
Global Time and Clock Synchronization in Networks



"Milestone" papers concerning clock synchronization:

L. Lamport and P. M. Melliar-Smith. Synchronizing clocks in the presence of faults. *Journal of the ACM*, 32(1):5278, July 1985

F. Cristian, H. Aghili, and R. Strong: Clock synchronization in the presence of omission and performance failures, and processor joins. In *Proc. of 16th International Symposium on Fault-Tolerant Computing Systems*, July 1986

H. Kopetz and W. Ochsenreiter: Clock synchronization in distributed real-time computer systems. *IEEE Transactions on Computers*, C-36(8):933940, August 1987.

D. L. Mills. Internet time synchronization: The network time protocol. *IEEE Transactions on Computers*, 39(10):14821493, October 1991

P. Verissimo, A. Casimiro, and L. Rodrigues. Cesiumspray: a precise and accurate global time service for large scale systems. *Journal of Real-Time Systems*, 12:243294, 1997

Good survey paper:

Emmanuelle Anceaume and Isabelle Puaut: Performance Evaluation of Clock Synchronization Algorithms, *Rapport de recherche N ° 3526*, Octobre 1998
ISSN 0249-6399



What is the function of time ?

Orientation:

Determine the position of an event in the continuum of time.

Regulation

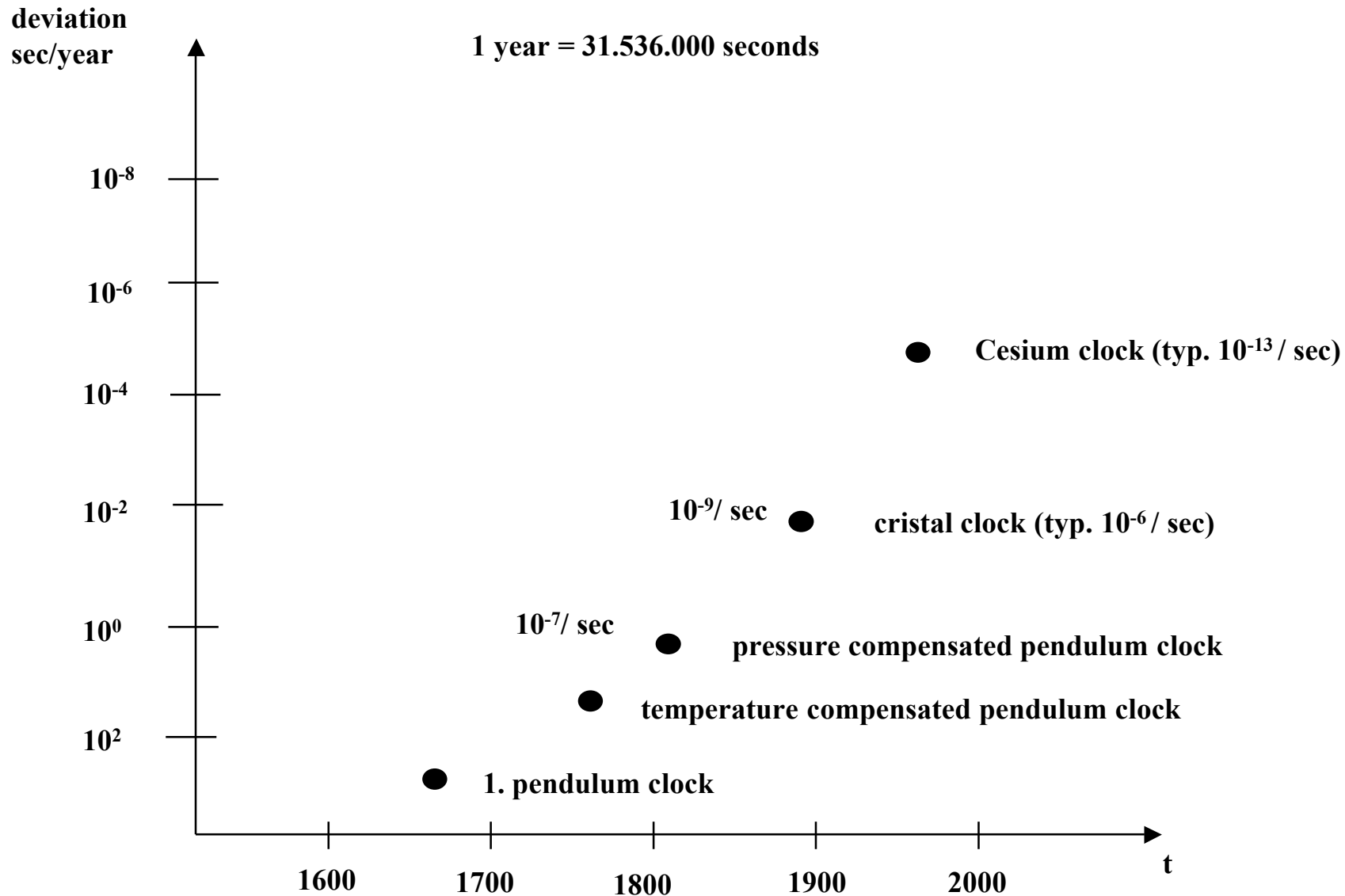
Enforcing a time regime.

Coordination

Synchronizing cooperative tasks.



Advances in Time Measurement





The Cesium-clock of the "Physikalisch-Technischen Bundesanstalt" at Braunschweig

D. Lehmann: "Ohne Uhren keine Zeit", in: Geo - Das neue Gesicht der Erde,
Nr.12, Dezember 1995



AOSI
IVS-EOS

Sommersemester 2008

5

J. Kaiser, IVS-EOS

Time standards

<p>■ Astronomische Zeit (AT) Physikalische Zeit (PT)</p>	<p>basiert auf der gleichförmigen Bewegung von Himmelskörpern basiert auf periodischen physikalischen Prozessen (Schwingungen)</p>
<p>Sonnendurchlauf: Sonnentag: Sonnensekunde: Mittlere Sonnensekunde:</p>	<p>Höchster Punkt der Sonne während eines Tages Intervall zwischen zwei aufeinanderfolgenden Sonnendurchläufen Der 1 / 86400 Teil eines Sonnentages Sonnensekunde gemittelt über eine große Anzahl von Sonnentagen Erster Zeitstandard (1820) basiert auf mittlerer Sonnensekunde</p>
<p>Zeitzone:</p>	<p>Gebiete für die dieselbe Zeit festgelegt ist. 1884 wird die Welt in 24 Zeitzone aufgeteilt. Die Zeitzone unterscheiden sich von UT (GMT) ganzzahlig um jeweils 1 Stunde.</p>
<p>UT (AT, 1833)</p>	<p>Universal Time (UT) Mittlere Sonnenzeit, gemessen am Greenwich 0-Meridian (GMT). Basiert auf der mittleren Länge eines Sonnentags, d.h. auf der Erdrotation</p>
<p>ET (AT, 1955)</p>	<p>Ephemeridenzeit (ET), basiert auf der Umlaufzeit der Erde um die Sonne. Harold Spencer Jones stellte 1939 fest, daß die Rotation der Erde variiert, die Umlaufzeit um die Sonne nicht. 1 Sekunde der ET wird festgelegt als der 1/31.566.925,9747 Teil des tropische Jahres, das am Mittag des 1. Januars 1900 begann. (Tropische Jahr: Periode zwischen zwei aufeinanderfolgenden Durchläufen der Sonne durch den Himmelsäquator in derselben Richtung.)</p>
<p>UT0 (AT, 1960)</p>	<p>Zeit, basierend auf den lokalen Beobachtungen verschiedener, über die Erde verteilter Observatorien.</p>
<p>UT1 (AT, 1960)</p>	<p>Zeit, basierend auf der Koordination der verschiedenen UT0-Zeiten (Mittlung).</p>
<p>UT2 (AT, 1960)</p>	<p>Nochmals, auf empirischer Basis korrigierte UT1-Zeit.</p>
<p>TAI (PT, 1961)</p>	<p>Temps Atomique International (TAI) basiert auf mehreren koordinierten Cäsium-Uhren. Fortlaufende Zeitzählung, beginnend mit dem 1. Januar 1958 0 Uhr UT2-Zeit (daher konsistent mit UT2).</p>
<p>UTC (PT, 1972)</p>	<p>Universal Time Coordinated (UTC) basiert auf TAI, wird aber ständig an UT2 angepaßt. Immer wenn UTC und UT2 mehr als 800 ms auseinander gedrifted sind, wird eine "Schaltsekunde" eingefügt. UTC beginnt am 1. Januar 1972. Seit dieser Zeit sind (bis 1992) 15 Schaltsekunden eingefügt worden. UTC ist damit eine an AT angepaßte physikalische Zeit. Genauigkeit: ca. 1 Sek / 300000 Jahre</p>



World wide, standardized time reference : *Universal Time Coordinated (UTC)*

distributed by:

- **land-based radio stations (0,1-10 msek)**
- **GPS - *Global Positioning System***
- **GEOS - *Geostationary Environmental Operational Satellites***

Time base in distributed systems:

- **a (single) UTC-based reference clock**
- **local clocks**
- **algorithms for clock synchronization**

Correspondence to perfect time is dependent on:

- **precision of the time signal**
- **distance to sender**
- **atmospheric conditions**
- **drift of local clocks**
- **predictability of network latencies (steadyness and tightness)**
- **the synchronization algorithm**



Physical model of time

Assumption: Existence of a uniform sequence of events suitable to represent a time reference

Properties:

Equivalence: All time stamps are equivalent in the sense that there is a linear formule converting a timestamp into another one.

Linearity: The conversion formulas are linear.
Different timestamps can be related by a proportional factor.

**Causality:
(potential)** Time is aligned according to an order of events which are related in a before/after relationship.

Reversal: Physical processes in general cannot be reversed.

Homogeneity: Time flow is uniform and does not contain holes.



Relation between internal and external time

To measure the occurrence of events in a system in the metric of the external physical time (e.g. TAI), there must be an internal representation of the physical time.

Internal physical time: Internal time reference that approximates the external physical time. All events are ordered in the global flow of time. The before/after relation between events is determined according to the internal time. Causally independent events are also temporally ordered. Given two events e_1 and e_2 at time t_{e1} and t_{e2} respectively. If on an internal time scale $t_{e1} \leq t_{e2}$ holds, this must also hold on an external time scale.

