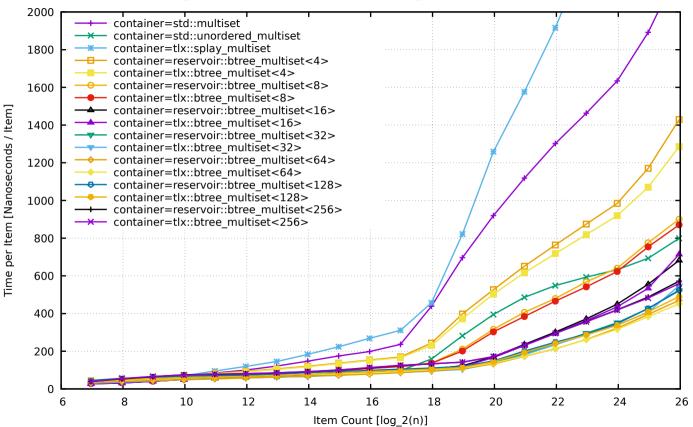
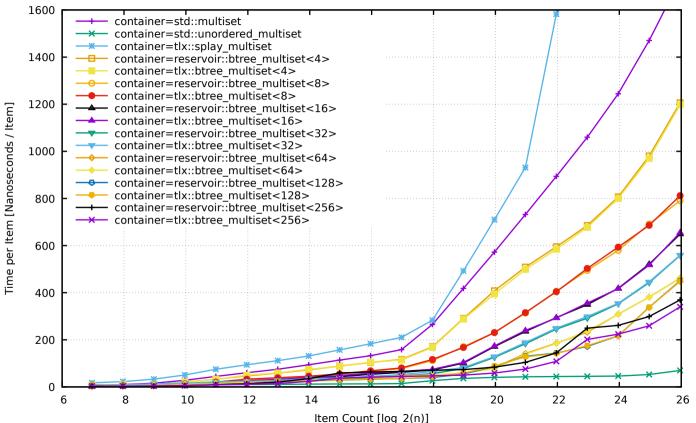
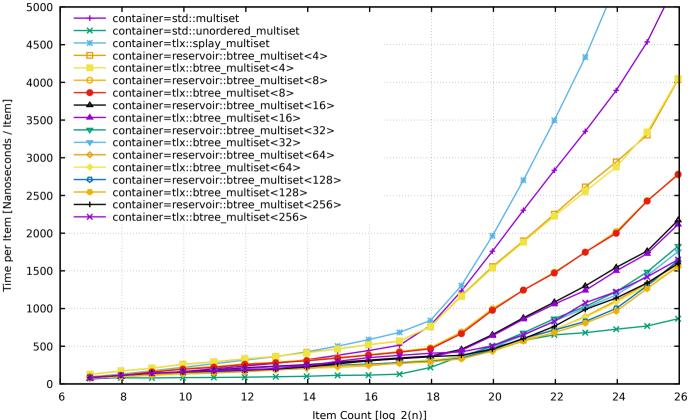
tlx::btree_multiset vs std::multiset vs std::unordered_multiset -- Insert Performance Test



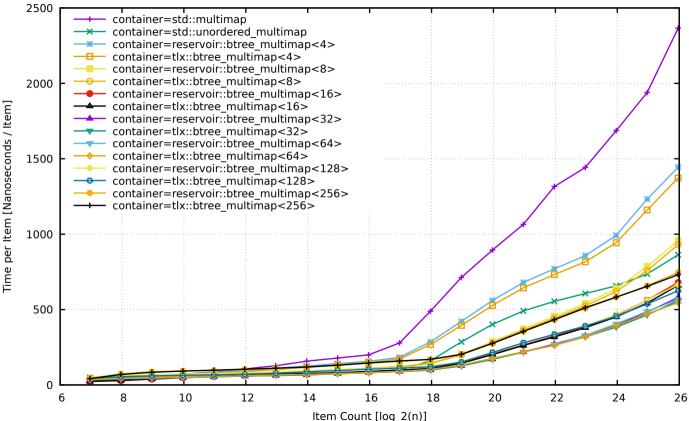
tlx::btree_multiset vs std::multiset vs std::unordered_multiset -- Find Performance Test



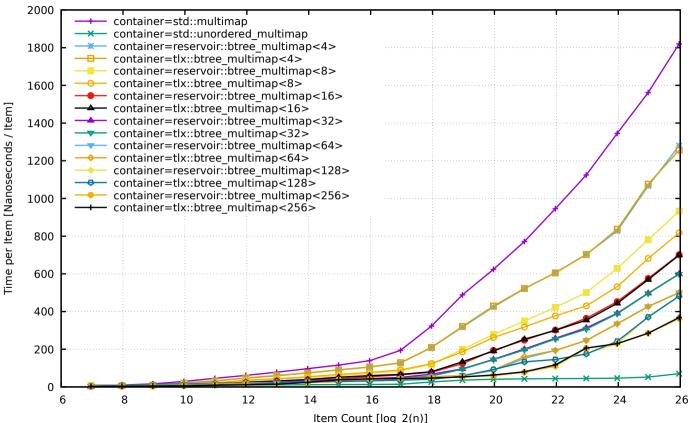
tlx::btree_multiset vs std::unordered_multiset -- Insert/Find/Delete Performance Test



tlx::btree multimap vs std::multimap vs std::unordered multimap -- Insert Performance Test



tlx::btree multimap vs std::multimap vs std::unordered multimap -- Find Performance Test



tlx::btree multimap vs std::multimap vs std::unordered multimap -- Insert/Find/Delete Performance Test container=std::multimap container=std::unordered_multimap container=reservoir::btree_multimap<4>
container=tlx::btree_multimap<4>
container=reservoir::btree_multimap<8> container=tlx::btree_multimap<8>
container=reservoir::btree_multimap<16>
container=tlx::btree_multimap<16> container=reservoir::btree_multimap<32> container=tlx::btree_multimap<32>
container=reservoir::btree_multimap<64> container=tlx::btree_multimap<64> container=reservoir::btree multimap<128> container=tlx::btree_multimap<128>
container=reservoir::btree_multimap<256> container=tlx::btree multimap<256> 10 12 14 16 18 20 22 24

Item Count [log 2(n)]

6000

5000

4000

3000

2000

1000

Time per Item [Nanoseconds / Item]