



60802 Dynamic Time Sync Error – Recommended Parameters & Correction Factors

David McCall (Intel)

March IEEE 802 Plenary – 802.1 TSN – IEC/IEEE 60802

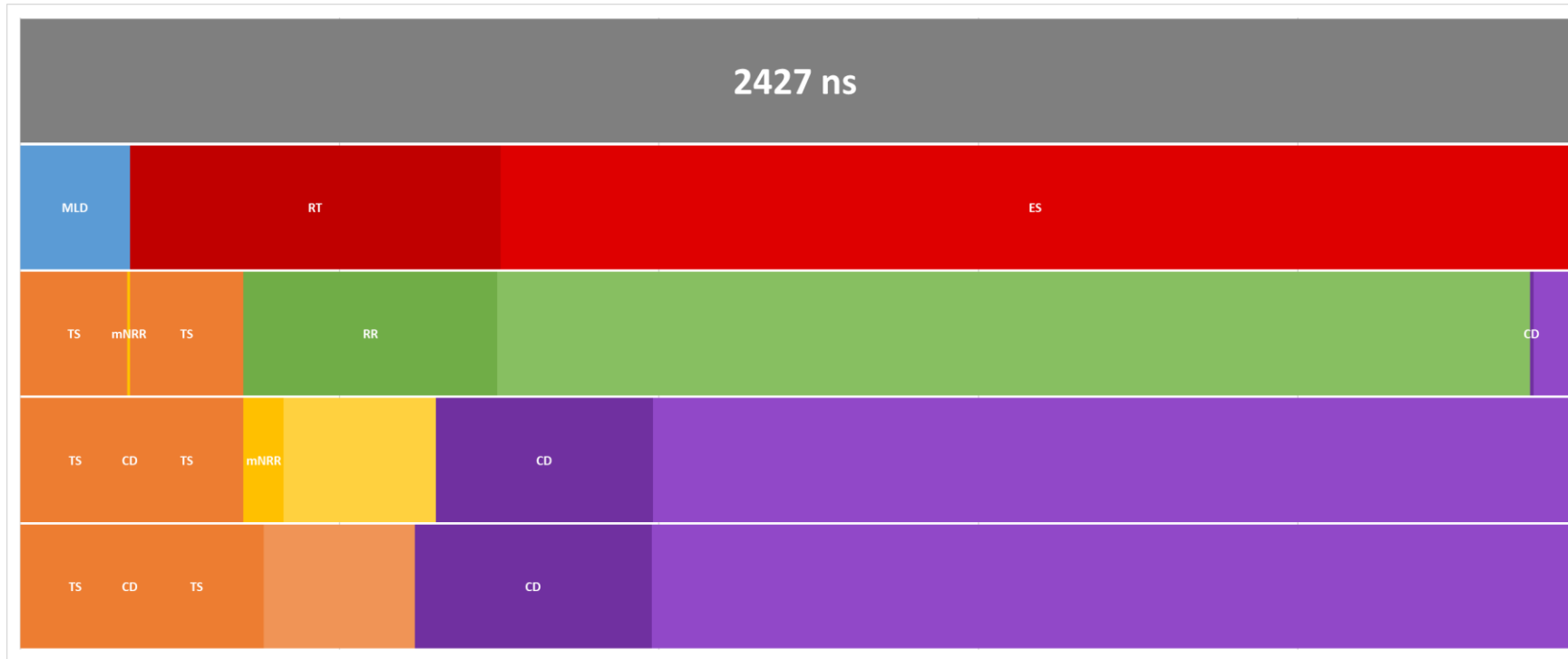
Abstract

- The Monte Carlo Analysis approach to modelling Dynamic Time Error (DTE) across long chains of networked devices was developed to assist the IEC/IEEE 60802 group meet the target of 1us Time Sync Error across 100 hops.
- Previous presentations...
 - [60802-McCall-et-al-Time-Sync-Error-Model-0921-v03.pdf](#)
 - [60802-McCall-Stanton-Time-Sync-Error-Model-and-Analysis-2021-11-v02.pdf](#)
 - [60802-McCall-Stanton-Time-Sync-Error-Model-and-Analysis-0222-v03.pdf](#)
 - [60802-McCall-Stanton-Time-Sync-Error-Model-and-Analysis-0322-v01.pdf](#)
 - [60802-McCall-Time-Sync-Monte-Carlo-Results-for-Time-Series-Comparison-0322-v01.pdf](#)
- In this contribution we:
 - Provide recommended parameters and correction factors to achieve the group's goals along; additional background information; suggested Time Series Simulations to validate the goals; and a suggested approach to normative and informative test for the specification.

Content

- Recommended parameters & correction factors
- Background – why these are the recommended settings
- Time Series Simulations to validate recommendations
- Normative & Informative text for the specification
- Proposed Algorithms

Error Breakdown Charts with ES – Example



Using MAXabs values for top level and breakdown as several of the distributions are not gaussian so 7σ doesn't work well.

| Input Errors | | |
|-----------------------------|---------|-------|
| GM Clock Drift Max | +1.5 | ppm/s |
| GM Clock Drift Min | -1.5 | ppm/s |
| GM Nodes w/ Clock Drift | 80% | |
| Clock Drift Max (non-GM) | +1.5 | ppm/s |
| Clock Drift Min (non-GM) | -1.5 | ppm/s |
| Non-GM Nodes w/ Clock Drift | 80% | |
| Timestamp Granularity TX | 4 | ±ns |
| Timestamp Granularity RX | 4 | ±ns |
| Dynamic Time Stamp Error TX | 8 | ±ns |
| Dynamic Time Stamp Error RX | 8 | ±ns |
| Input Parameters | | |
| pDelay Interval | 31.25 | ms |
| Sync Interval | 125 | ms |
| pDelay Turnaround Time | 1 | ms |
| residenceTime | 1 | ms |
| Input Correction Factors | | |
| Mean Link Delay Averaging | 0 | % |
| NRR Drift Correction | 0 | % |
| RR Drift Correction | 0 | % |
| pDelayResponse → Sync | 0 | % |
| mNRR Smoothing N | 1 | |
| mNRR Smoothing M | 1 | |
| Configuration | | |
| Hops | 100 | |
| Runs | 100,000 | |

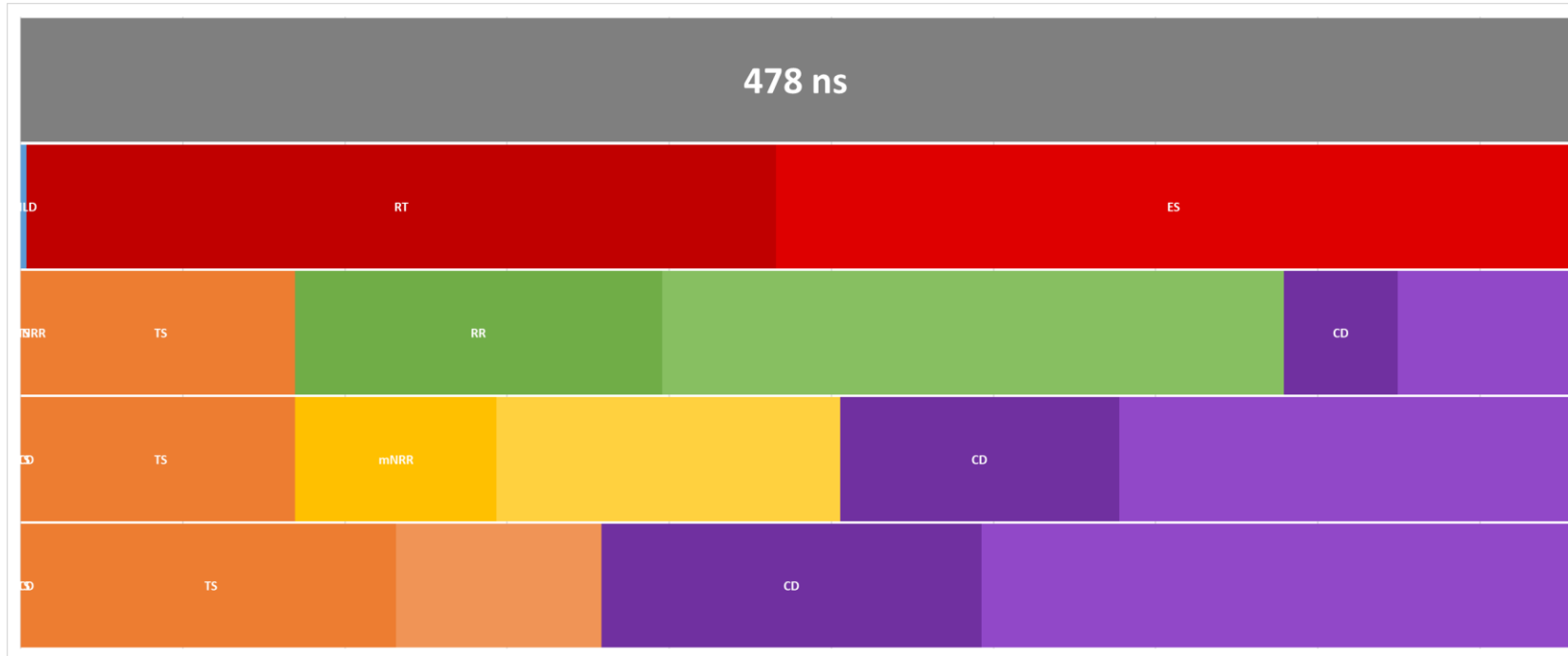
Notes

- Added “Fraction of Nodes with Drift” factor
- In Time Series Simulation temp ramps up...is held stable...ramps down...is held stable.
- % of time stable, i.e. no Clock Drift, is 20%
- This factor matches Monte Carlo more closely to Time Series