Internal synchronization: Synchronization of internal clocks .

External Synchronization: Synchronization to the external physical time.



Physical Clocks

Physical clock: Time reference that is based on periodic physical processes like oscillations of a pendulum, a crystal or an atom to measure the progression of time.

Characterization of physical clocks:

Frequency/Rate: f Number of periodic events per time frame. (oscillations, clock pulses)
Metric: Hz, KHz, MHz, GHz, . . .

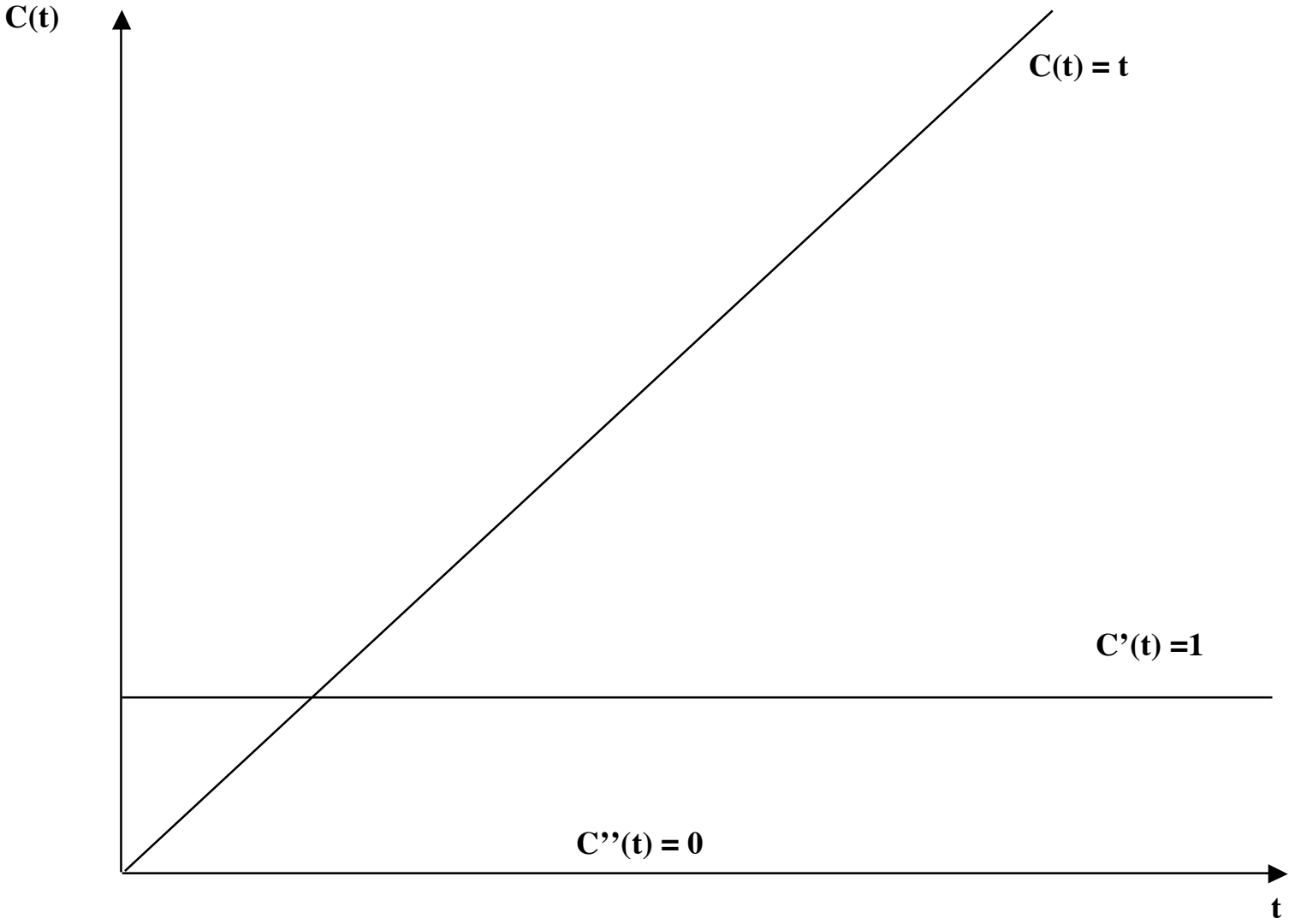
Period $p=1/f$ Time interval between two consecutive periodic events.
Metric: sec, ms, μ s, ns . .

Granularity: g time interval between two clock pulses (corresponds to a period).
Can only be determined by a clock of higher granularity. The granularity determines the lower bound of the temporal distance between two events that are not recognized as simultaneous events.

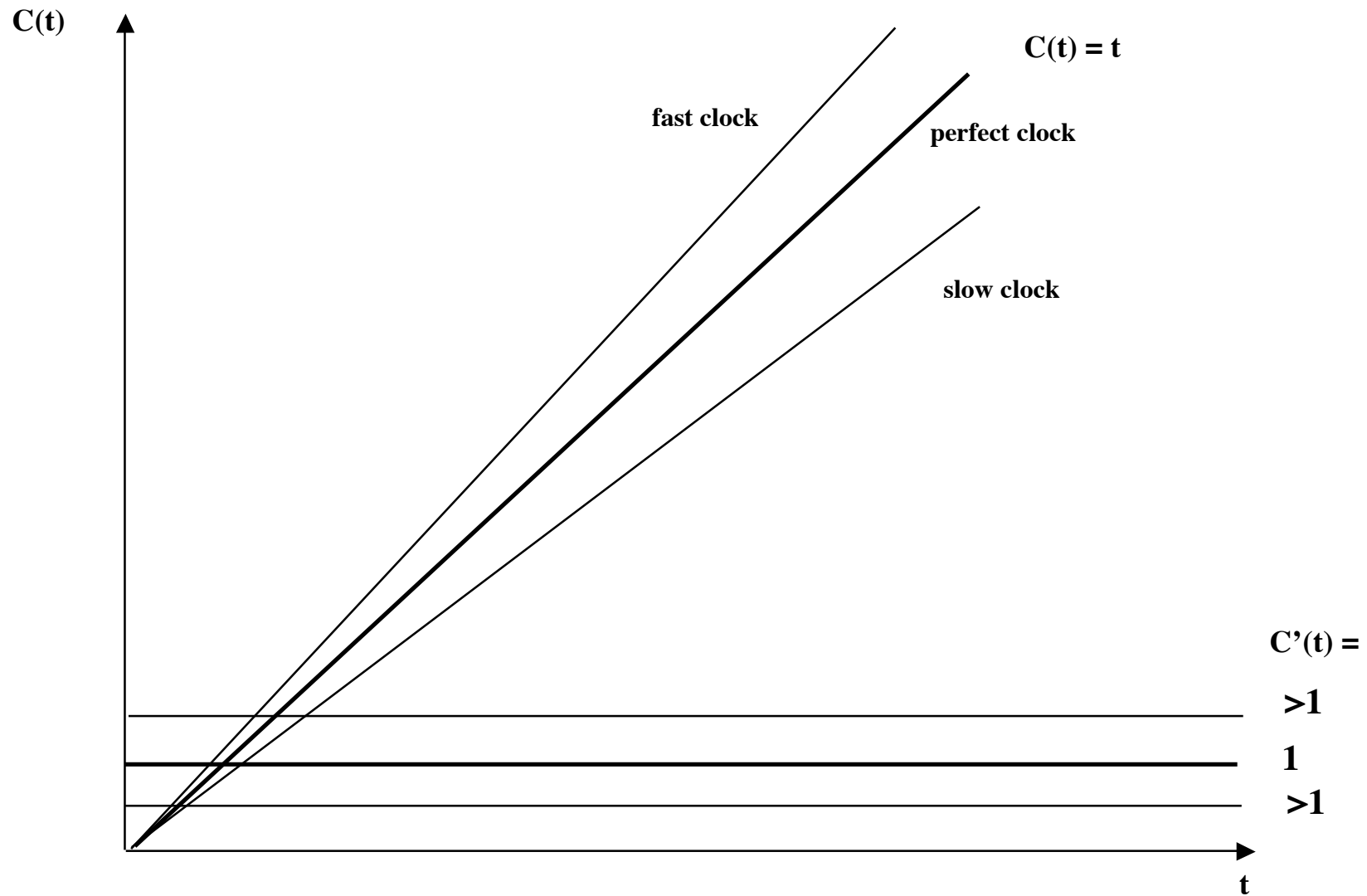
Metric: sec, ms, μ s, ns . . .



The perfect Clock



Clock with constant Drift Rate



Terminology

Let $C(t)$ be the function that maps the time instance t of the reference time to the internal clock.

Def. 1:

a clock is a reference clock or perfect clock if : $\forall t: C(t) = t$

Def. 2:

a clock is correct at time t_0 , if $C(t_0) = t_0$ is satisfied.

Def. 3:

a clock rate is perfect at t_0 , if at t_0 : $C'(t) = 1$ ($dC(t) / dt = 1$)

$C'(t)$ denotes the (rate) stability of a clock.

