

Scalable and Relocatable Interlocking Device

Dr. Natsuki Terada

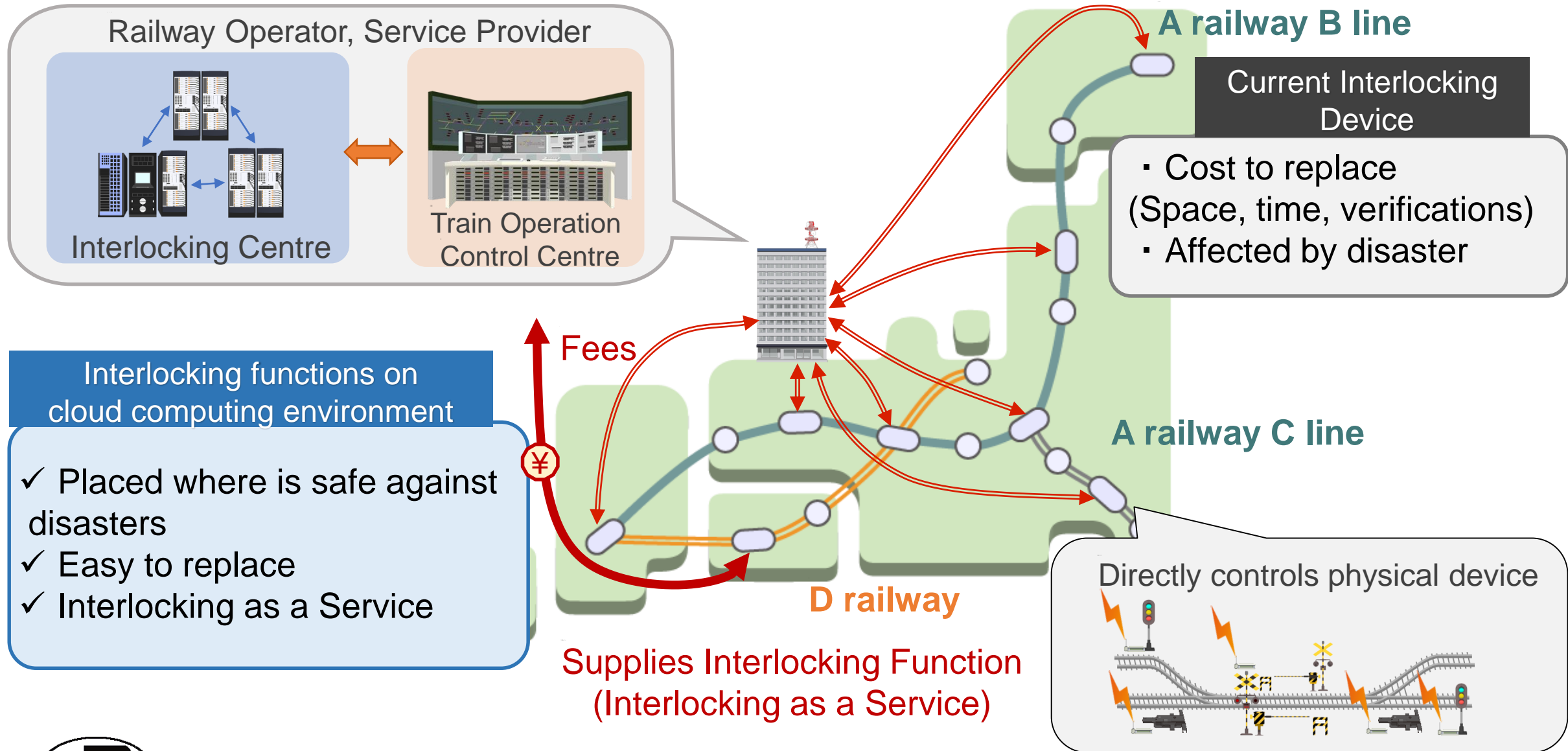
Railway Technical Research Institute



Contents

- Concept of interlocking device on cloud computing environment
- Comparison the concept with current technology
- Proposal of scalable and relocatable interlocking architecture
- Proposal of asynchronous interlocking logic

Image of “Interlocking as a Service”