Recommended Settings

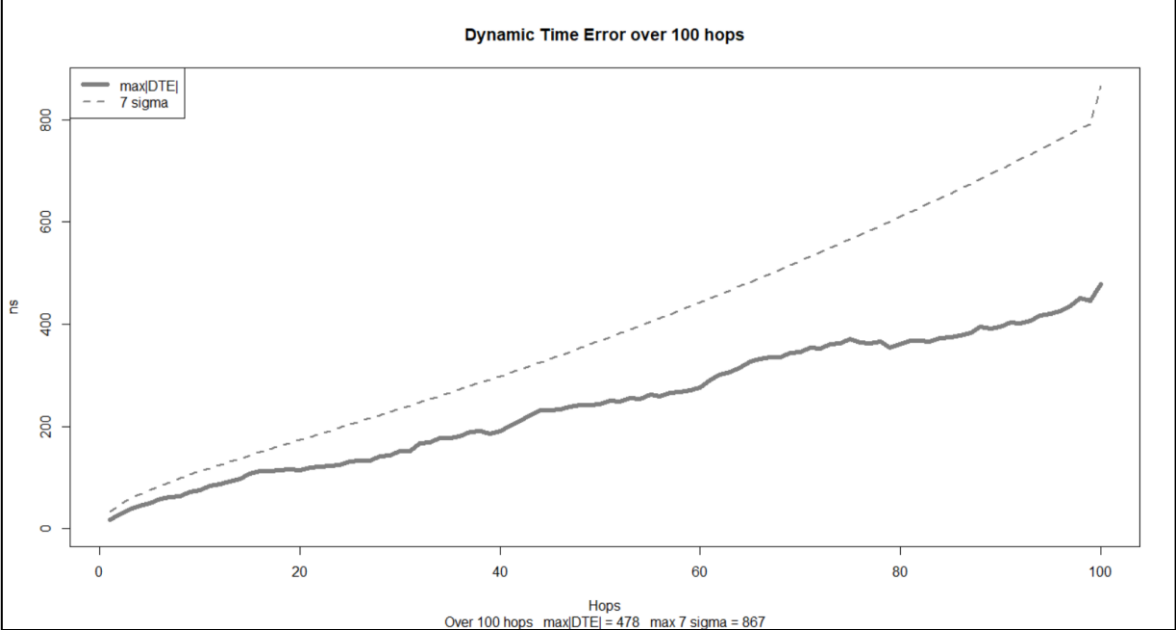
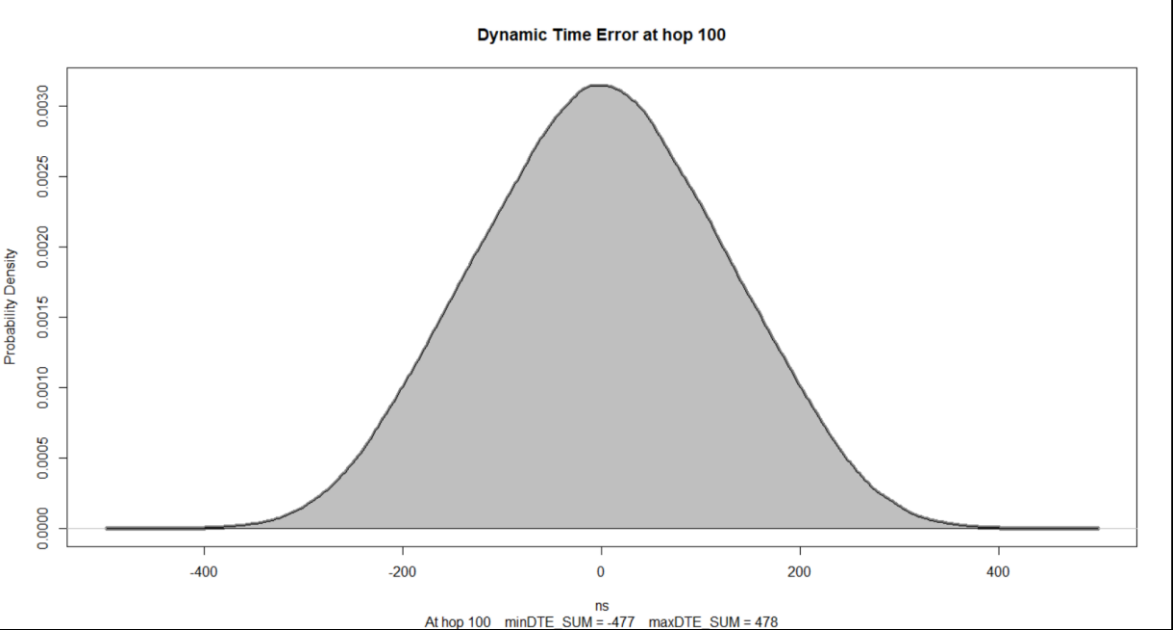
Input parameters & correction factors

Recommended Settings



| Input Errors | | |
|-----------------------------|---------|-------|
| GM Clock Drift Max | +1.5 | ppm/s |
| GM Clock Drift Min | -1.5 | ppm/s |
| GM Nodes w/ Clock Drift | 80% | |
| Clock Drift Max (non-GM) | +1.5 | ppm/s |
| Clock Drift Min (non-GM) | -1.5 | ppm/s |
| Non-GM Nodes w/ Clock Drift | 80% | |
| Timestamp Granularity TX | 4 | ±ns |
| Timestamp Granularity RX | 4 | ±ns |
| Dynamic Time Stamp Error TX | 4 | ±ns |
| Dynamic Time Stamp Error RX | 4 | ±ns |
| Input Parameters | | |
| pDelay Interval | 125 | ms |
| Sync Interval | 125 | ms |
| pDelay Turnaround Time | 10 | ms |
| residenceTime | 10 | ms |
| Input Correction Factors | | |
| Mean Link Delay Averaging | 98 | % |
| NRR Drift Correction | 90 | % |
| RR Drift Correction | 90 | % |
| pDelayResponse → Sync | 0 | % |
| mNRR Smoothing N | 3 | |
| mNRR Smoothing M | 1 | |
| Configuration | | |
| Hops | 100 | |
| Runs | 100,000 | |

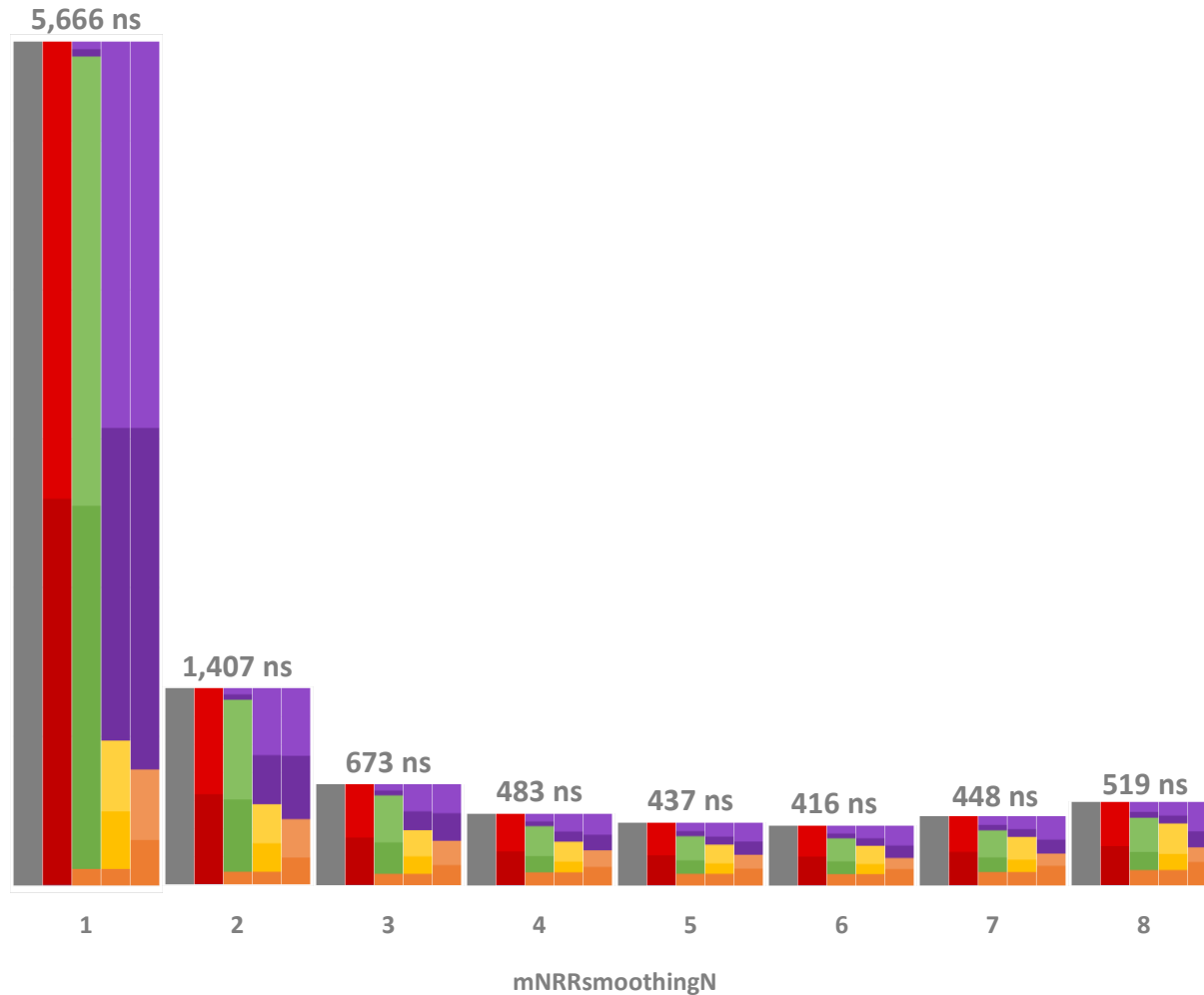
DTE



Background Information

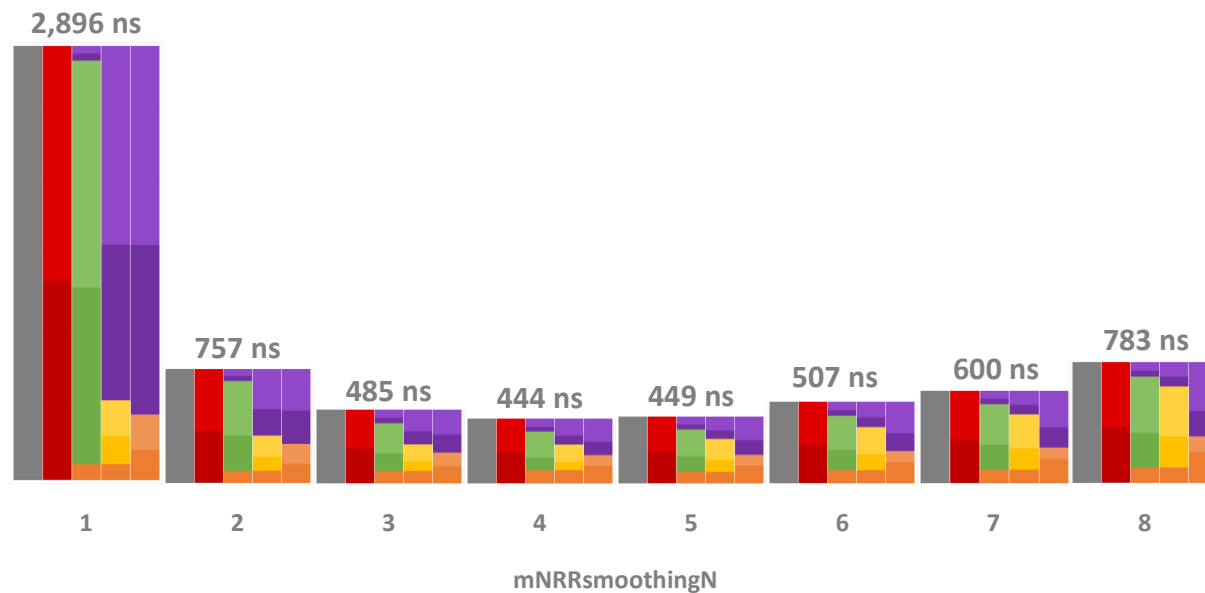
Why these recommended settings?