We consider 3 cases:

$dC/dt > 1$: *fast clock*

$dC/dt = 1$: *perfect clock*,

$dC/dt < 1$: *slow clock*

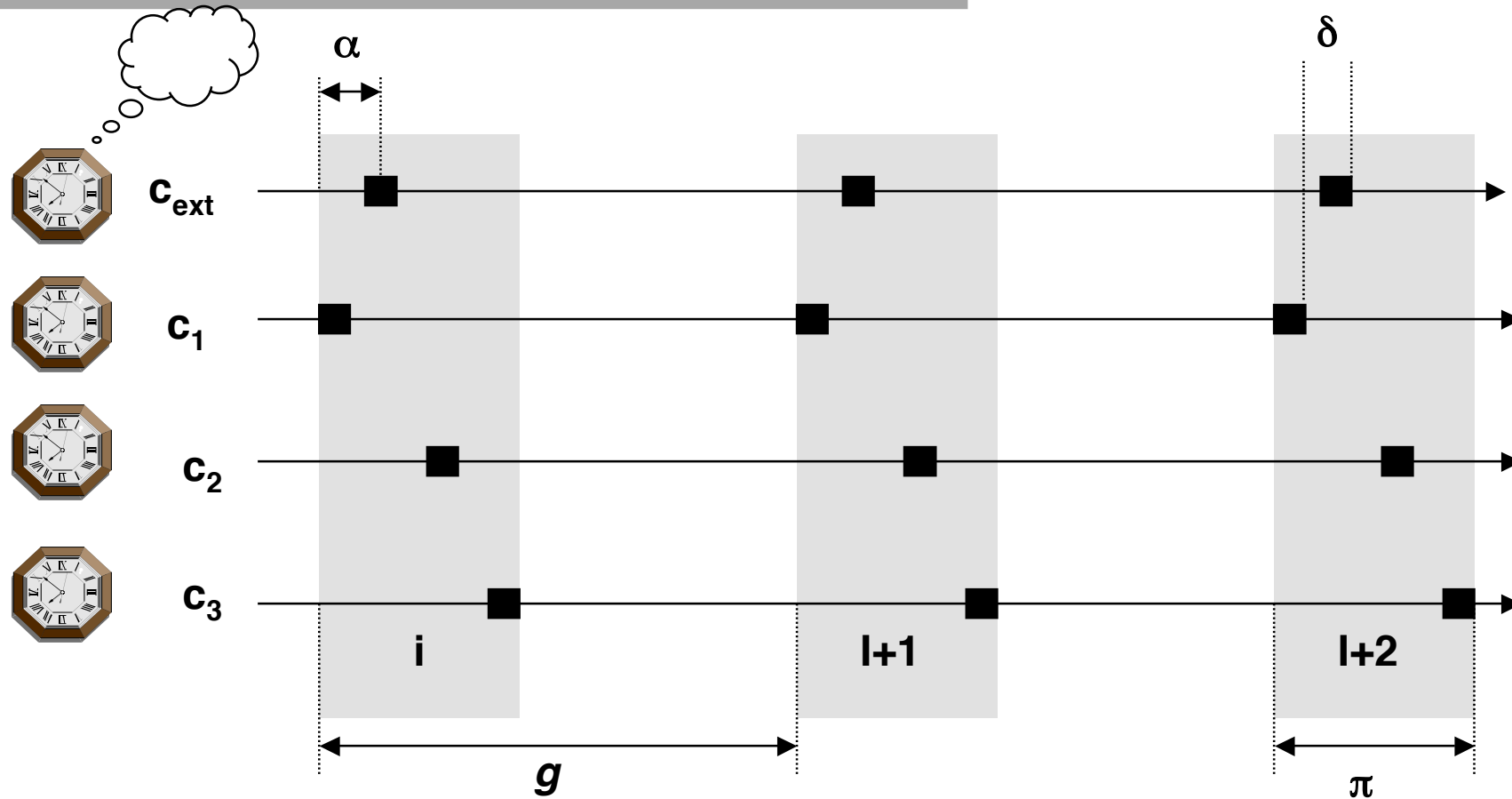
Drift(rate) r : deviation of the clock/time

The drift is assumed to be bounded by:

$$1-\rho \leq dC/dt \leq 1+\rho$$



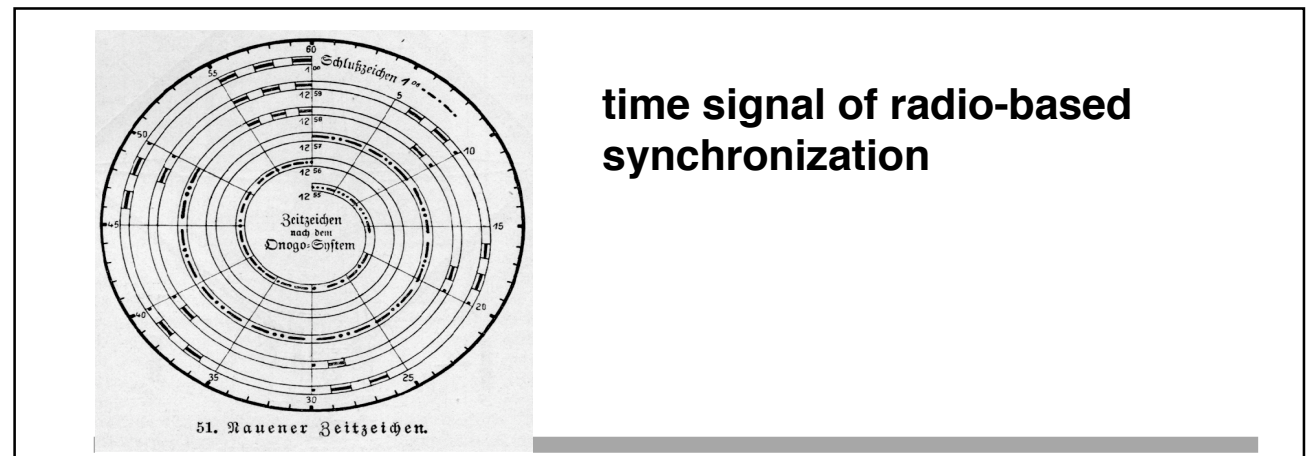
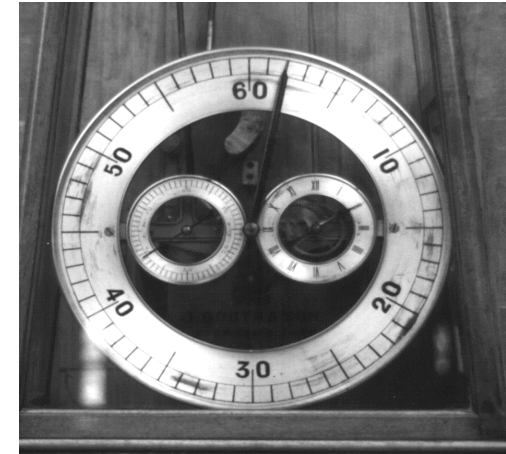
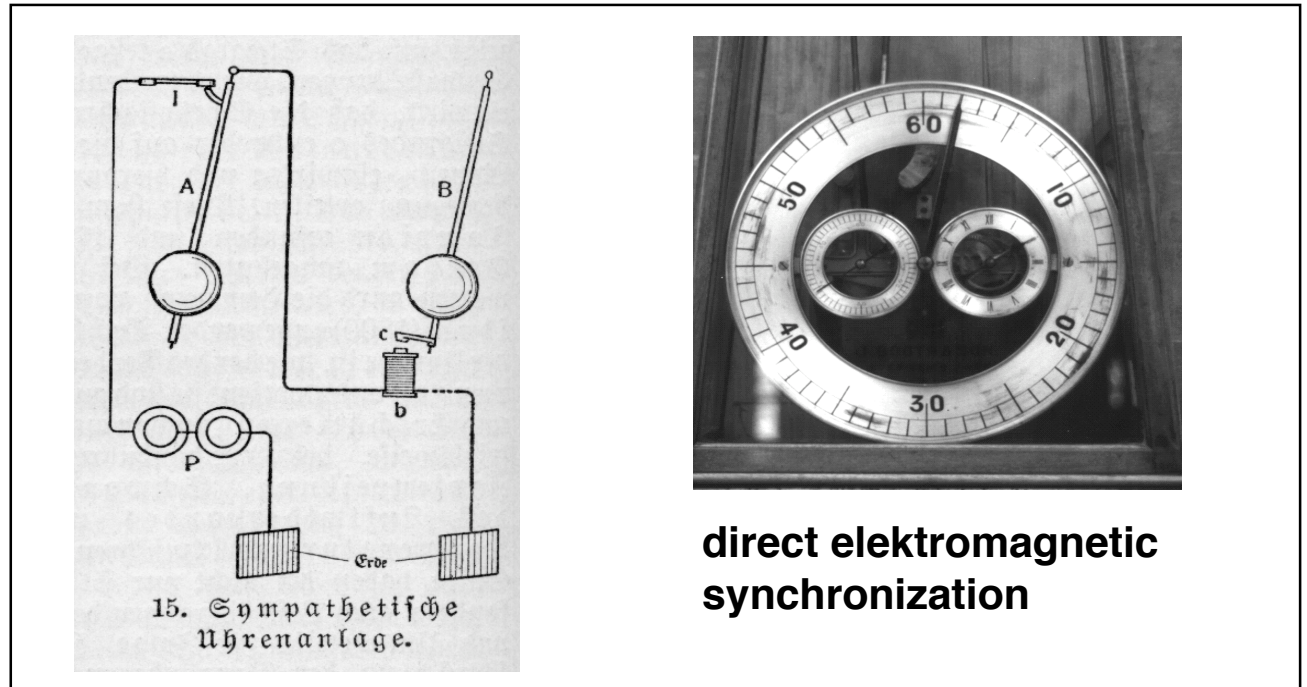
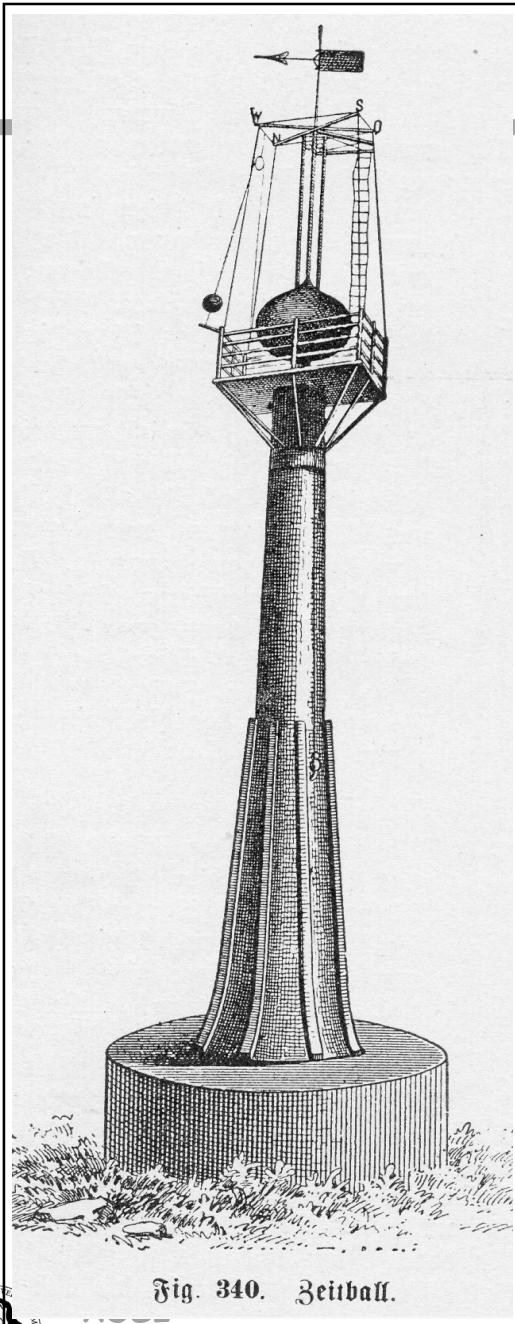
Relation between the main distributed clock parameters



α : accuracy
 π : precision
 g : granularity
 δ : offset



Clock Synchronization



Synchrony between Clocks

Two clocks c_i and c_j are synchronous at time T , if:

$$|c_i(T) - c_j(T)| < \delta$$

measure

objective

frequency synchronization

stability/min. skew

time synchronization

accuracy / min. offset



When to re-synchronize ?

