AmLight ExP
Americas Lightpaths Express & Protect

CI Lunch and Learn / July 7th, 2023

Handling Microbursts @ AmLight – Part 2 of 2

Jeronimo Bezerra, Italo Valcy, David Miranda, David Ramirez, Renata Frez

<sdn@amlight.net>

Outline

- Part 1 – The challenge of detecting microbursts (April/2023)
  - What is a burst? When is a burst *micro*? Detecting microbursts: what's the challenge?
  - Some fundamentals about tools and protocols
  - The AmLight INT Collector 2.0: Our adaptive approach
  - Full talk: https://youtu.be/1x-aVZTyyiM

- **Part 2 – Why and when is detecting microbursts important?**
  - **When is a microburst an issue?**
    - **The Vera Rubin Observatory Use case**
  - **Lessons Learned, Conclusions, and Future**

Recap: What is a network microburst?

Network microbursts are sporadic bursts of traffic that occurs in *very short* timescales

"very short" varies per vendor and per author:

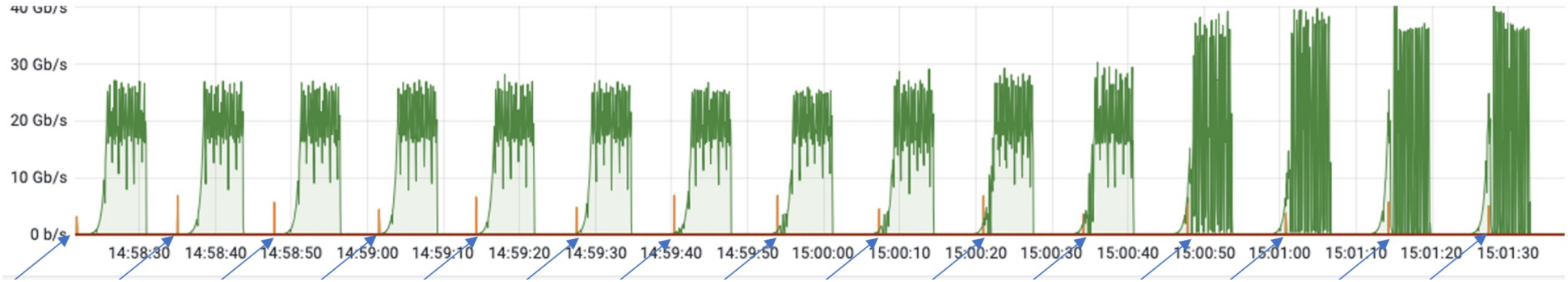
- Cisco and Facebook: In a **microsecond** time-scale
- Huawei, Arista, and most authors: In a **millisecond** time-scale
- Mine: In a time-scale my network monitoring system can't detect

Detecting microbursts is a complex activity due to the granularity required to observe those events:

- Most Network Monitoring Systems (NMS) or protocols (SNMP, NetFlow) were not made for it.

Not all microbursts are malicious by nature, but they can impact interface buffers and lead to packet drops and poor network performance.

Recap: What is a network microburst?



AmLight ExP
Americas Lightpaths Express & Protect

Research Questions

- In Part 2, answering "*why and when is detecting microbursts important?*" is the main goal.
- Our research questions:
  - *How short does a microburst have to be to become a problem?*
  - *How do TCP Congestion Control/Avoidance Algorithms react to microbursts?*
  - *When during a data transfer is a microburst the most dangerous?*
  - *How many TCP retransmits should I expect depending on the size of the microburst?*

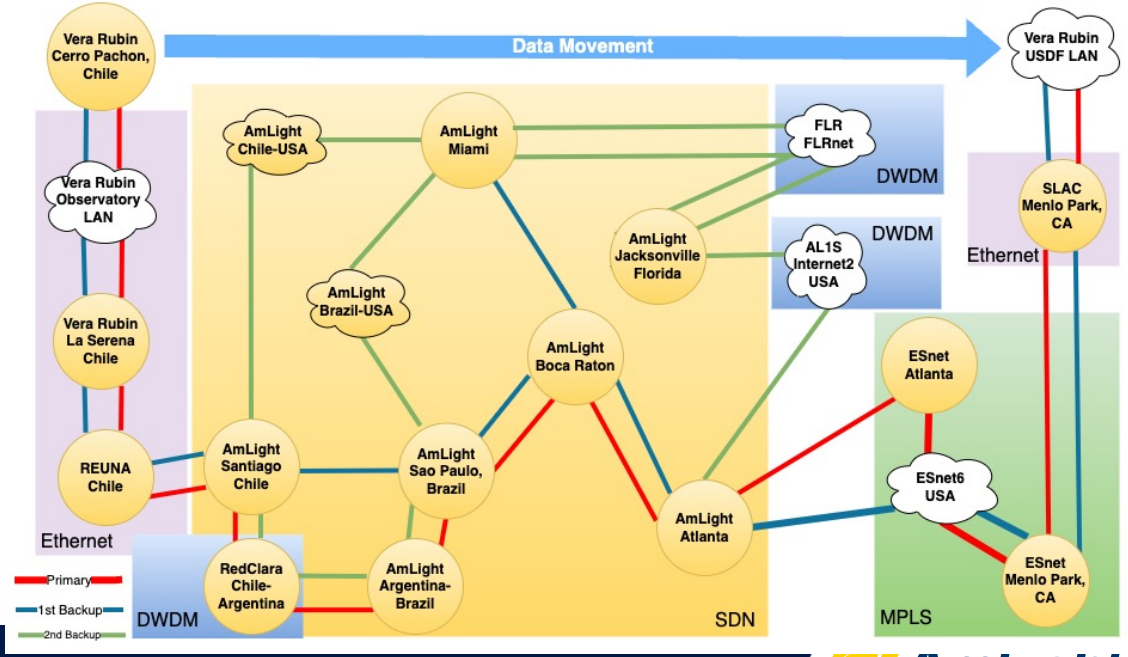
We modeled the Vera Rubin Observatory network *modus operandi* as our use case.

The Use Case: Vera Rubin Obs's operation

Vera Rubin is a large-aperture, wide-field, ground-based optical telescope under *installation* in northern Chile. ETD: Q42024

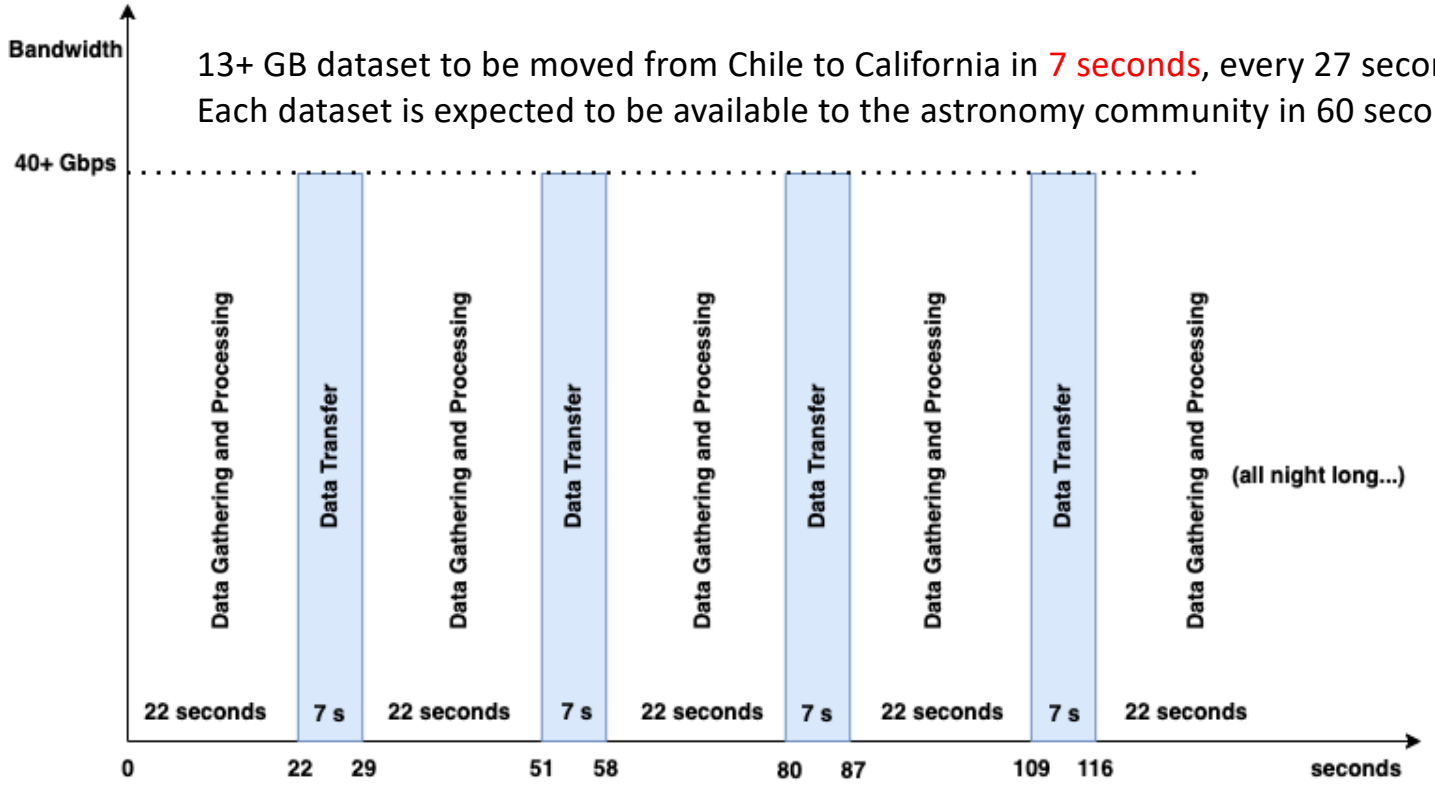
The Long-Haul Network (LHN) connecting Summit to the US Data Facility (USDF) is built over infrastructure provided by Rubin Obs., REUNA, RedClara, RNP, Rednesp, FIU/AmLight, FLR, ESnet, Internet2, and SLAC.

https://rubinobservatory.org/slideshows/welcome



The Use Case: Vera Rubin Obs's operation [2]

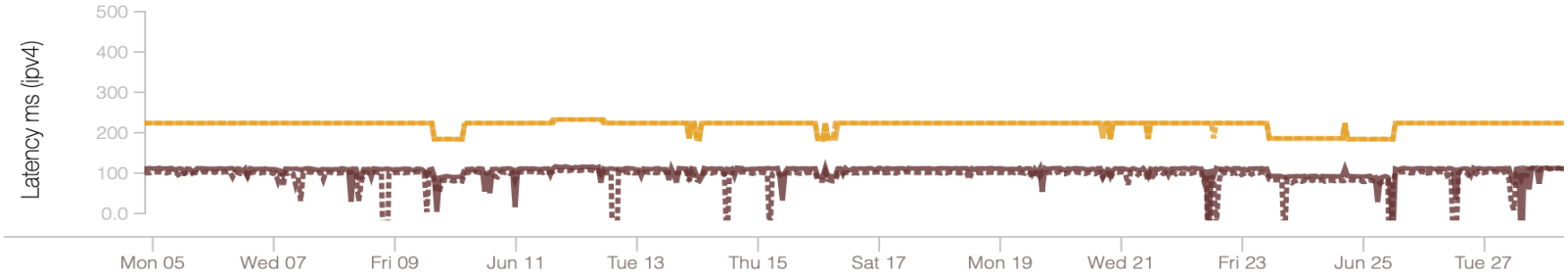
13+ GB dataset to be moved from Chile to California in 7 seconds, every 27 seconds, all night long.
Each dataset is expected to be available to the astronomy community in 60 seconds.



The Use Case: Vera Rubin Obs's operation [3]

Network/Data Transfer Challenges:

- High RTT from the Summit to USDF: ~220ms (see below)
- Complex troubleshooting: A packet loss of $1x10^{-3}$ is enough to compromise a Rubin Observatory 7-second-over-220ms data transfer.
  - Microbursts, damaged components (transceivers, connectors), dirty fiber, misconnected patch cords, etc., can lead to packet loss.



Playing with microbursts: The Experiment Methodology

- **Response Variables/Metrics: Flow Completion Time (FCT):**
  - **Target: Understanding how microbursts can impact the goal of transferring 13GB under 7 seconds over long-haul topologies.**

Methodology:

- Sender Node (SN) will send traffic to Receiver Node (RN) using iperf3 v3.9.
  - Tests are memory-to-memory. One stream.
- We will use two RTTs for experimentation: 209ms and 301ms.
  - 209ms: traffic will flow from Miami to Sao Paulo and back, from SN to RN and vice-versa
  - 301ms: traffic will flow from Miami to Chile via Sao Paulo, and back, from SN to RN and vice-versa
  - 1ms: direct connection in Miami for tuning only
  - All routes were tested with the packet generator for RFC2544, and we found no bit errors.
- Each experiment will have from 5 to 20 repetitions, depending on the goal.
- We simulate Vera Rubin datasets by using iperf3 option –n (-n 13G) to send 13 Gbytes of data.
- We will use TCP CCAs HTCP and BBR (Not BBRv2!)

