



Demonstrating time compliance to regulators and other non-technologists

Peter Lankford
Founder and Director, STAC®

www.STACresearch.com

What is STAC?

- STAC facilitates the STAC Benchmark Council:
 - ~300 financial firms and ~50 tech vendors
 - Establishes standard technology benchmarks and testing software
 - Promotes dialog
 - Includes the STAC-TS Working Group (time sync)



What is STAC?

- STAC also performs independent benchmark audits

The collage shows five overlapping STAC Report covers, each with a unique title and configuration:

- Report 1:** *STAC Report* STAC-A3 / Spark / Levey Xenon 3.2 / GCP n1-standard-64. Configuration: Spark 1.6.1 with Levey Xenon 3.2.0 on 5 x Google Cloud Platform n1-standard-64, each with 2TB local SSD.
- Report 2:** *STAC Report* STAC-M0 / Adb+LS / Lenovo ThinkSystem SR650 / Intel SSD P4800 & 4800. Configuration: Kx Systems kdb+ 3.5 on Lenovo ThinkSystem SR650 with Intel Xeon Platinum 8180 and Intel SSD P4600 & P4500.
- Report 3:** *STAC Report* STAC-A2 / Dell C4130 / NVIDIA P100 GPU. Configuration: NVIDIA CUDA 8.0 on a Dell PowerEdge C4130 with 4x NVIDIA Tesla P100 GPUs and 2x Intel Xeon E5-2690v4 @ 2.6GHz.
- Report 4:** *STAC Report™* STAC-A2 Pack B. Configuration: UDP over 10GbE using OpenOnLoad on RHEL 6.6 with Solarflare SFN 8522-PLUS Adapters on HPE ProLiant XL170r Gen9 Trade & Match Servers (SUT ID: SFC170206). **STAC-N1™ Benchmarks β1**. Tested by: STAC. Test date: 06 Feb 2017. Report 1.0, 07 Feb 2017.
- Report 5:** *STAC Report* STAC-A2 Pack A. Configuration: RHEL 7.2 on 2x 28-core Intel Xeon Platinum 8180, 4x 1.4TB Intel SSD P4800, 2x 1.4TB Intel SSD P4600, 1x Lenovo ThinkSystem SR650.

Each report cover includes the STAC logo, a 'Stack under test' section with a color-coded bar chart, and a disclaimer: 'THESE TESTS PROPOSED OR APPROVED BY THE STAC BENCHMARK COUNCIL (SEE WWW.STACBENCHMARK.COM) - BE SURE TO CHECK THE VERSION OF ANY SPECIFICATION USED IN A REPORT. DIFFERENT VERSIONS MAY NOT YIELD RESULTS THAT CAN BE COMPARED TO ONE ANOTHER.'