Maximum drift rate ρ :

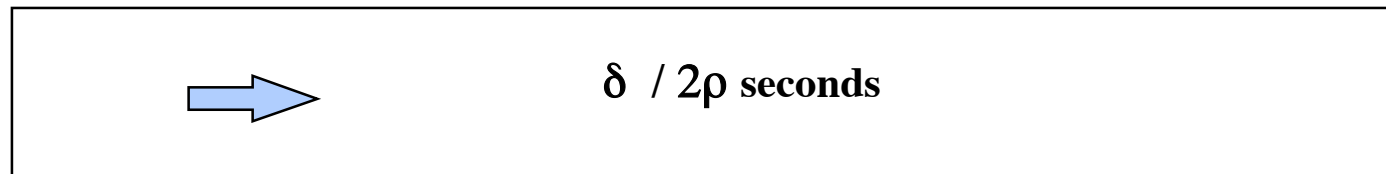
$$1 - \rho \leq dC/dt \leq 1 + \rho$$

constant ρ is a part of the physical clock specification; typical: 10^{-5} to 10^{-6}

difference of two un-synchronized clocks Δt :

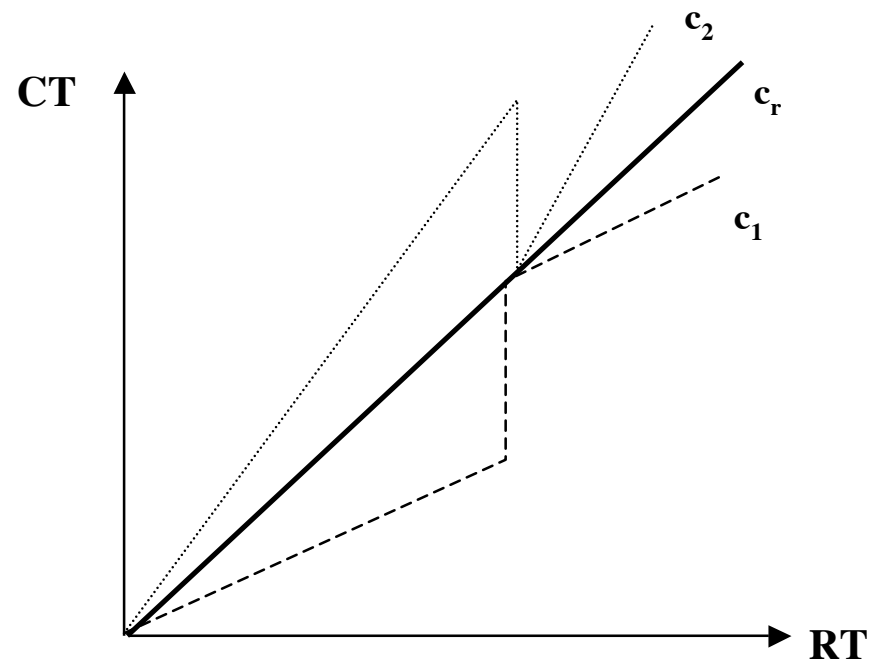
$$\leq 2\rho\Delta t$$

re-synchronization interval if two clocks should not deviate more than δ seconds after



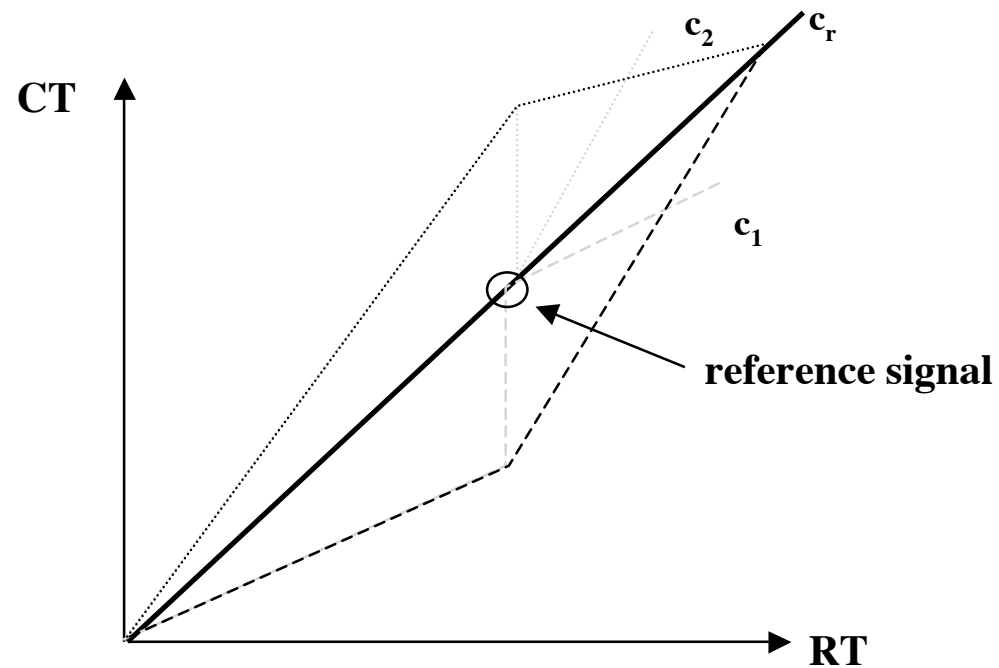
How to synchronize ?

Adapting clock values



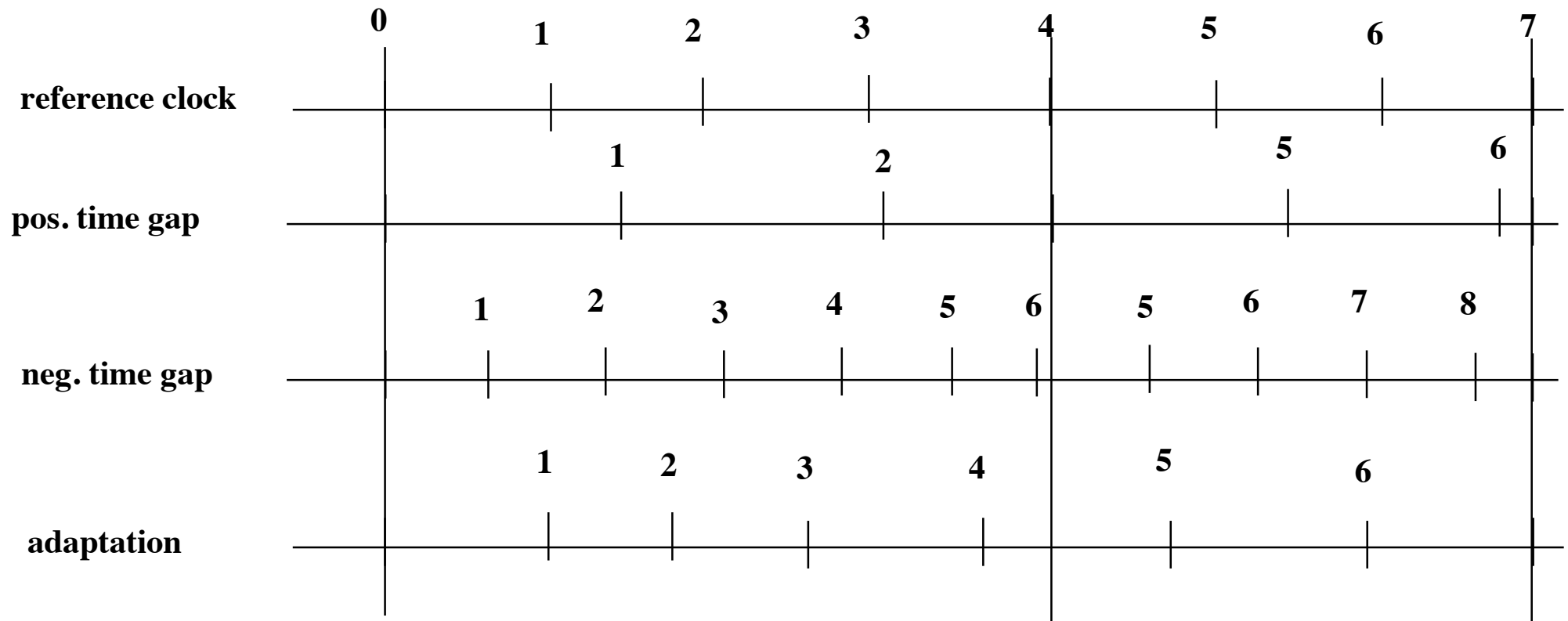
How to synchronize ?

Adapting clock rates



How to synchronize ?