The Experiment Methodology [2]

- **Response Variables/Metrics: Flow Completion Time (FCT) (ideal under 7 seconds)**

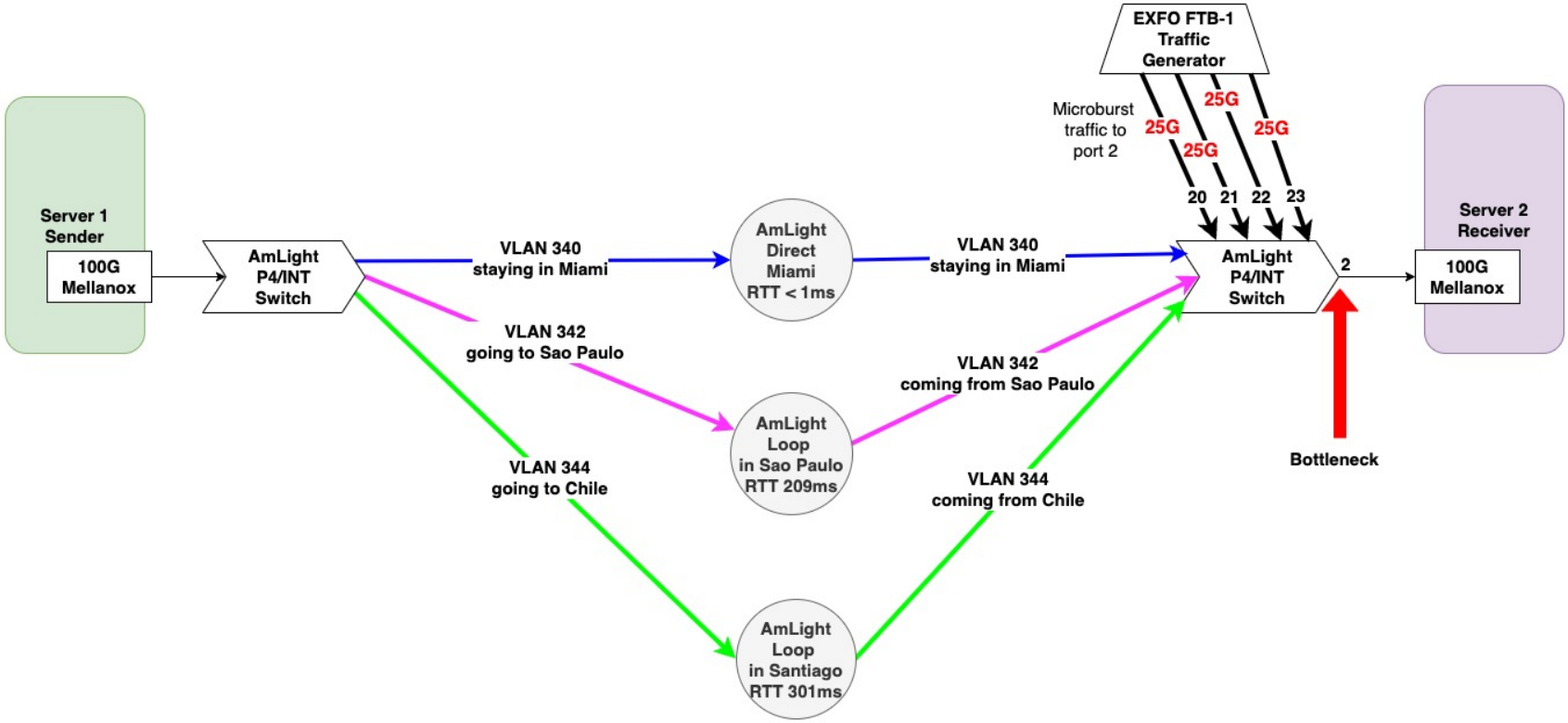
- We used a traffic generator to create a 25Gbps microburst that will be sent out via 4 x 40G interfaces (incast), totaling 100Gbps of traffic. (see slide 44)
- Experiments used microbursts of 25ms, 50ms, 100ms, 500ms, 1000ms, and 2000ms.
- Iperf3 traffic and the four 25Gbps microburst flows share the 100GE port #2.

- **Tuning, host configs, and microburst creation are provided at the end of the presentation.**

Disclaimer!

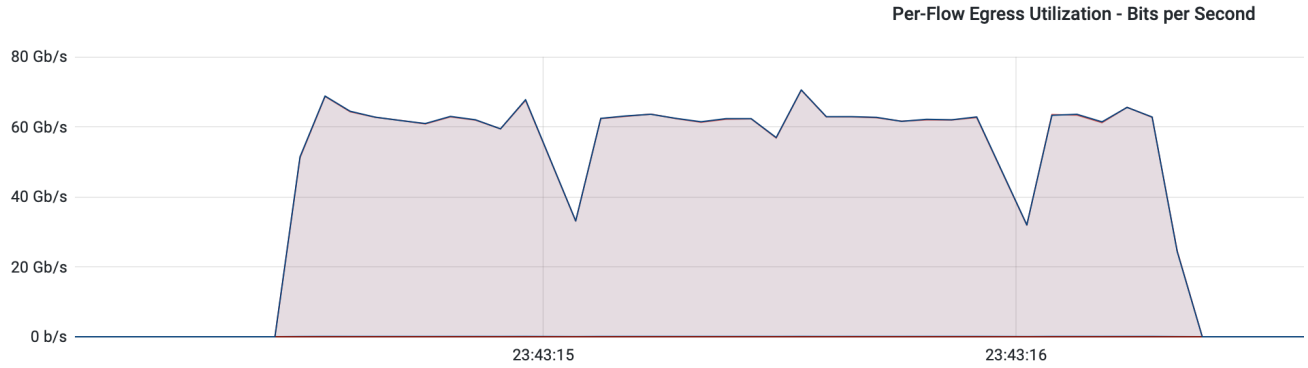
- Vera Rubin Data Management won't use iperf3 for data movement.
- Vera Rubin Data Management will use 10x sender nodes and 10x receiver nodes, each with 10G NICs, not just one sender and receiver like our testbed.
- This talk is merely informational. It is not our goal to influence how DM or any network operation should be performed.

The Experiment Methodology [3]



The Experiment Methodology [4]

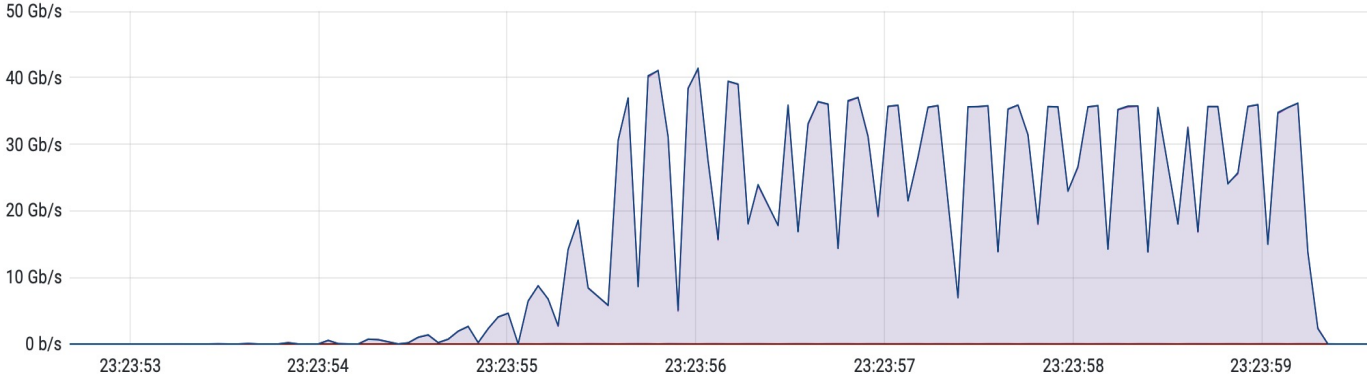
- Baseline for 1ms RTT
  - *Just for tuning and understanding*
- *With 1ms RTT and 1 single core, we reached peaks of 65Gbps and 59.6Gbps on average.*
- *Iperf3 traffic with no special options, just –n 13G.*
- *Used TCP HTCP*




```
Thu Jul  6 23:43:15 2023 [ ID] Interval           Transfer     Bitrate         Retr  Cwnd
Thu Jul  6 23:43:15 2023 [  5]   0.00-1.00   sec  6.93 GBytes  59.5 Gbits/sec   45   3.97 MBytes
Thu Jul  6 23:43:16 2023 [  5]   1.00-1.87   sec  6.07 GBytes  59.7 Gbits/sec    4   5.36 MBytes
Thu Jul  6 23:43:16 2023 - - - - - - - - - - - - - - - - - - - - - - - - -
Thu Jul  6 23:43:16 2023 [ ID] Interval           Transfer     Bitrate         Retr
Thu Jul  6 23:43:16 2023 [  5]   0.00-1.87   sec  13.0 GBytes  59.6 Gbits/sec   49             sender
Thu Jul  6 23:43:16 2023 [  5]   0.00-1.87   sec  13.0 GBytes  59.5 Gbits/sec                  receiver
```

The Experiment Methodology [5]

- Baseline for 209 ms RTT
- *With 209ms RTT and 1 single core, we reached peaks of 41Gbps and 25+ Gbps on average after TCP Slow Start.*
- *Iperf3 traffic with no special options, just –n 13G.*
- *Used TCP HTCP*

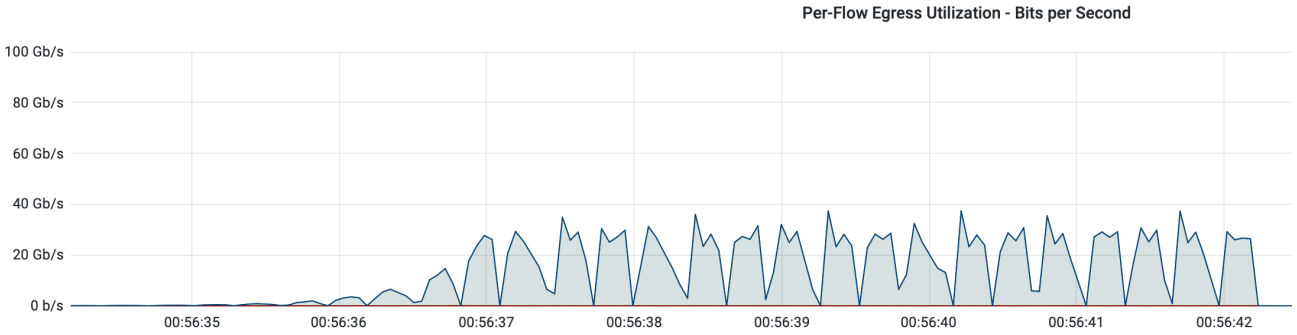



```
Thu Jul  6 23:23:53 2023 [ ID] Interval           Transfer     Bitrate         Retr  Cwnd
Thu Jul  6 23:23:53 2023 [  5]   0.00-1.00   sec  21.2 MBytes   178 Mbits/sec    0   8.72 MBytes
Thu Jul  6 23:23:54 2023 [  5]   1.00-2.00   sec   191 MBytes  1.60 Gbits/sec    0   63.7 MBytes
Thu Jul  6 23:23:55 2023 [  5]   2.00-3.00   sec  1.92 GBytes  16.5 Gbits/sec    0   1.40 GBytes
Thu Jul  6 23:23:56 2023 [  5]   3.00-4.00   sec  3.28 GBytes  28.2 Gbits/sec    0   1.40 GBytes
Thu Jul  6 23:23:57 2023 [  5]   4.00-5.00   sec  3.35 GBytes  28.7 Gbits/sec    0   1.40 GBytes
Thu Jul  6 23:23:58 2023 [  5]   5.00-6.00   sec  3.41 GBytes  29.3 Gbits/sec    0   1.40 GBytes
Thu Jul  6 23:24:00 2023 [  5]   6.00-6.28   sec   854 MBytes  25.9 Gbits/sec    0   1.40 GBytes
Thu Jul  6 23:24:00 2023 - - - - - - - - - - - - - - - - - - - - - - - - -
Thu Jul  6 23:24:00 2023 [ ID] Interval           Transfer     Bitrate         Retr
Thu Jul  6 23:24:00 2023 [  5]   0.00-6.28   sec  13.0 GBytes  17.8 Gbits/sec    0             sender
Thu Jul  6 23:24:00 2023 [  5]   0.00-6.49   sec  13.0 GBytes  17.2 Gbits/sec                  receiver
```


AmLight ExP
Americas Lightpaths Express & Protect

The Experiment Methodology [6]

- Baseline for 301ms RTT
- *With 301ms RTT and 1 single core, we reached peaks of 39Gbps and 20+ Gbps on average after TCP Slow Start.*
- *Iperf3 traffic with no special options, just –n 13G.*
- *Used TCP HTCP*