pDelay Interval 31.25ms



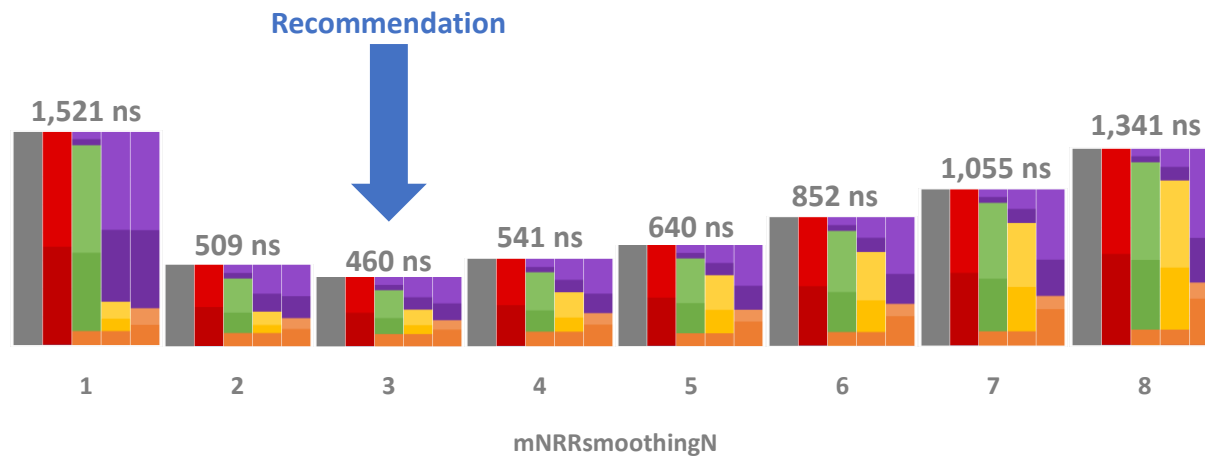
| Input Errors | | |
|-----------------------------|----------|-------|
| GM Clock Drift Max | +1.5 | ppm/s |
| GM Clock Drift Min | -1.5 | ppm/s |
| GM Nodes w/ Clock Drift | 80% | |
| Clock Drift Max (non-GM) | +1.5 | ppm/s |
| Clock Drift Min (non-GM) | -1.5 | ppm/s |
| Non-GM Nodes w/ Clock Drift | 80% | |
| Timestamp Granularity TX | 4 | ±ns |
| Timestamp Granularity RX | 4 | ±ns |
| Dynamic Time Stamp Error TX | 4 | ±ns |
| Dynamic Time Stamp Error RX | 4 | ±ns |
| Input Parameters | | |
| pDelay Interval | 31.25 | ms |
| Sync Interval | 125 | ms |
| pDelay Turnaround Time | 10 | ms |
| residenceTime | 10 | ms |
| Input Correction Factors | | |
| Mean Link Delay Averaging | 98 | % |
| NRR Drift Correction | 90 | % |
| RR Drift Correction | 90 | % |
| pDelayResponse → Sync | 0 | % |
| mNRR Smoothing N | Variable | |
| mNRR Smoothing M | 1 | |
| Configuration | | |
| Hops | 100 | |
| Runs | 100,000 | |

pDelay Interval 62.5ms



| Input Errors | | |
|-----------------------------|----------|-------|
| GM Clock Drift Max | +1.5 | ppm/s |
| GM Clock Drift Min | -1.5 | ppm/s |
| GM Nodes w/ Clock Drift | 80% | |
| Clock Drift Max (non-GM) | +1.5 | ppm/s |
| Clock Drift Min (non-GM) | -1.5 | ppm/s |
| Non-GM Nodes w/ Clock Drift | 80% | |
| Timestamp Granularity TX | 4 | ±ns |
| Timestamp Granularity RX | 4 | ±ns |
| Dynamic Time Stamp Error TX | 4 | ±ns |
| Dynamic Time Stamp Error RX | 4 | ±ns |
| Input Parameters | | |
| pDelay Interval | 62.5 | ms |
| Sync Interval | 125 | ms |
| pDelay Turnaround Time | 10 | ms |
| residenceTime | 10 | ms |
| Input Correction Factors | | |
| Mean Link Delay Averaging | 98 | % |
| NRR Drift Correction | 90 | % |
| RR Drift Correction | 90 | % |
| pDelayResponse → Sync | 0 | % |
| mNRR Smoothing N | Variable | |
| mNRR Smoothing M | 1 | |
| Configuration | | |
| Hops | 100 | |
| Runs | 100,000 | |

pDelay Interval 125ms



| Input Errors | | |
|-----------------------------|----------|-------|
| GM Clock Drift Max | +1.5 | ppm/s |
| GM Clock Drift Min | -1.5 | ppm/s |
| GM Nodes w/ Clock Drift | 80% | |
| Clock Drift Max (non-GM) | +1.5 | ppm/s |
| Clock Drift Min (non-GM) | -1.5 | ppm/s |
| Non-GM Nodes w/ Clock Drift | 80% | |
| Timestamp Granularity TX | 4 | ±ns |
| Timestamp Granularity RX | 4 | ±ns |
| Dynamic Time Stamp Error TX | 4 | ±ns |
| Dynamic Time Stamp Error RX | 4 | ±ns |
| Input Parameters | | |
| pDelay Interval | 125 | ms |
| Sync Interval | 125 | ms |
| pDelay Turnaround Time | 10 | ms |
| residenceTime | 10 | ms |
| Input Correction Factors | | |
| Mean Link Delay Averaging | 98 | % |
| NRR Drift Correction | 90 | % |
| RR Drift Correction | 90 | % |
| pDelayResponse → Sync | 0 | % |
| mNRR Smoothing N | Variable | |
| mNRR Smoothing M | 1 | |
| Configuration | | |
| Hops | 100 | |
| Runs | 100,000 | |

Other pDelay Intervals

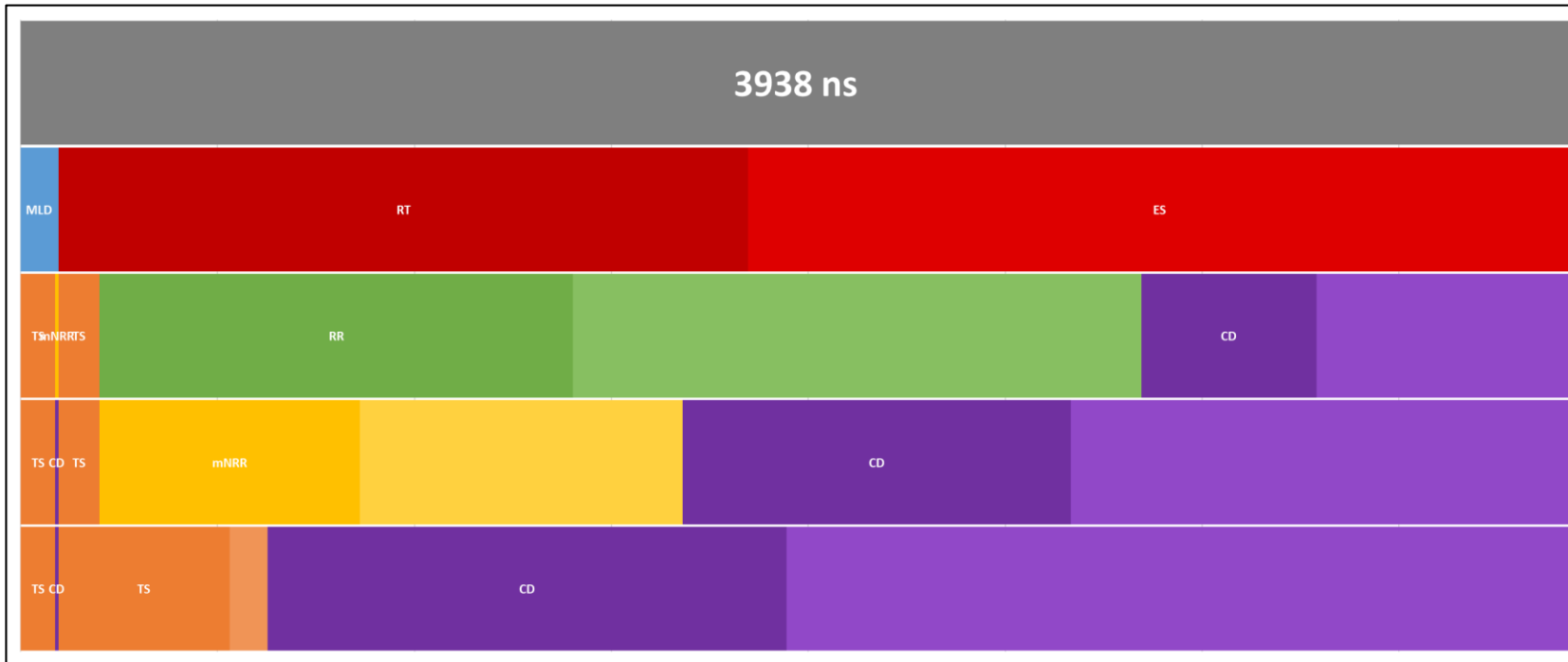
- Best maxABS DTE with 250ms pDelayInterval: 533ns (N = 2)
- Best maxABS DTE with 500ms pDelayInterval: 852ns (N = 1)

