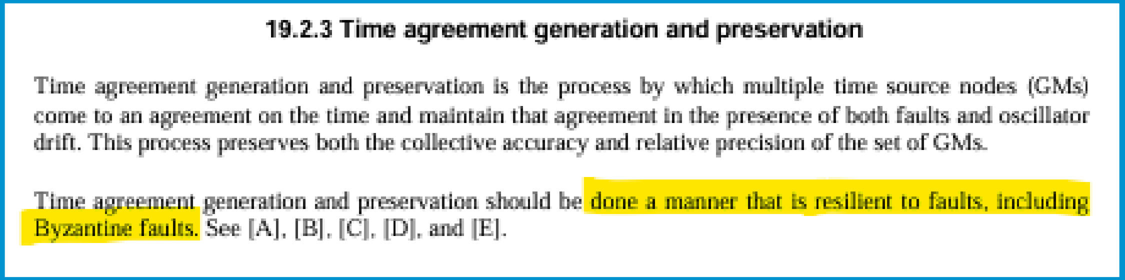
Fault-Tolerant Grandmaster Clock Synchronization Needs for Highly-Reliable Systems

P802.1ASed

Dan Finnegan | Wilfried Steiner

IEEE 802.1 Interim – Sept 2024

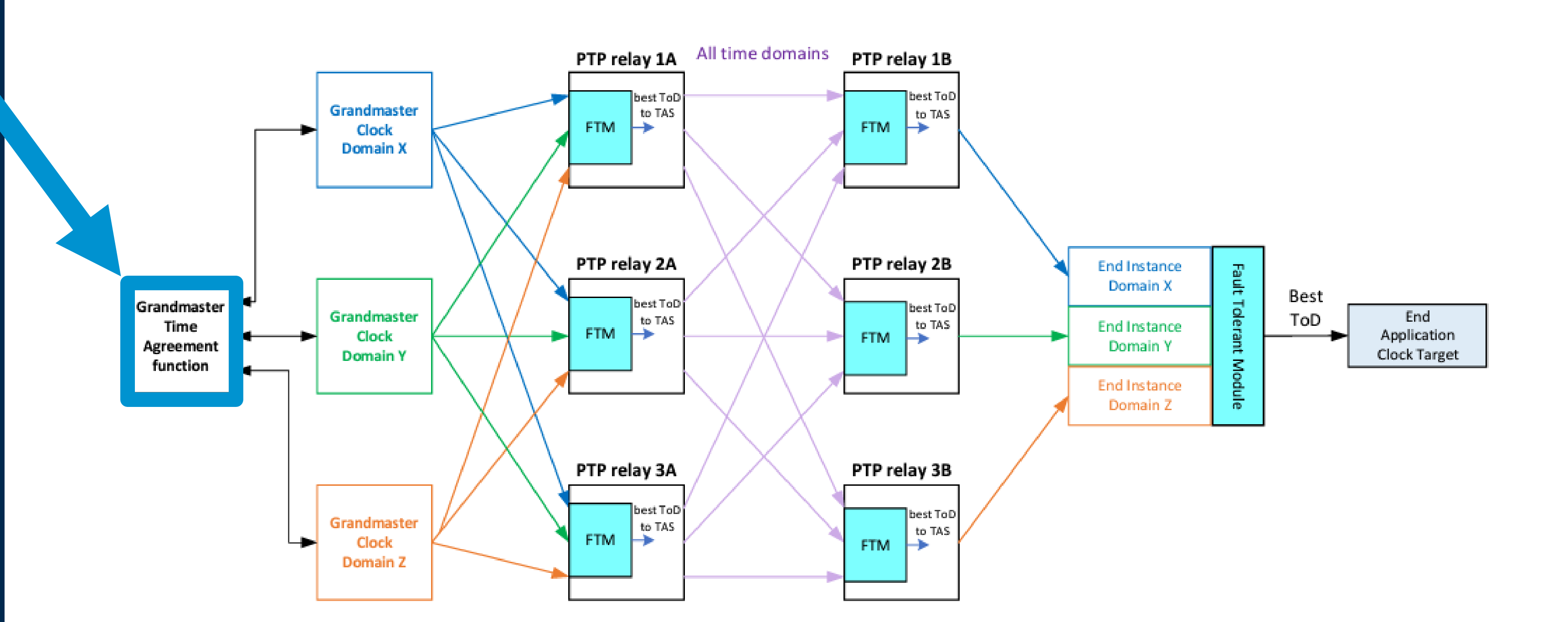
Basic 802.1DP Time-Aware Network



19.2.3 Time agreement generation and preservation

Time agreement generation and preservation is the process by which multiple time source nodes (GMs) come to an agreement on the time and maintain that agreement in the presence of both faults and oscillator drift. This process preserves both the collective accuracy and relative precision of the set of GMs.

