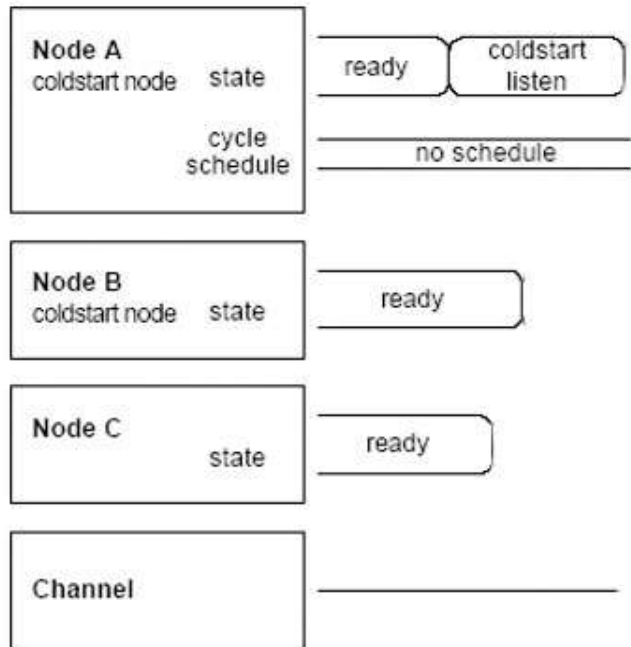
# Non - coldstart Nodes Status.

A non-coldstart node requires  
at least two startup frames  
from distinct nodes for integration.

Non - coldstart nodes don't  
send startup frames during Startup.

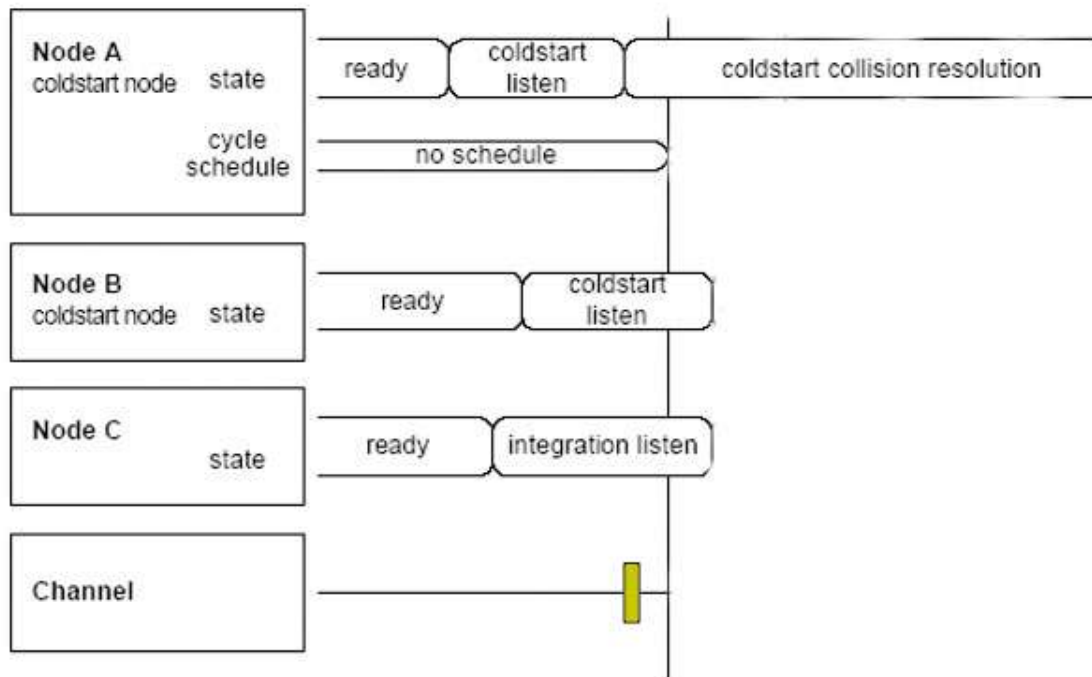


# Cluster Startup.



 : CAS symbol       : startup frame of node A       : startup frame of node B       : frame of node C