- counter correction
- drift correction



Clock synchronization via messages in communication networks

- **time server**
- **Master- Slave synchronisation**
- **Decentralized and co-operative sync.**



Server-based clock synchronization

Cristian's Algorithm:

Every node synchronizes its time base periodically with a time server.

period: $\frac{\delta}{2\rho}$ to keep error below δ



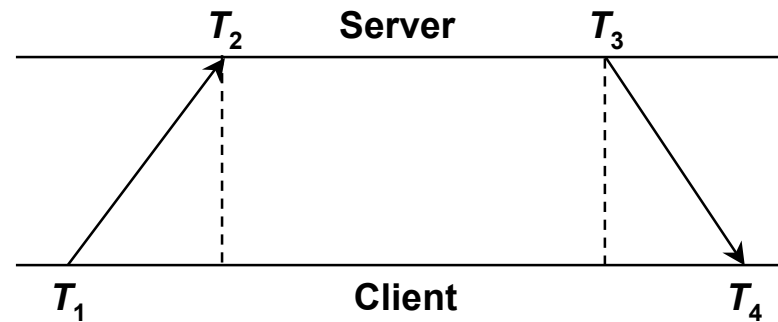
- send request for a time stamp Z to the time server
- calculate a correction factor K to correct the local clock

Problems:

- Monotony properties of time
- Measurements are based on roundtrips ($T2-T1$). A roundtrip may vary depending on network conditions.
 - > simple correction: $[(T2-T1)/2 - Z]$
 - > Refinement 1: statistical methods to calculate the average of $T2-T1$.
 - > Refinement 2: statistical methods to estimate the delay in the server.

Calculating offset and delay

Offsets o
Delay d



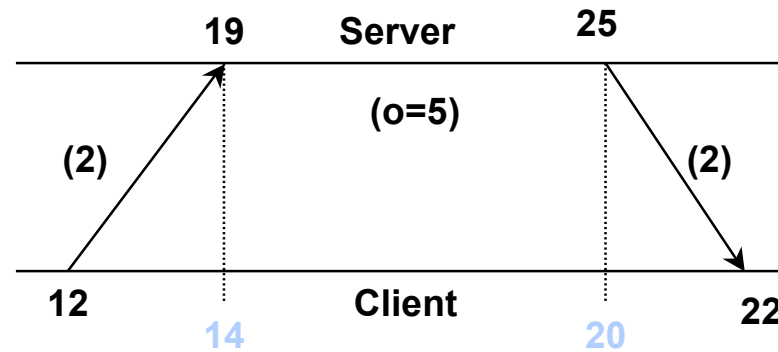
$$o = [(T_2 - T_1) + (T_3 - T_4)] / 2$$

$$d = (T_4 - T_1) - (T_3 - T_2)$$



Calculating offset and delay

Offsets o
Delay d



$$o = [(19 - 12) + (25 - 22)] / 2$$

$$o = 5$$

$$d = (22 - 12) - (25 - 19)$$

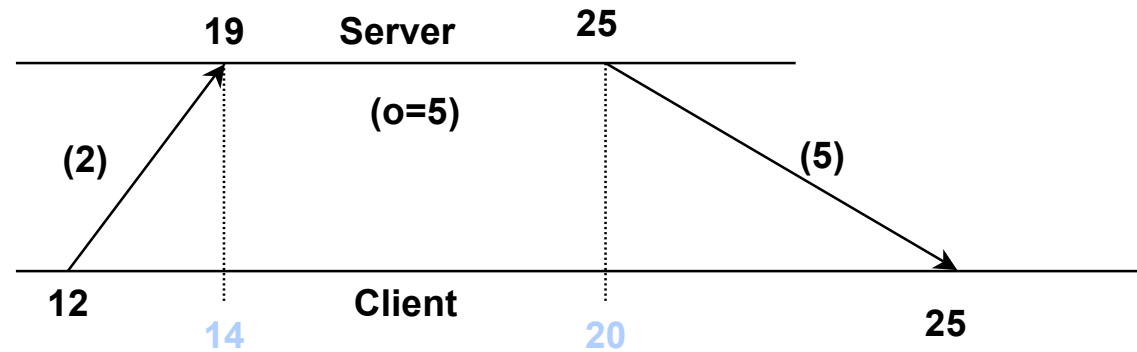
$$d = 4$$



Calculating offset and delay

Offsets: o

Delay: d

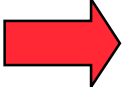





$$o = [(19 - 12) + (25 - 25)] / 2$$
$$o = 3,5$$

$$d = (25 - 12) - (25 - 19)$$
$$d = 7$$



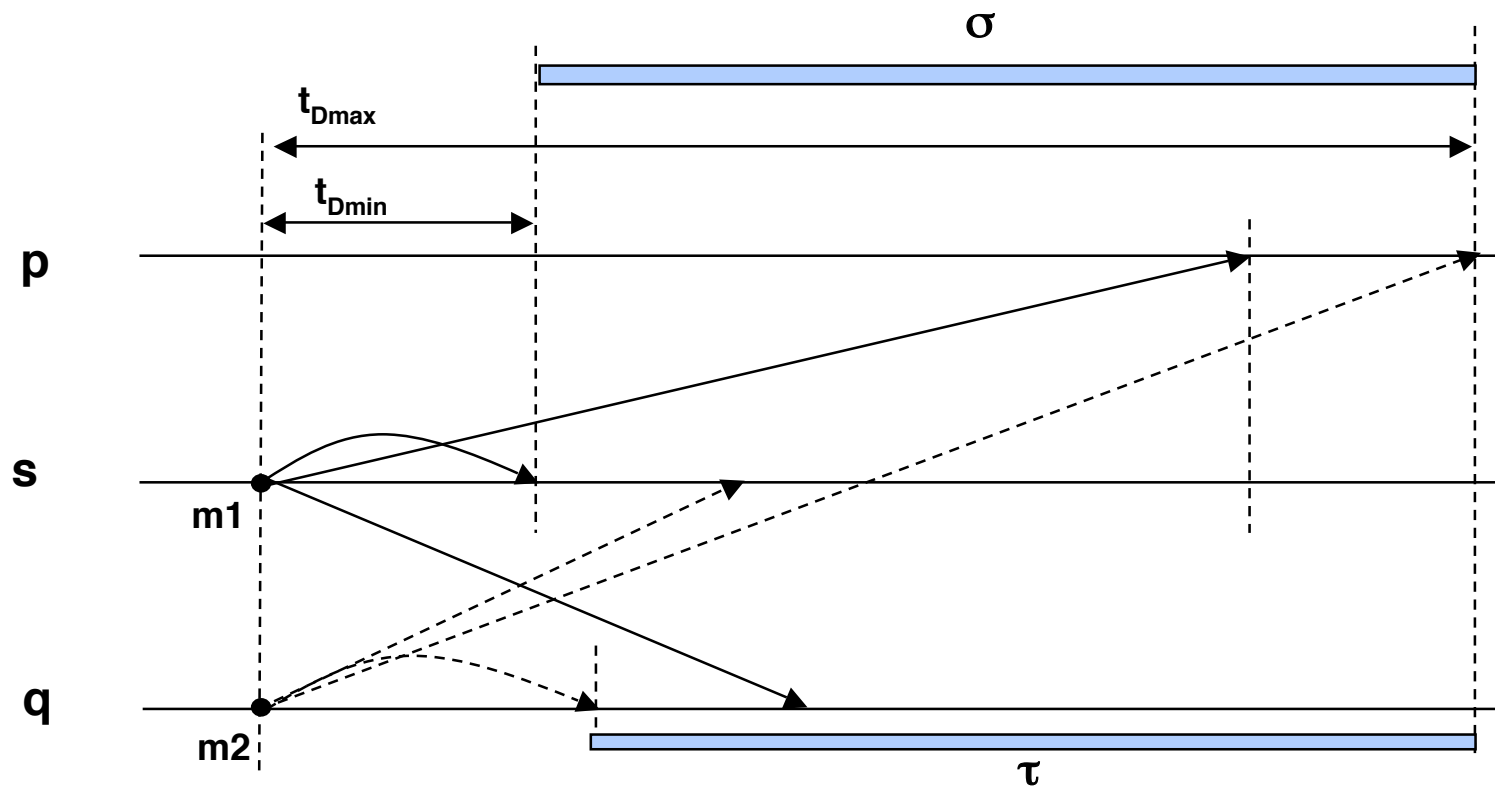
Parameter for the calculation of time

-  **message delays in the communication network
(steadyness + tightness)**
-  **Time until the message is sent
Delays in the Reply Queue**
-  **Delay when reading the local clock
Delay in the Request Queue**
-  **Physical parameters of the server clock:
Drift, Offset**

Network Synchrony Properties

Steadyness σ : measures the maximal difference of delivery times of **DIFFERENT** messages.

Tightness τ : measures the difference of transmission times of **ONE** message to **DIFFERENT** nodes.



NTP Architecture, Protocol and Algorithms



David L. Mills
University of Delaware
<http://www.ee.udel.edu/~mills>

David L. Mills : „Internet Time Synchronization: The Network Time Protocol“, IEEE TC on Communications, 39(10), 1991

<http://www.ee.udel.edu/~mills/ntp.html> (20.5.07)



The *Network Time Protocol* (Internet ~~NTP~~)

Properties:

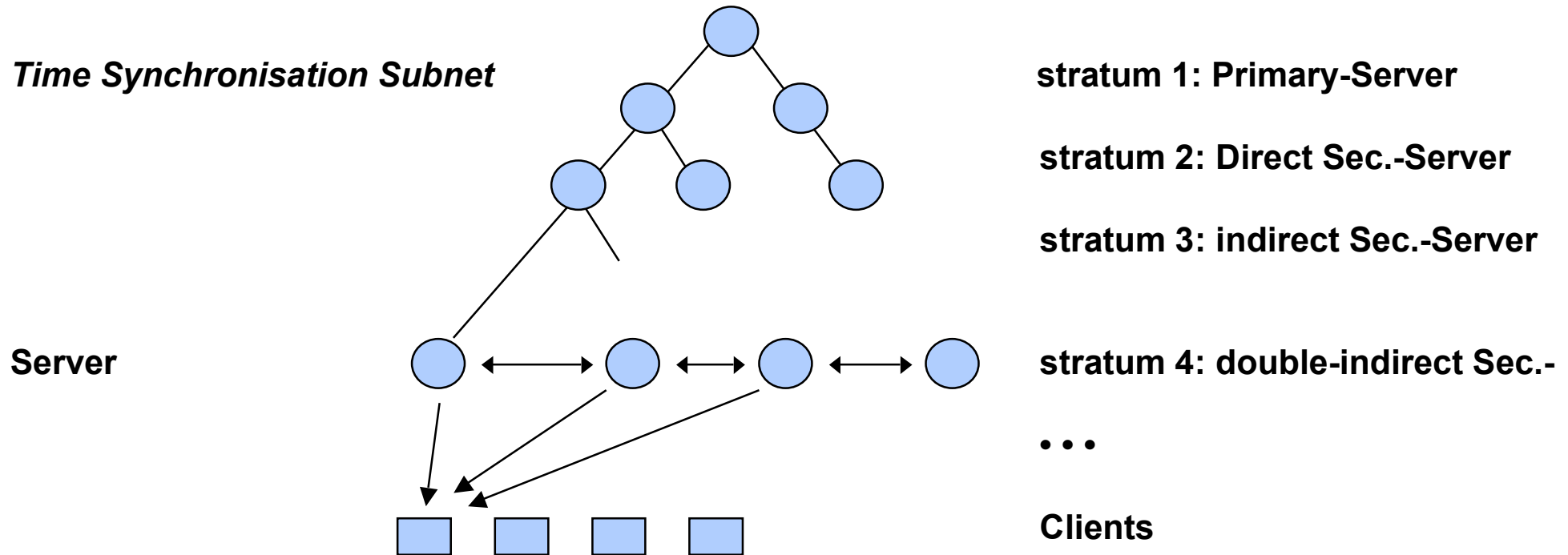
- Synchronization of physical time in LANs/WANs with UTC;
Statistical techniques to:
 - consider the variance and delay of message transmissions
 - consider the quality of time on different time servers
- High availability by redundant servers and connections
- Scalability concerning the number of time servers, clients and the synchronization rate (adapted to Internet characteristics).
- Authentication of messages and time information to protect time services against manipulation