Time agreement generation and preservation should be done a manner that is resilient to faults, including Byzantine faults. See [A], [B], [C], [D], and [E].



Fault Tolerant Timing Module (FTTM)

- 802.1DP compliant devices that are intended for Synchronous functions are expected to use 802.1AS-2020 mechanisms to synchronize a time domain with a Grand Master (GM) clock
  - Multiple independent time domains required to provide fault tolerance

- FTTM is an application layer function to be implemented in all time aware bridges and end stations that support multiple time domains
- FTTM manages the selection of a clock source from among multiple independent time domains

- Time Error Accumulation
  - FTTM uses a "max allowable skew" between two independent time domains (GMs) based on the sum of the total possible error that can accumulate from a non-faulty GM time sender to receiver
  - Two GMs are "paired" and therefore "in sync" if the delta between the two is less than the theoretical max error that could accumulate over the pathway for a non-malfunctioning GM

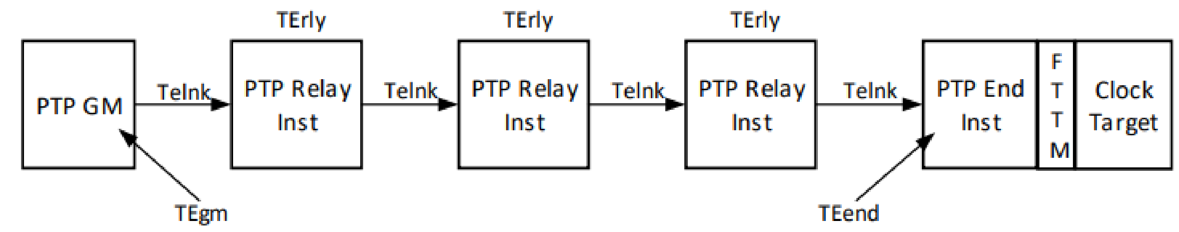


Figure 7-1—Time error accumulation across a network

State Machine for FTTM

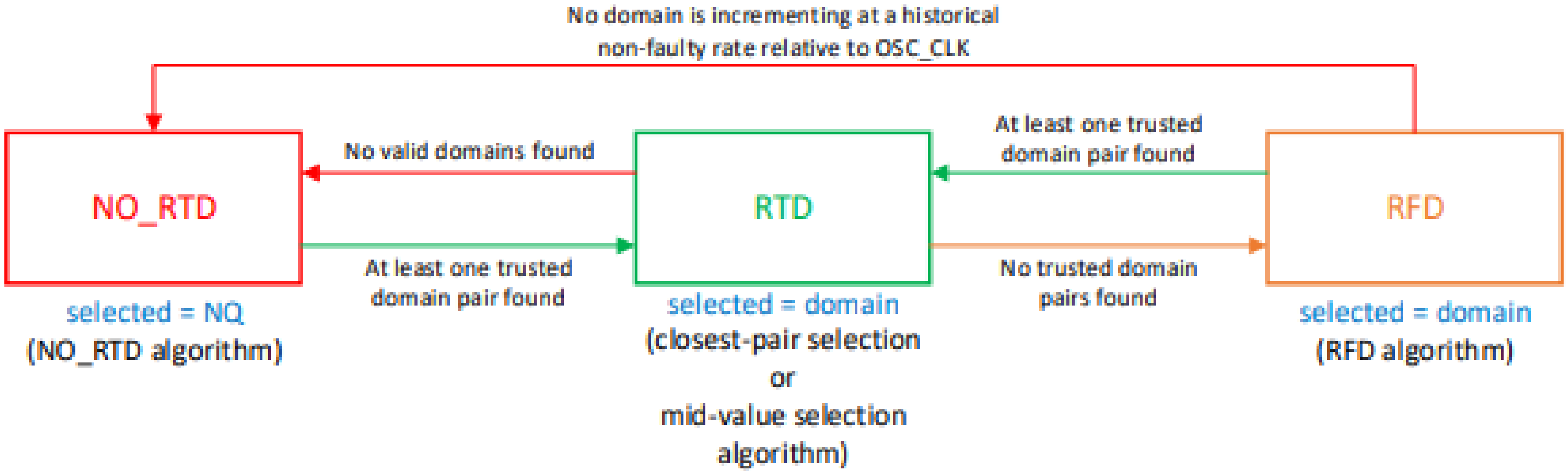

No domain is incrementing at a historical non-faulty rate relative to OSC_CLK
NO_RTD