Advantages of Interlocking Device with Cloud Computing Environment

- **Reduced cost of replacement**
 - No change on the interlocking table,
 - No change of physical connection of field devices.
- **Smaller size of the device** at signal house
- **Resilient against Disasters**
 - The logical part can be placed where is safe from disasters.
 - The backup system can be possible for huge disasters.
 - Quick recovery. e.g. Temporal recovery with radio connection
- **More friendly to Newer Train Control Systems**



e.g. Radio based Train Control

Advantages of Interlocking Device with Cloud Computing Environment

- **Interlocking as a Service**
 - A new business model providing interlocking device
 - For small companies, manufacturer and/or large companies provide interlocking device as a service.
 - For large companies, profit can be obtained from spare equipment.
- **Standardized interlocking logic provided as packages**
 - For typical track layouts of small stations, interlocking logic can be provided as a package.
 - Verification can be extensively reduced.

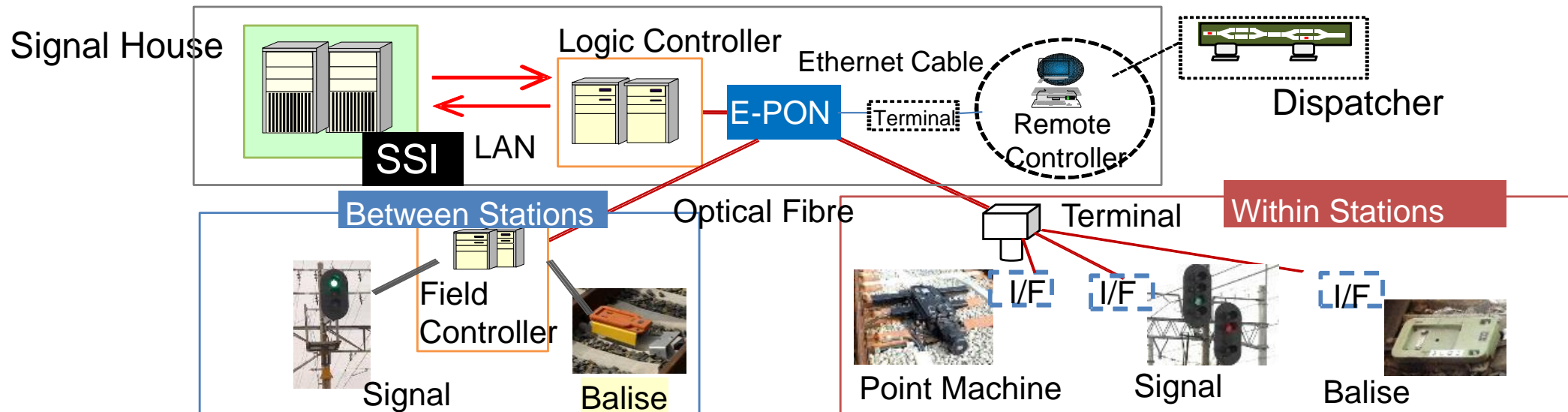
Disadvantages of Interlocking Device with Cloud Computing Environment

- **Larger influences** when interlocking centre is in trouble
 - Backup/redundant systems will ease these troubles.
- **Sustainability of service** from service provider
 - When general purpose processor can be used, many provider may provides service. However, some provider may easily suspend.
- **Security**
 - Required especially when public line and/or radio is used.
 - For crackers, it is enough to stop services. (e.g. DoS attacks)

Observation of Current Technology

Network oriented Interlocking (by East Japan Railway Company)

- Instead of peer to peer connection, network connection is used.
- Each field device has network interface and connected via optical fibres
- Currently, Logic Unit are distributed to each stations