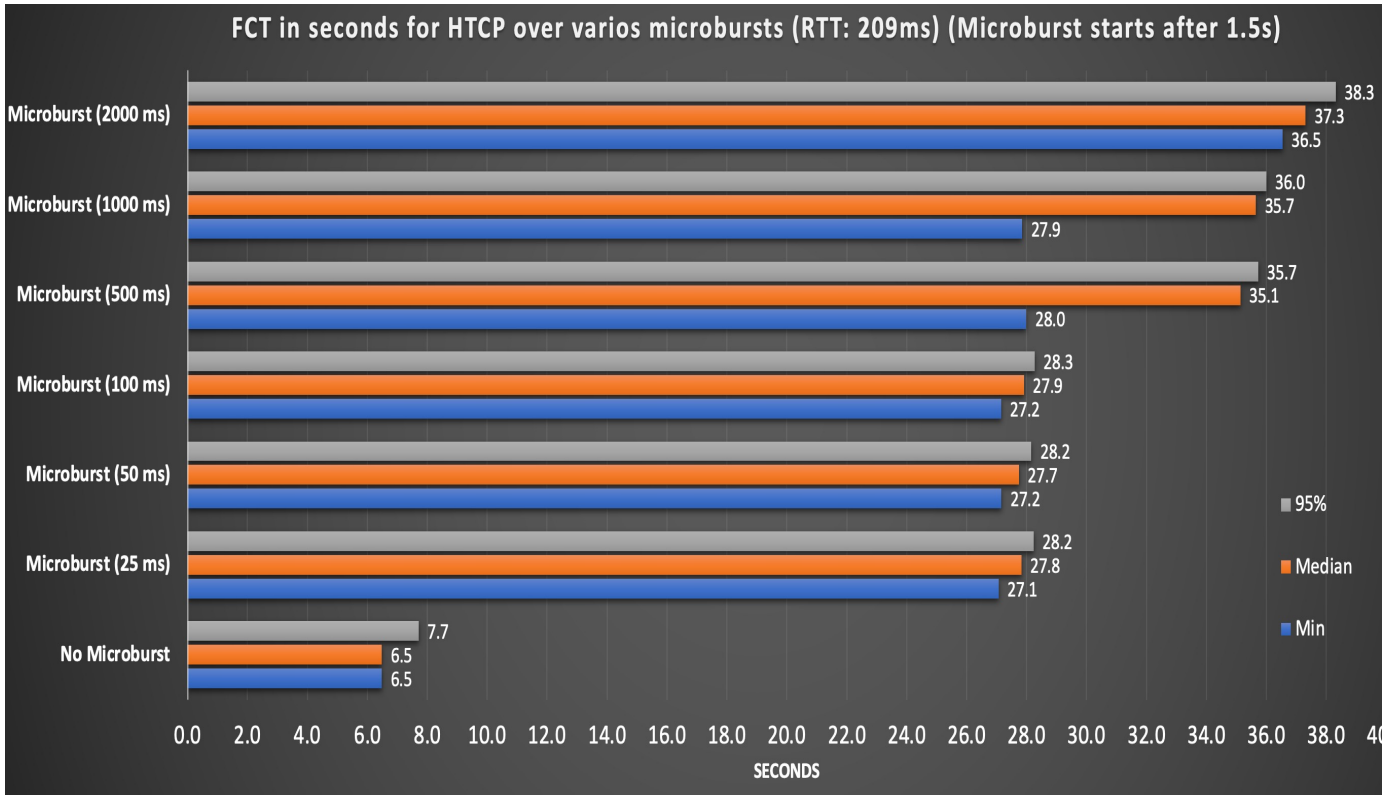

```
Fri Jul  7 00:56:33 2023 [ ID] Interval           Transfer     Bitrate         Retr  Cwnd
Fri Jul  7 00:56:33 2023 [  5]   0.00-1.00   sec  1.97 MBytes  16.5 Mbits/sec    0    250 KBytes
Fri Jul  7 00:56:34 2023 [  5]   1.00-2.00   sec  5.70 MBytes  47.9 Mbits/sec    0   2.28 MBytes
Fri Jul  7 00:56:35 2023 [  5]   2.00-3.00   sec  56.2 MBytes   472 Mbits/sec    0   18.8 MBytes
Fri Jul  7 00:56:36 2023 [  5]   3.00-4.00   sec   605 MBytes  5.08 Gbits/sec    0    198 MBytes
Fri Jul  7 00:56:37 2023 [  5]   4.00-5.00   sec  1.80 GBytes  15.5 Gbits/sec    0   1.40 GBytes
Fri Jul  7 00:56:38 2023 [  5]   5.00-6.00   sec  2.21 GBytes  19.0 Gbits/sec    0   1.40 GBytes
Fri Jul  7 00:56:39 2023 [  5]   6.00-7.00   sec  2.35 GBytes  20.2 Gbits/sec    0   1.40 GBytes
Fri Jul  7 00:56:40 2023 [  5]   7.00-8.00   sec  2.37 GBytes  20.3 Gbits/sec    0   1.40 GBytes
Fri Jul  7 00:56:41 2023 [  5]   8.00-9.00   sec  2.25 GBytes  19.4 Gbits/sec    0   1.40 GBytes
Fri Jul  7 00:56:43 2023 [  5]   9.00-9.55   sec  1.37 GBytes  21.5 Gbits/sec    0   1.40 GBytes
Fri Jul  7 00:56:43 2023 - - - - - - - - - - - - - - - - - - - - - - - - -
Fri Jul  7 00:56:43 2023 [ ID] Interval           Transfer     Bitrate         Retr
Fri Jul  7 00:56:43 2023 [  5]   0.00-9.55   sec  13.0 GBytes  11.7 Gbits/sec    0             sender
Fri Jul  7 00:56:43 2023 [  5]   0.00-9.85   sec  13.0 GBytes  11.3 Gbits/sec                  receiver
```

- *Question 1: How short does a microburst have to be to become a problem to Vera Rubin data transfers?*

- *Question 2: How do TCP Congestion Control/Avoidance Algorithms react to microbursts?*

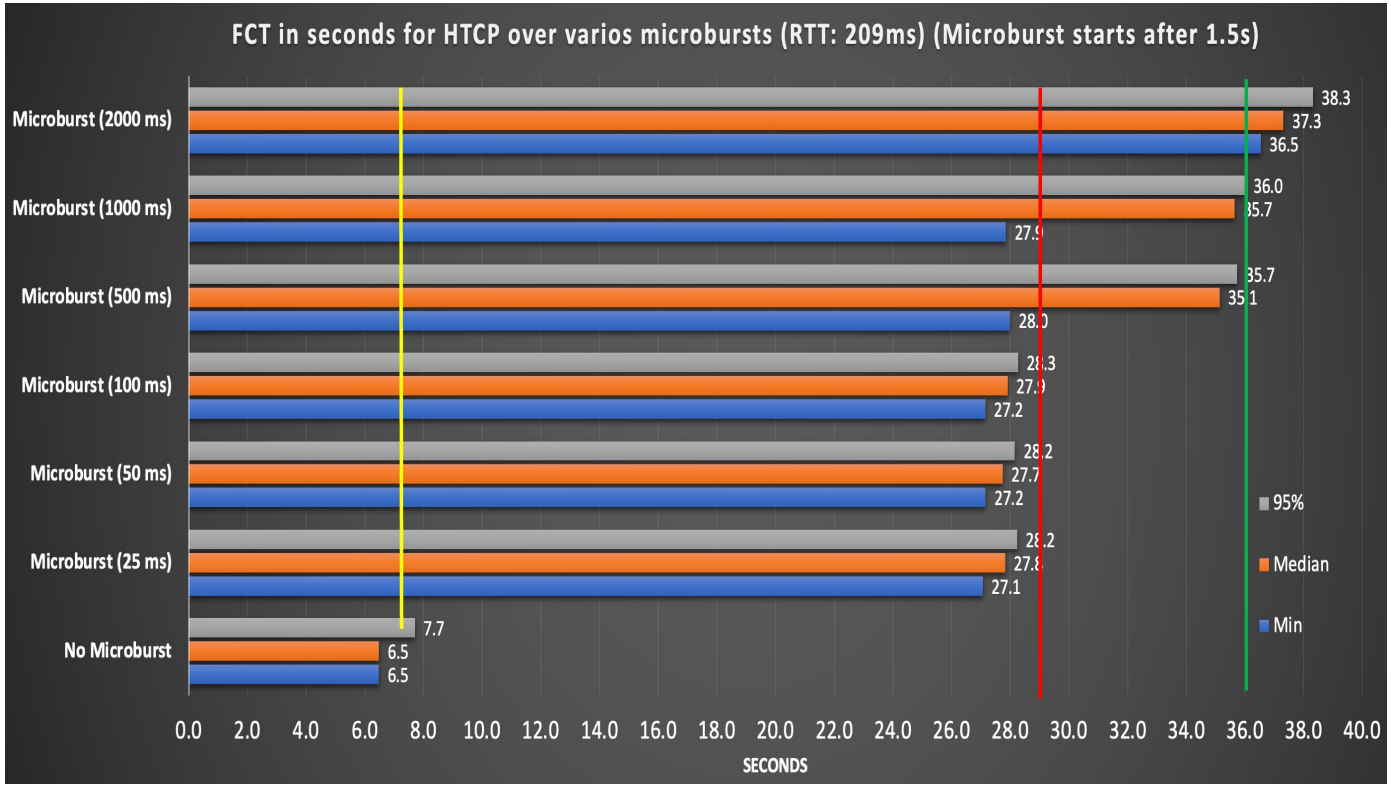
Flow Completion Time vs Microbursts vs HTCP (RTT 209ms)

- CCA: HTCP
- Baseline with no microbursts:
  - FCT of 6.5 seconds
- Microbursts generated after 1.5s:
  - 25, 50, 100, 500, 1000, 2000ms



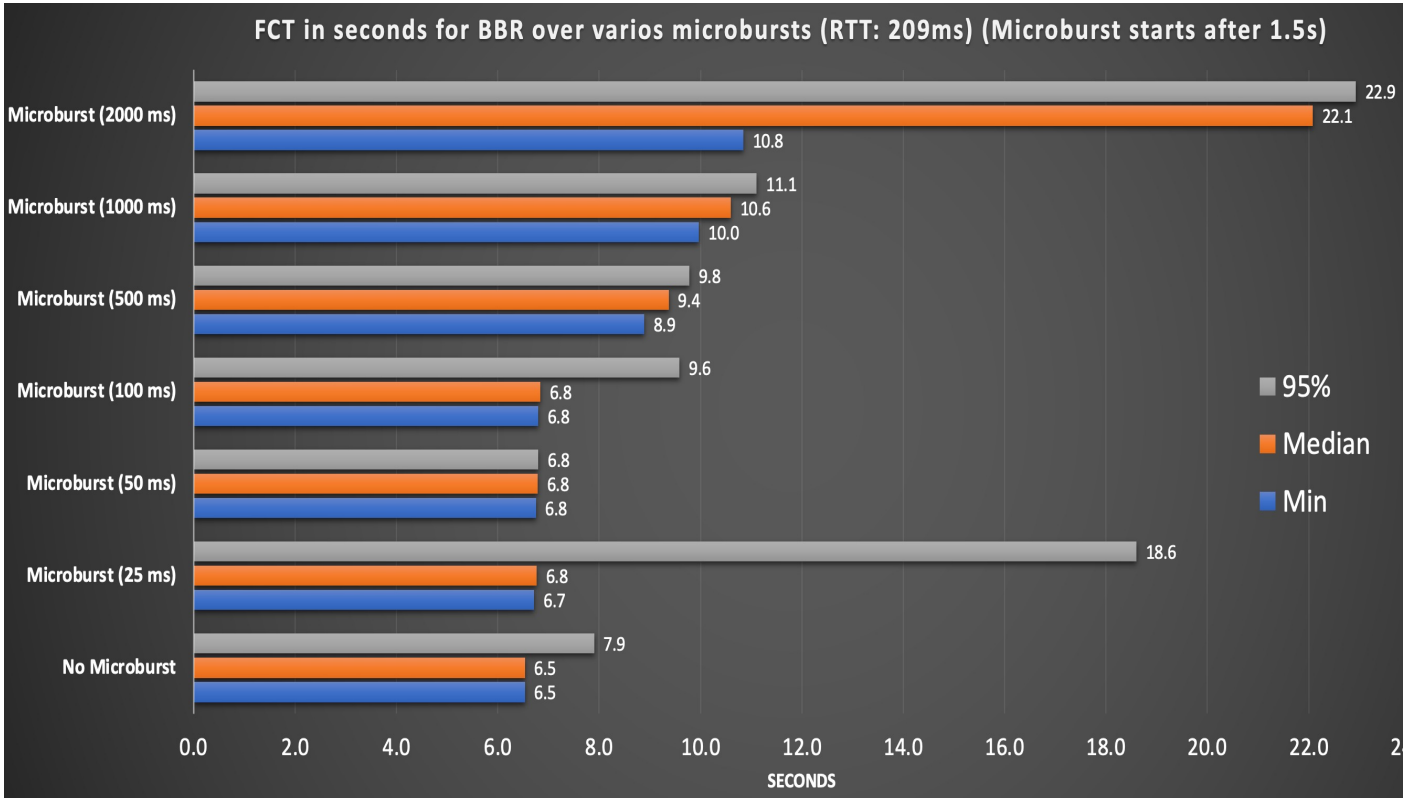
Flow Completion Time vs Microbursts vs HTCP (RTT 209ms)

- CCA: HTCP
- Baseline with no microbursts:
  - FCT of 6.5 seconds
- Microbursts generated after 1.5s:
  - 25, 50, 100, 500, 1000, 2000ms
- Findings:
  - Extreme poor performance!
  - FCT collides with the next data transfer window (cascade effect!)
  - 29 second marks the next data transfer window (red line)
  - 36 second marks the end of the next data transfer window (green line)