# Cluster Startup.



: CAS symbol



: startup frame of node A

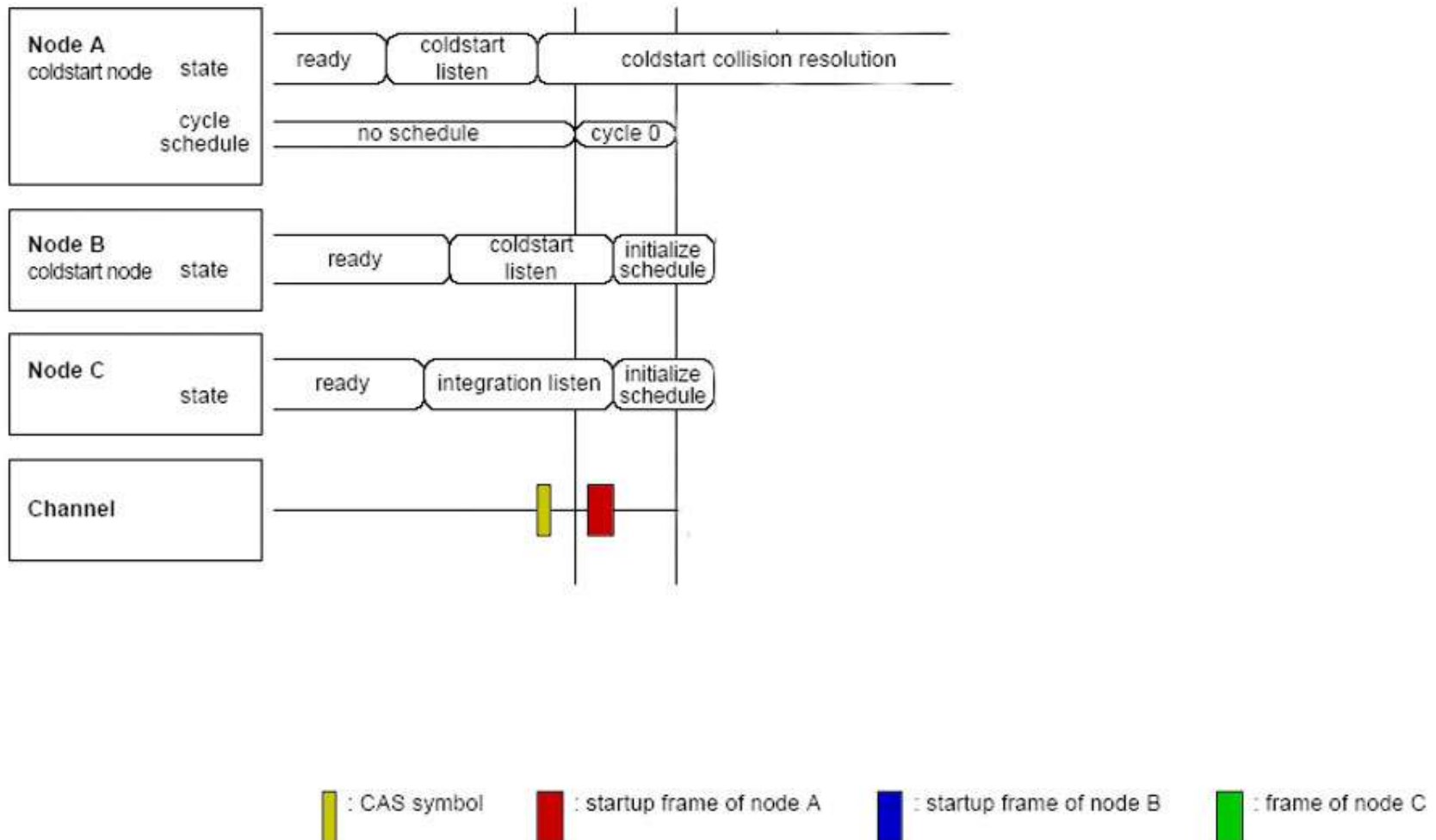


: startup frame of node B