The *Network Time Protocol* (Internet NTP)

Hierarchy of time servers:

- Primary server: UTC-clock
- Secondary server: directly or indirectly connected with a primary server. May not have a UTC-clock.



The *Network Time Protocol* (Internet NTP)

Reconfiguration

- Clock failure in a primary server → server becomes secondary server

Communication

- Cristian's algorithm: RPC-based communication, server send time-stamped replies
- on lower strata, symmetric relations between servers mutually exchanging time information
- in LANs: Multicast is used to distribute time information

High precision by exploiting the history of time messages:

Every server adds to the time message t_A

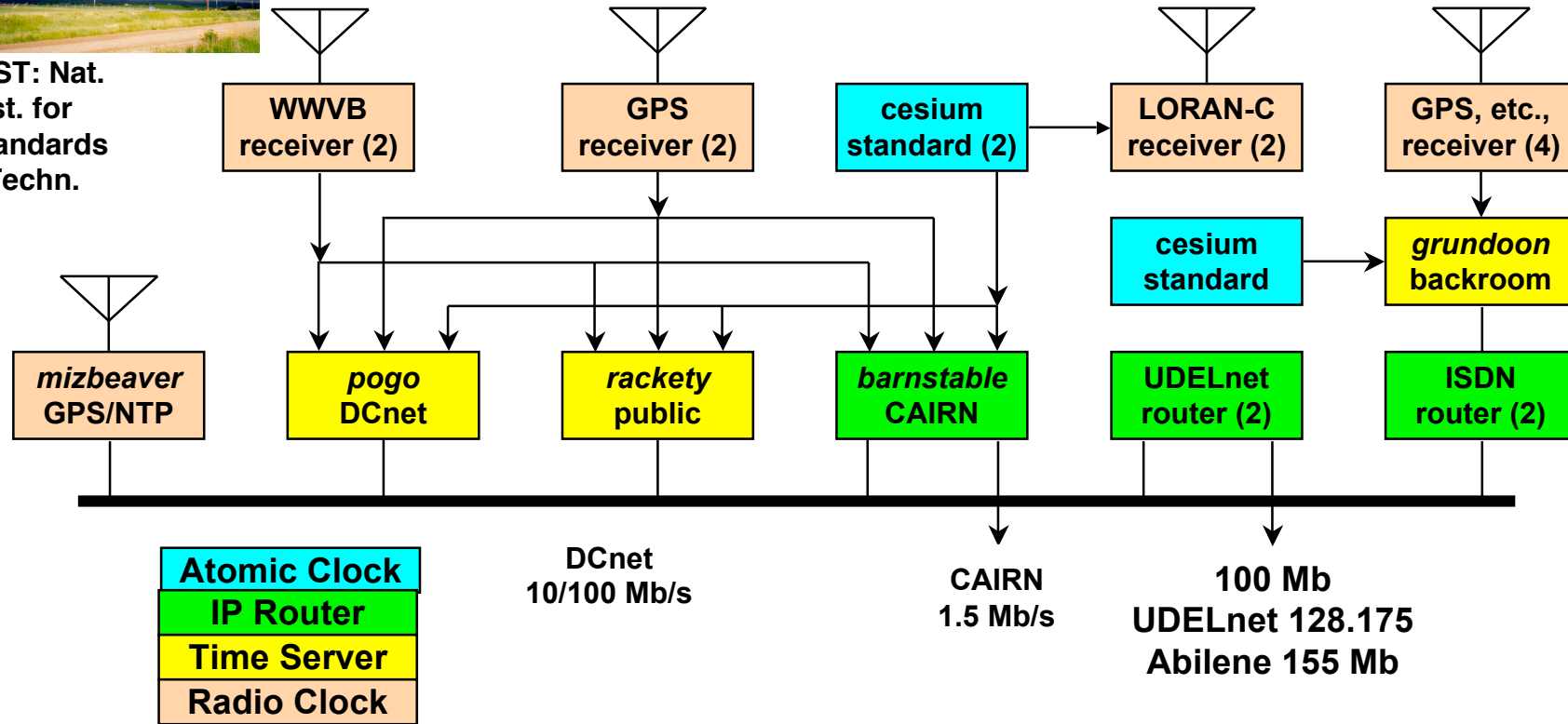
- the time stamp when the last request r_L was received
- the time stamp when the corresponding reply t_L has been sent
- the time stamp when the actual message t_A has been sent



DCnet timekeeping facilities (January 2001)



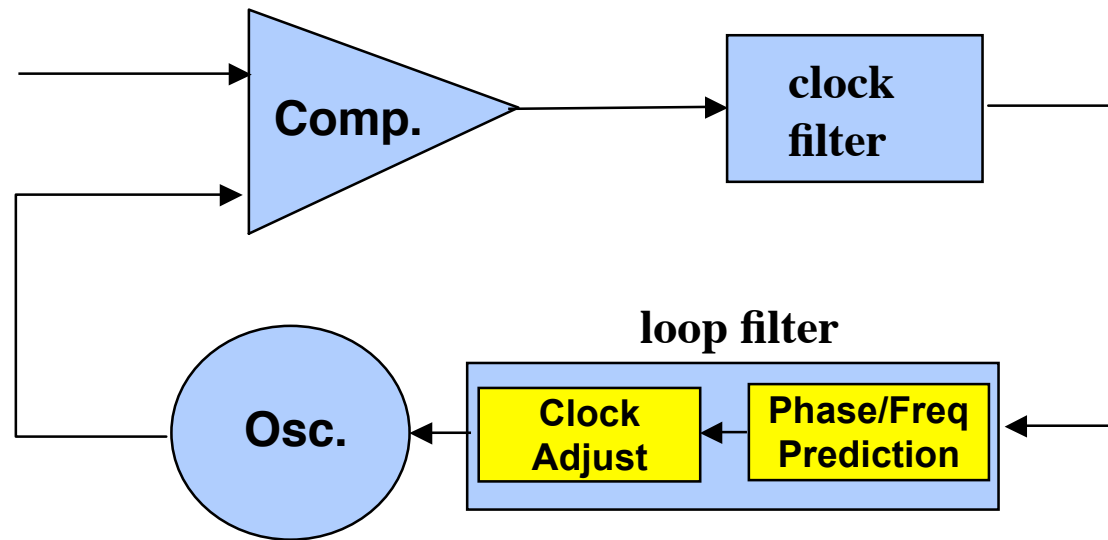
NIST: Nat. Inst. for Standards & Techn.



D. Mills/02



PLL-Modell of NTP



NTP controls the "oszillator frequency"

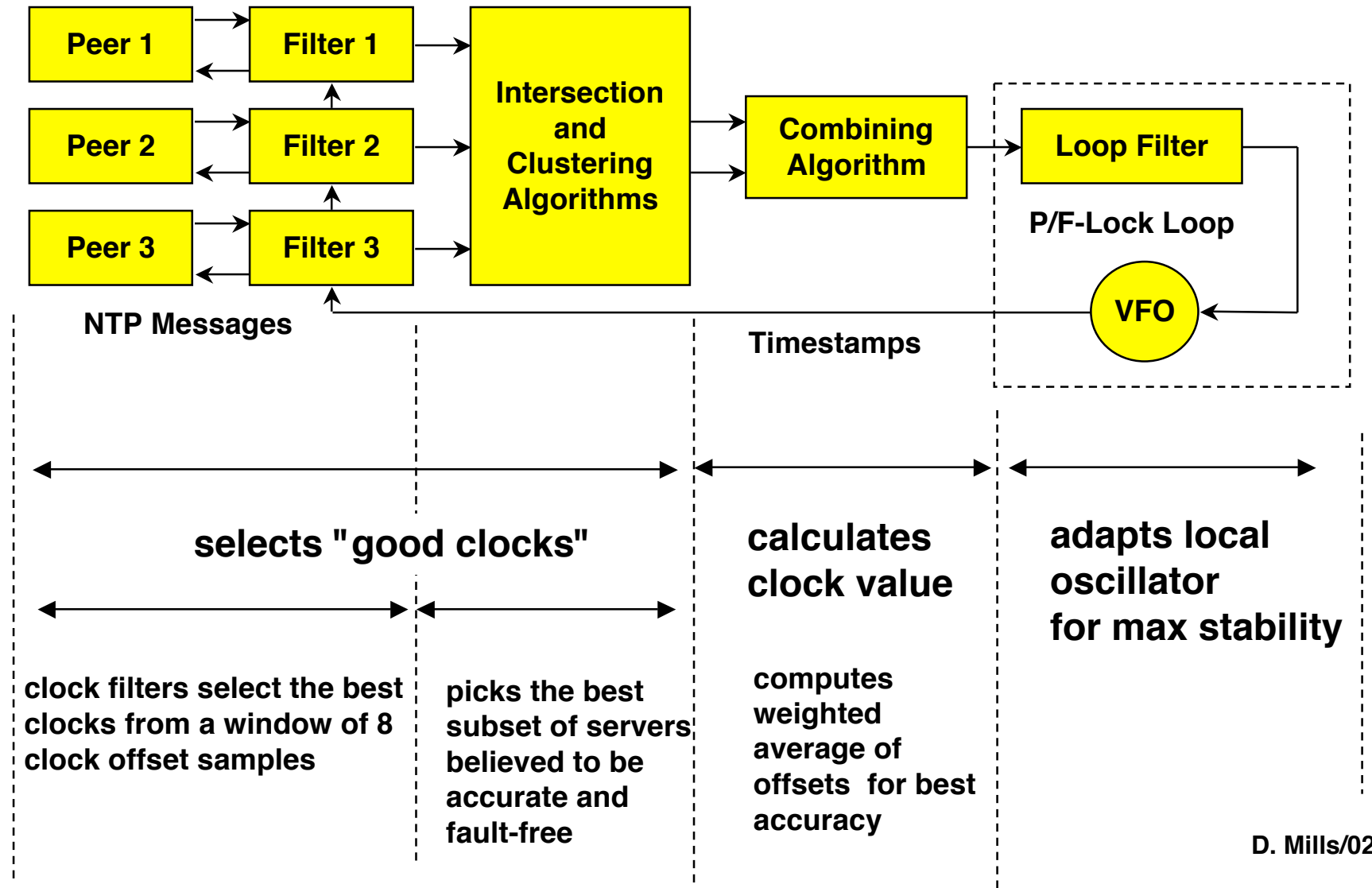
clock filter: selects appropriate time messages from the set of received time messages

