International Timing & Sync Forum 2022

#ITSF2022

PTP+Squared - an Overlay Network

Ian Gough

Co-Founder Timebeat.app

#ITSF2022

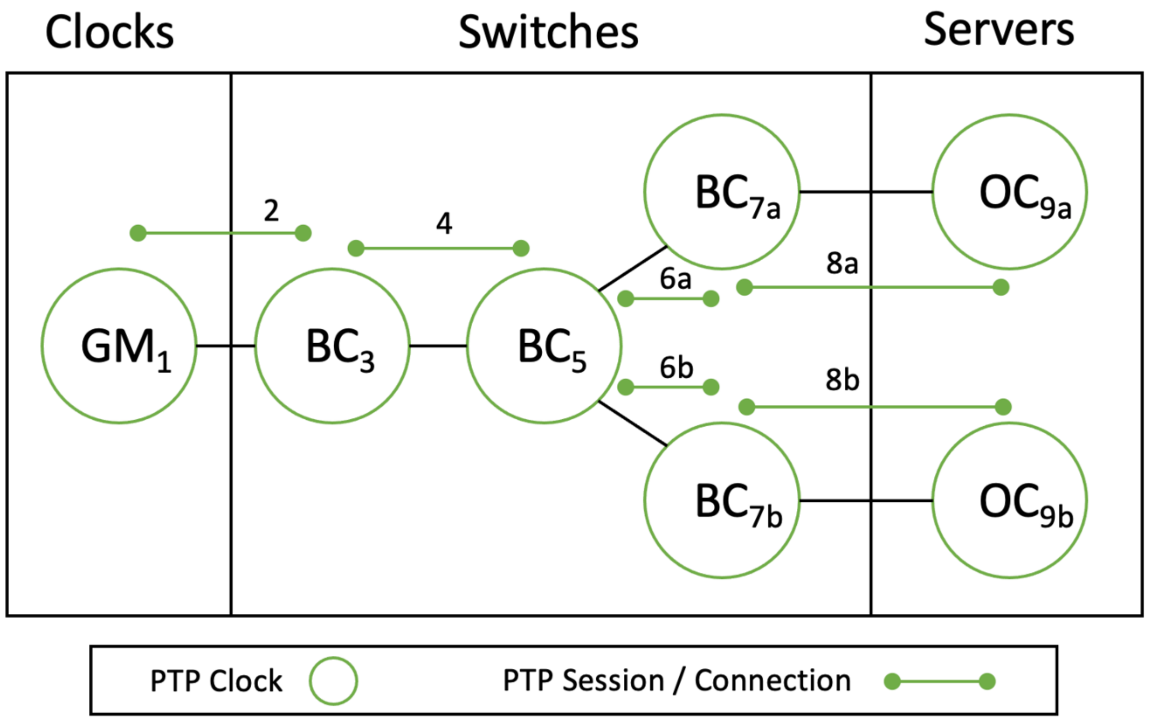
IEEE1588/PTP is a Great Achievement!*

- The success of getting PTP support built-in to most commodity hardware network controllers MACs and PHYs is quite an achievement
- The architects of IEEE1588-2008 have achieved some important milestones
    - For example; dishonest practices in broker-dealers has been significantly curtailed by MiFID regulations - in no small part because sequencing is now a recorded fact. A leads to B leads to C.
    - Their work allows for great accuracy and the impact that this has had is tangible in many sectors
- The way PTP works is brilliant
    - PTP solves real problems in the real world

*However…. There are a few unresolved issues…. ☺

...Let’s talk about them…

Boundary clock distribution is literally nothing but single points of failure and unquantifiable error



