

Server Implementations of HTTP/2 Priority

Kazu Yamamoto
@kazu_yamamoto



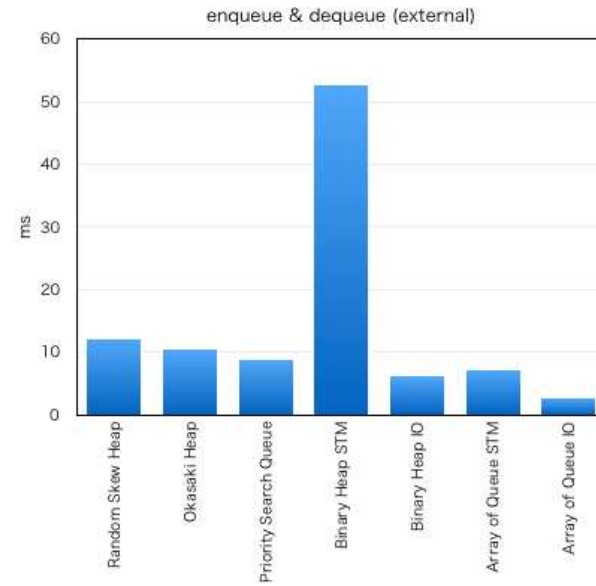
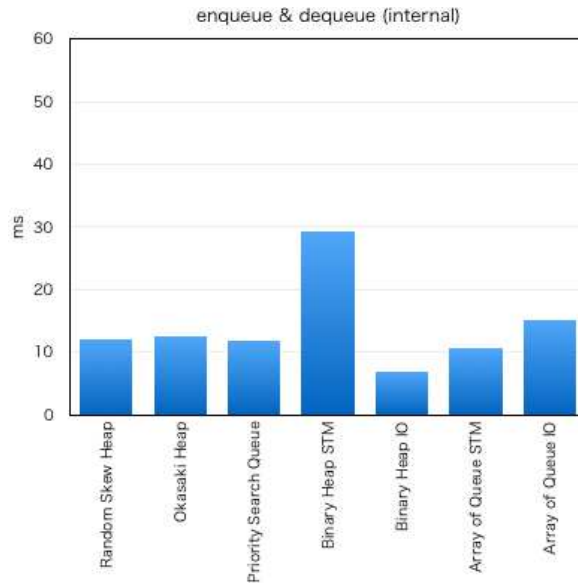
History

- h2o (in C)
 - Kazuho Oku
 - Array of Queue (external)
 - Enqueue $O(1)$, dequeue $O(1)$, delete $O(1)$
 - Deficit and delete information is managed outside
- nhttp2 (in C)
 - Tatsuhiro Tsujikawa
 - Array of Queue IO (external)
 - Enqueue $O(\log N)$, dequeue $O(\log N)$, delete $O(\log N)$
 - Deficit and delete information is managed outside
- Warp (in Haskell)
 - Kazu Yamamoto
 - Random Skew Heap IO
 - Enqueue $O(\log N)$, dequeue $O(\log N)$, delete $O(N \log N)$
 - No deficit and delete information

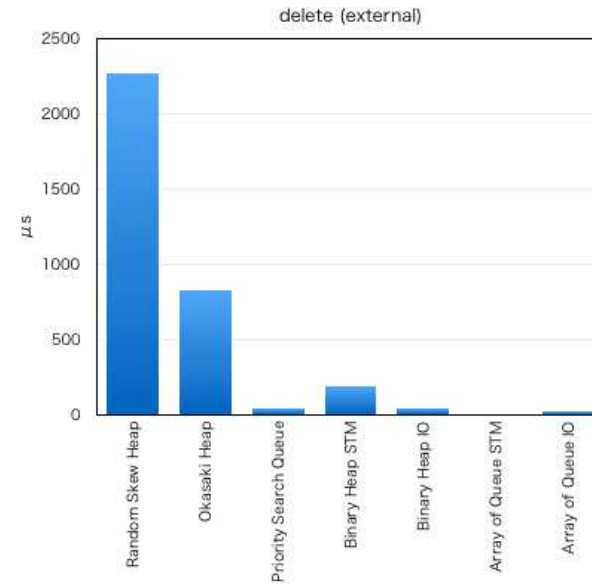
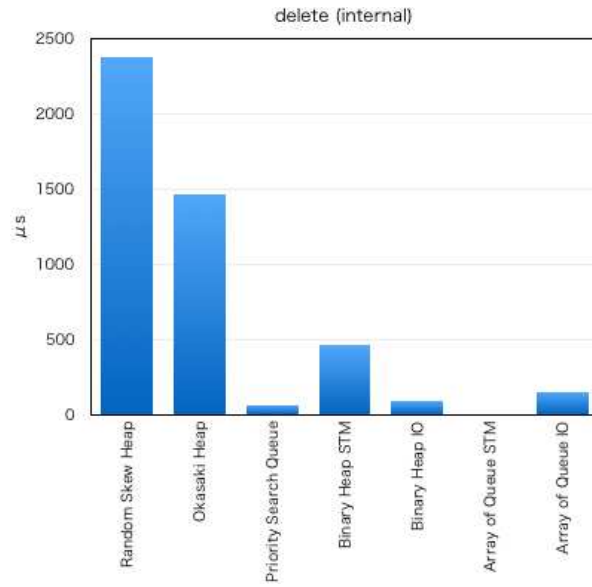
Comparison

- 13 implementations
 - Random Skew Heap
 - Okasaki Heap (internal)
 - Okasaki Heap (external)
 - Priority Search Queue (internal)
 - Priority Search Queue (external)
 - Binary Heap STM(Software Transactional Memory) (internal)
 - Binary Heap STM (external)
 - Binary Heap IO (internal)
 - Binary Heap IO (external)
 - Array of Queue STM (internal)
 - Array of Queue STM (external)
 - Array of Queue IO (internal)
 - Array of Queue IO (external)

Benchmark on enqueue & dequeue



Benchmark of delete



Conclusion

- Binary Heap IO (nghttp2) is the first choice for most programming language
- Array of Queue IO (h2o) is the next choice if you are not satisfied with the performance
- Priority Search Queue (current Warp) is recommended for highly concurrent programming language such as Haskell

Further Reading

- I'm writing a blog article on this topic
- "Understanding HTTP/2 prioritization"
 - <https://speakerdeck.com/summerwind/2-prioritization>
 - Moto Ishizawa