No valid domains found
RTD
At least one trusted domain pair found
RFD
At least one trusted domain pair found
No trusted domain pairs found
selected = NQ
(NO_RTD algorithm)
selected = domain
(closest-pair selection or mid-value selection algorithm)
selected = domain
(RFD algorithm)

Figure 7-6—Default state-machine for FTTM

- NO_RTD – No redundant time domain found
- RTD – Redundant time domain found
- RFD – Redundant frequency (provided by local oscillator) used to maintain time domain

Grandmaster Time Agreement Function (GTAF)

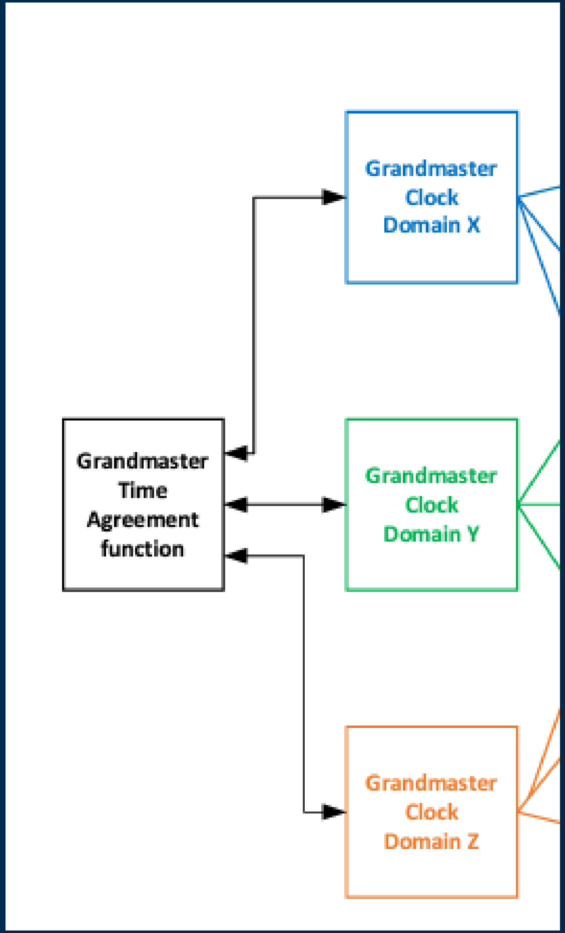
- FTTM expects as an input multiple independent but synchronized time domains
  - Requires multiple independent Grandmaster clocks

  AND
  - These Grandmaster clocks must be synchronized

- FTTM requires at least two “paired” GMs for normal function
  - A breakdown in the synchronization of the GMs will result in a failure of the FTTM to identify valid Redundant Time Domains

- The actual degree of fault tolerance provided by the FTTM can be no greater than the reliability of the synchronization of the input GMs

- To use Synchronous 802.1DP TSN for critical functions:
  <u>Grandmaster Time Agreement Function must be Fault-Tolerant!</u>


Grandmaster Time Agreement function
Grandmaster Clock Domain X
Grandmaster Clock Domain Y
Grandmaster Clock Domain Z