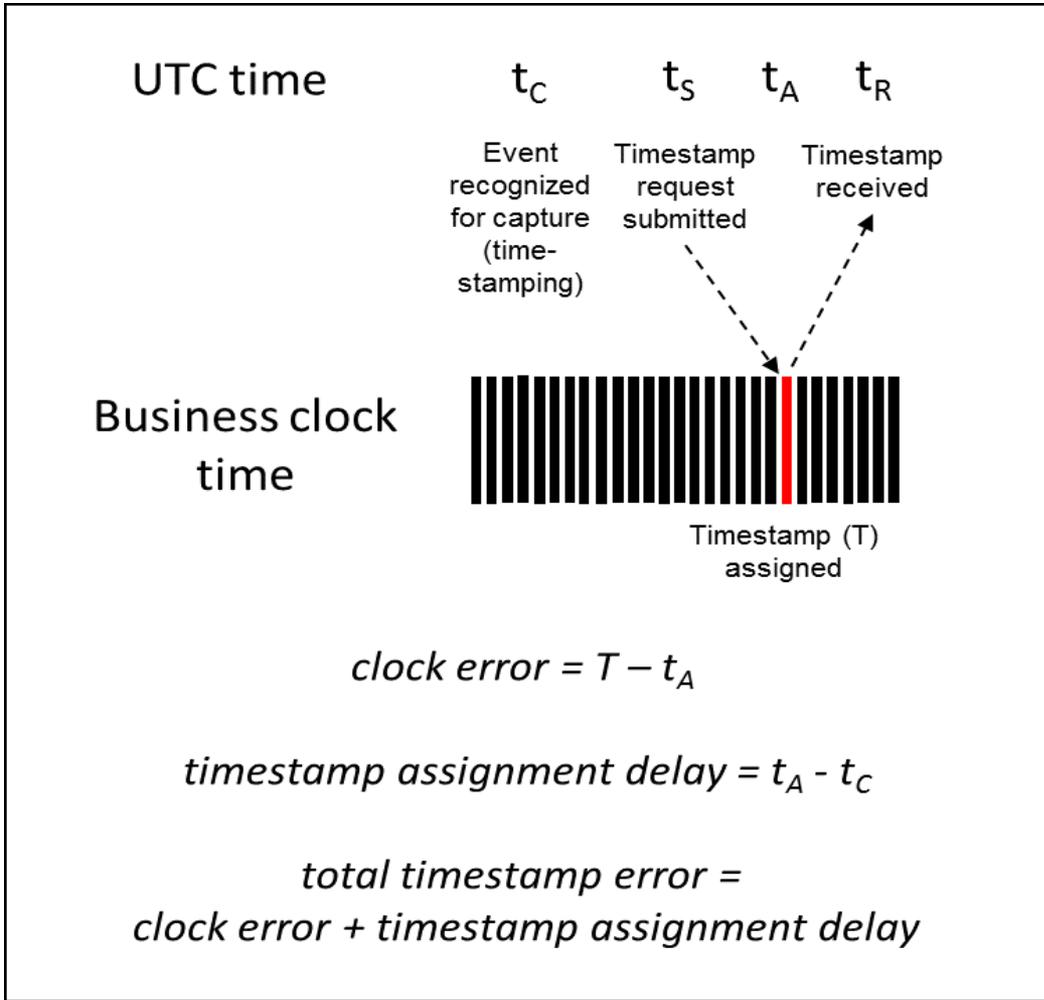


RTS 25 compliance challenges

- Some issues get a lot of attention
 - How should I get accurate time to my sites?
 - How should I synchronize my host clocks?
 - Should I use network capture?
- Other issues sometimes get less attention than they should
 - For example..