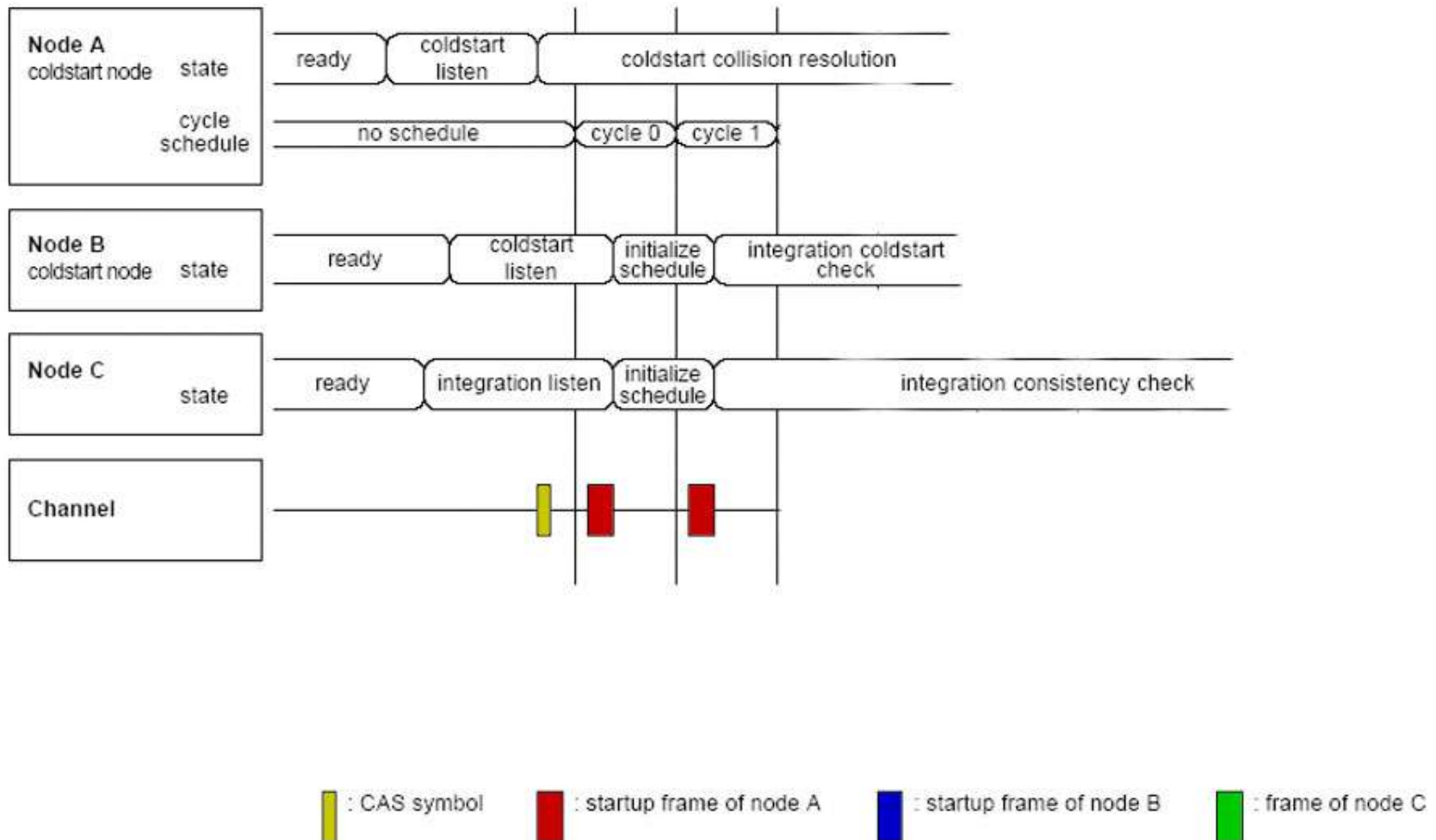


: frame of node C

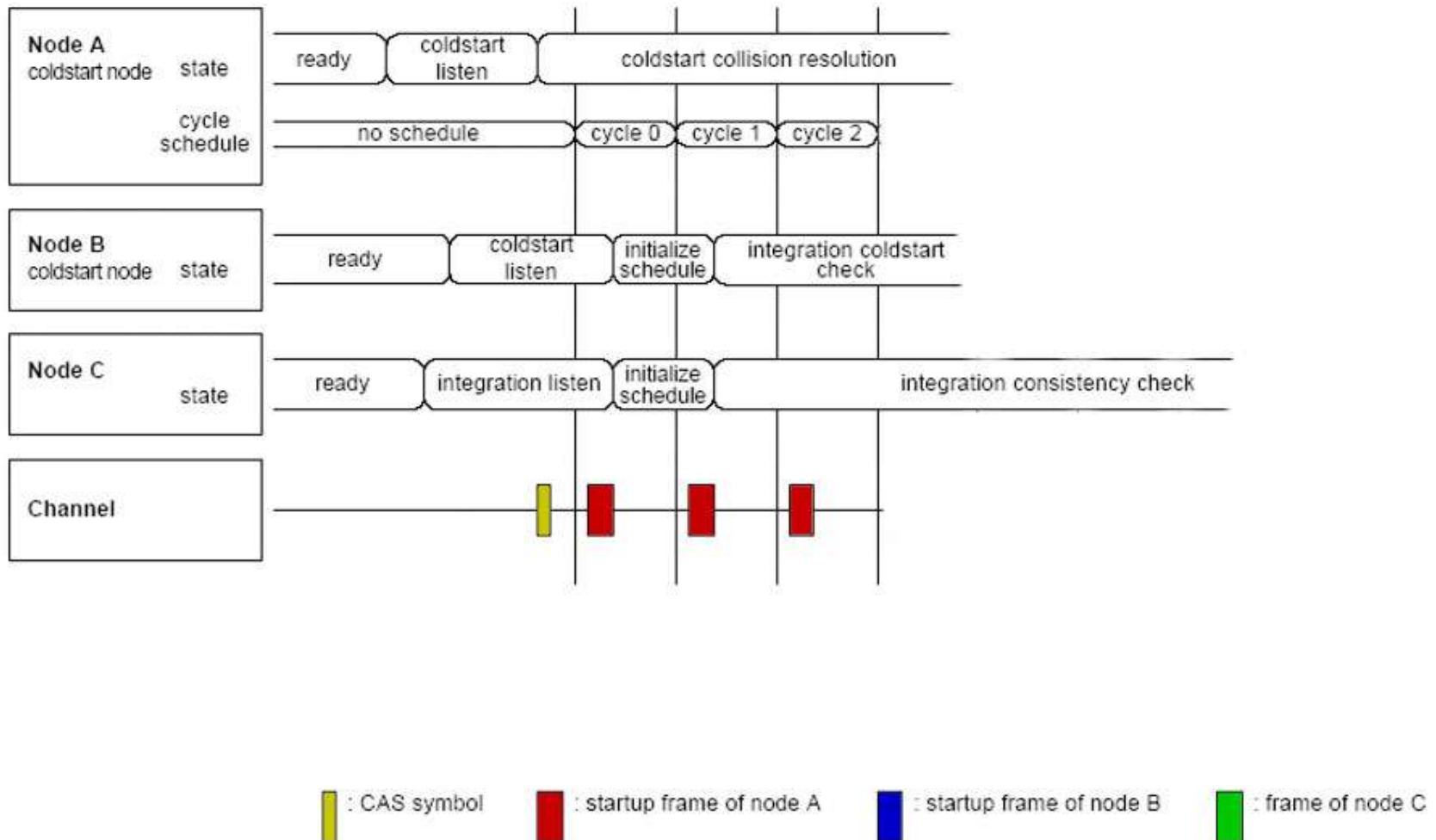
# Cluster Startup.