Fault Tolerance for GM Time Agreement Function

- Large Commercial Aircraft avionics must have no more than one catastrophic failure in 1,000,000,000 hours
  - 10^{-9} probability of catastrophic failure for 1 hour exposure is typical for commercial aircraft
- We cannot exhaustively test to this level. We must have architectures that can be ***proven*** to meet the requirement using components that are less dependable than the 10^{-9} failure probability
- Many failures which seem “unbelievable” can happen with a probability much greater than 10^{-9}
- An example of an “unbelievable” failure type that actually occurs is known as the “Byzantine General’s Problem”

Byzantine Generals Problem

A type of electronic failure

Described in academic literature as a parable about Byzantine-era generals attacking a city, with possible lying traitors in their midst (hence the moniker "lying")

Concise practical definitions:

- Byzantine fault: any fault presenting different symptoms to different observers
  - "Observers" here are not required to be human, but can be any portion of the system that reacts to faults

Tolerating these faults is difficult

- General fault tolerance mechanisms (e.g. simple redundancy) don't work, and in fact make the exposure worse (more "lying generals", more opportunity for inconsistent fault symptoms)
- Proper solutions can place huge demands on system resources
- Ignoring the problem doesn't make it go away!

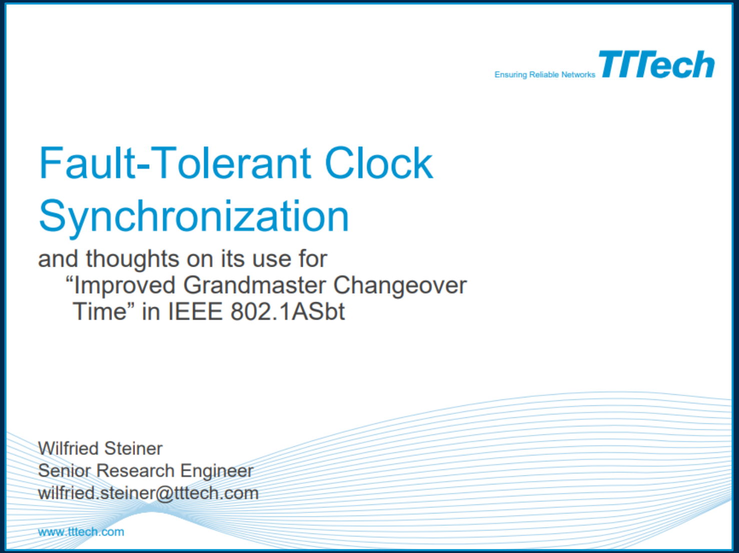
Byzantine Fault-Tolerance

(3f+1) clocks are necessary to tolerate the Byzantine Failure of f clocks (e.g., f=2 → 7 clocks)

- Highly reliable aerospace systems typically require a system failure rate to be in the order of 10^{-9} failures/h or lower.
- Only a distributed fault-tolerant computer system with well-defined fault-containment units (FCUs) achieves the ultrahigh reliability requirement.
- FCUs may fail in a Byzantine failure mode and the distributed computer system must be designed to mitigate this failure mode.

- **In order to tolerate f Byzantine failures *n=3f+1* nodes are needed.**

- The concept to Byzantine fault tolerance has been introduced in: Lamport, L., Shostak, R., Pease, M. (1982). *The Byzantine Generals Problem*. Comm. ACM TOPLAS. Vol. 4 (3). (pp.382-401).



Ensuring Reliable Networks TTTech

Fault-Tolerant Clock Synchronization

and thoughts on its use for
"Improved Grandmaster Changeover Time" in IEEE 802.1ASbt