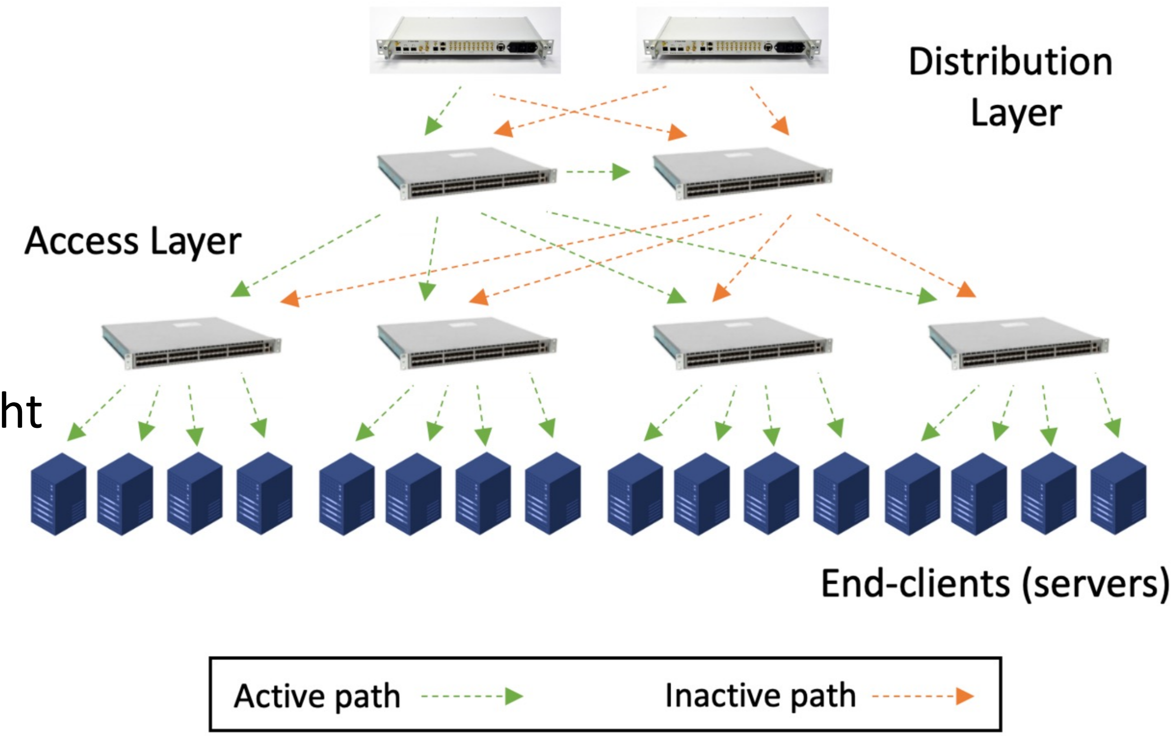
Everything green is a single-point-of-failure. Every :

- Clock
- Switch
- Host
- PTP Session / Connection between them!

The single-parent principle is everywhere

Boundary clock distribution and BMC

- Here is a typical PTP deployment
- It is unknown if nodes have the same time
- In the real world, when we measure “no, they don’t”
- The best master clock concept of self announcing how good the quality of the time distributed is “not a good strategy”™
- It’s hard not to think that BMC was an afterthought in PTP
    - It’s also optional
- Problems changed when we went from ms->us->ns->ps
    - What used to be ignorable and insignificant is now the opposite