loop filter: calculates the correction factor for the oszillator

D. Mills/02



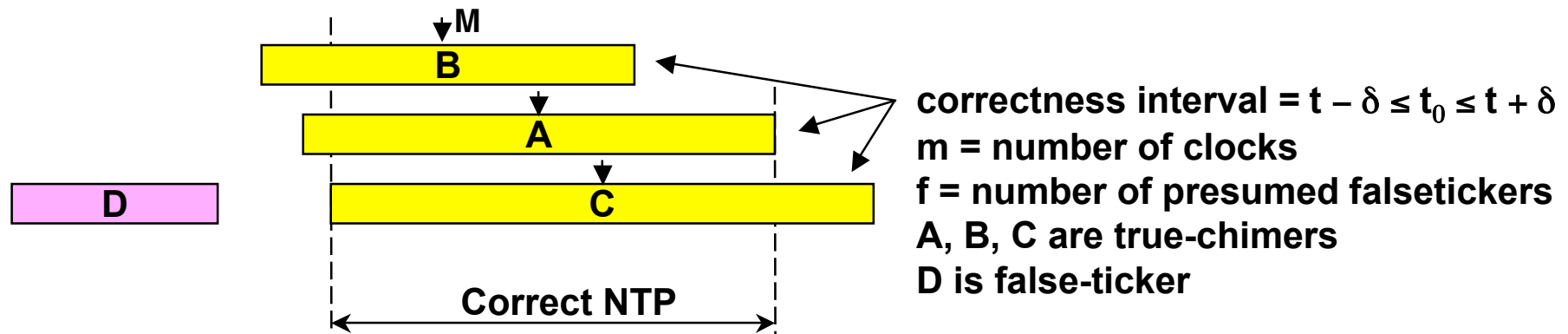
How NTP works



D. Mills/02



Intersection algorithm

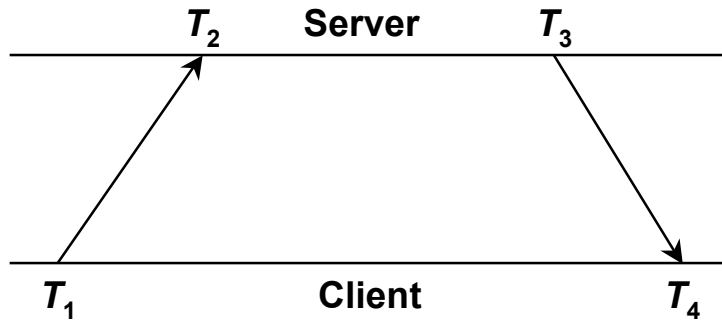


- NTP algorithm requires the *midpoint of the intervals* to be in the intersection
 - Initially, set falsetickers f and counters c and d to zero
 - Scan from far left endpoint: add one to c for every lower endpoint, subtract one for every upper endpoint, add one to d for every midpoint
 - If $c \geq m - f$ and $d \geq m - f$, declare success and exit procedure
 - Do the same starting from the far right endpoint
 - If success undeclared, increase f by one and try all over again
 - if $f \geq m/2$, declare failure

D. Mills/02

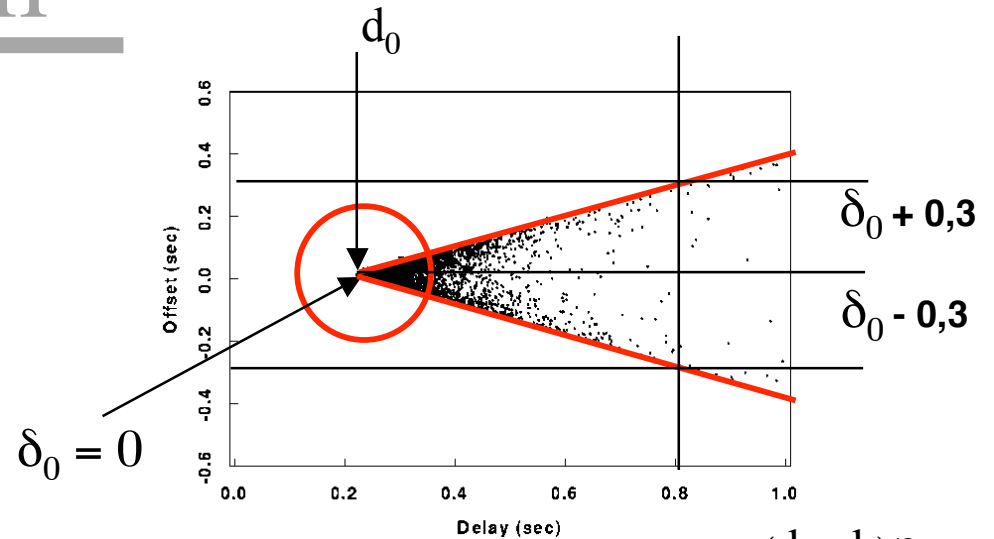


Clock filter algorithm



$$\text{offset} : \delta = \frac{1}{2}[(T_2 - T_1) + (T_3 - T_4)]$$

$$\text{delay} : d = (T_4 - T_1) - (T_3 - T_2)$$



$$\begin{aligned} & (d - d_0)/2 \\ &= (0,8 - 0,2)/2 \\ &= 0,3 \end{aligned}$$

- The most accurate offset δ_0 is measured at the lowest delay d_0 (apex of the wedge scattergram).
 - The correct time must lie within the wedge $\theta_0 \pm (d - d_0)/2$.
 - The d_0 is estimated as the minimum of the last eight delay measurements and (d_0, δ_0) becomes the offset and delay output.
 - Each output can be used only once and must be more recent than the previous output.
- The distance metric λ is based on delay, frequency tolerance and time since the last measurement.

D. Mills/02



Fault-tolerant convergence algorithms:

- Fault-tolerant Midpoint and
- Fault-Tolerant Average

– Fault-tolerant Midpoint:

The k largest and the k smallest values will not be considered.

The average is computed from the values $k+1$ und $n-k$:

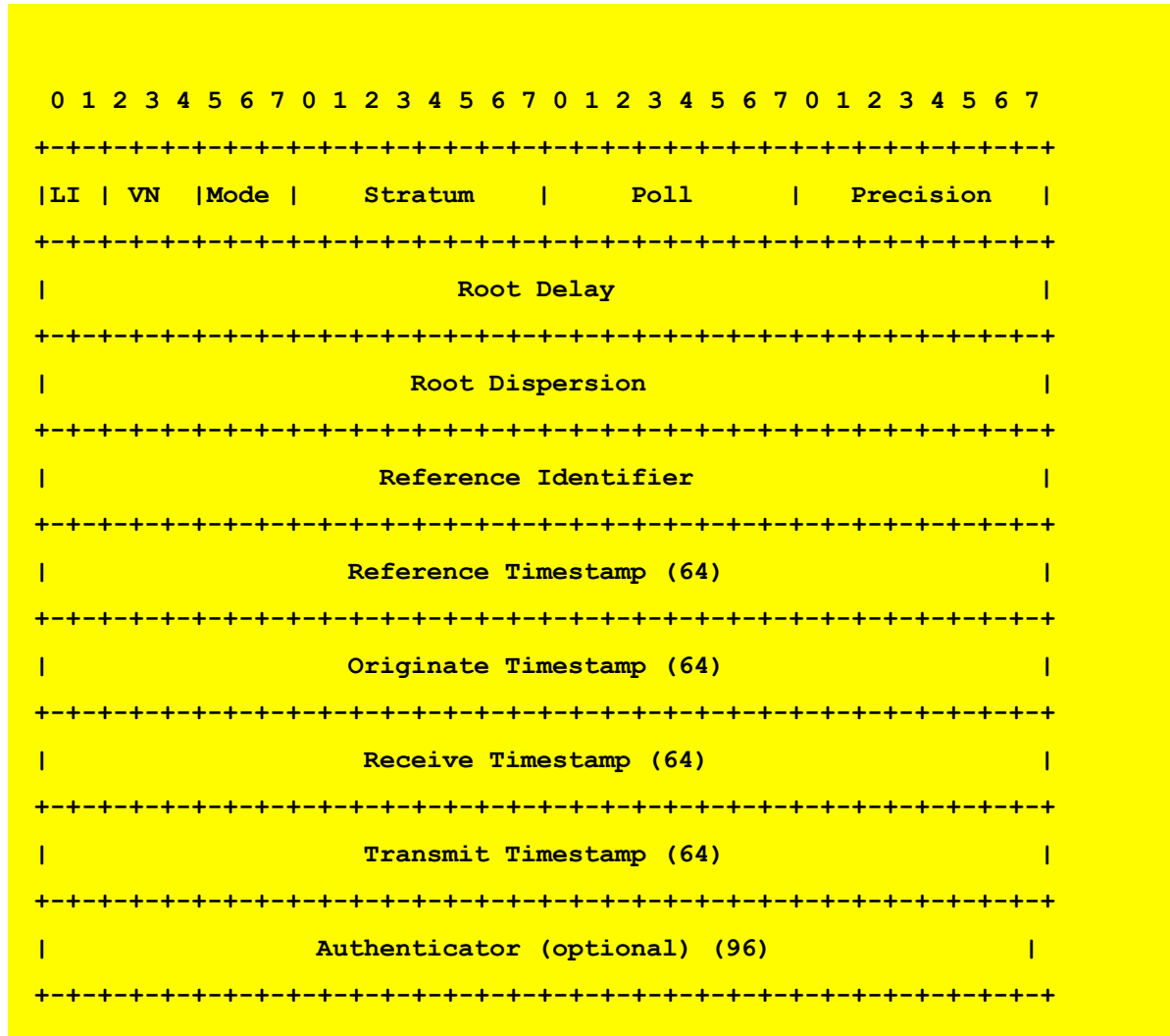
- Midpoint: $(\text{max_value} + \text{min_value}) / 2$
- this is NOT the MEAN value of the sorted list of values!!!!