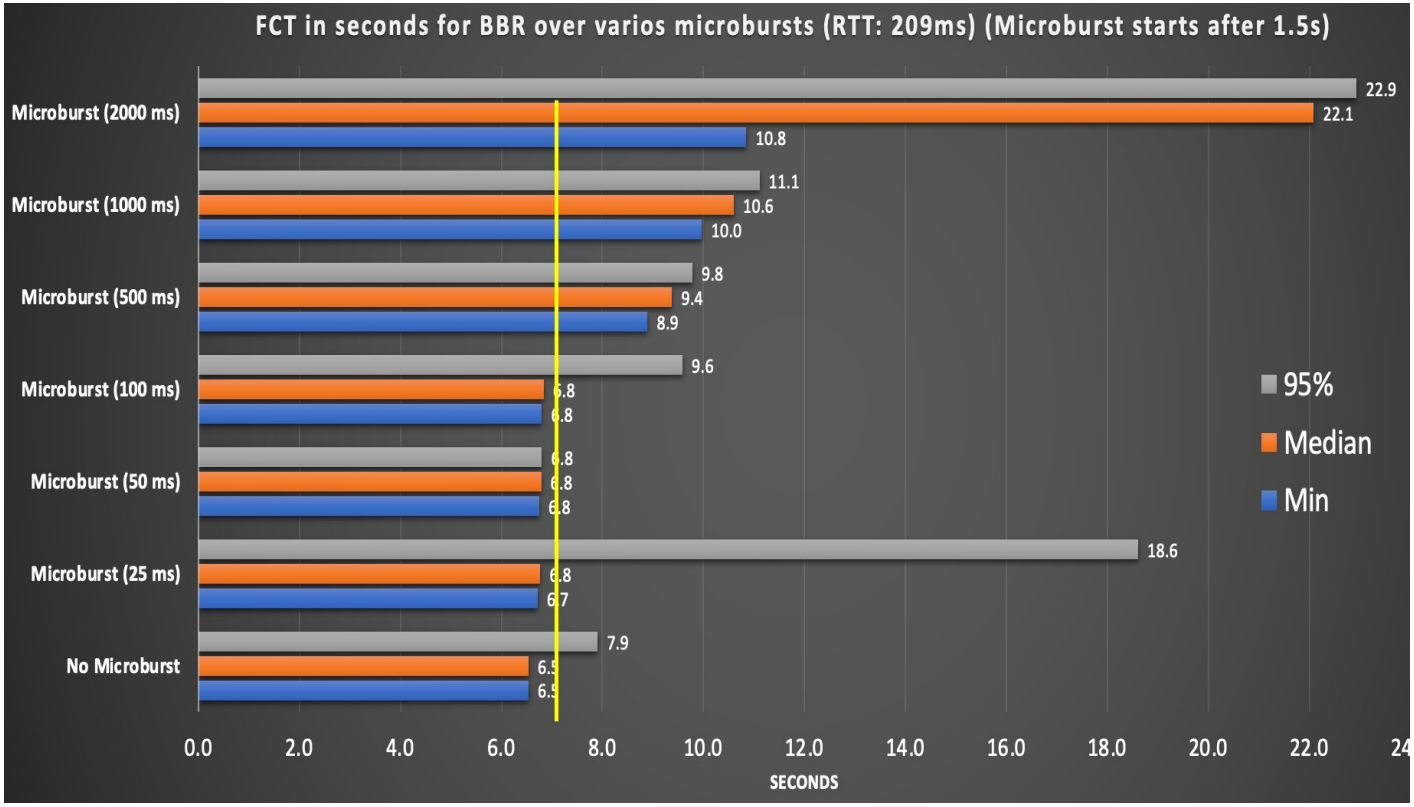
Flow Completion Time vs Microbursts vs BBR (RTT 209ms)

- CCA: BBR
- Baseline with no microbursts:
  - FCT of 6.5 seconds
- Microbursts generated after 1.5s:
  - 25, 50, 100, 500, 1000, 2000ms

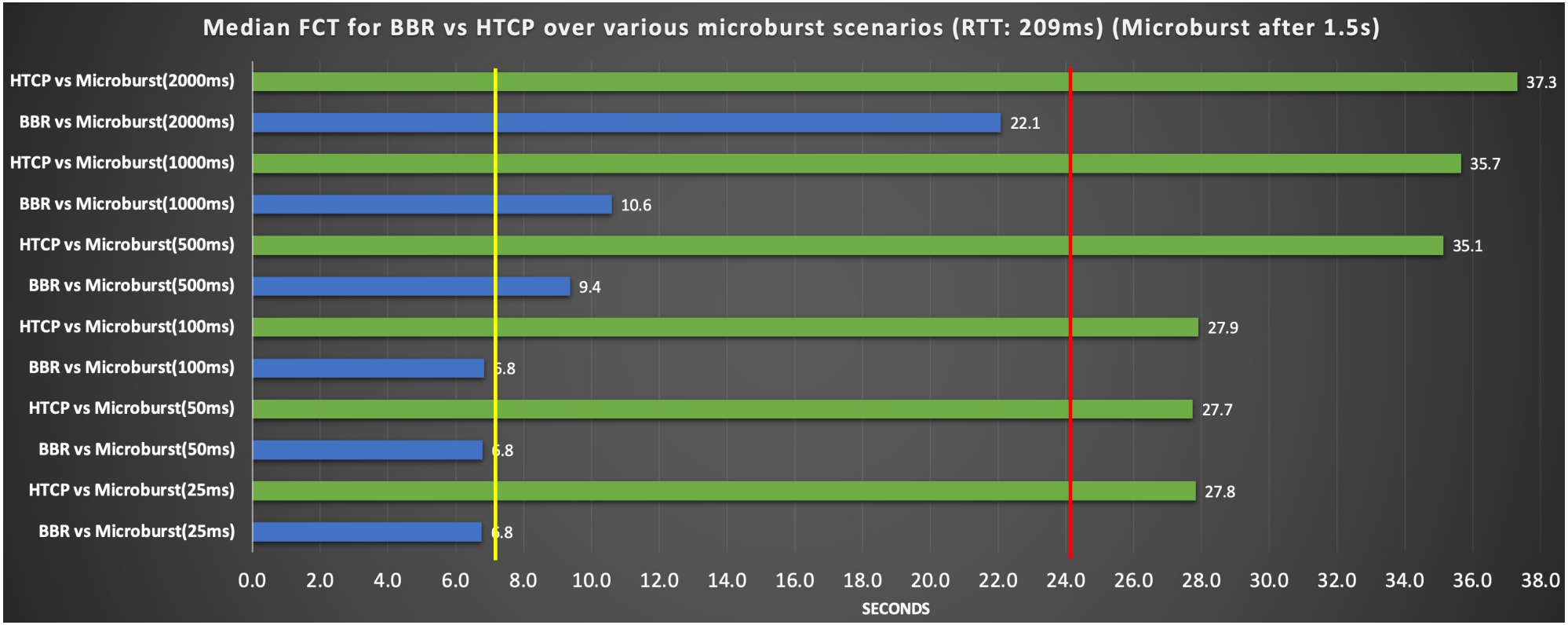


Flow Completion Time vs Microbursts vs BBR (RTT 209ms)

- CCA: BBR
- Baseline with no microbursts:
  - FCT of 6.5 seconds
- Microbursts generated after 1.5s:
  - 25, 50, 100, 500, 1000, 2000ms
- Findings:
  - BBR handled microbursts up to 100ms.
  - Interesting results for microbursts lasting up to 1000ms
  - FCT does NOT collide with the next data transfer window even with microbursts lasting 2000ms.

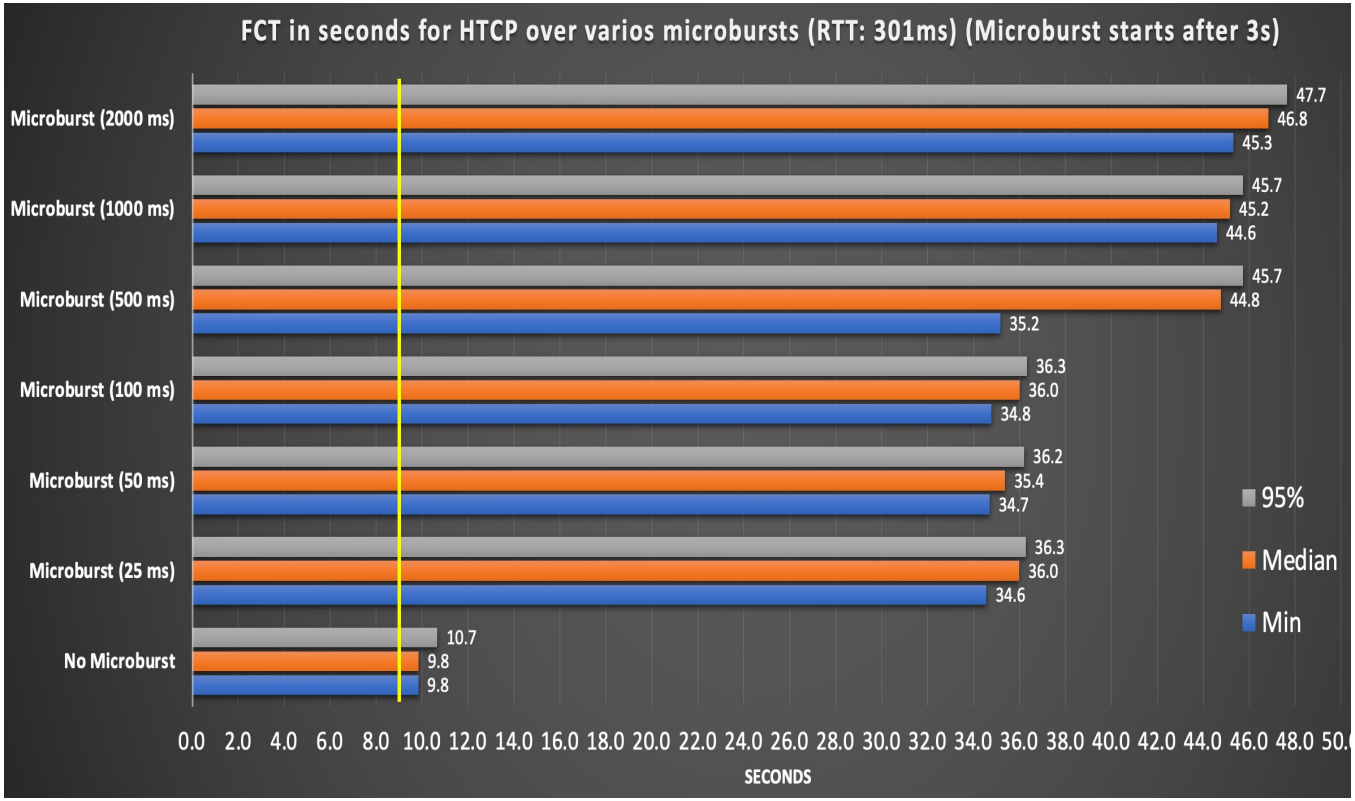


FCT vs. BBR vs. HTCP (RTT 209ms) - Comparison



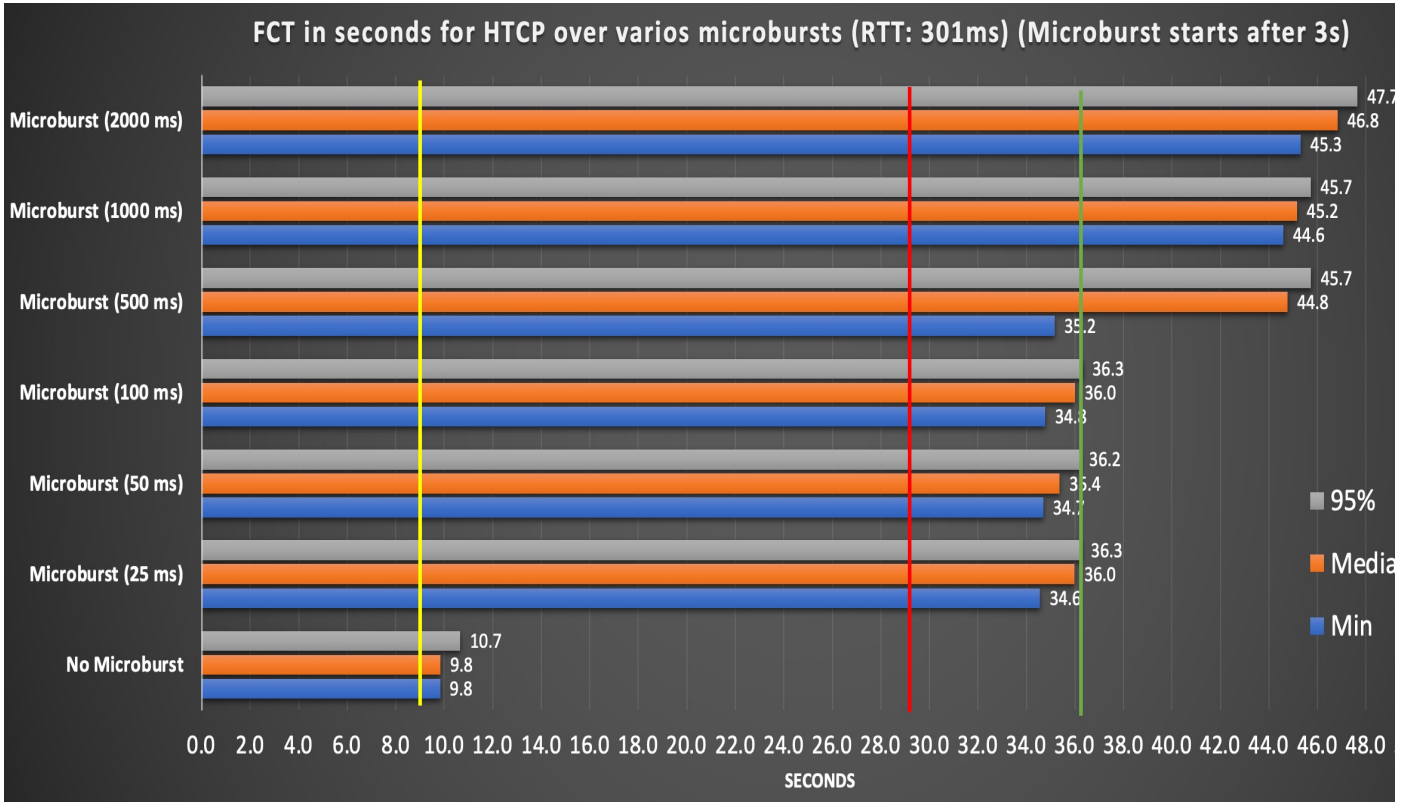
Flow Completion Time vs Microbursts vs HTCP (RTT 301ms)