Time Series Simulations To Validate Recommendation

Time Series Simulations to Validate Recommendation

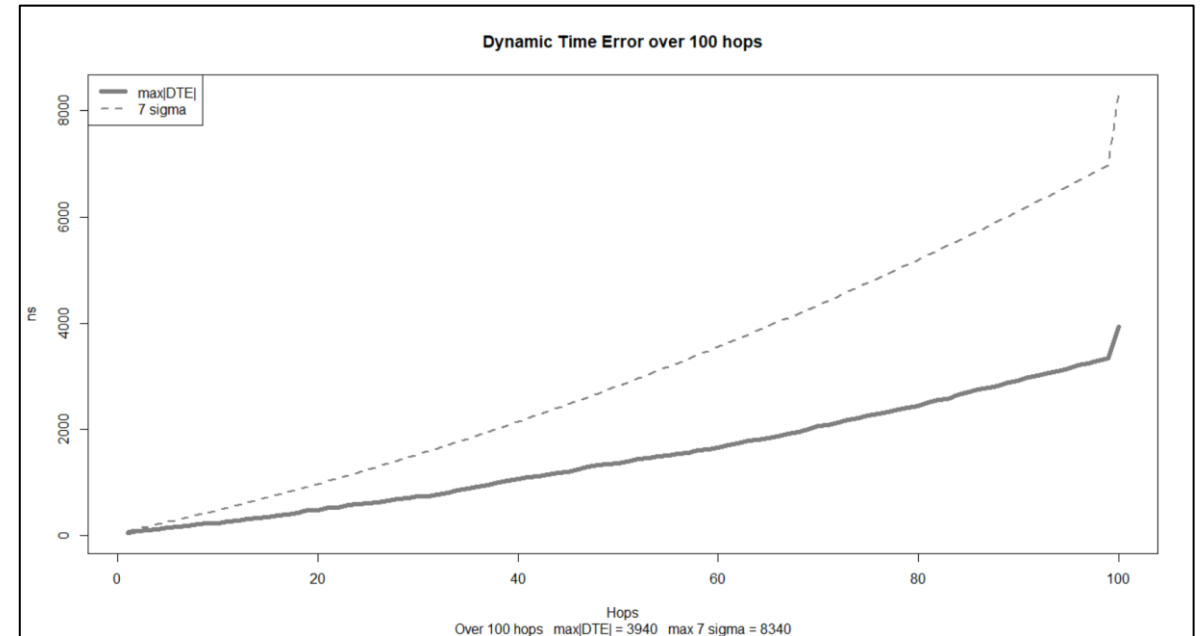
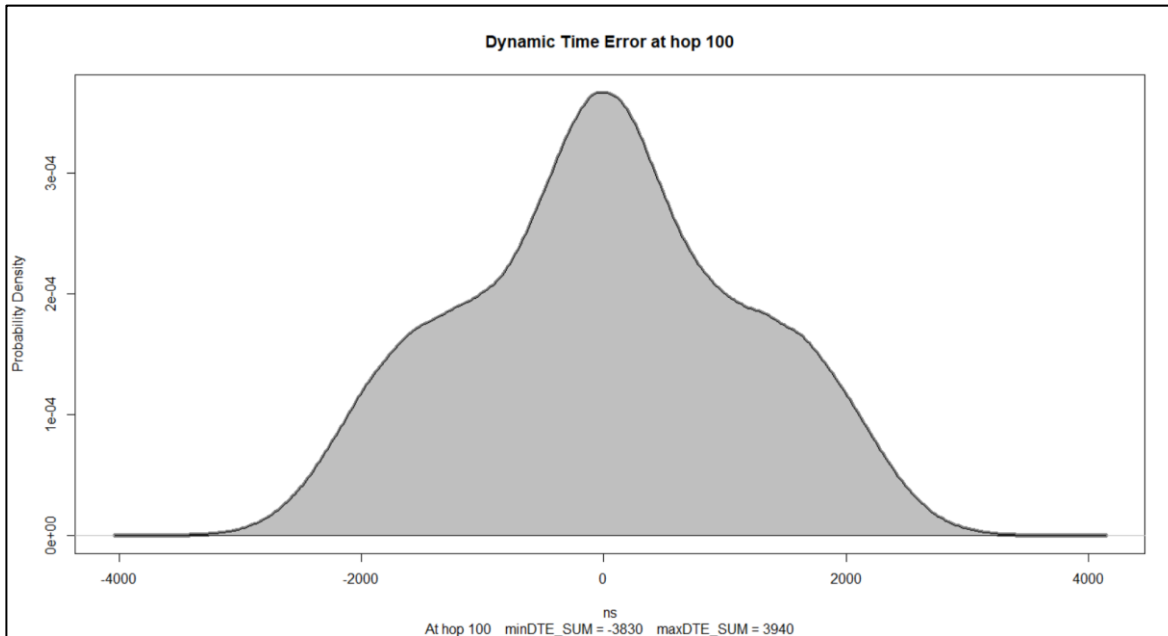
- Four Simulations
 - No Algorithmic Compensation
 - NRR Drift Correction
 - RR Drift Correction + NRR Drift Correction
 - **Recommended Settings:**
Mean Link Delay Averaging + RR Drift Correction + NRR Drift Correction
- Detail on next slides

No Algorithms

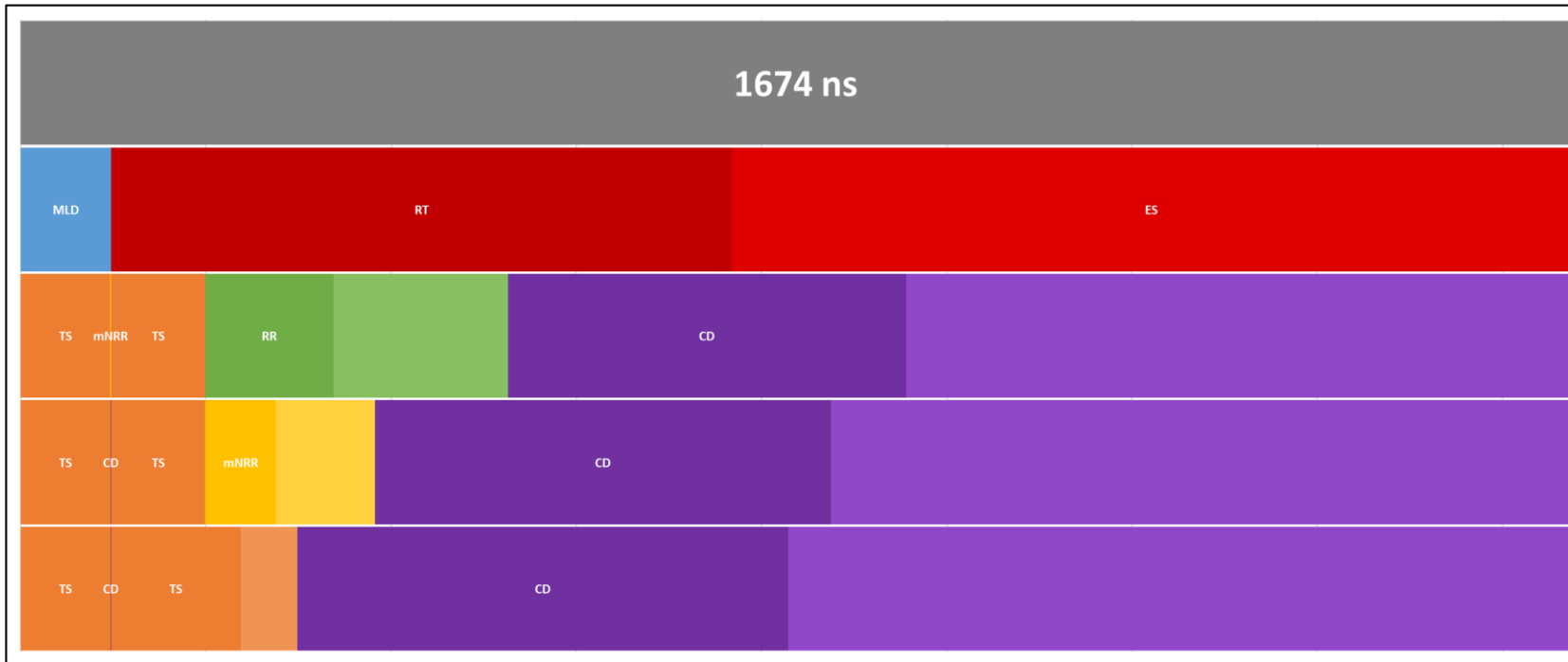


| Input Errors | | |
|-----------------------------|---------|-------|
| GM Clock Drift Max | +1.5 | ppm/s |
| GM Clock Drift Min | -1.5 | ppm/s |
| GM Nodes w/ Clock Drift | 80% | |
| Clock Drift Max (non-GM) | +1.5 | ppm/s |
| Clock Drift Min (non-GM) | -1.5 | ppm/s |
| Non-GM Nodes w/ Clock Drift | 80% | |
| Timestamp Granularity TX | 4 | ±ns |
| Timestamp Granularity RX | 4 | ±ns |
| Dynamic Time Stamp Error TX | 4 | ±ns |
| Dynamic Time Stamp Error RX | 4 | ±ns |
| Input Parameters | | |
| pDelay Interval | 125 | ms |
| Sync Interval | 125 | ms |
| pDelay Turnaround Time | 10 | ms |
| residenceTime | 10 | ms |
| Input Correction Factors | | |
| Mean Link Delay Averaging | 0 | % |
| NRR Drift Correction | 0 | % |
| RR Drift Correction | 0 | % |
| pDelayResponse → Sync | 0 | % |
| mNRR Smoothing N | 3 | |
| mNRR Smoothing M | 1 | |
| Configuration | | |
| Hops | 100 | |
| Runs | 100,000 | |