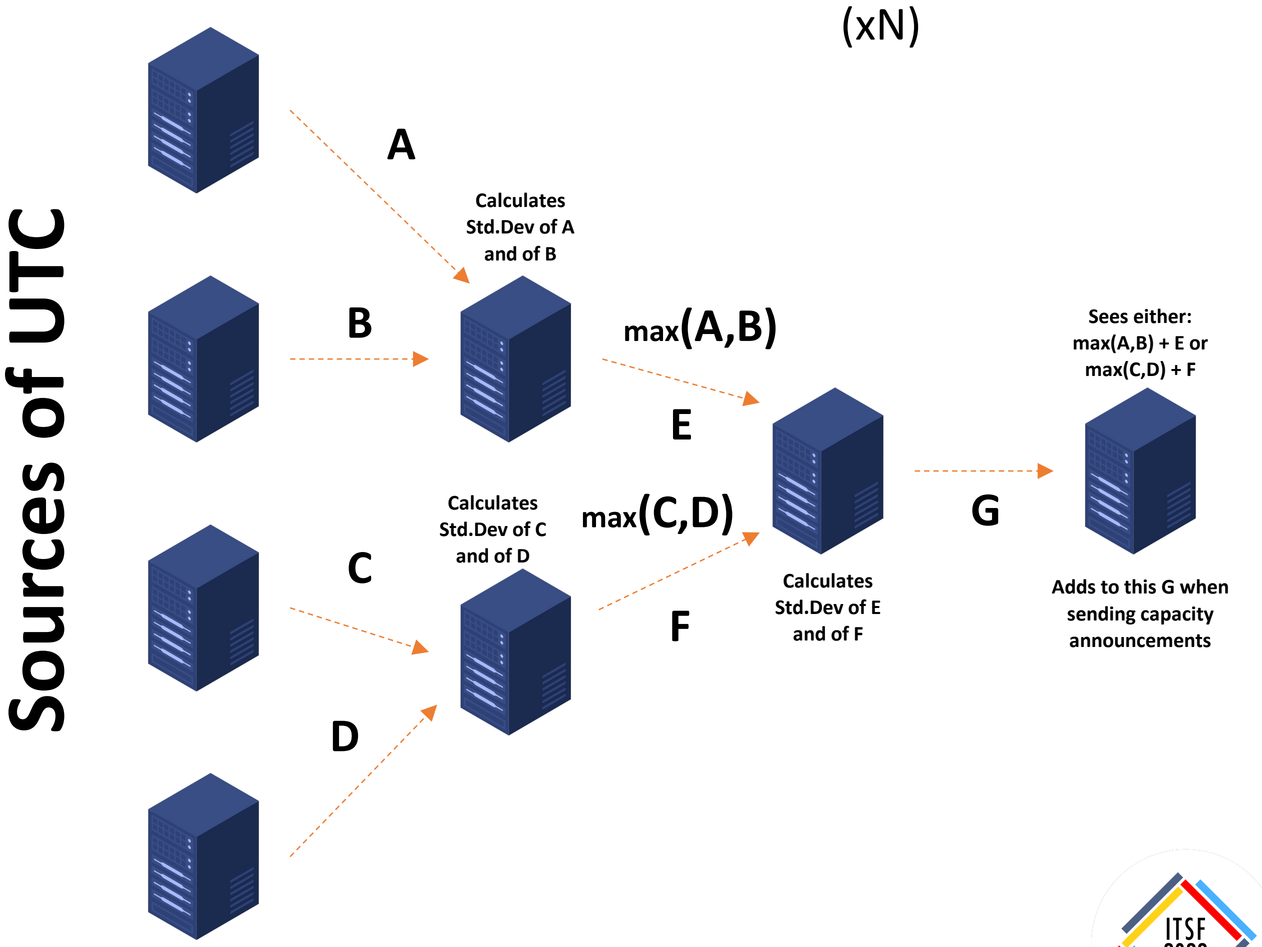


We thought about how to solve these problems…

... Introducing PTP+Squared…

Suppose we turn the tree upside down….

- Now any server has multiple paths to sources of time
- All of the paths can be compared in real time
- Error of variation can be communicated downstream in a cumulative fashion
- In that way if any part of the tree becomes congested it will be visible and time can be taken from a better path


(xN)
Sources of UTC
A
B
C
D
Calculates Std.Dev of A and of B
Calculates Std.Dev of C and of D
max(A,B)
max(C,D)
E
F
Calculates Std.Dev of E and of F
G
Sees either: max(A,B) + E or max(C,D) + F
Adds to this G when sending capacity announcements