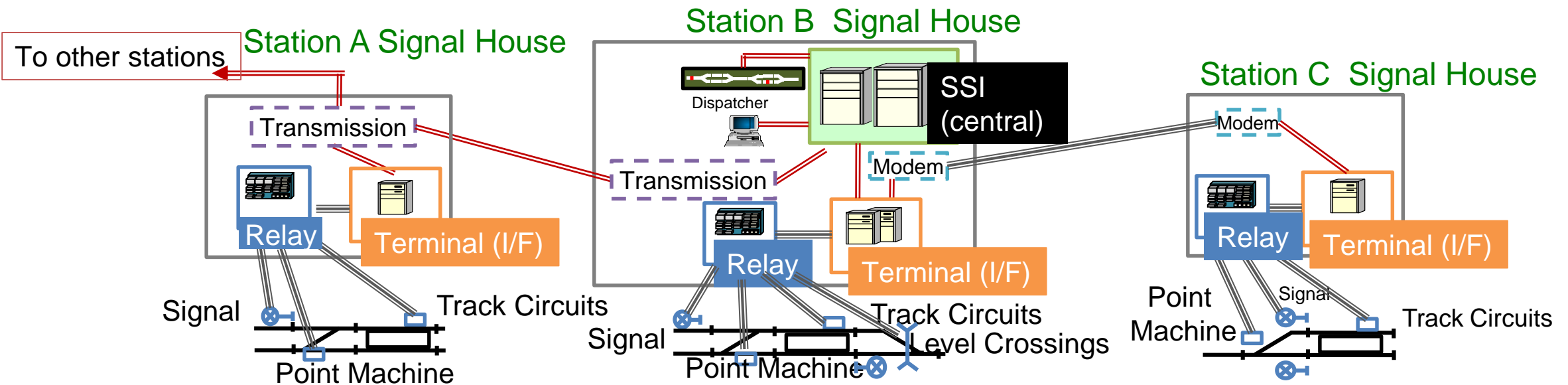


Moving logic units into cloud environment leads to cloud interlocking

Observation of Current Technology

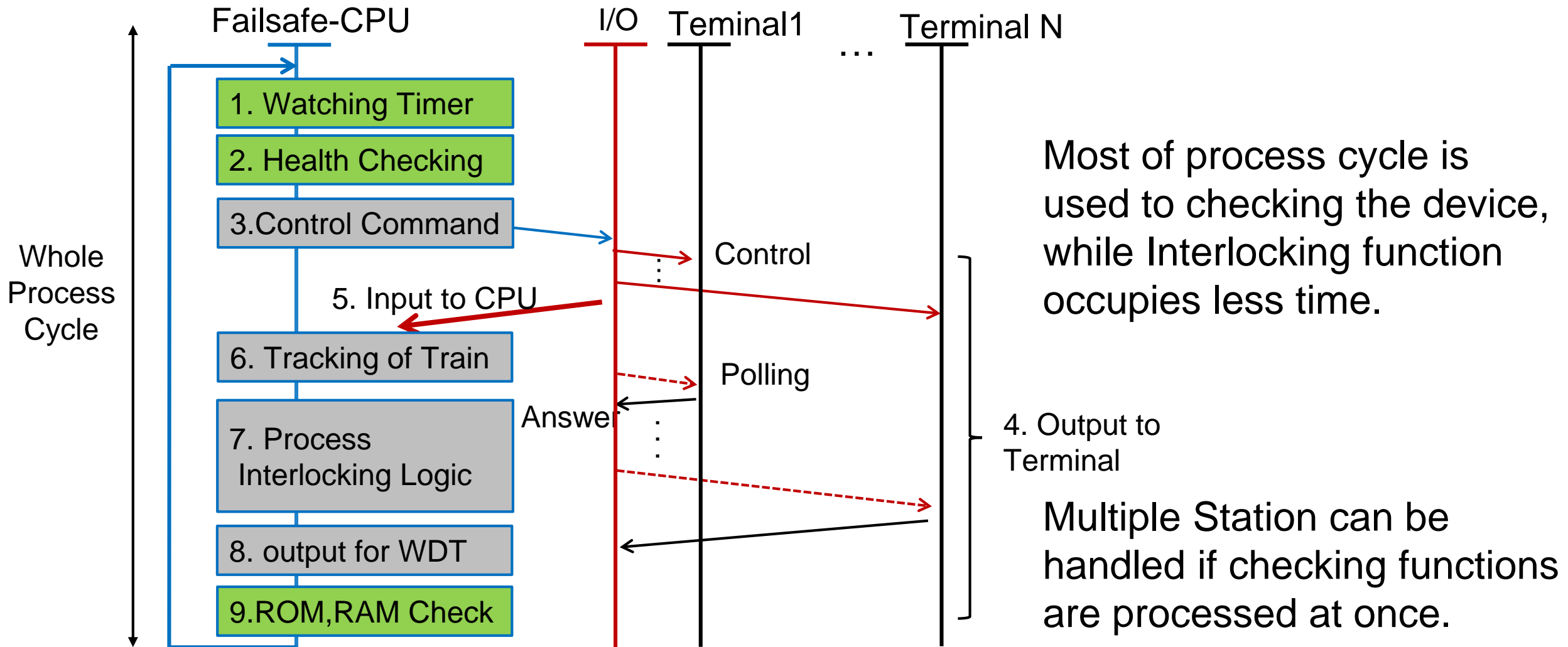
Centralized Interlocking (by West Japan Railway Company)