No Algorithms

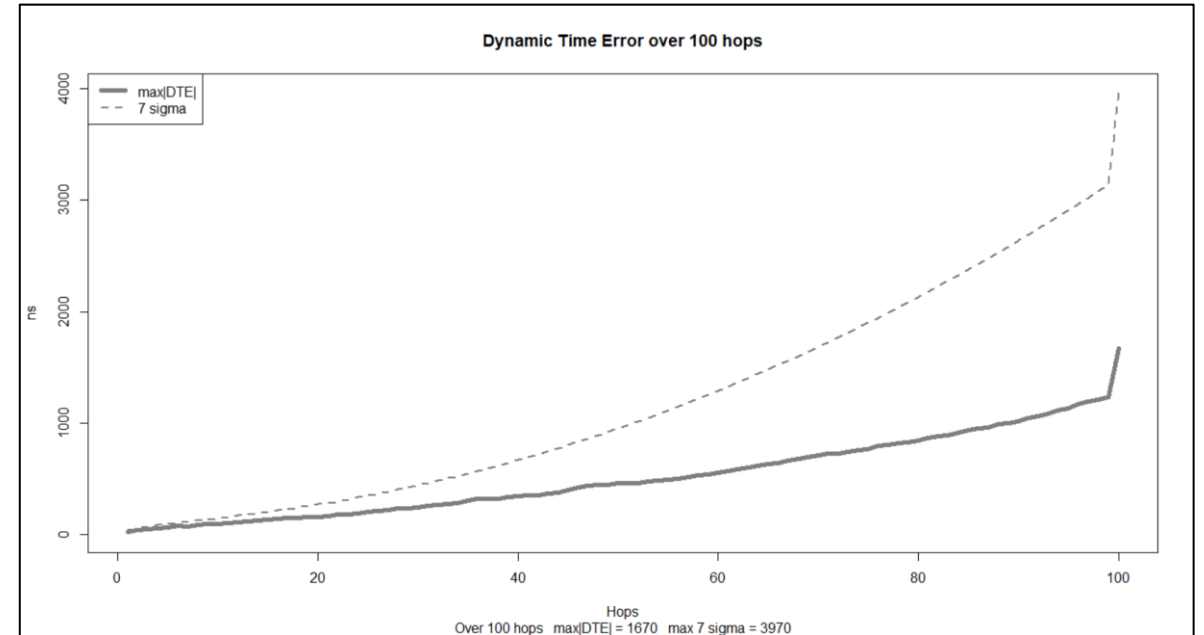
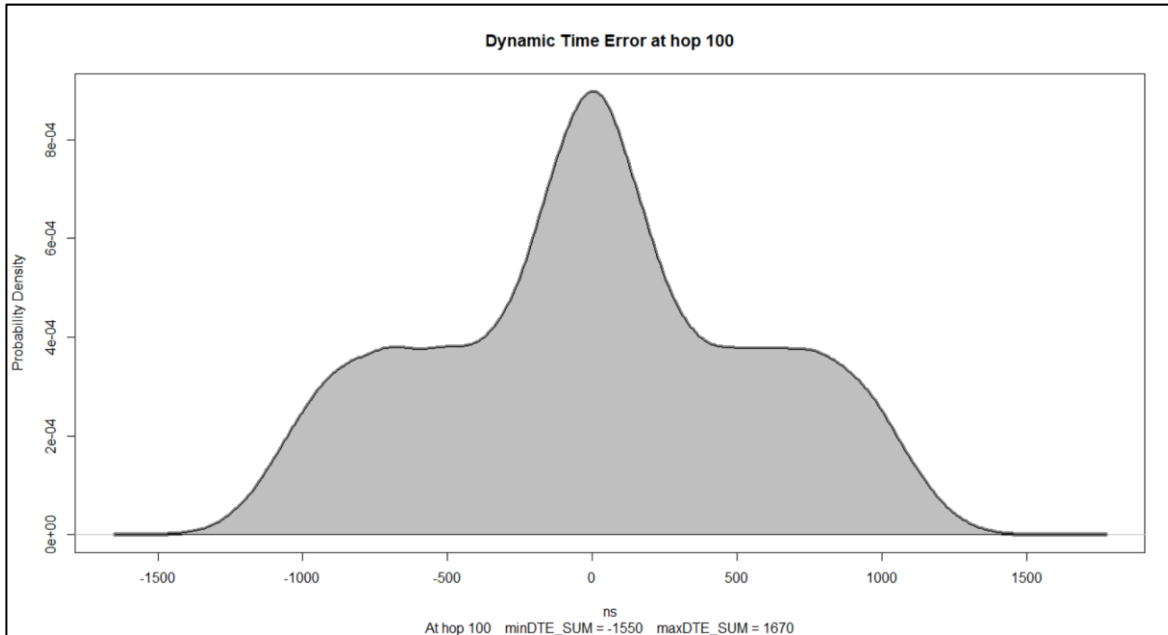


NRR Drift Correction

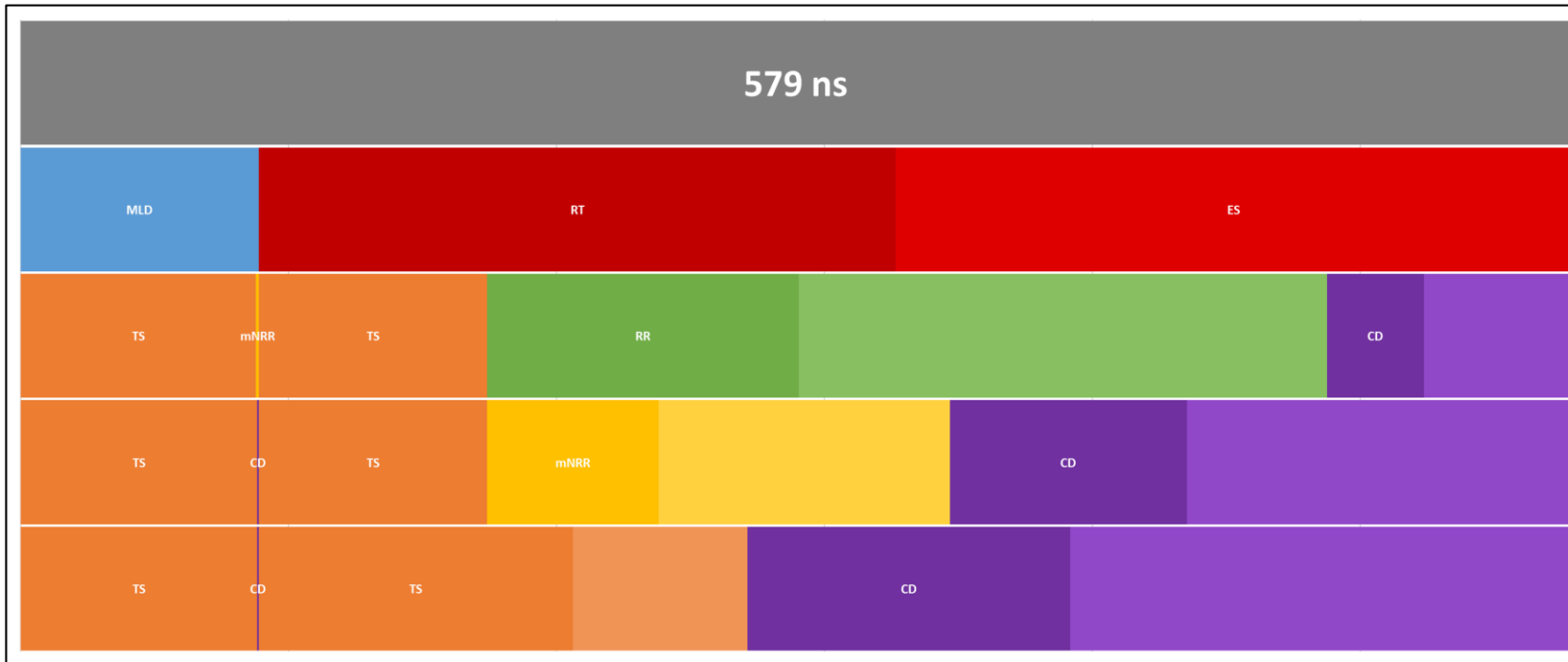


| Input Errors | | |
|-----------------------------|---------|-------|
| GM Clock Drift Max | +1.5 | ppm/s |
| GM Clock Drift Min | -1.5 | ppm/s |
| GM Nodes w/ Clock Drift | 80% | |
| Clock Drift Max (non-GM) | +1.5 | ppm/s |
| Clock Drift Min (non-GM) | -1.5 | ppm/s |
| Non-GM Nodes w/ Clock Drift | 80% | |
| Timestamp Granularity TX | 4 | ±ns |
| Timestamp Granularity RX | 4 | ±ns |
| Dynamic Time Stamp Error TX | 4 | ±ns |
| Dynamic Time Stamp Error RX | 4 | ±ns |
| Input Parameters | | |
| pDelay Interval | 125 | ms |
| Sync Interval | 125 | ms |
| pDelay Turnaround Time | 10 | ms |
| residenceTime | 10 | ms |
| Input Correction Factors | | |
| Mean Link Delay Averaging | 0 | % |
| NRR Drift Correction | 90 | % |
| RR Drift Correction | 0 | % |
| pDelayResponse → Sync | 0 | % |
| mNRR Smoothing N | 3 | |
| mNRR Smoothing M | 1 | |
| Configuration | | |
| Hops | 100 | |
| Runs | 100,000 | |

NRR Drift Correction

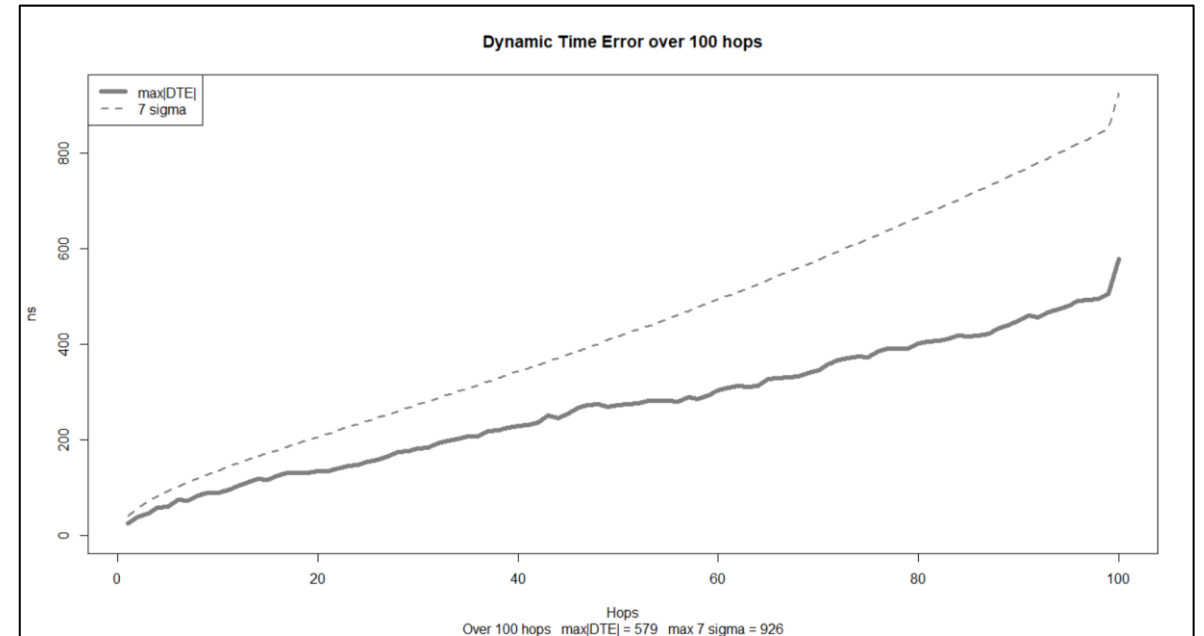
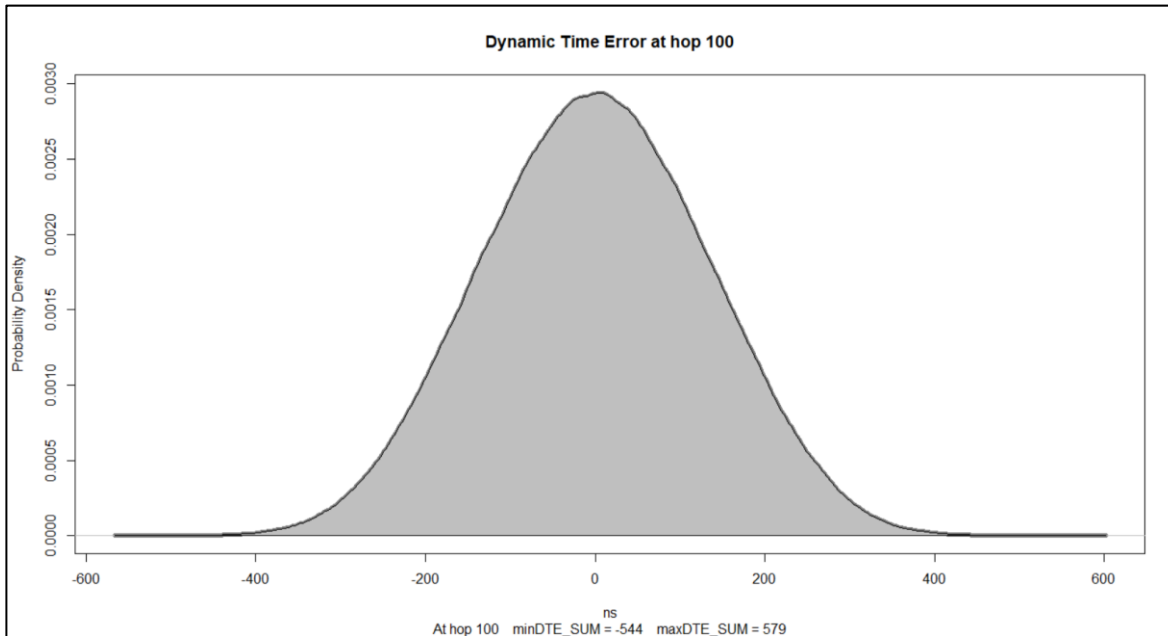