- CCA: HTCP
- Baseline with no microbursts:
  - FCT of 9.8 seconds
- Microbursts generated after 3s:
  - 25, 50, 100, 500, 1000, 2000ms



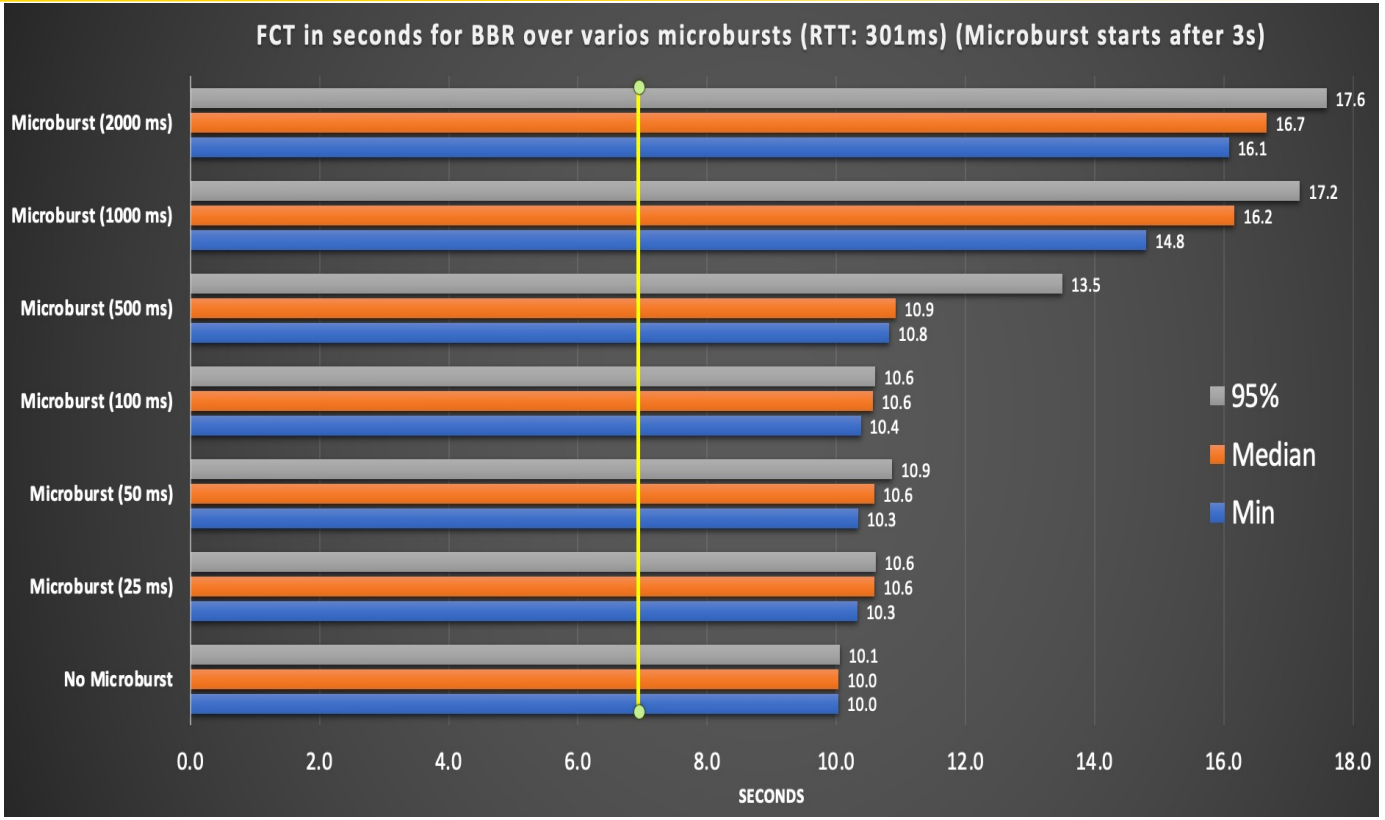
Flow Completion Time vs Microbursts vs HTCP (RTT 301ms)

- CCA: HTCP
- Baseline with no microbursts:
  - FCT of 9.8 seconds
- Microbursts generated after 3s:
  - 25, 50, 100, 500, 1000, 2000ms
- Findings:
  - Extreme poor performance!
    - Average 9 seconds longer than 201ms.
  - FCT collides with the next data transfer window (cascade effect!)
  - 27 second marks the next data transfer window (red line)
  - 34 second marks the end of the next data transfer window (green line)

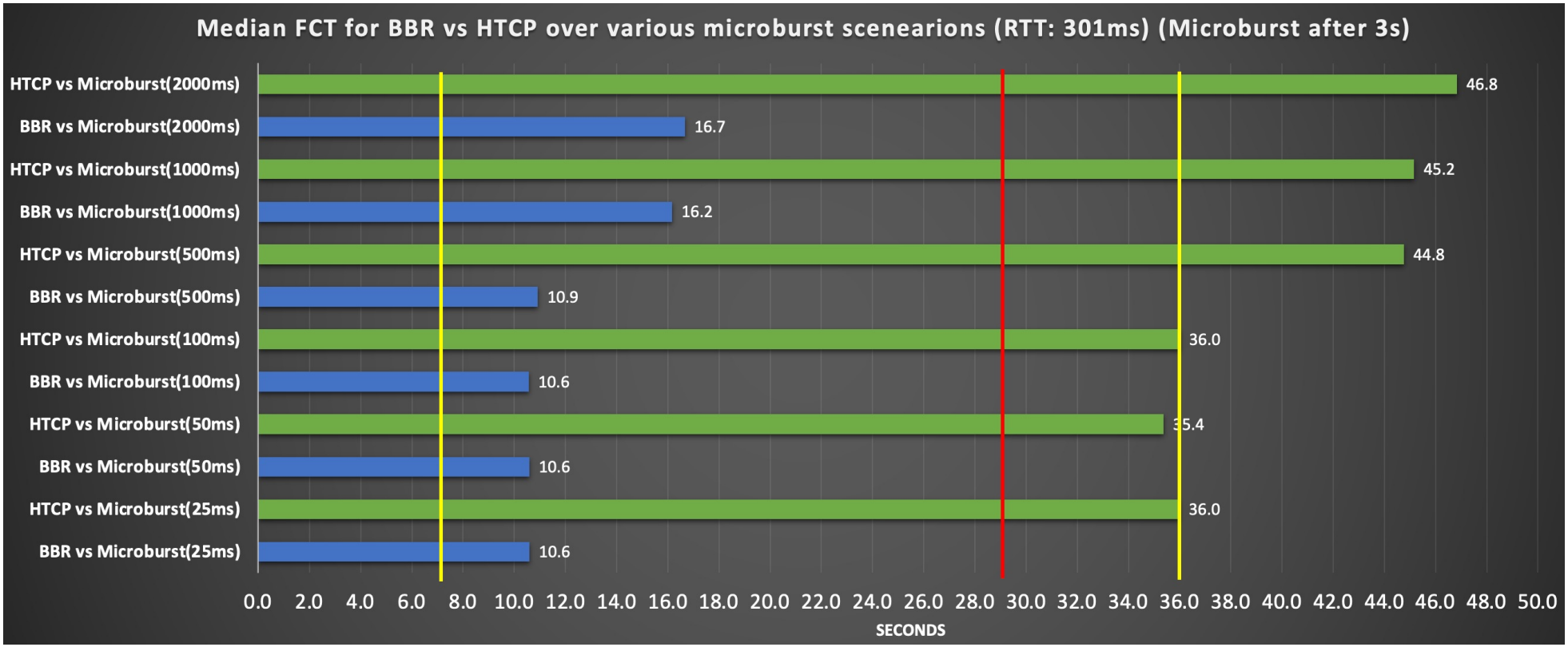


Flow Completion Time vs Microbursts vs BBR (RTT 301ms)

- CCA: BBR
- Baseline with no microbursts:
  - FCT of 10 seconds
- Microbursts generated after 3s:
  - 25, 50, 100, 500, 1000, 2000ms
- Findings:
  - Better performance than HTCP but going over the 7-second limit.
    - Average 4 seconds longer than 209ms.
  - Interesting results for microbursts lasting up to 500ms
  - FCT does NOT collide with the next data transfer window even with microbursts lasting 2000ms.



FCT vs. BBR vs. HTCP (RTT 301ms) - Comparison



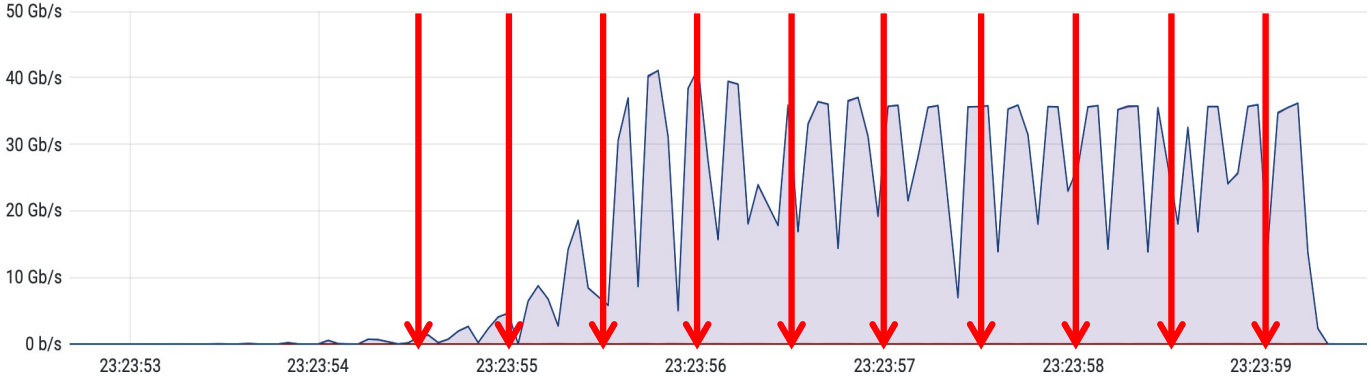
Answering Question 1 and Question 2

- *Question 1: How short does a microburst have to be to become a problem to Vera Rubin data transfers?*
- *Question 2: How do TCP Congestion Control/Avoidance Algorithms react to microbursts?*
- Without properly addressing the TCP Slow Start, neither HTCP nor BBR managed to complete the data transfers under 7 seconds for RTT of 301ms.
- When using HTCP, **all** microbursts affected the FCT due to packet drops caused by full queue occupancy on port 2.
- More robust, BBR managed to handle microburst up to 500ms with *acceptable* FCT.
  - BBR proved to be 20x more tolerant to microbursts than HTCP.

Question 3: When during the TCP Slow Start is a microburst the most impactful?