- A whole line is treated as one station and controlled by one interlocking device
- Terminals/interfaces are installed in the signal house at each station, and connected to field devices with metal cables.
- When the central interlocking device is stopped, operation of whole line stops.

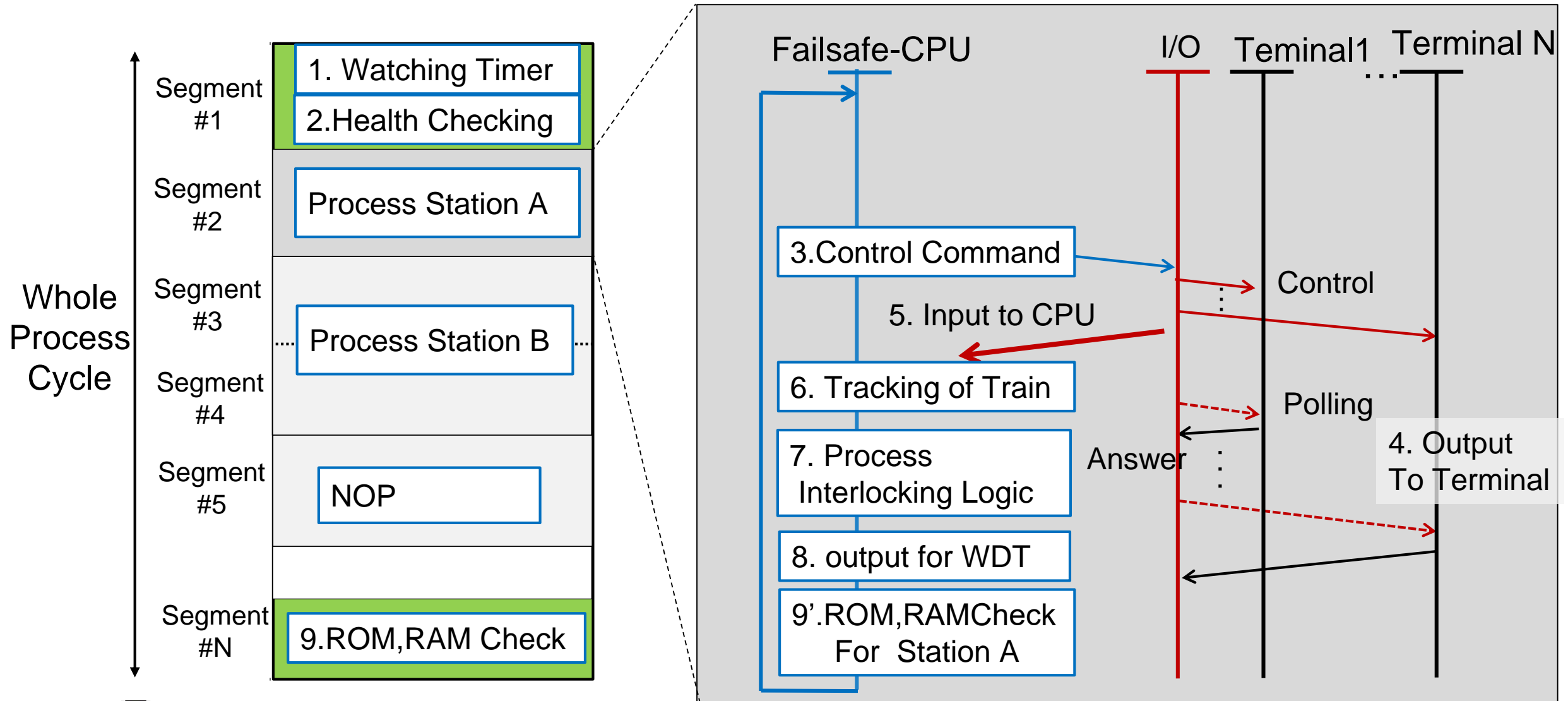


Moving central SSI into cloud environment leads to cloud interlocking

Observation of Current Process for Interlocking



Proposal of Segment Based Processing



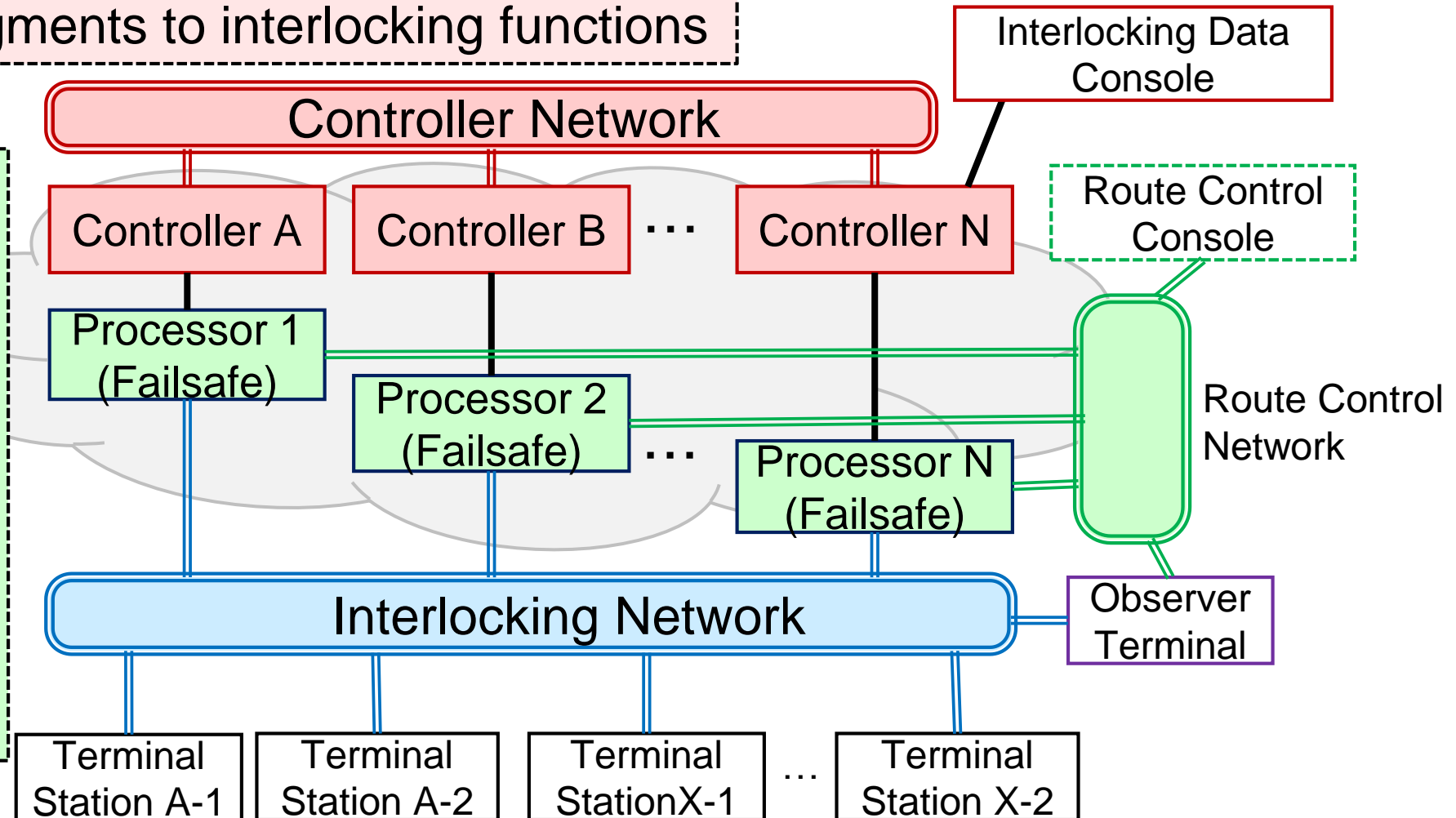
Proposed Architecture of Scalable and Relocatable Interlocking Device