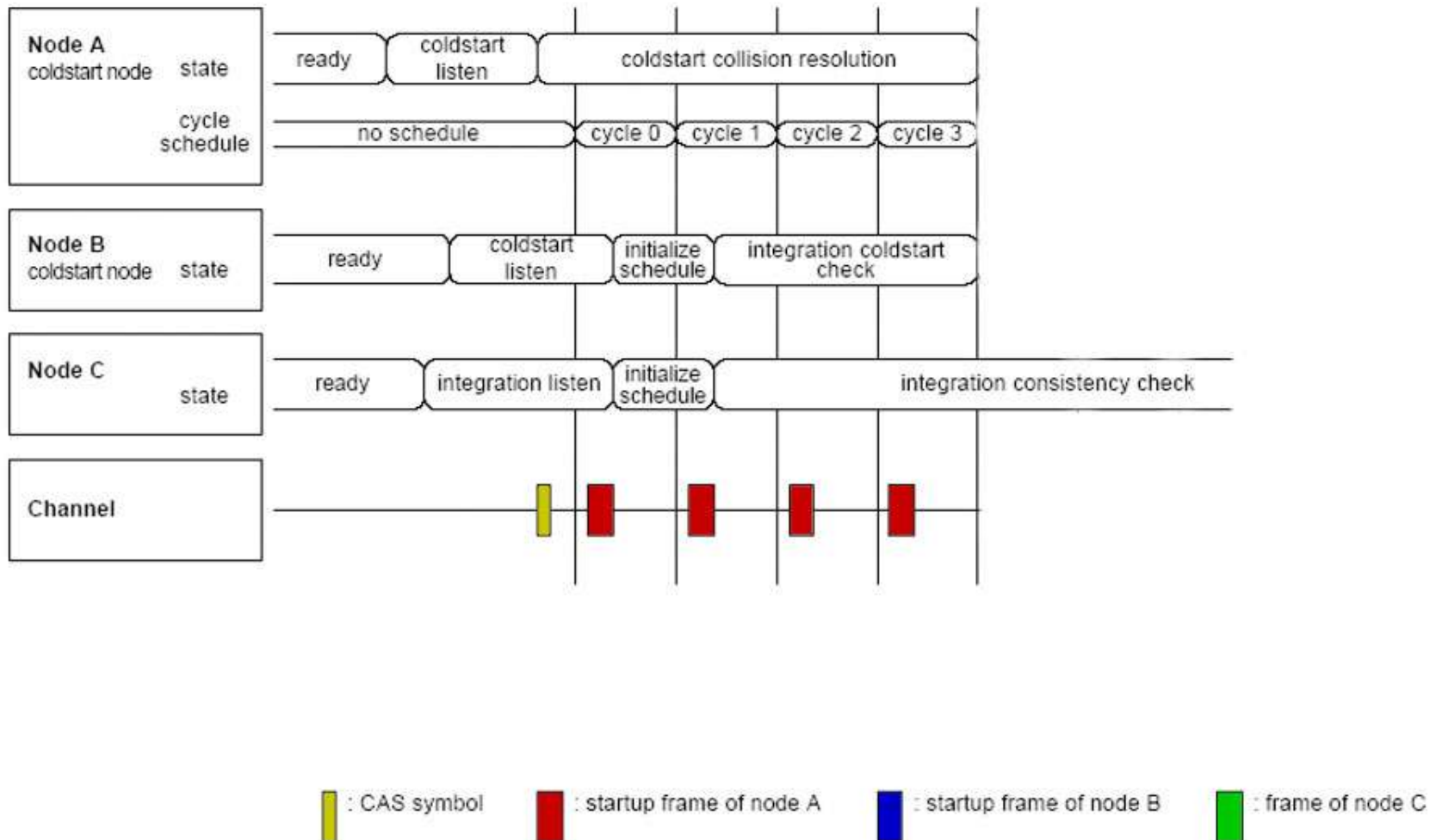
# Cluster Startup.