Wilfried Steiner
Senior Research Engineer
wilfried.steiner@tttech.com

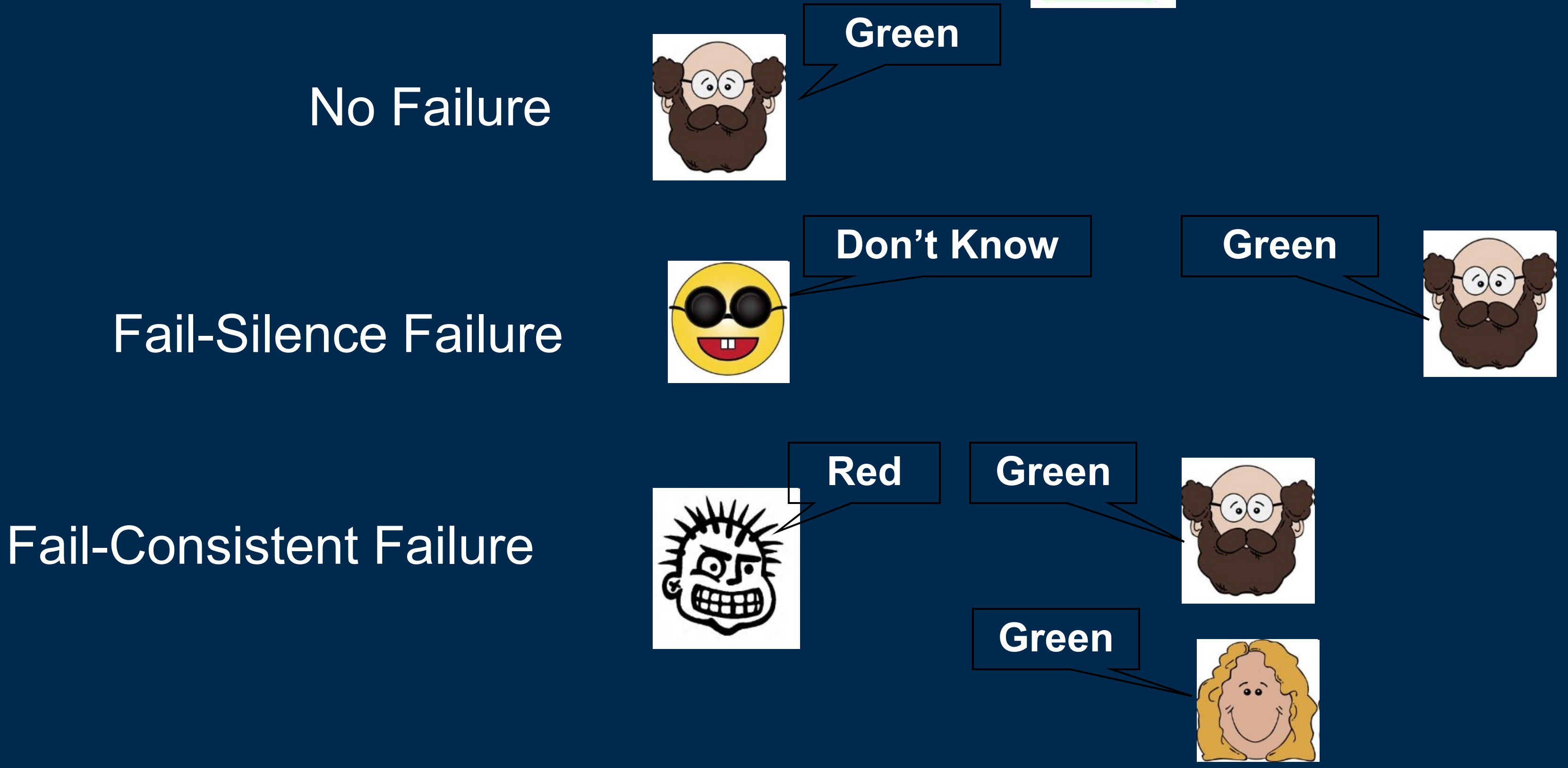
www.tttech.com

https://www.ieee802.org/1/files/public/docs2012/new-avb-wsteiner-fault-tolerant-clock-synchronization-0112-v01.pdf

Byzantine Fault-Tolerance

Situation:

What is the color of the house?



No Failure

Green

Fail-Silence Failure

Don’t Know

Green

Fail-Consistent Failure

Red

Green

Green

Byzantine Fault-Tolerance

Situation:

What is the color of the house?

