Unzipping Baskets with TBB

Zhe Zhang
University of Nebraska-Lincoln
Agenda

• Motivation
• Introduction of TBB
• Implementation
• Thoughts on Improvement
• Future work
Motivation

- The experiment was run on 4-core VM.
- Pthread does not linearly scale decompression speed.
- Decompressing threads are competed for lock to update buffers and its status.
Thread Building Block (TBB)

- TBB is a C++ template designed by Intel for parallel programming on multicore processors.
- It adopts work stealing mechanism.
Implementation

• Each task is dedicated to decompressing a single basket.
• Using unlimited cache to hold unzipped buffers.
  – Removed \((fTotalUnzipBytes < fUnzipBufferSize)\)
• Creating TBB tasks when cache miss happens in the main thread.
Performance between Pthread and TBB

Comparison between Pthread and TBB

- The experiment was run on 4-core VM.
- Current implementation of TBB is slower than Pthread.
- Benchmark with more events seem close the gap.
- 4000 Events: 1.71x CPU Time
- 10000 Events: 1.54x CPU Time
Thoughts on Improvement(I)

• In Pthread implementation, the main thread decompress baskets in three scenarios:
  – Basket size is too small (< 256 bytes)
  – Basket size is too large (> 4*fUnzipBufferSize)
  – Cache miss occurs

• In TBB, a task could compact multiple small baskets together.
Thoughts on Improvement(II)

• In Pthread implementation, when cache miss happens, a background thread pick up a zipped basket to work on.
• In TBB, when cache miss happens, UnzipCacheTBB function repeatedly creates tasks for the same baskets.
• Need to find a good spot for the function to avoid unnecessary task overhead.
Thoughts on Improvement(III)

• In Pthread implementation, the background threads pick up the baskets which are close to the requested basket (determined by fLastReadPos).

• In TBB, tasks are randomly appended to the queue, executing sequence is undetermined.
Future Work

• Dynamically assign baskets to a task.
• Minimizing the number of tasks created.
• Assigning different priorities to different tasks.
Questions ?