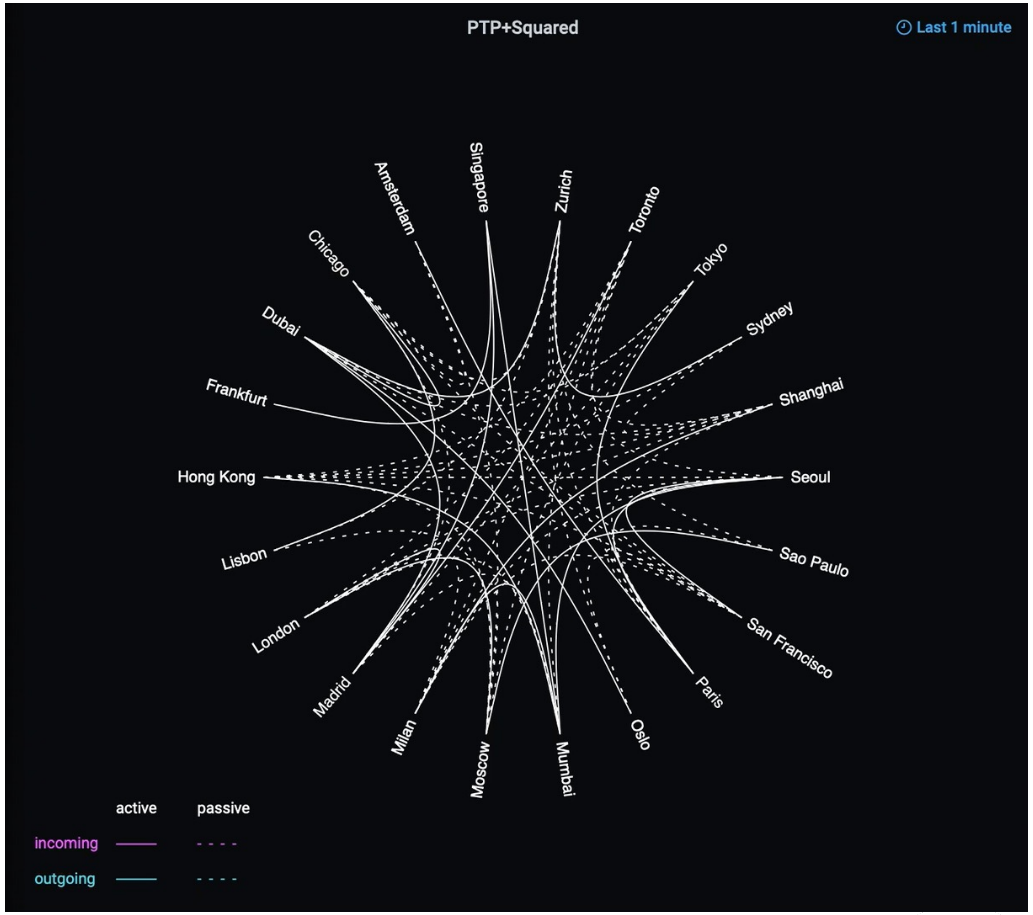
How do we create the reverse tree / mesh structure?

- How is it done? & What does it look like?

- We used a peer-2-peer distributed hash table to create an overlay network to PTP.

- The overlay network sets up unicast sessions between PTP participants in the grid
  - we call them “grid nodes” and “root nodes”

- Everything is completely dynamic and in real time
  - We used D3.js to visualize it, it’s cool ☺