Static Situation – One Truth

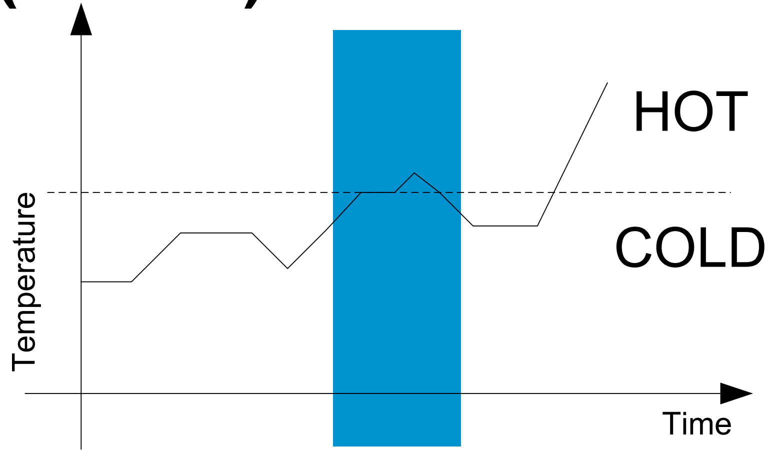
Situation:

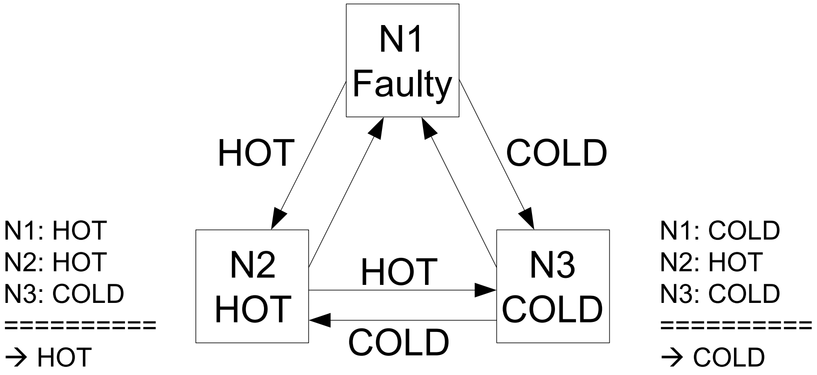
What is the color of the ball ?

Dynamic Situation – More than One Truth

Byzantine Fault-Tolerance (cont.)

A distributed system that measures the temperature of a vessel shall raise an alarm when the temperature exceeds a certain threshold.
The system shall tolerate the arbitrary failure of one node.
How many nodes are required?
How many messages are required?





N1: HOT
N2: HOT
N3: COLD

==========

→ HOT

N1: COLD
N2: HOT
N3: COLD

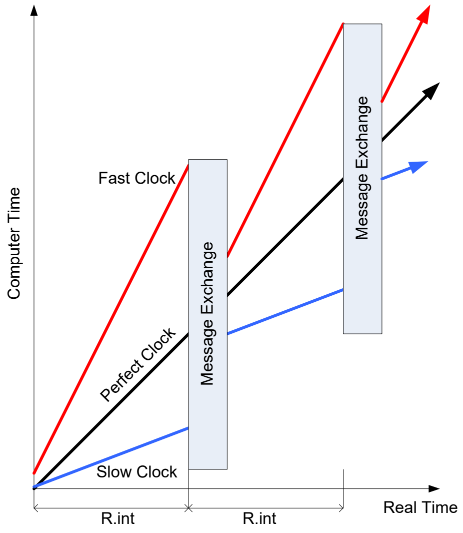
==========

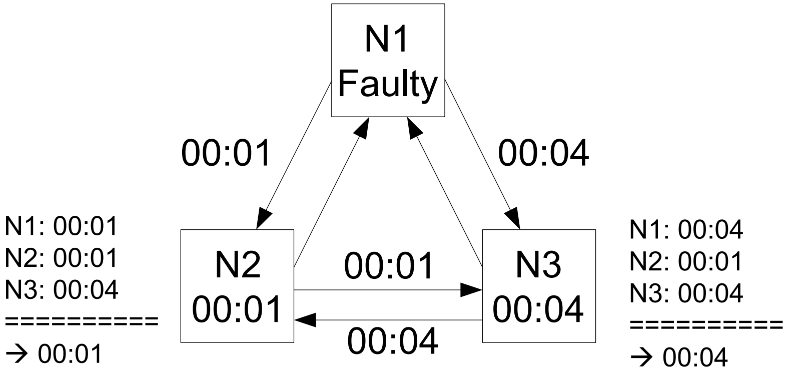
→ COLD

<u>In general, three nodes are insufficient</u> to tolerate the arbitrary failure of a single node.
The two correct nodes are not always able <u>to agree on a value</u>.
A decent body of scientific literature exists that address this problem of dependable systems, in particular dependable communication.