NRR & RR Drift Correction

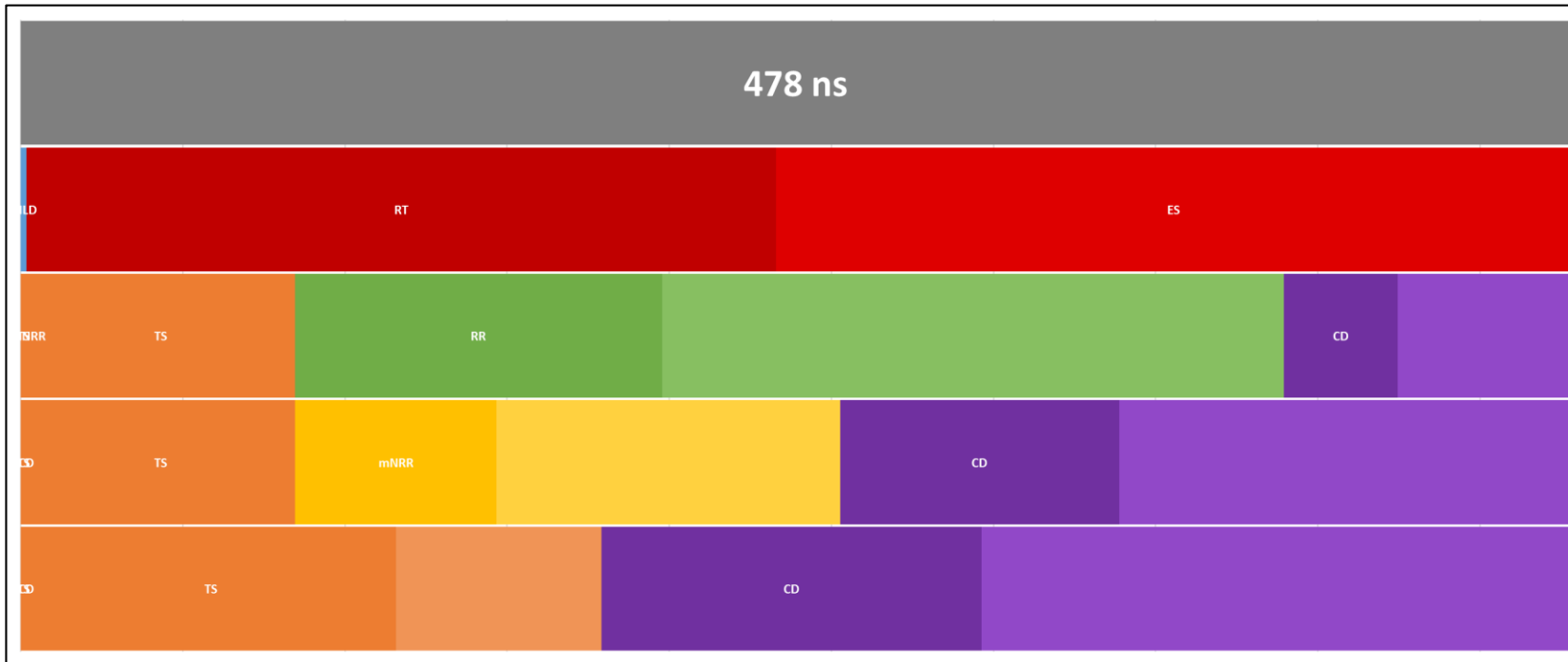


| Input Errors | | |
|--------------------------------|---------|-------|
| GM Clock Drift Max | +1.5 | ppm/s |
| GM Clock Drift Min | -1.5 | ppm/s |
| GM Nodes w/ Clock Drift | 80% | |
| Clock Drift Max (non-GM) | +1.5 | ppm/s |
| Clock Drift Min (non-GM) | -1.5 | ppm/s |
| Non-GM Nodes w/ Clock Drift | 80% | |
| Timestamp Granularity TX | 4 | ±ns |
| Timestamp Granularity RX | 4 | ±ns |
| Dynamic Time Stamp Error TX | 4 | ±ns |
| Dynamic Time Stamp Error RX | 4 | ±ns |
| Input Parameters | | |
| pDelay Interval | 125 | ms |
| Sync Interval | 125 | ms |
| pDelay Turnaround Time | 10 | ms |
| residenceTime | 10 | ms |
| Input Correction Factors | | |
| Mean Link Delay Averaging | 0 | % |
| NRR Drift Rate Correction | 90 | % |
| RR Drift Rate Error Correction | 90 | % |
| pDelayResponse → Sync | 0 | % |
| mNRR Smoothing N | 3 | |
| mNRR Smoothing M | 1 | |
| Configuration | | |
| Hops | 100 | |
| Runs | 100,000 | |

NRR & RR Drift Correction

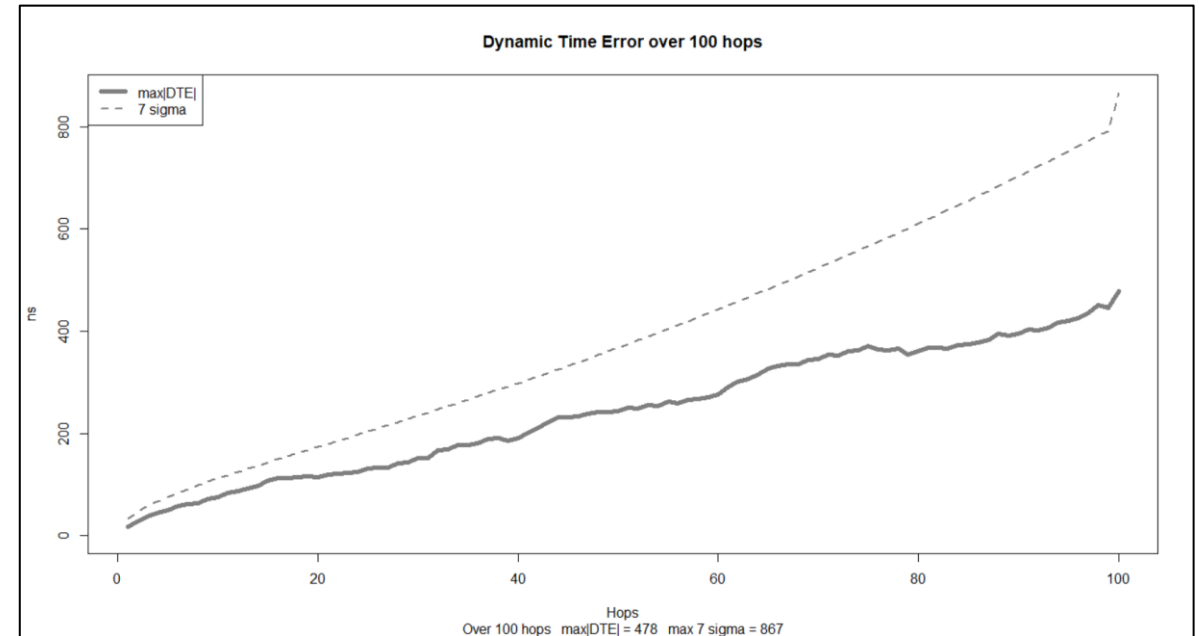
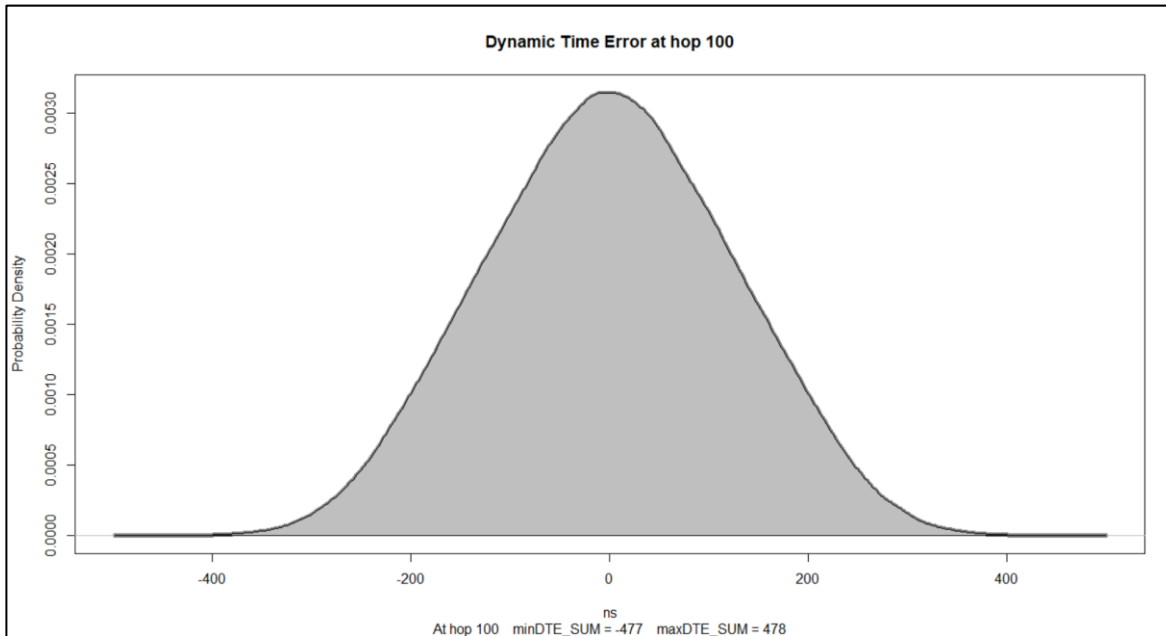