# Cluster Startup.

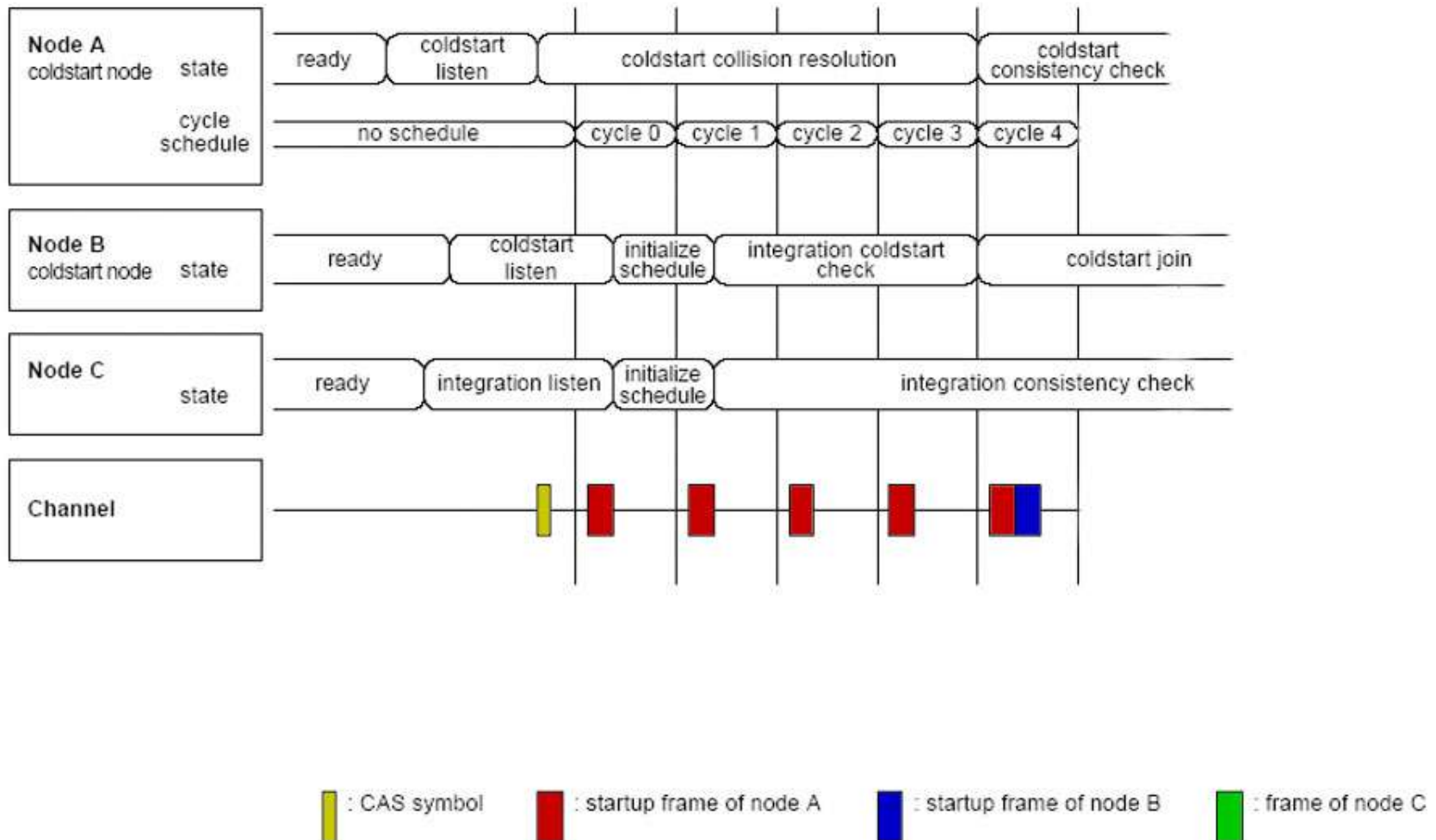


# Cluster Startup.