Application-level error

Error in application timestamps that is independent of clock error

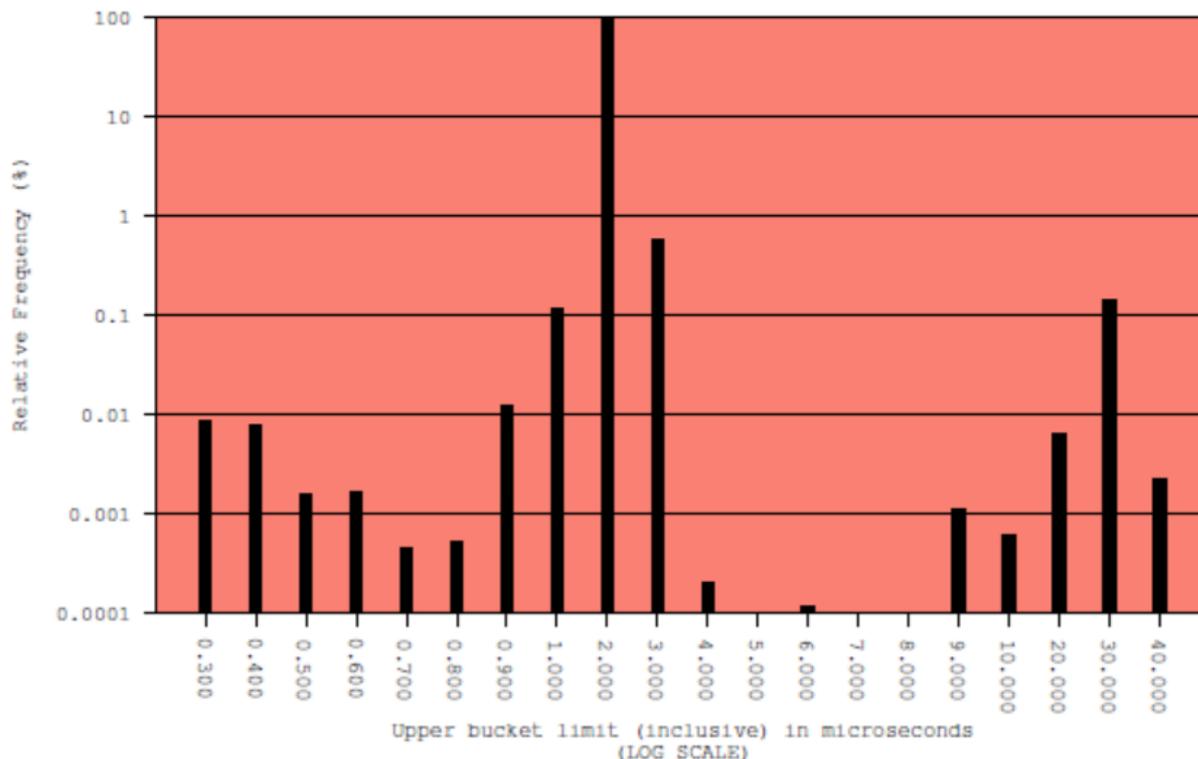


Not-uncommon distributions of app-level error

Percentiles (μsec)

Percentile	Error
Max	21,177.625
99.9999%	10,012.557
99.999%	32.836
99.99%	27.150
99.9%	21.833
99%	1.773
95%	1.640

Log-log histogram



- Takeaway: You need to test application-level error carefully

(BTW, I grabbed these from different reports. They are not from the same system.)

Holdover of host clocks

- What happens if a daemon dies? Or a connection to the upstream clock is lost?
- Unfortunately, a lot:

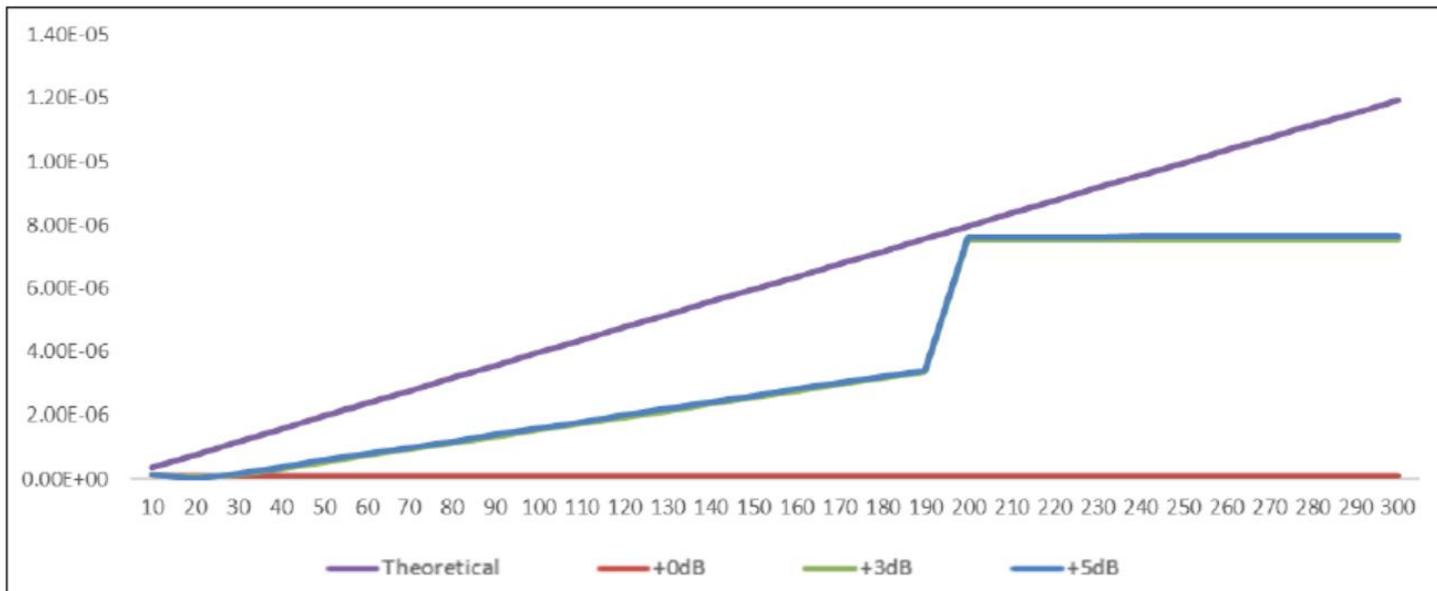
Time (s)	Offset (us)	
	Server A	Server B
60	0.4	1.6
120	1.3	1.4
300	5.8	3.6
600	15.4	15.3
1200	44.5	50.4
1800	75.8	83.0



- Takeaway: You need to test host-clock holdover and track daemon health in production carefully

GNSS spoofing and jamming

- Vulnerabilities of GNSS (e.g., GPS) are well known
- Will regulators treat those issues as exceptions?
- What are the potential impacts on our architecture?