Byzantine Fault-Tolerance (cont.)

A distributed system in which all nodes are equipped with local clocks, all clocks shall become and remain synchronized.
The system shall tolerate the arbitrary failure of one node.
How many nodes are required?
How many messages are required?





<u>In general, three nodes are insufficient</u> to tolerate the arbitrary failure of a single node.
The two correct nodes are not always able <u>to bring their clocks into close agreement</u>.
A decent body of scientific literature exists that address this problem of fault-tolerant clock synchronization.

A Byzantine Fault-Tolerant Grandmaster Time Agreement Function must…

- FTTM is suitable for inclusion in a Byzantine Fault-Tolerant System IF and ONLY IF the Grandmaster Time Agreement Function is also Byzantine Fault Tolerant

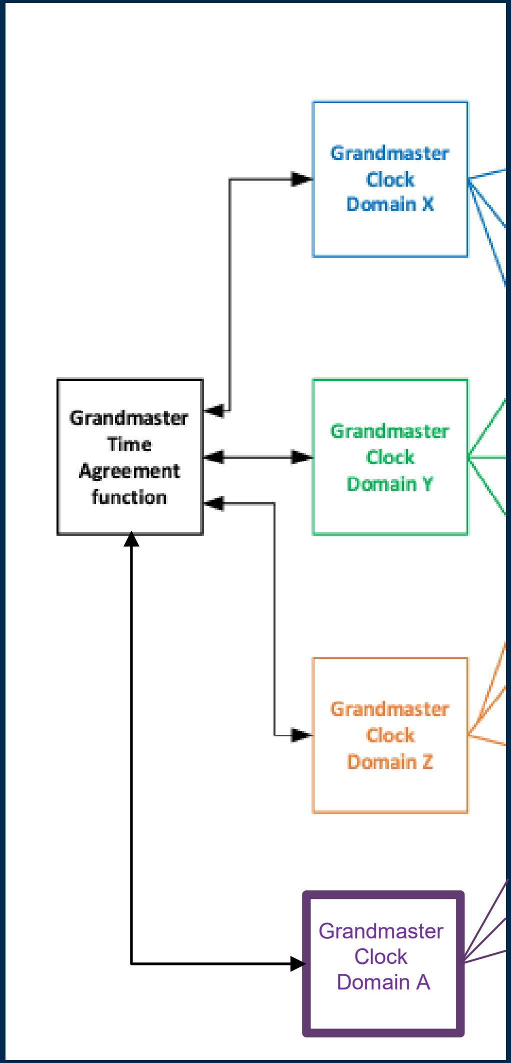
- *Recall: 3f+1 clocks are necessary to tolerate the Byzantine Failure of f clocks*
  - For single Byzantine Fault-Tolerance, the GTAF must synchronize 3*1+1 = 4 clocks
  - For double Byzantine Fault-Tolerance, the GTAF must synchronize 3*2+1 = 7 clocks

BYZANTINE FAULT-TOLERANT GTAF MUST INCLUDE AT LEAST 4 CLOCKS

Grandmaster Time Agreement Function Possible Options

A High-Integrity GTAF is necessary…
Let's not re-invent the wheel but build on proven solutions!

- PPS Synchronization of GMs
  - Low to no Fault-Tolerance - suitable for low-criticality applications
- 802.1ASdm Hot-Standby Synchronization of GMs
  - Provides FT but not Byzantine – not suitable for high-criticality applications
- TTP (or other serial x-channel data links) Synchronization of GMs
  - Additional hardware required
- SAE AS6802-based synchronization of GMs
  - Software-only solution possible
  - Only needs to be employed between GM nodes
  - Already selected for DAL A highly reliable systems with 10^{-9} criticality
  - Existing patent:
    - US9331805B2 *Network and Method for Implementing a High-Availability Grand Master Clock*


Grandmaster Time Agreement function
Grandmaster Clock Domain X
Grandmaster Clock Domain Y
Grandmaster Clock Domain Z
Grandmaster Clock Domain A

TTTech