Recommended Settings: MLD Averaging + RR & NRR Drift Correction



| Input Errors | | |
|--------------------------------|---------|-------|
| GM Clock Drift Max | +1.5 | ppm/s |
| GM Clock Drift Min | -1.5 | ppm/s |
| GM Nodes w/ Clock Drift | 80% | |
| Clock Drift Max (non-GM) | +1.5 | ppm/s |
| Clock Drift Min (non-GM) | -1.5 | ppm/s |
| Non-GM Nodes w/ Clock Drift | 80% | |
| Timestamp Granularity TX | 4 | ±ns |
| Timestamp Granularity RX | 4 | ±ns |
| Dynamic Time Stamp Error TX | 4 | ±ns |
| Dynamic Time Stamp Error RX | 4 | ±ns |
| Input Parameters | | |
| pDelay Interval | 125 | ms |
| Sync Interval | 125 | ms |
| pDelay Turnaround Time | 10 | ms |
| residenceTime | 10 | ms |
| Input Correction Factors | | |
| Mean Link Delay Averaging | 98 | % |
| NRR Drift Rate Correction | 90 | % |
| RR Drift Rate Error Correction | 90 | % |
| pDelayResponse → Sync | 0 | % |
| mNRR Smoothing N | 3 | |
| mNRR Smoothing M | 1 | |
| Configuration | | |
| Hops | 100 | |
| Runs | 100,000 | |

Recommended Settings: MLD Averaging + RR & NRR Drift Correction



Normative & Informative Text

Normative Text

- Normative requirement on Local Clock drift relative to TAI: 1.5 ppm/s
 - Already in Günter's planned submission
- Normative requirement on “tolerance” of upstream drift of...
 - Local Clock of upstream node (NRR Drift Compensation)
 - GM (RR Drift Compensation)
- For tolerance...
 - Vary upstream Local Clock or GM with sinusoidal variation
 - ± 20 ppm with maximum slope of ± 1.5 ppm/s
 - No more than 10% of DTE error at individual node vs amount of error without algorithmic correction

Informative Text

- Description of algorithms that allow the requirements to be met.
- Based on prior contribution.
- Is this needed for draft 1.4?

Proposed Algorithms

Conventions

- $p-1$ is back in time
 - $mNRR(p)$ is the most recent $mNRR$ calculation
 - $mNRR(p-1)$ is the $mNRR$ calculation one prior (in time) to the most recent
- $n-1$ is back up the chain of nodes
 - $RR(n)$ is the RR calculation for the current node's Local Clock (against the GM)
 - $RR(n-1)$ is the RR same calculation for the node one step up the chain
- If n and/or p are omitted the value is the most recent for the current node

Note

- Two sets of equations are presented for NRR and RR Drift Correction
 - “a” assumes RR and NRR are represented as ratios that are multiplied together
 - “b” assumes RR and NRR are represented as ppm values and added together
 - This introduces an error, but if ppm values are small (one or two digits) the error is similarly small and can be ignored in many practical applications

NRR Drift Correction – Time Series – 1a Ratio

Correction factor applied to RR to account for clock drift between the Local Clock and the Local Clock (or GM) in the previous node (i.e. NRR) during time between NRR measurement and Sync message

- Assumes clock drifts linearly over the period of interest.
- Calculations based on Local Clock timing; mNRRsmoothingN = 3
- Calculate Drift Rate (mNRR as Ratio)

$$NRR_{driftRate}(n) = \frac{1}{Time_{effectiveNRRmeasure}(p) - Time_{effectiveNRRmeasure}(p-1)} \times \frac{mNRR(p)}{mNRR(p-1)} \quad \text{ratio/s}$$

$$= \frac{1}{\left(\frac{t_{4pDelayResp}(p) + t_{4DelayResp}(p-3)}{2}\right) - \left(\frac{t_{4DelayResp}(p-1) + t_{4DelayResp}(p-4)}{2}\right)} \times \frac{mNRR(p)}{mNRR(p-1)} \quad \text{ratio/s}$$

$$= \frac{2 \times (mNRR(p))}{mNRR(p-1) \times (t_{4DelayResp}(p) - t_{4DelayResp}(p-1) + t_{4DelayResp}(p-3) - t_{4DelayResp}(p-4))} \quad \text{ratio/s}$$

NRR Drift Correction – Time Series – 2a Ratio

Correction factor applied to RR to account for clock drift between the Local Clock and the Local Clock (or GM) in the previous node (i.e. NRR) during time between NRR measurement and Sync message

- Where previously

$$RR(n) = RR(n - 1) \times mNRR(n) \quad \text{ratio}$$

- Now

$$RR(n) = RR(n - 1) \times mNRR(p) \times \left(NRR_{driftRate}(n) \cdot \left(t_{1syncOut}(n) - Time_{effectiveNRRmeasure}(p) \right) \right) \quad \text{ratio}$$

$$= RR(n - 1) \times mNRR(p) \times \left(NRR_{driftRate}(n) \cdot \left(t_{1syncOut}(n) - \left(\frac{t_{4pDelayResp}(p) + t_{4DelayResp}(p - 3)}{2} \right) \right) \right) \quad \text{ratio}$$

RR Drift Correction – Sync Messaging – Time Series – 1a Ratio

Correction factor applied to Correction Field during processing of Sync message to account for drift between Local Clock & GM during time from GM's transmit of initial Sync message

- Applied during processing of Correction Field at all nodes.
 - Residence Time and Mean Link delay for Bridges; Mean Link Delay only for End Stations.
 - Note: where a device functions as both Bridge and End Station, there are two version of the Correction Field; one for local use (MLD only); one for transmitted Sync messaging (MLD & RT).
- Assumes clock drifts linearly over the period of interest.
- Calculations based on Local Clock timing (apart from correctionField)
- Calculate Drift Rate

$$RR_{driftRate}(n) = \frac{1}{t_{1syncOut}(p) - t_{1syncOut}(p-1)} \times \frac{RR(p)}{RR(p-1)}$$

ratio

- $t_{1syncOut}(p)$ is the timestamp for when the current node (n) transmits Sync to the next node in the chain.
- $RR(p)$ is the Rate Ratio calculated when Sync is transmitted

RR Drift Correction – Sync Messaging – Time Series – 2a Ratio

Correction factor applied to Correction Field during processing of Sync message to account for drift between Local Clock & GM during time from GM's transmit of initial Sync message

- Where previously

$$\text{correctionField}(n) = \text{correctionField}(n - 1) + RR(n) \cdot (\text{residenceTime} + \text{meanLinkDelay}) \quad \mathbf{ns}$$

$$\text{correctionField}(n) = \text{correctionField}(n - 1) + RR(n) \cdot RR_{\text{driftCorrection}} \cdot (\text{residenceTime} + \text{meanLinkDelay}) \quad \mathbf{ns}$$

$$RR_{\text{driftCorrection}} = RR_{\text{driftRate}}(n) \times \left(\frac{\text{correctionField}(n - 1)}{RR(n)} + \text{residenceTime} + \text{meanLinkDelay} \right) \quad \mathbf{ppm}$$

- correctionField is in terms of GM Clock
- residenceTime is only applied to Correction Field when generating Sync messaging for TX to next node in the chain

RR Drift Correction – ES – Time Series – 1a Ratio

Correction factor applied to applied RR at End Station during time between arrival of Sync messages

- Applied at End Stations between arrival of Sync Messages.
- Assumes clock drifts linearly over the period of interest.
- Calculations based on Local Clock timing.
- Calculate Drift Rate

$$RR_{driftRate}(n) = \frac{1}{t_{2syncIn}(p) - t_{2syncIn}(p-1)} \times \frac{RR(p)}{RR(p-1)} \quad \text{ratio}$$

- $t_{2syncIn}(p)$ is the timestamp for when the current node (n) receives Sync to the previous node in the chain.
- $RR(p)$ is the Rate Ratio calculated when Sync is transmitted

RR Drift Correction – ES – Time Series – 2a Ratio

Correction factor applied to applied RR at End Station during time between arrival of Sync messages

- Where previously...

$$RR_{applied} = RR(n)$$

Ratio

...and remained constant until next Sync message arrives

- Now...

$$RR_{applied} = RR(n) \cdot RR_{driftRate}(n) \cdot \left(\frac{correctionField(n-1)}{RR(n)} + timeElapsedSinceSync \right)$$

Ratio

...and therefore constantly changing, albeit linearly.

- Application to Time Series will involve quadratic equations

NRR Drift Correction – Time Series – 1b ppm

Correction factor applied to RR to account for clock drift between the Local Clock and the Local Clock (or GM) in the previous node (i.e. NRR) during time between NRR measurement and Sync message

- Assumes clock drifts linearly over the period of interest.
- Calculations based on Local Clock timing; mNRRsmoothingN = 3
- Calculate Drift Rate (mNRR as Ratio)

$$NRR_{driftRate}(n) = \frac{(mNRR(p) - mNRR(p - 1))}{Time_{effectiveNRRmeasure}(p) - Time_{effectiveNRRmeasure}(p - 1)} \quad \text{ppm/s}$$

$$= \frac{(mNRR(p) - mNRR(p - 1))}{\left(\frac{t_{4pDelayResp}(p) + t_{4DelayResp}(p - 3)}{2}\right) - \left(\frac{t_{4DelayResp}(p - 1) + t_{4DelayResp}(p - 4)}{2}\right)} \quad \text{ppm/s}$$

$$= \frac{2 \times (mNRR(p) - mNRR(p - 1))}{(t_{4DelayResp}(p) - t_{4DelayResp}(p - 1) + t_{4DelayResp}(p - 3) - t_{4DelayResp}(p - 4))} \quad \text{ppm/s}$$

NRR Drift Correction – Time Series – 2b ppm

Correction factor applied to RR to account for clock drift between the Local Clock and the Local Clock (or GM) in the previous node (i.e. NRR) during time between NRR measurement and Sync message

- Where previously

$$RR(n) = RR(n - 1) + mNRR(n) \quad \text{ppm}$$

- Now

$$RR(n) = RR(n - 1) + mNRR(p) + \left(NRR_{driftRate}(n) \cdot \left(t_{1syncOut}(n) - Time_{effectiveNRRmeasure}(p) \right) \right) \quad \text{ppm}$$

$$= RR(n - 1) + mNRR(p) + \left(NRR_{driftRate}(n) \cdot \left(t_{1syncOut}(n) - \left(\frac{t_{4pDelayResp}(p) + t_{4DelayResp}(p - 3)}{2} \right) \right) \right) \quad \text{ppm}$$