TCP Slow Start vs. microburst

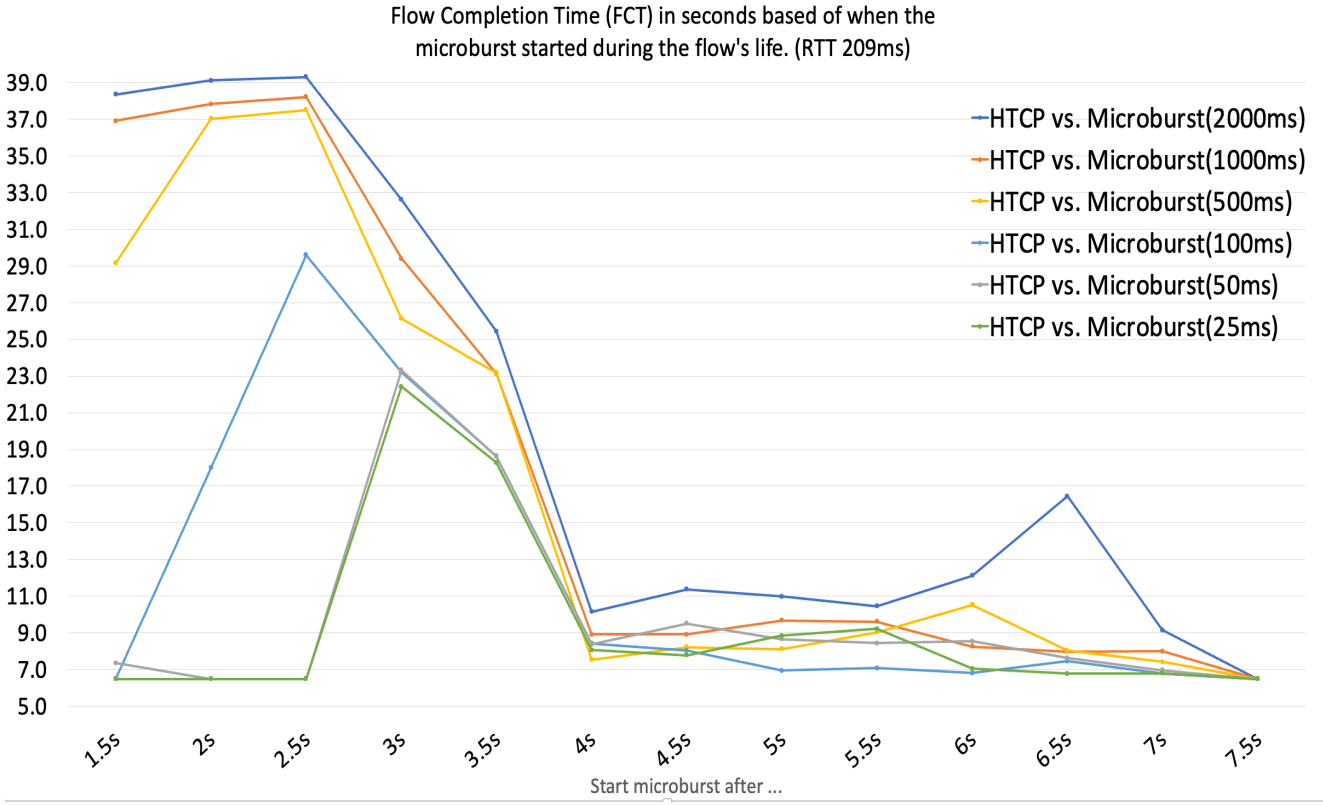
- Iperf3 takes ~1 second to start (control).
- Once iperf3 starts, on average, it takes 3-4 seconds to achieve full bandwidth for the tuning performed
- The goal is to understand the impact of a microburst during the TCP Slow Start phase.
- We will create microbursts every 0.5s from 1.5 to 7.5 seconds (red lines on the graph).
  - Some microbursts happened after the flow is over when RTT is 209ms.




```
Thu Jul  6 23:23:53 2023 [ ID] Interval           Transfer     Bitrate         Retr  Cwnd
Thu Jul  6 23:23:53 2023 [  5]   0.00-1.00   sec  21.2 MBytes   178 Mbits/sec    0   8.72 MBytes
Thu Jul  6 23:23:54 2023 [  5]   1.00-2.00   sec   191 MBytes  1.60 Gbits/sec    0   63.7 MBytes
Thu Jul  6 23:23:55 2023 [  5]   2.00-3.00   sec  1.92 GBytes  16.5 Gbits/sec    0   1.40 GBytes
Thu Jul  6 23:23:56 2023 [  5]   3.00-4.00   sec  3.28 GBytes  28.2 Gbits/sec    0   1.40 GBytes
Thu Jul  6 23:23:57 2023 [  5]   4.00-5.00   sec  3.35 GBytes  28.7 Gbits/sec    0   1.40 GBytes
Thu Jul  6 23:23:58 2023 [  5]   5.00-6.00   sec  3.41 GBytes  29.3 Gbits/sec    0   1.40 GBytes
Thu Jul  6 23:24:00 2023 [  5]   6.00-6.28   sec   854 MBytes  25.9 Gbits/sec    0   1.40 GBytes
Thu Jul  6 23:24:00 2023 - - - - - - - - - - - - - - - - - - - - - - - - -
Thu Jul  6 23:24:00 2023 [ ID] Interval           Transfer     Bitrate         Retr
Thu Jul  6 23:24:00 2023 [  5]   0.00-6.28   sec  13.0 GBytes  17.8 Gbits/sec    0             sender
Thu Jul  6 23:24:00 2023 [  5]   0.00-6.49   sec  13.0 GBytes  17.2 Gbits/sec                  receiver
```

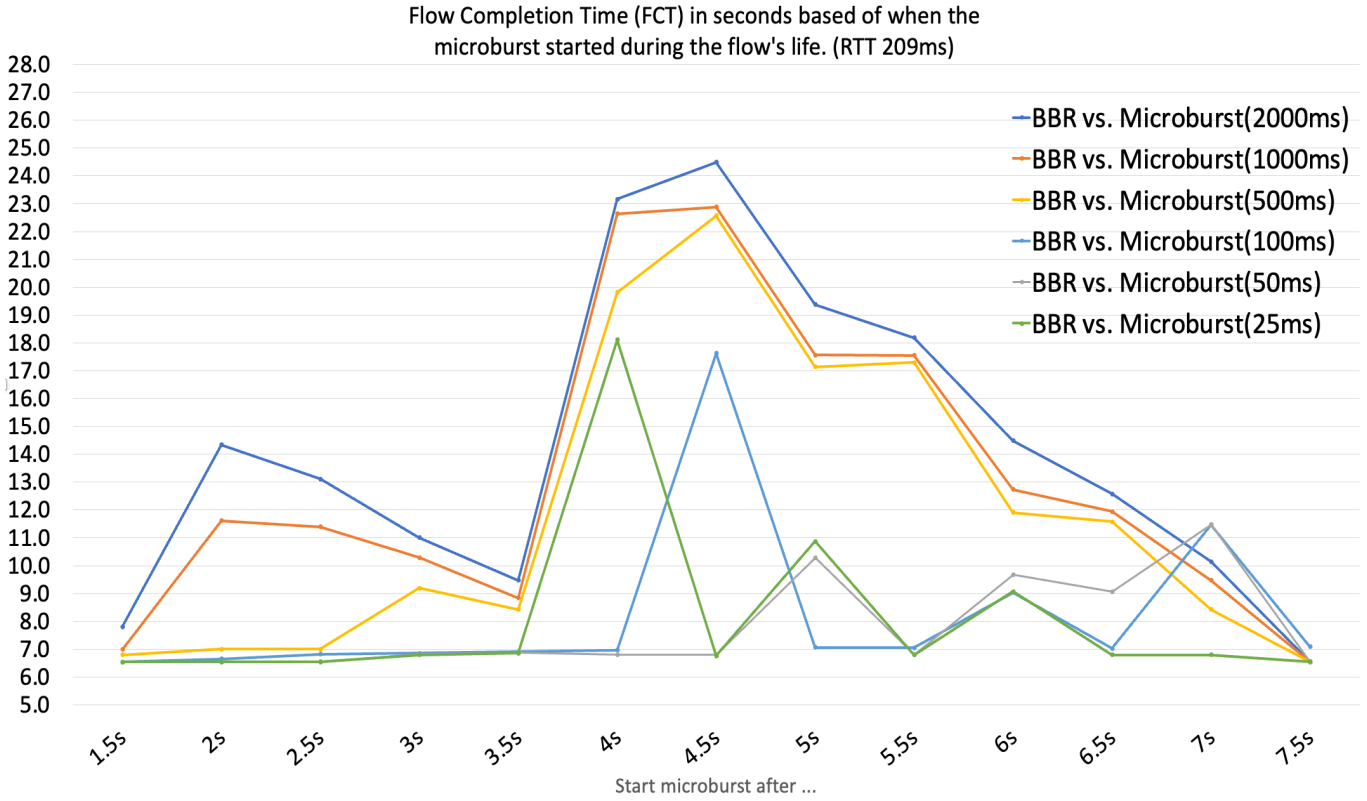
HTCP – Microbursts starting 1.5 to 7.5 seconds after iperf3 with 209ms RTT [2]

- HTCP struggles the most with microbursts happening at the beginning, from 1.5s to 4s
- The duration of the microburst, as expected, has a direct impact: the longer the burst, longer the FCT.
- After 4 seconds, the impact of microbursts are minimized by the TCP Congestion Control window being almost fully established.
- Highest FCT was 39 seconds for microbursts of 2000ms happening after 2.5 seconds of the beginning of the flow.

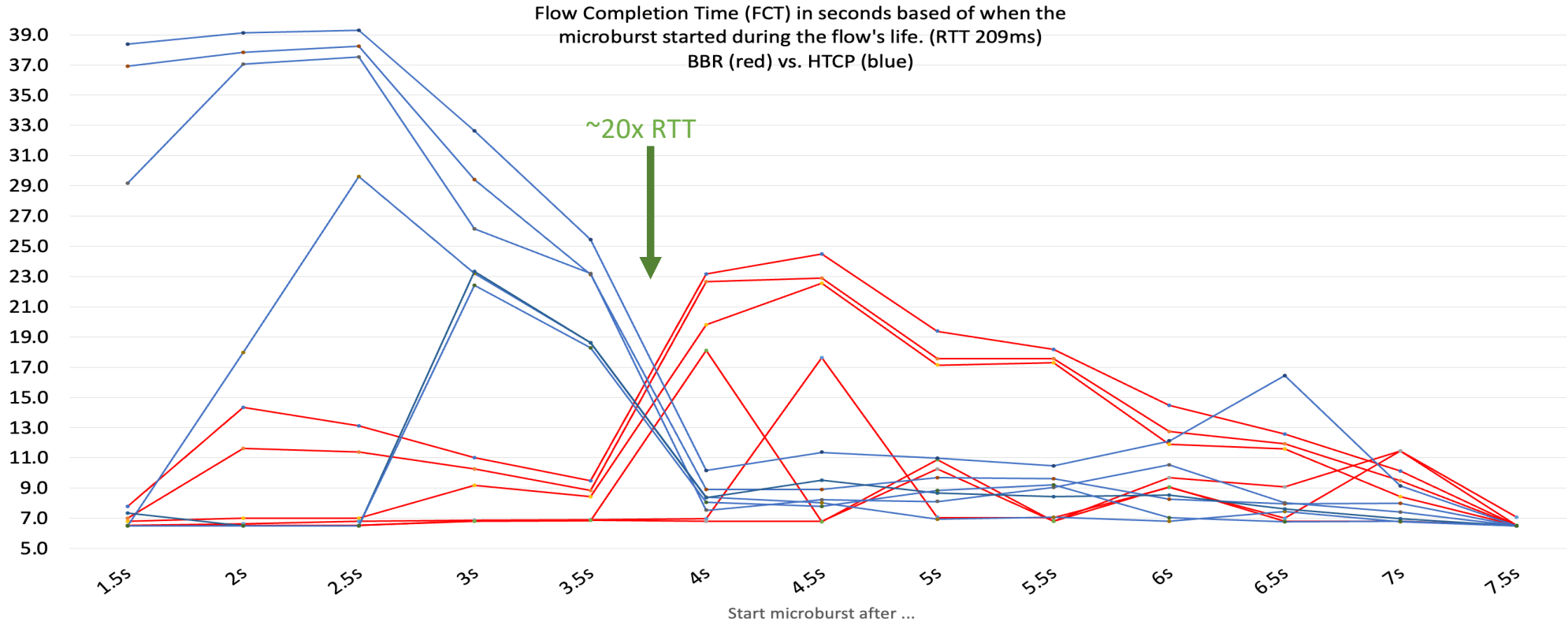


BBR – Microbursts starting 1.5 to 7.5 seconds after iperf3 with 209ms RTT [2]

- BBR struggles the most with microbursts happening after 3.5s of the beginning of the flow.
- The duration of the microburst, as expected, has a direct impact: the longer the burst, longer the FCT.
- After 5.5 seconds, the impact of microbursts are minimized by the TCP Congestion Control window being almost fully established.
- Highest FCT was 25 seconds for microbursts of 2000ms happening after 4.5 seconds of the beginning of the flow.
- We don't know yet why BBR performs as observed from seconds 1.5 to 3.5.