– Fault-tolerant Average:

The k largest and the k smallest values will not be considered. The average will be computed from the remaining values.



NTP Packet Header



LI : Leap indicator, indicates a „leap second“ because of UT/UTC difference

VN: Version number of NTP Impl.

Mode: Operational smode, e.g. client or server

Stratum:

1: primary reference

2-15: secondary reference

16-255: reserved

Poll: Max. interval between consecutive time messages.

Precision: precision of local clock

Root Delay: Roundtrip Delay

Root Dispersion: Statistical error relative to the primary reference

Ref. ID: ID of the reference clock (e.g. DCF77)

Ref.TS: time of last synchronization

Originate, Rx. Tx.: Client req., Server Rx. Sever Tx to calculate roundtrip.



Authenticator: for optional authentication of time stamps.

Control characteristic

Phase error

100ms

corrected after 39 Min.

overshoot (overcompensation) 7ms after 54 Min.

stabilizes to error < 1ms after 6h

Frequency error

50 ppm

1 ppm after 16h

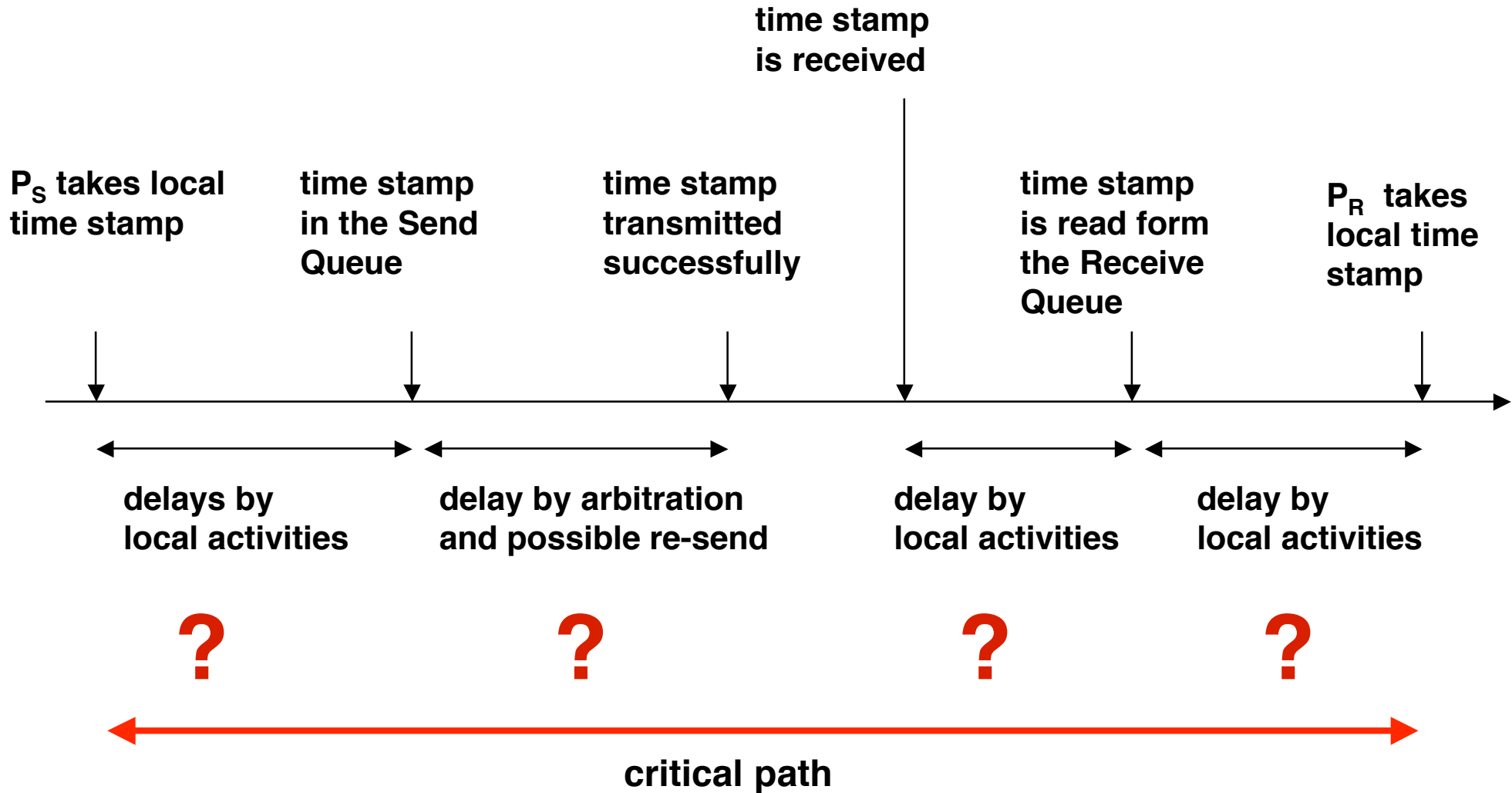
0,1 ppm after 26 h

=> measurements: 99% of all time information deviates less than 30 msek from UTC.

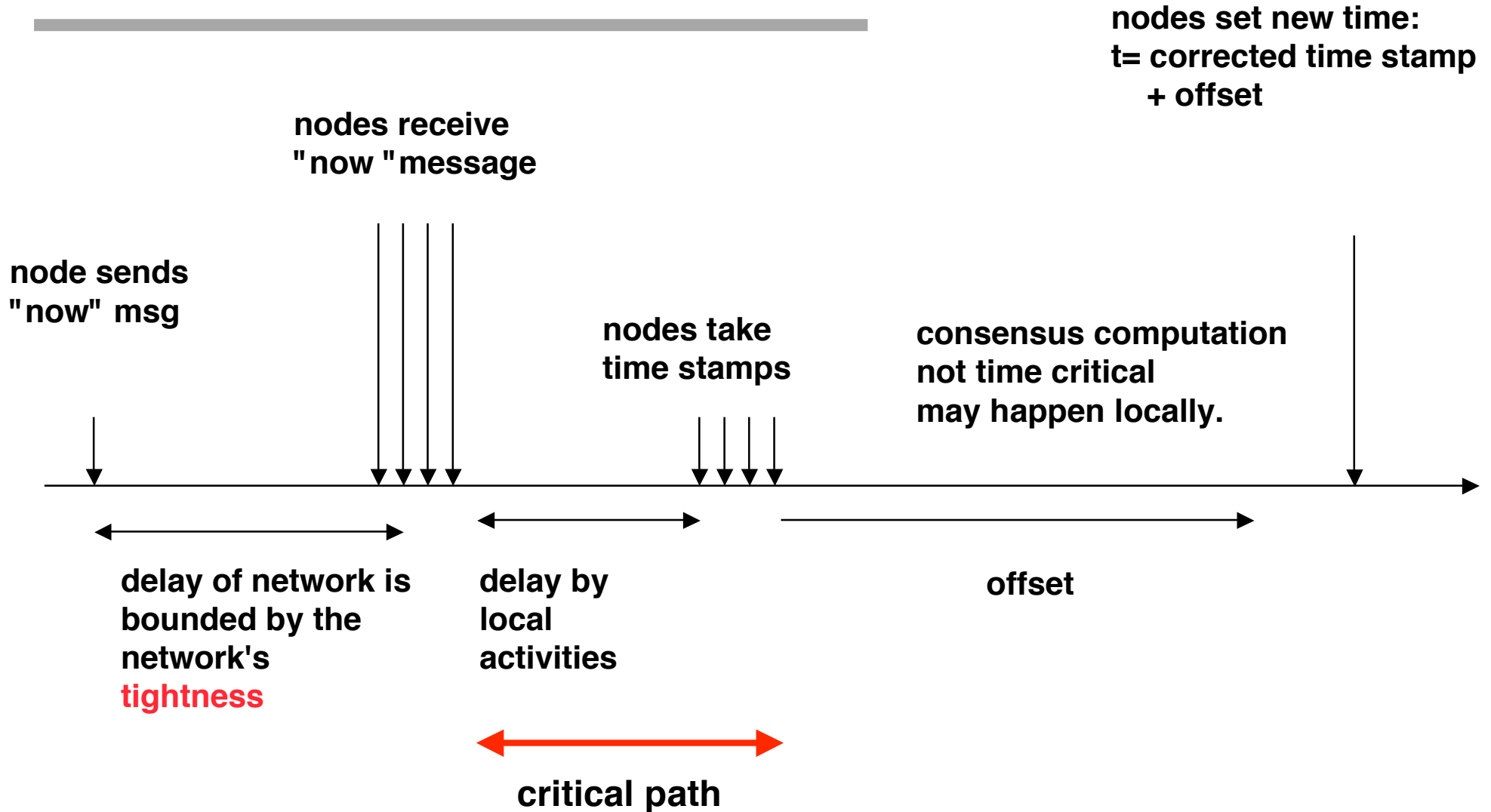
If the clock is outside the capture of the PLL, the clock will be adjusted rather than the Oscillator frequency.



Practical Considerations:



Practical Considerations:



The Berkeley Algorithm

R. Gusella, S. Zati: "The accuracy of the clock synchronization achieved by TEMPO in Berkeley Unix 4.3 BSD." IEEE Trans. Softw. Eng., (15)7, July 1989