Controllers allocate segments to interlocking functions

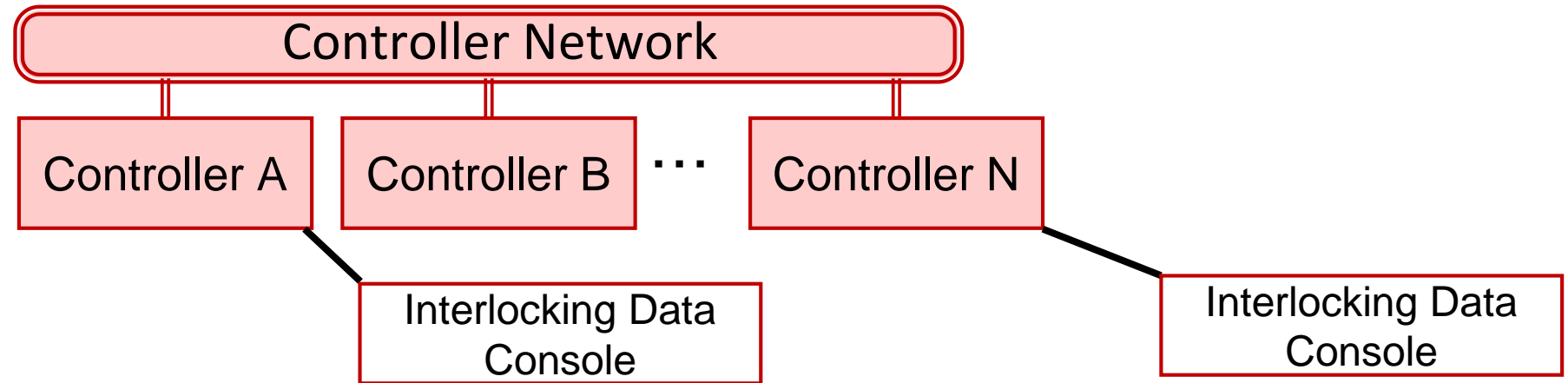
Each processor has several segments to execute interlocking functions.

No restriction of location.
(Relocatable)

Can be increased when capacity is required.
(Scalable)



Architecture of Controller



Roll of Controllers

- allocate segments to interlocking functions
- share allocation table
- upload the interlocking tables from the console to the processors
- observes the health of the processors and their own health