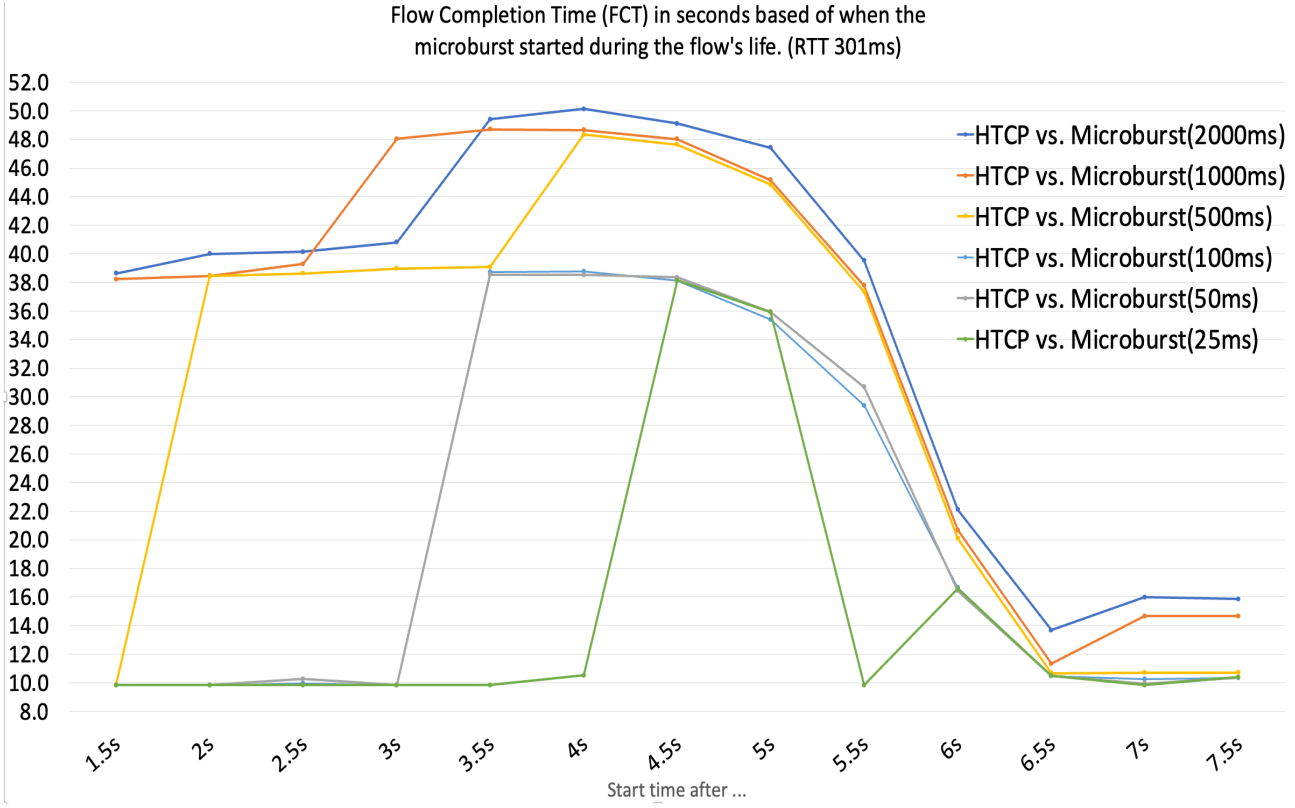
Comparing BBR vs. HTCP over 209ms RTT



AmLight ExP
Americas Lightpaths Express & Protect

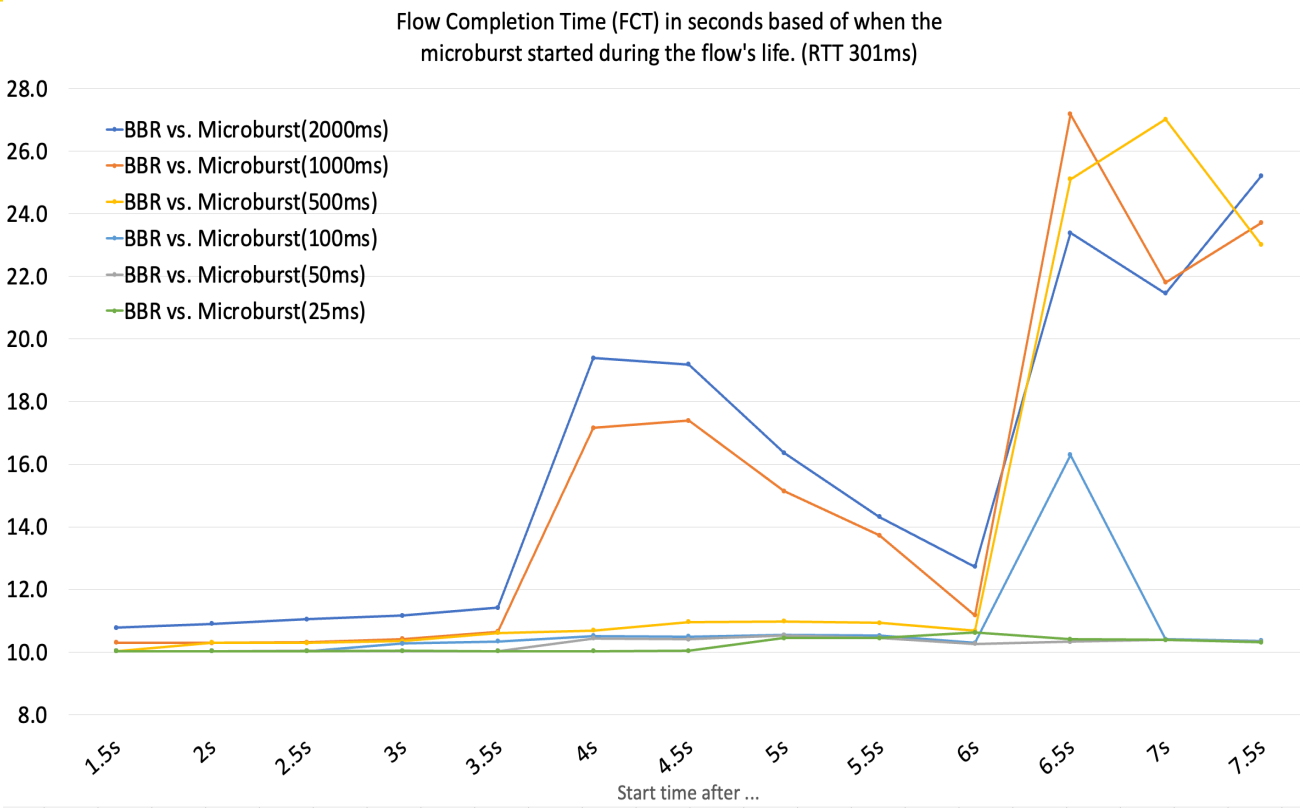
HTCP – Microbursts starting 1.5 to 7.5 seconds after iperf3 with 301ms RTT [2]

- HTCP struggles the most with microbursts happening at the beginning, from 1.5s to 5.5s
- The duration of the microburst, as expected, has a direct impact: the longer the burst, longer the FCT.
- After 5.5 seconds, the impact of microbursts are minimized by the TCP Congestion Control window being almost fully established.
- Highest FCT was 50 seconds for microbursts of 2000ms happening after 4 seconds of the beginning of the flow.

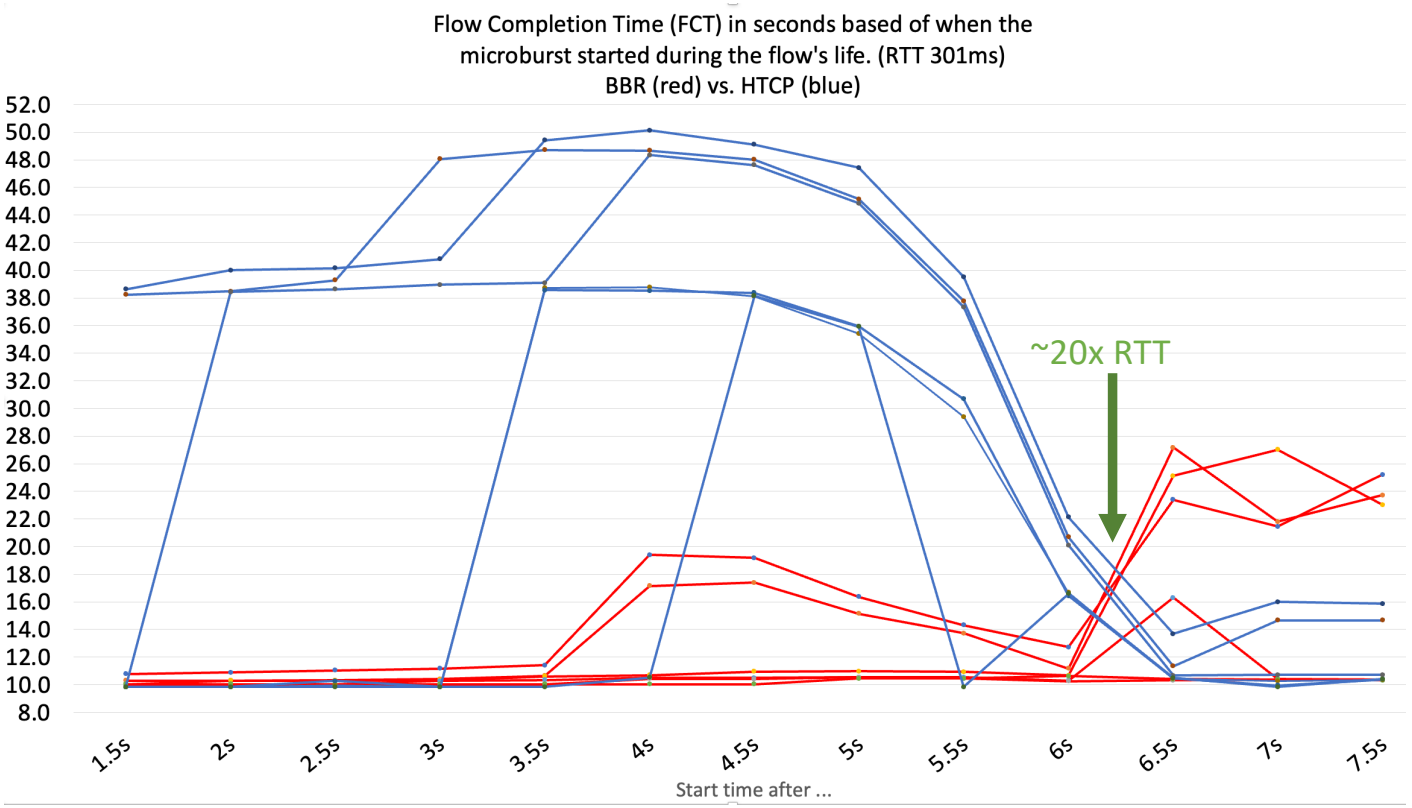


BBR – Microbursts starting 1.5 to 7.5 seconds after iperf3 with 301 ms RTT [2]

- BBR struggles the most with microbursts happening after 6s of the beginning of the flow.
- The duration of the microburst, as expected, has a direct impact: the longer the burst, longer the FCT.
- The impact of microbursts are minimum until 3.5s of the beginning of the flow.
- Highest FCT was 27 seconds for microbursts of 2000ms happening after 6.5 seconds of the beginning of the flow.
- We don't know yet why BBR performs as observed from second 1.5 to 6.



Comparing BBR vs. HTCP over 301 ms RTT



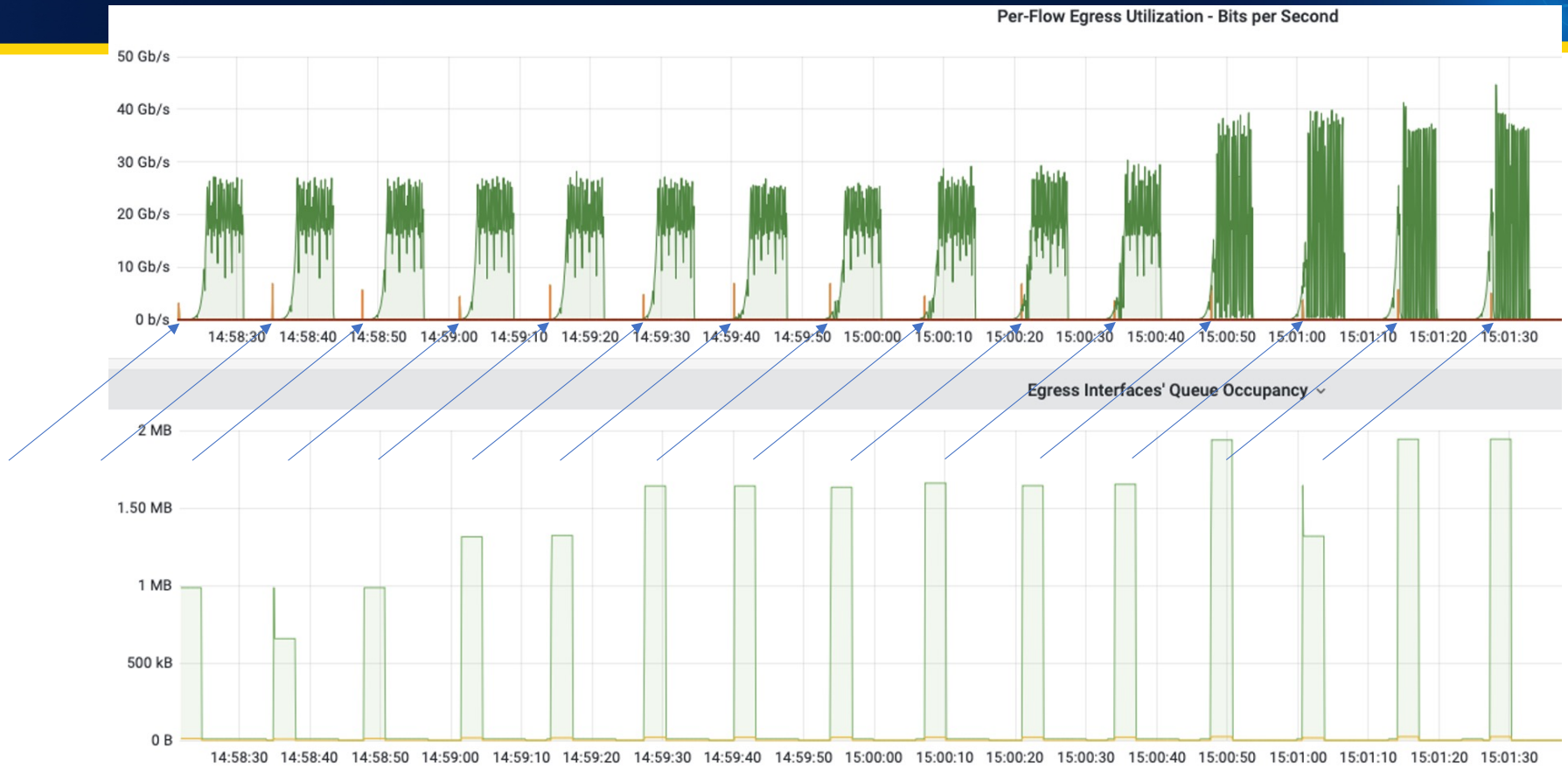
AmLight ExP
Americas Lightpaths Express & Protect