Courtesy Spectracom,
a member of STAC-TS

Example: Test the impact of a Frequency Offset on a disciplined oscillator

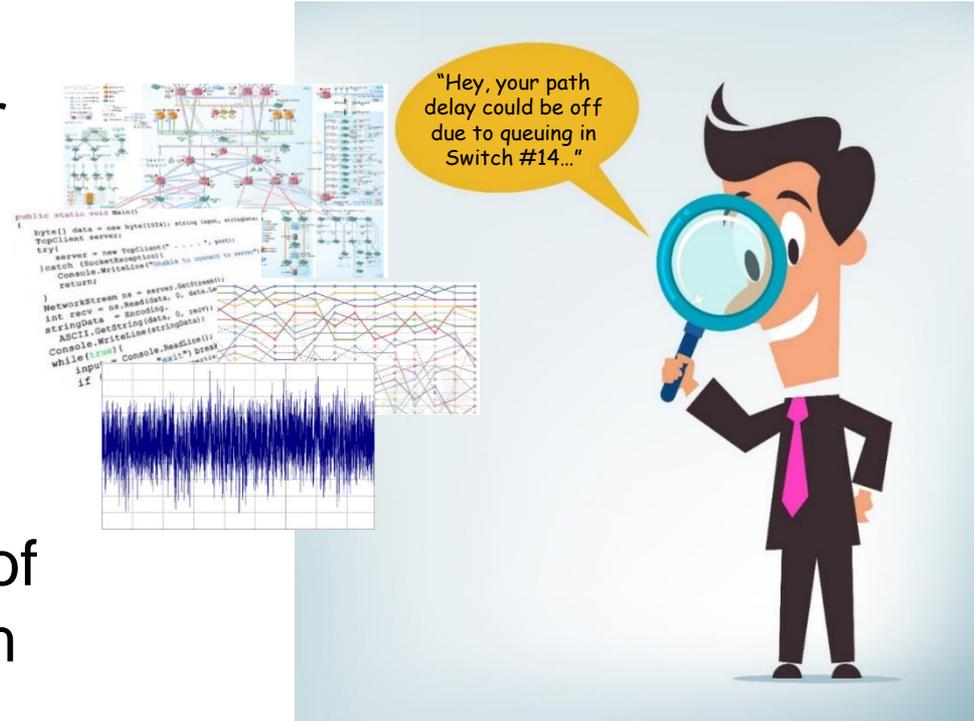
But these are just challenges
with complying

The thing about RTS 25...

- Firms must not only comply; they must demonstrate that they comply
- That's trickier than with other regs

Regulators themselves can't judge a technical implementation*

- And remember: the burden of proof is on the regulated firm
- This is a recipe for confusion and cost



* For that matter, neither can most compliance teams or senior execs

The key to demonstrating compliance

- The key is not a checklist of technologies
 - “We use GPS in every location with PTP to every server” (or whatever)
- There are many ways that great technologies can yield bad results
 - Trust me, we see it every day!
- The key is:
 - Testing
 - Monitoring
- These tell you how things actually work

Why is monitoring important for compliance?

- Can't rely just on testing
 - Testing says what can happen
 - Monitoring says what actually has happened
- Testing isn't perfect
 - E.g., lab conditions may not have anticipated a production scenario
- ESMA says so
 - “Relevant and proportionate monitoring of the system should be required...”

Why is testing important for compliance?

- Can't rely solely on monitoring
 - Driving by the rear-view mirror is not best practice
 - Some things can't be monitored, e.g.:
 - Application-level error
 - Error in holdover
- Can't rely on manufacturer specs
 - Sometimes wrong, usually ambiguous
 - Many solutions have no manufacturer to turn to
- ESMA says so
 - “Relevant and proportionate testing of the system should be required...”

So what are some best practices for testing?