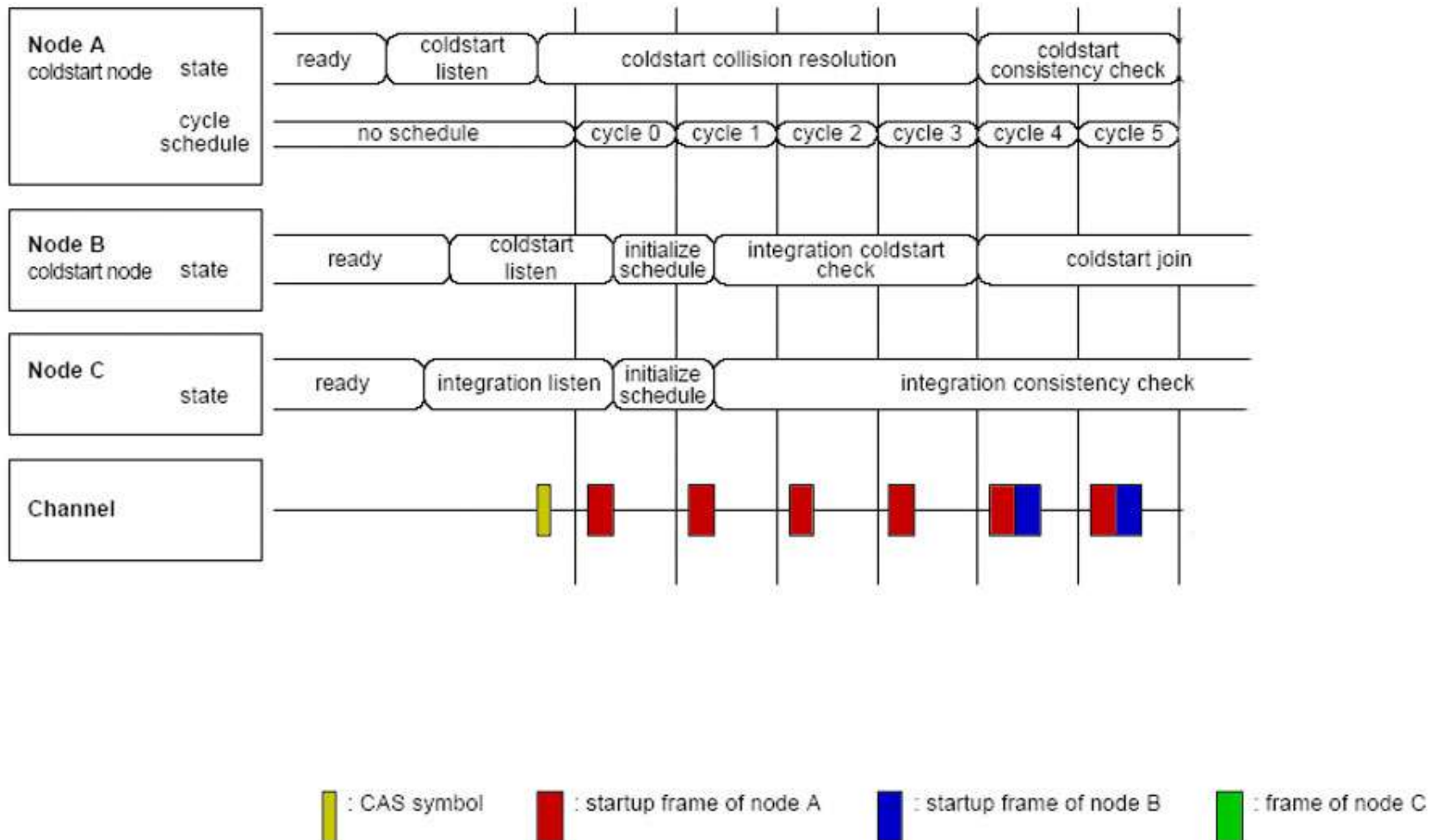




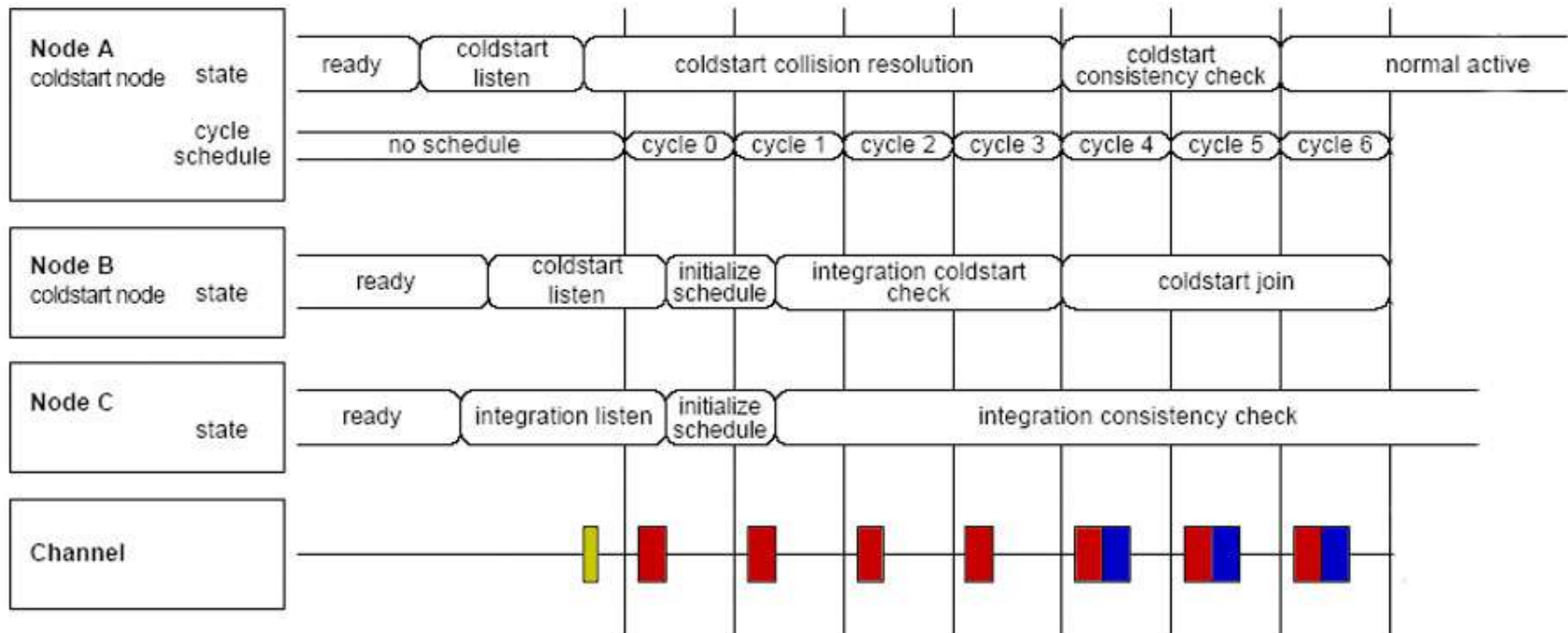
# Cluster Startup.