Answering Question 3

- *Question 3: When during the TCP Slow Start is a microburst the most impactful?*

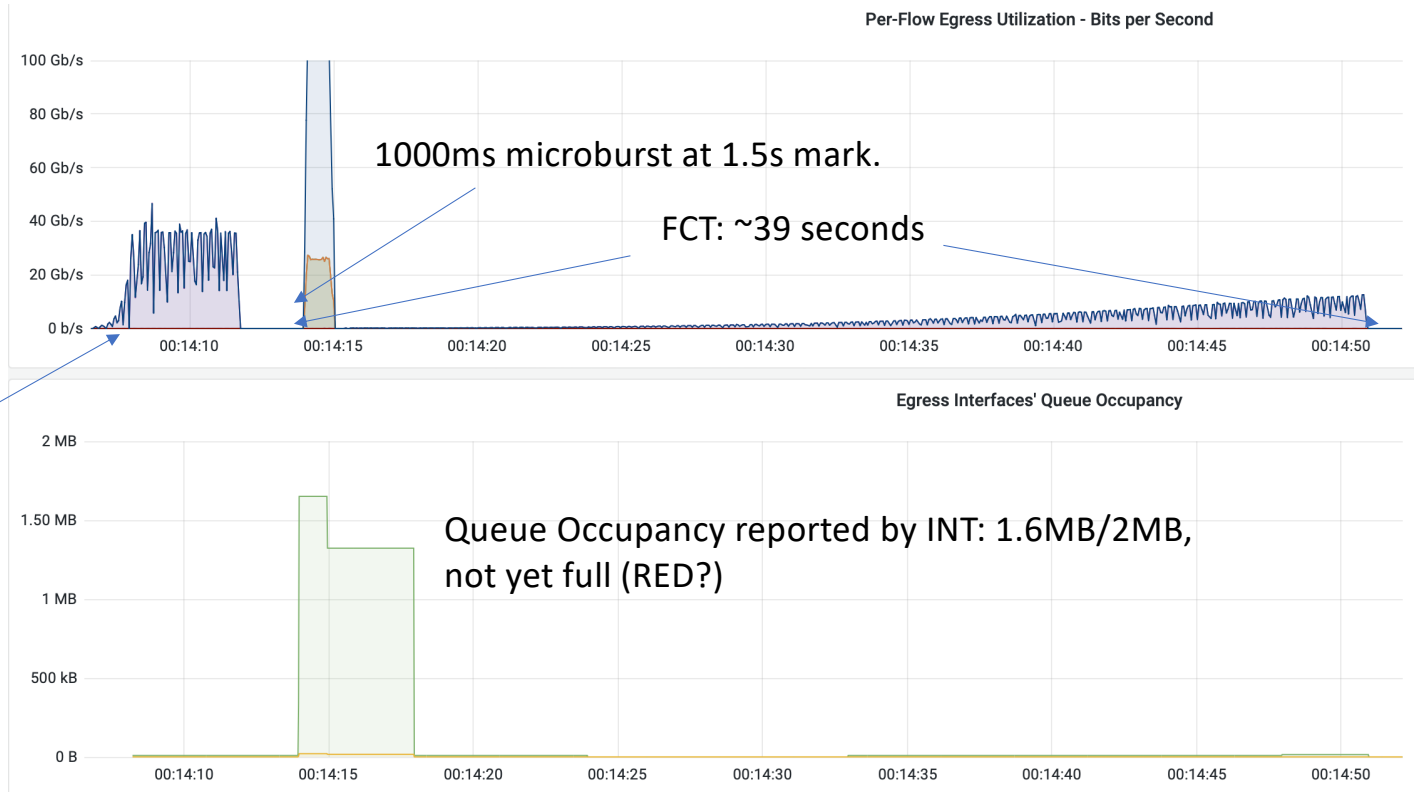
- Microbursts can **dramatically** impact the FCT depending on when it happens during the life of the TCP flow.
- For HTCP, the impact is primarily during the early stages of the TCP Slow Start process while BBR is primarily impacted after the Slow Start process is “over”.
- Figure in the next slide shows a microburst during various phases of the same TCP data transfer and the port #2’s buffer utilization.

A 25ms microburst happening at different moments (HTCP vs. 209ms) (vertical yellow line)



No microburst vs 1000ms microburst (HTCP, 1.5s mark, 209ms RTT)

Only 12 TCP retransmissions were observed!



No microburst, 7s FCT

Question 4: How many retransmits should I expect depending on the size of the microburst?

Microbursts vs TCP Retransmissions

- On average, BBR has 6-10x more TCP retransmits than HTCP.
- For HTCP, although for the 2-second mark has only 15 retransmits, because it was at the early phases of the TCP Slow Start, it has led to a much longer FCT (40s) than the rest.
- For BBR, the number of retransmits is proportional to the FCT, especially for RTT 301ms.
- Results need to be better analyzed. However, results show that small number of reported TCP retransmissions can be worse than big numbers depending of when they happen:
  - Results shows that 15 retransmits at 2s mark is worse than 32,939 retransmits at 4s mark.

		209 ms		301 ms	
	Mark (s)	Median	95% Percentile	Median	95% Percentile
HTCP	2	15	10,882	1	5
	2.5	21	456	2	7,416
	3	658	2,714	2	13
	3.5	8,733	16,097	15	33
	4	32,939	41,649	57	6,043
	4.5	26,095	36,681	293	958
	5	24,017	38,625	977	2,938
	5.5	24,971	37,760	2,890	6,901
	6	27,187	50,373	10,438	23,955
	6.5	21,590	49,132	20,324	32,006
	7	23,708	38,355	17,420	31,151
	7.5	0	18,619	17,201	30,731
BBR	2	21	343,757	0	12
	2.5	168	113,229	2	32
	3	1,068	5,064	4	87
	3.5	7,089	245,130	16	63,251
	4	188,826	387,750	23	359
	4.5	322,152	399,855	212	1,171
	5	233,541	347,600	836	3,406
	5.5	103,833	396,492	4,062	65,322
	6	157,384	192,791	13,583	53,242
	6.5	91,172	143,773	87,339	284,489
	7	58,551	271,062	112,704	301,796
	7.5	0	234,852	42,217	295,720

Lessons Learned

- TCP is hard to troubleshoot!
  - eBPF and new tools help (ss, for instance)
  - We still lack tools to show why something happened, not just that it happened.
- Overtuning could become an issue for FCT
  - Too high Initcwnd and tcp.wmem can lead to small TCP retransmissions that affect the overall FCT
- Microbursts shouldn’t be ignored with FCT is a concern.
- Use cases where FCT is key should address the TCP Slow Start in advance (something like option –O)

Future Work

- More tests!
  - Understanding the real possibility of cascade events
  - Test with BBRv2, CCAs-based on INT, UDP/QUIC
  - Identify the ideal Initial Congestion Window

- Interaction with the iperf3 development community:
  - Seeing iperf3 results in the microscope that INT created led to many questions ☺

- How we plan to mitigate microbursts after seeing the results:
  - Change the AmLight's Traffic Engineering and Prioritization policy to use Queue 0 for bursty flows and make Queue 1 the Best Effort queue.
  - We will use the Behavior, Anomaly, and Performance Manager (BAPM) to redirect the flows to the proper queue based on INT reports.
    - Goal: Lowering the odds of having a cascade event.

Acknowledgement

- This effort to understand the impact of microbursts is supported by
  - NSF CC* Q-Factor (Award #2018754) (next CI Lunch and Learn talk!)
  - NSF IRNC AmLight-ExP (Award #OAC-2029283)
  - Vera Rubin Observatory
  - Rednesp

- The team:
  - Italo Valcy, Sr. Network Engineer
  - David Miranda, Q-Factor Sr. Software Engineer
  - David Ramirez, Software Engineer
  - Renata Frez, Network Enginer
  - Dr. Julio Ibarra, AmLight-ExP's PI
  - Jeronimo Bezerra, AmLight-ExP's Co-PI and Q-Factor's PI.

AmLight ExP
Americas Lightpaths Express & Protect

CI Lunch and Learn / July 7th, 2023

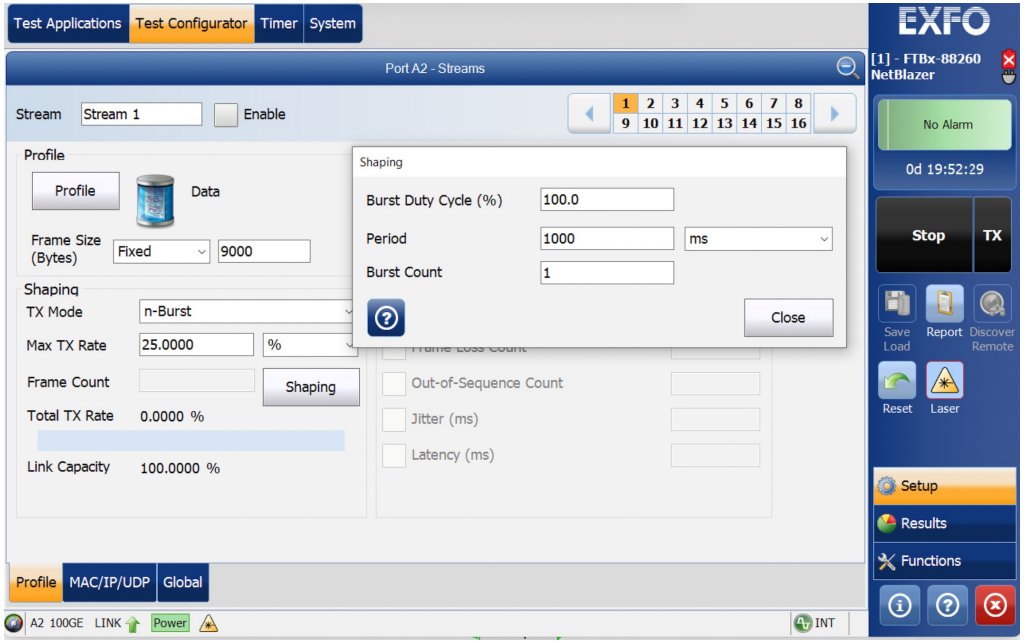
Handling Microbursts @ AmLight – Part 2 of 2

Jeronimo Bezerra, Italo Valcy, David Miranda, David Ramirez, Renata Frez

sdn@amlight.net

The microbursts

- We used EXFO FTB-1 NetBlazer traffic generator to create the microbursts using TrafficGen application.
- Microbursts were set to 25Gbps.
- We used 9,000-byte packets.
- EXFO NetBlazer supports SCPI API.
  - A Python wrapper was created to automate the test routines.



The hosts

- CPU:
  - model name: Intel(R) Xeon(R) Gold 6346 CPU @ 3.10GHz
  - cpu MHz : 3604.871
  - cache size : 36864 KB
- Memory:
  - MemTotal: 131611164 kB
- Network Card:
  - Mellanox MLX5
  - firmware-version: 16.27.1016 (MT_0000000012)
- OS:
  - Debian 11.3
  - Kernel: 5.10.0-14-amd64

The tuning

- Source: https://fasterdata.es.net/host-tuning/
- Use of NUMA 0
  - NIC and PCI can talk directly
- Sysctl –w for TCP memory options (mem, wmem, rmem).
  - Limited wmem to 1GB to avoid oversubscription that was leading to TCP retransmits
- Ethtool –X weight
  - Redirect packets to specific vCPU in the same NUMA as the NIC
- CPU set to performance (BIOS)
- IP route
  - Linux's default TCP Initial Congestion Window (IW) is set to 10x MSS (91,480 Bytes).
  - For our tests, we achieved optimum result setting IW to 1000x MSS (9,140,000 Bytes)
    - Larger the IW, faster TCP achieves highest throughput lowering the FCT.
    - Higher values for IW led to TCP retransmissions or stalling TCP performance (stuck around 14Gbps).