RR Drift Correction – Sync Messaging – Time Series – 1b ppm

Correction factor applied to Correction Field during processing of Sync message to account for drift between Local Clock & GM during time from GM's transmit of initial Sync message

- Applied during processing of Correction Field at all nodes.
 - Residence Time and Mean Link delay for Bridges; Mean Link Delay only for End Stations.
 - Note: where a device functions as both Bridge and End Station, there are two version of the Correction Field; one for local use (MLD only); one for transmitted Sync messaging (MLD & RT).
- Assumes clock drifts linearly over the period of interest.
- Calculations based on Local Clock timing (apart from correctionField)
- Calculate Drift Rate

$$RR_{driftRate}(n) = \frac{RR(p) - RR(p - 1)}{t_{1syncOut}(p) - t_{1syncOut}(p - 1)}$$

ppm

- $t_{1syncOut}(p)$ is the timestamp for when the current node (n) transmits Sync to the next node in the chain.
- $RR(p)$ is the Rate Ratio calculated when Sync is transmitted

RR Drift Correction – Sync Messaging – Time Series – 2b ppm

Correction factor applied to Correction Field during processing of Sync message to account for drift between Local Clock & GM during time from GM's transmit of initial Sync message

- Where previously

$$correctionField(n) = correctionField(n - 1) + \left(1 + \frac{RR(n)}{10^6}\right) \cdot (residenceTime + meanLinkDelay) \quad \mathbf{ns}$$

- Now

$$correctionField(n) = correctionField(n - 1) + \left(1 + \frac{RR(n) + RR_{driftCorrection}}{10^6}\right) \cdot (residenceTime + meanLinkDelay) \quad \mathbf{ns}$$

$$RR_{driftCorrection} = RR_{driftRate}(n) \times \left(\frac{correctionField(n - 1)}{RR(n)} + residenceTime + meanLinkDelay\right) \quad \mathbf{ppm}$$

- correctionField is in terms of GM Clock

RR Drift Correction – ES – Time Series – 1a Ratio

Correction factor applied to applied RR at End Station during time between arrival of Sync messages

- Applied at End Stations between arrival of Sync Messages.
- Assumes clock drifts linearly over the period of interest.
- Calculations based on Local Clock timing.
- Calculate Drift Rate

$$RR_{driftRate}(n) = \frac{RR(p) - RR(p - 1)}{t_{2syncIn}(p) - t_{2syncIn}(p - 1)}$$

ppm

- $t_{2syncIn}(p)$ is the timestamp for when the current node (n) receives Sync to the previous node in the chain.
- $RR(p)$ is the Rate Ratio calculated when Sync is transmitted

RR Drift Correction – ES – Time Series – 2b ppm

Correction factor applied to applied RR at End Station during time between arrival of Sync messages

- Where previously...

$$RR_{applied} = RR(n)$$

ppm

...and remained constant until next Sync message arrives

- Now...

$$RR_{applied} = RR(n) + RR_{driftRate}(n) \cdot \left(\frac{correctionField(n-1)}{RR(n)} + timeElapsedSinceSync \right)$$

ppm

...and therefore constantly changing, albeit linearly.

- Application to Time Series will involve quadratic equations

Mean Link Delay Averaging

- Wired connection link delay is very stable
- pDelay measurements can be noisy due to Timestamp Errors
- It should be possible to average out errors over time
 - Low bandwidth IIR filter...but need to be careful about start-up behaviour

Mean Link Delay Averaging – Possible Algorithm

- For p^{th} pDelay measurement since initialisation...

if $p \leq 1000, F = X$

$$MeanLinkDelay(1) = pDelay(1)$$

if $p > 1000, F = 1000$

$$MeanLinkDelay(p) = \frac{(MeanLinkDelay(p - 1) \times (F - 1)) + pDelay(X)}{F}$$

- So, for example...

$$MeanLinkDelay(100) = \frac{(MeanLinkDelay(99) \times (99)) + pDelay(100)}{100}$$

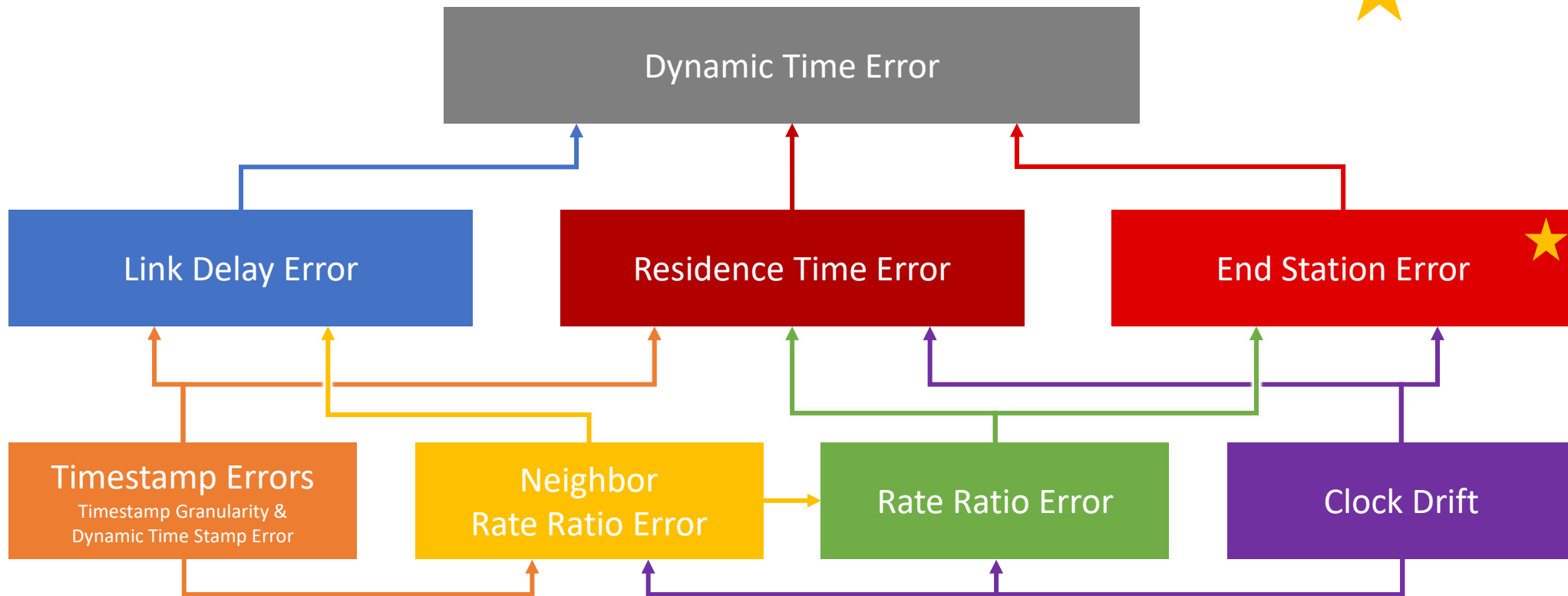
$$MeanLinkDelay(10500) = \frac{(MeanLinkDelay(10499) \times (999)) + pDelay(10500)}{1000}$$

- Reset F if pDelay deviates too much from current MeanLinkDelay?
 - Deviates too much...repeatedly?

Thank You

Time Sync – Errors to Model

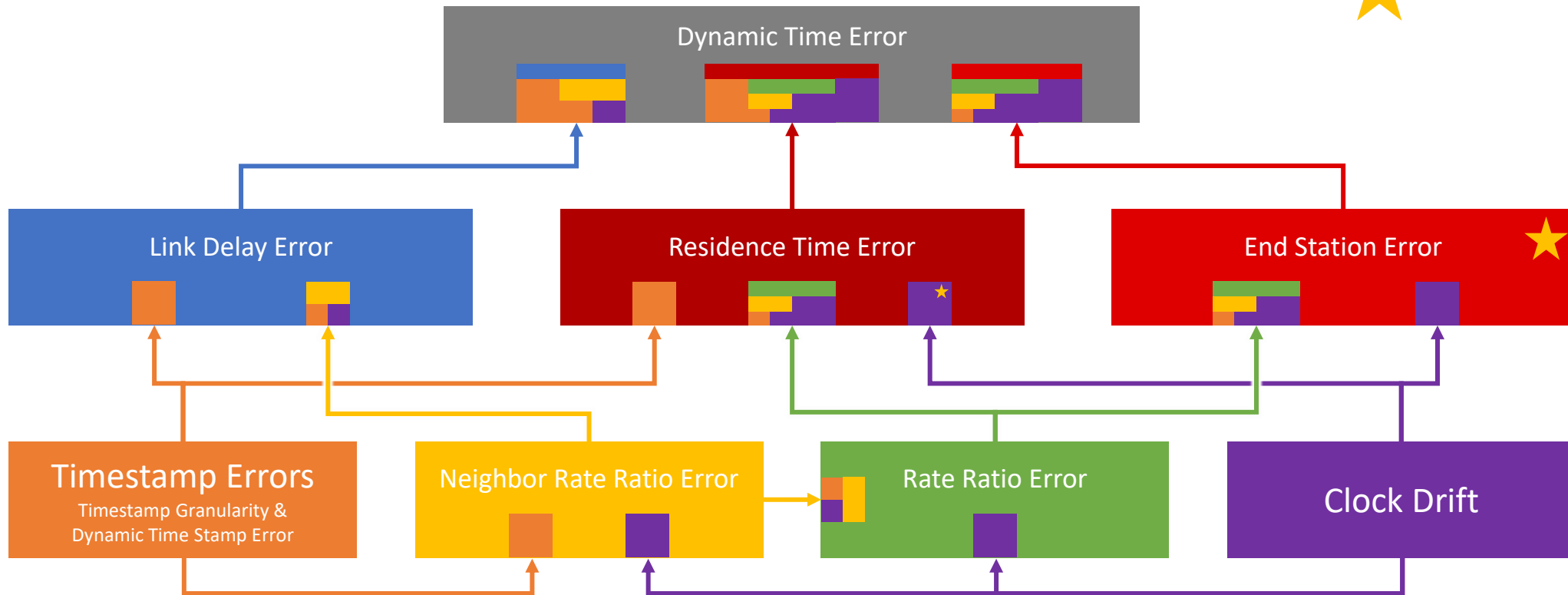
 = Addition since Nov 2021



All errors in this analysis are caused by either **Clock Drift or **Timestamp Errors****

Time Sync – How Errors Add Up

★ = Addition since Nov 2021



All errors in this analysis are caused by either **Clock Drift** or **Timestamp Errors**