# Cluster Startup.

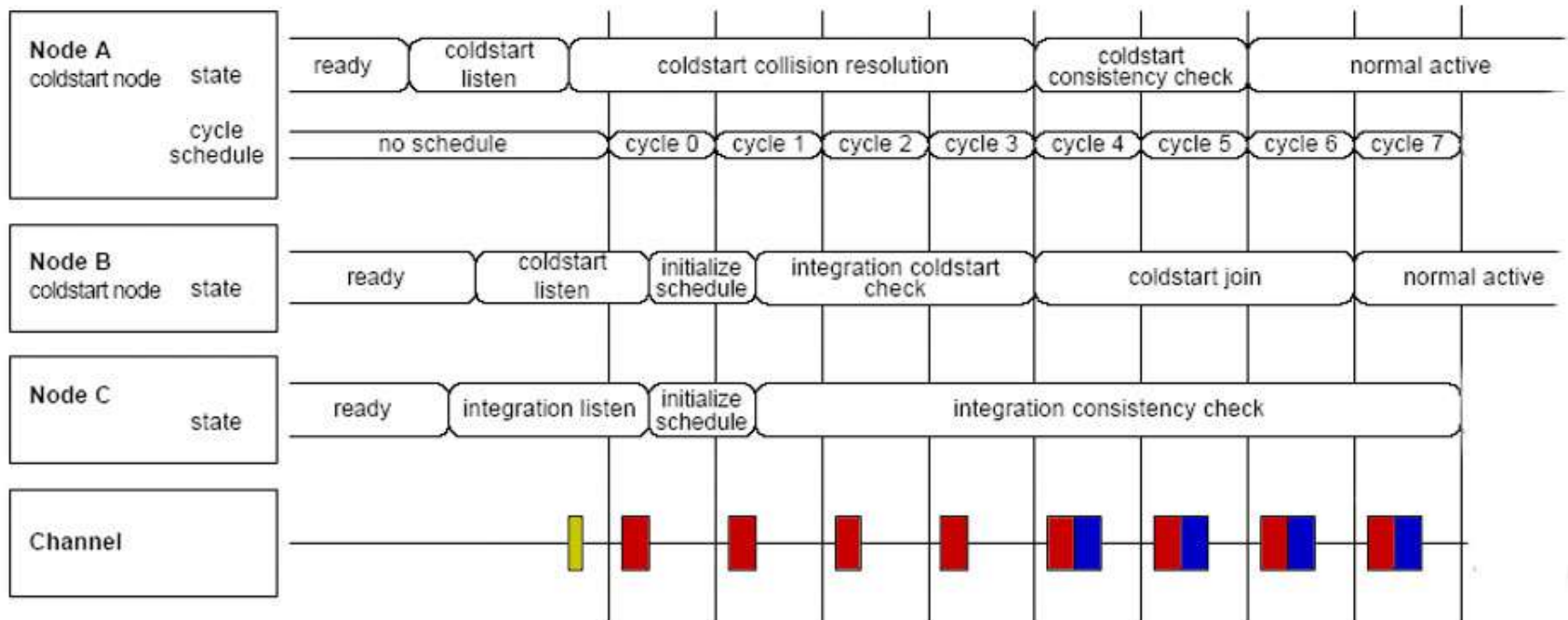


# Cluster Startup.



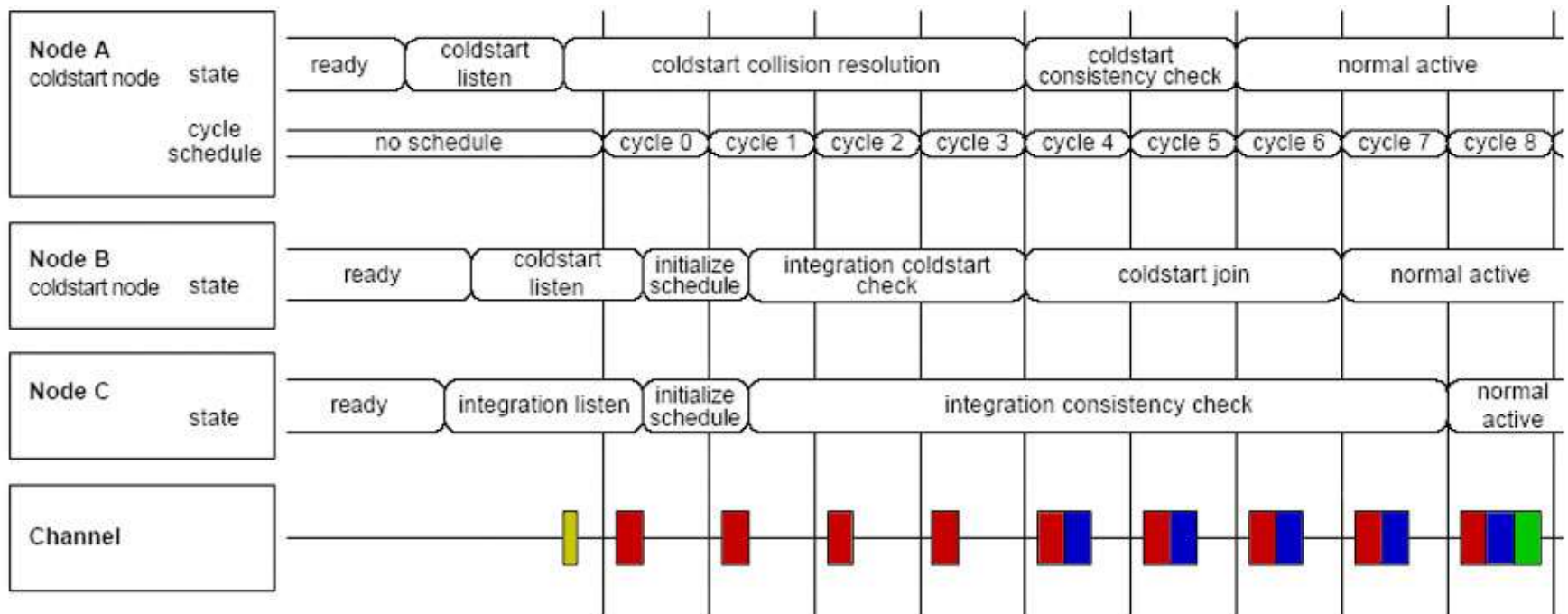
: CAS symbol
  : startup frame of node A
  : startup frame of node B
  : frame of node C

# Cluster Startup.



: CAS symbol
  : startup frame of node A
  : startup frame of node B
  : frame of node C

# Cluster Startup.



□ : CAS symbol

□ : startup frame of node A

□ : startup frame of node B

□ : frame of node C