- Test all unique configurations
- For each, cover all links in the traceability chain
- Test conditions at least as bad as production
 - Including foreseeable exception scenarios
- Use non-parametric statistics
- Integrate test data with monitoring data
- Very nice to have:
 - Automate execution and analysis of tests
 - Automate end-to-end traceability analysis

Best practices and standards

- Best practices evolve as the industry learns and compares notes
- A standards process codifies best practices
 - And updates them as best practices evolve
- Standards reduce costs
- Standards give regulators a reference point

The purpose of STAC-TS

- Provide testing standards and tools that reflect industry best practices
 - Software for traceability reporting
 - Software for load gen, measurement, analytics
- Enables firms to:
 - Justify traceability at any point in time
 - “Self certify” or get audits (e.g., annual compliance certification)
- Also provides basis for STAC to publish results using the standards

STAC-TS taxonomy

STAC-TS.CE6.STEADY
STAC-TS.CE6.SPOOF
STAC-TS.CE6.JAM
STAC-TS.CE6.LOSS
STAC-TS.CE6.RECOV

Time distribution to site
(GPS, GNSS, PTP from NL, etc.)

Enterprise time distribution
(infrastructure for NTP, PTP, PPS, etc.)

STAC-TS.CE7.STEADY
STAC-TS.CE7.SPOOF
STAC-TS.CE7.JAM
STAC-TS.CE7.LOSS
STAC-TS.CE7.RECOV

STAC-TS.NTE1
STAC-TS.NTE2
STAC-TS.PSE1
STAC-TS.PSE2
STAC-TS.CAP1
STAC-TS.CAP2
STAC-TS.CAP3

Network timestamping
(switches, NICs, capture cards, appliances, etc.)

Application timestamping
(APIs, C++, Java, .Net, VMs, etc.)

STAC-TS.PE1
STAC-TS.PE2
STAC-TS.ALE
STAC-TS.GRAN
STAC-TS.RES
STAC-TS.AVN1
STAC-TS.AVN2
STAC-TS.AVN3

STAC-TS.CE1.STEADY
STAC-TS.CE1.HLDVR
STAC-TS.CE1.RECOV

STAC-TS.CE2

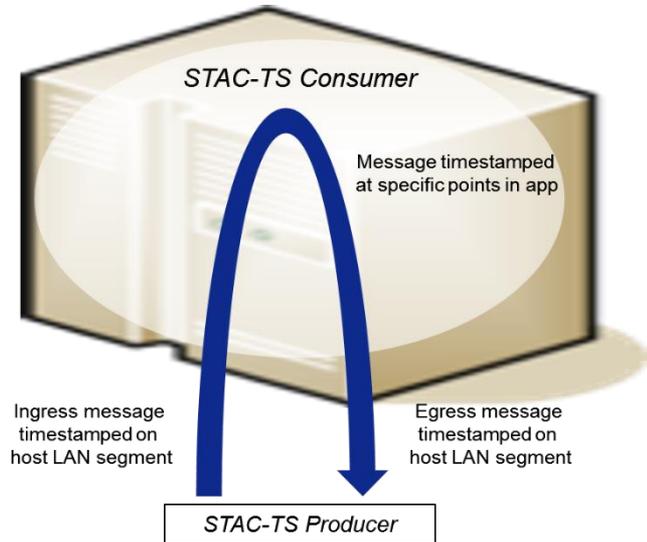
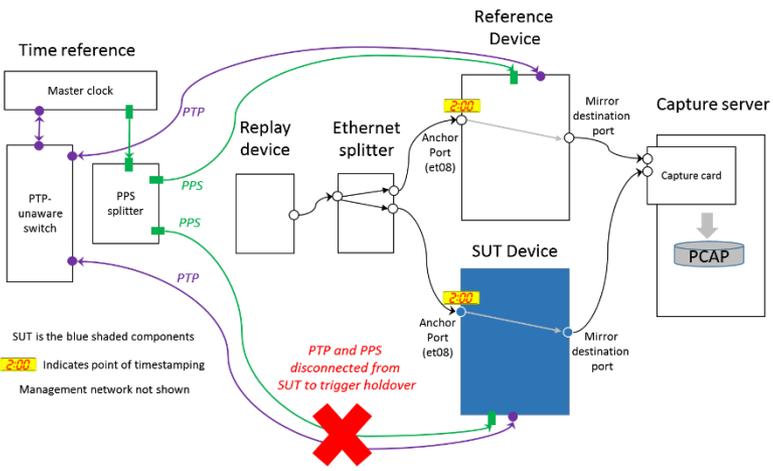
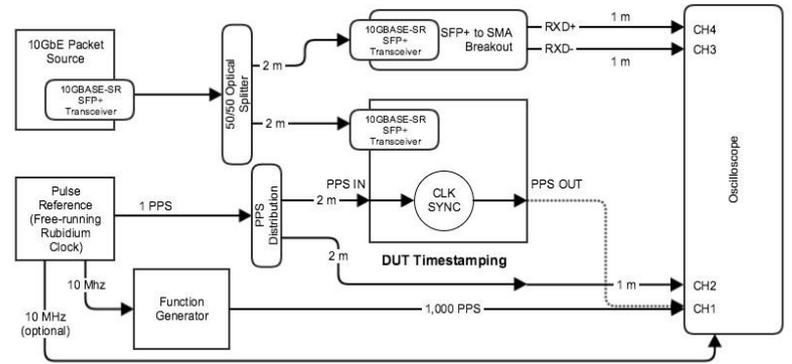
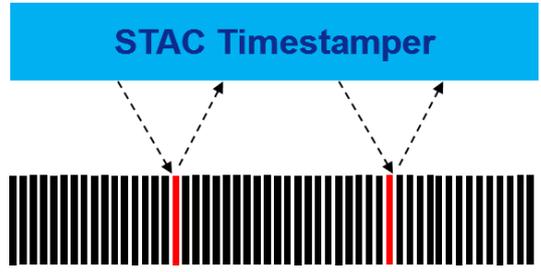
STAC-TS.AVN4