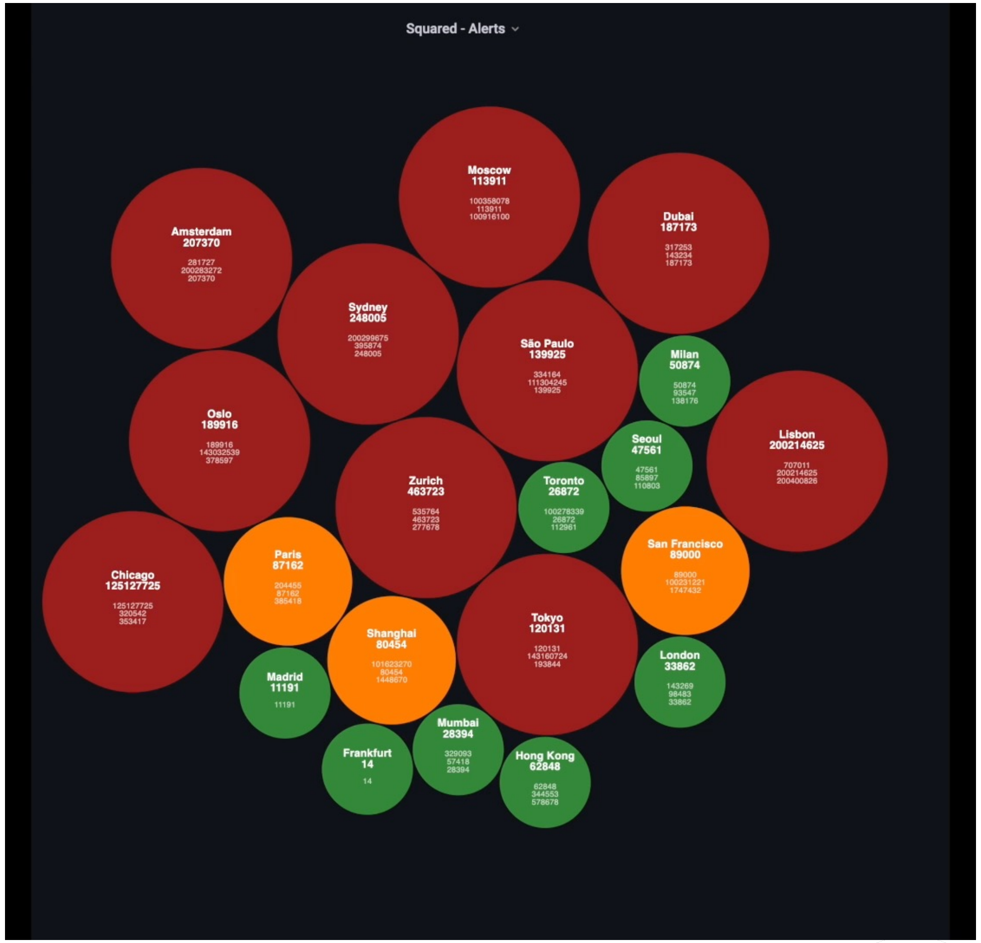
Error is quantifiable and we compare paths

- Squared nodes showing their cumulative error (from source to server).

- Above certain thresholds the bubbles grow and turn orange and red.

- This is 8x faster than real-time, so 5 1/2 minutes condensed into 40 seconds.

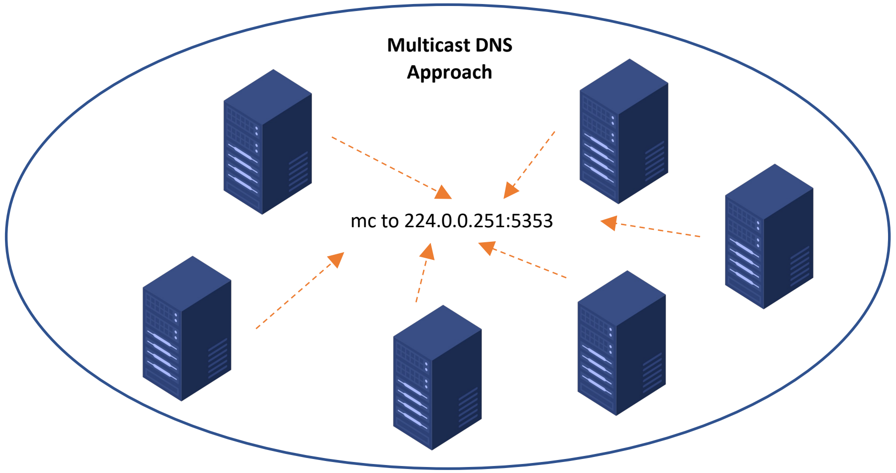
- Here the mesh goes from largely bad to largely good in that time.