# Theorem.

- Assumptions:
  - very node has unique time slot in communication cycle to transmit the satrtup frame.
- Conclusion: Given algorithm solves the leader election problem.

- Proof:

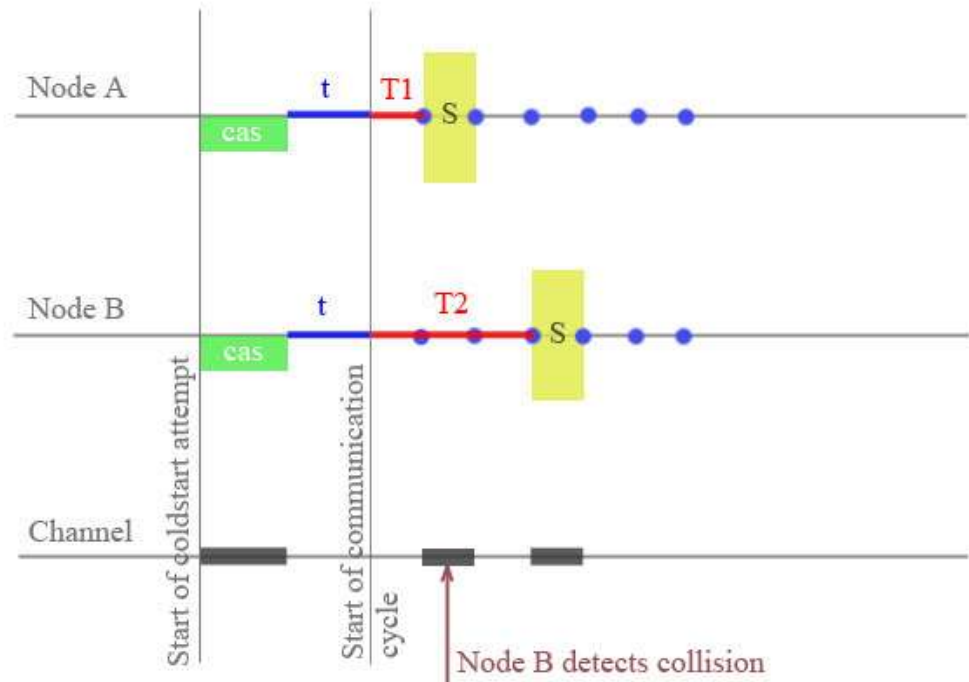
$$[T1 \neq T2] \Rightarrow (t + T1) \neq (t + T2)$$

1 case:

$$\text{if } \left( t_A^{CAS} = t_B^{CAS} \right) \text{ then } t_A^S \neq t_B^S$$

2 case:

$$t_A^{CAS} \neq t_B^{CAS}$$



Thank you for attantion.

Any questions?