And more

And more

STAC-TS goal: Right tool for the job

Example: STAC-TS.ALE - A quick but thorough way to assess application-level error



Example: STAC-TS.AVN - An easy way to prove compliance of an entire solution

STAC Traceability Report (in development)

- Reports the accuracy of a timestamping point using its traceability chain
- Links the traceability chain to source data
- Integrates test and monitoring data
- Can draw from internal STAC-TS results and results on STAC site
- Run in batch to create STAC Traceability Survey

STAC Traceability Report Generated by: Firm XYZ Report date: 15 May 2018

NORMAL OPERATING RANGE

Timestamping Point ID: 334 Timestamping Point Class: 62
 Accuracy with respect to: APPLICATION EVENTS clock_gettime(CLOCK_REALTIME)
 C++ 11
 OS build 274

STAC Traceability Report Generated by: Firm XYZ Report date: 15 May 2018

INTERVAL ANALYSIS

Time Interval
 2018-04-17T08:00:00Z to 2018-04-19T16:30:00Z

Timestamping Point ID: 334 Timestamping Point Class: 62
 Accuracy with respect to: APPLICATION EVENTS clock_gettime(CLOCK_REALTIME)
 C++ 11
 OS build 274
 Server build 736
 PTP client x.y.z
 PTP LAN: segment A22

Highest Error Magnitude (microseconds)			Error Range (microseconds)		
99%	99.99%	Max	99%	99.99%	Max
2	6	81	0 +/- 1	3 +/- 3	40 +/- 41

Traceability Decomposition (microseconds)					
Component	99%	99.99%	Max	Exceptions	Source/Hyperlink
Platform type X.Y - Application-level error	1 +/- 1	3 +/- 3	40 +/- 40	None	STAC-TS.ALE results
Platform type X.Y, PTP solution A, network segment A22 - Host clock error	0 +/- 0	0 +/- 0	0 +/- 1	None	Loop stats (host daemon)
Acme PTP boundary clock model B2m	0 +/- 0	0 +/- 0	0 +/- 0	None	STAC-TS.CE1 results
Acme Grand Master Clock v2.4 as PTP master	0 +/- 0	0 +/- 0	0 +/- 0	None	STAC-TS.CE4 results
GPS signal	0 +/- 0	0 +/- 0	0 +/- 0	None	GPS bulletin

Summary

- Complying with RTS25 is not the only challenge
- Demonstrating compliance is the other
- Think about how to persuade a non-technologist that you comply
- Follow standards where they exist
- If you're interested in STAC-TS, see www.STACresearch.com/TS