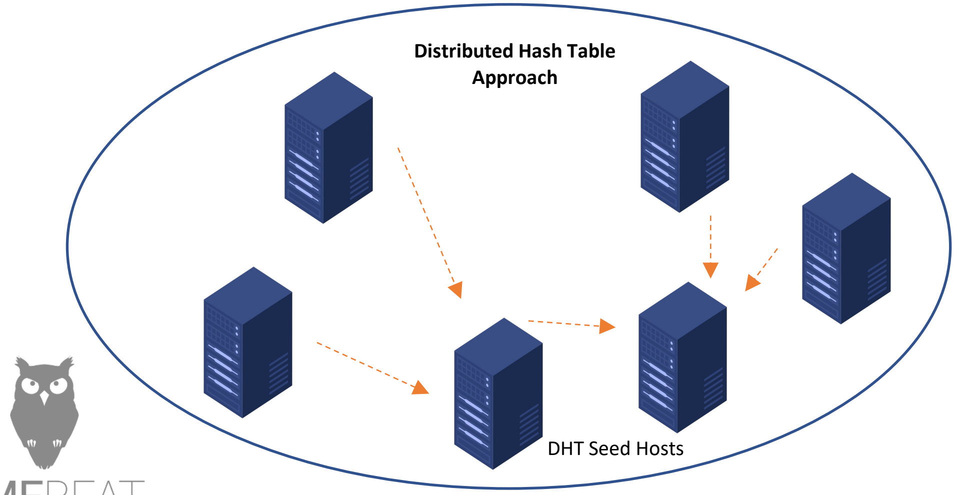

Squared - Alerts ⌄
Moscow 113911 100358078 113911 100916100
Amsterdam 207370 281727 200283272 207370
Dubai 187173 317253 143234 187173
Sydney 248005 200299675 395874 248005
São Paulo 139925 334164 111304245 139925
Milan 50874 50874 93547 138176
Oslo 189916 189916 143032539 378597
Seoul 47561 47561 85897 110803
Lisbon 200214625 707011 200214625 200400826
Zurich 463723 535764 463723 277678
Toronto 26872 100278339 26872 112961
Chicago 125127725 125127725 320542 353417
Paris 87162 204455 87162 385418
San Francisco 89000 89000 100231221 1747432
Shanghai 80454 101623270 80454 1448670
Tokyo 120131 120131 143160724 193844
Madrid 11191 11191
London 33862 143269 98483 33862
Mumbai 28394 329093 57418 28394
Frankfurt 14 14
Hong Kong 62848 62848 344553 578678

TIMEBEAT

ITSF 2022

Two types of peer discovery mechanisms





The two approaches are not mutually exclusive, and nodes can use one or both in the same PTP+Squared network, but pure DHT is nearly always the better approach.

Paths created dynamically using either:

- mDNS (rfc6762)
  - Quick and easy: zero config
  - Not particularly scalable
- DHT
  - Requires a few nodes whose IP and public key are known
  - Highly scalable

What is the attraction

Benefits

- This model can support millions of nodes in terms of scale and still have remarkably good accuracy
- The scaling model becomes vertical instead of horizonal.
  - Scale is achieved through many nodes that each do negligible work
  - Even with 3 hops between time source and final node (a cumulative error of +/-15ns can be achieved) this can easily support tens of thousands of nodes
- Time is compared from different nodes at all levels of the network. Verification of the time received is completely built into the backbone of the system (very much unlike BMCA)
- Everything uses signed / verified / authenticated public-private key X.509 type stuff and is ultra secure
- I.e. when the system speaks with a node or receives a message, it is certain™ of the sender

- PTP+Squared takes IEEE1588 PTP systems from a "blind faith" model into a verified "trust based" model.

THANK YOU