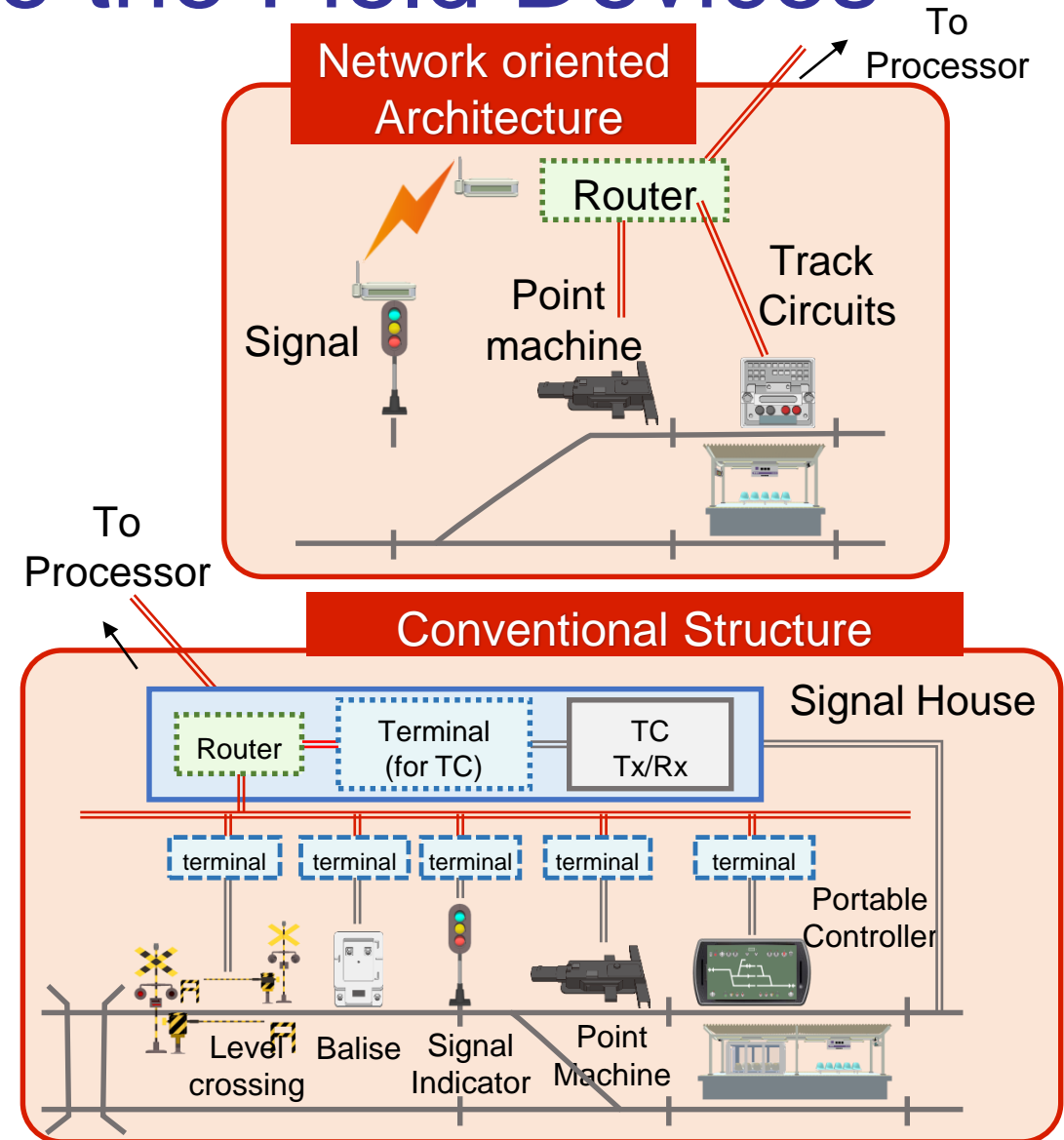
One of the controllers acts as the master

Architecture of Terminal to the Field Devices

Fail-safe function is required to the terminal to the field device considering the case there is no response from the processors

Two Options:

- Field device has its own network interface and fail-safe function.
(same as network oriented interlocking)
- Terminal for field devices are placed at station, and it checks the health of input/output
(same as centralized interlocking)



Interlocking Logics Considering Delay

- Information (input) must be kept in order.
- When the timeout is set and the delay is kept strictly ...?
- Our proposal to allow **asynchronous process**
 - add **time-stamp** and/or **sequence number** to the inputs/outputs
 - wait until all of required inputs at the certain time arrive

Relay		Current State	Adopted/Output Time	Input Time / Seq. No.	
21TPR	Input	Down	0:00:15	0:00:16	5
5CTPR	Input	Up	0:00:15	0:00:15	4
5OTPR	Input	Down	0:00:15	0:00:16	5
8FSR	Output	Up	0:00:16	-	-

Interlocking Logics Considering Delay

- Our proposal to allow asynchronous movement (continued)
 - Do output when output value can be determined without unreached value.
(by three value logic considering indeterminacy)
 - Undetermined \vee True = True
 - Undetermined \wedge False = False
 - Timeout is not uniformly set,
but **independently set** for each logical calculation.
Acceptable timeouts are further study.

Conclusion

- We propose the architecture of scalable and relocatable structure of interlocking functions
 - CPU cycle is divided into segments, and controllers allocate the segments to processing interlocking functions.
- We also propose asynchronous interlocking logic
- The cost of replacement will be reduced

Further Items to be Studied

- Management of the device ID
- Bandwidth required for the network
- Format of the messages between processor and field devices
- Architecture to realise interlocking with non-failsafe processors