Index Construction

Introduction to Information Retrieval

Informatics 141 / CS 121
Donald J. Patterson

Content adapted from Hinrich Schütze
http://www.informationretrieval.org

Overview

- Introduction
- Hardware
- BSBI - Block sort-based indexing
- SPIMI - Single Pass in-memory indexing
- Distributed indexing
- Dynamic indexing
- Miscellaneous topics

Terms

- Inverted index
  - (Term, Document) pairs
  - building blocks for creating Vector Space Models
- Index construction (or indexing)
  - The process of building an inverted index from a corpus
- Indexer
  - The system architecture and algorithm that constructs the index

Indices

The index is built from term-document pairs

(TERM,DOCUMENT)

(1998,www.cnn.com)
(Every,www.cnn.com)
(Her,www.cnn.com)
(I,www.cnn.com)
(I'm,www.cnn.com)
(Jensen's,www.cnn.com)
(Julie,www.cnn.com)
(Letter,www.cnn.com)
(Most,www.cnn.com)
(all,www.cnn.com)
(allegedly,www.cnn.com)
(back,www.cnn.com)
(before,www.cnn.com)
(brings,www.cnn.com)
(brothers,www.cnn.com)
(could,www.cnn.com)
(days,www.cnn.com)
(dead,www.cnn.com)
(death,www.cnn.com)
(everything,www.cnn.com)
(for,www.cnn.com)
(from,www.cnn.com)
(full,www.cnn.com)
(happens,www.cnn.com)
(haunts,www.cnn.com)
(have,www.cnn.com)
(hear,www.cnn.com)
(her,www.cnn.com)
(husband,www.cnn.com)
(if,www.cnn.com)
(it,www.cnn.com)
(killing,www.cnn.com)
(letter,www.cnn.com)
(nothing,www.cnn.com)
(now,www.cnn.com)
(of,www.cnn.com)
(pray,www.cnn.com)
(read,,www.cnn.com)
(saved,www.cnn.com)
(sister,www.cnn.com)
(stands,www.cnn.com)
(story,www.cnn.com)
(the,www.cnn.com)
(they,www.cnn.com)
(time,www.cnn.com)
(trial,www.cnn.com)
(wonder,www.cnn.com)
(wrong,www.cnn.com)
(wrote,www.cnn.com)

Indices

The index is built from term-document pairs

(TERM,DOCUMENT)
(1998,www.cnn.com)
(Every,www.cnn.com)
(Her,www.cnn.com)
(I,www.cnn.com)
(I'm,www.cnn.com)
(Jensen's,www.cnn.com)
(Julie,www.cnn.com)
(Letter,www.cnn.com)
(Most,www.cnn.com)
(all,www.cnn.com)
(allegedly,www.cnn.com)
(back,www.cnn.com)
(before,www.cnn.com)
(brings,www.cnn.com)
(brothers,www.cnn.com)
(could,www.cnn.com)
(days,www.cnn.com)
(dead,www.cnn.com)
(death,www.cnn.com)
(everything,www.cnn.com)
(for,www.cnn.com)
(from,www.cnn.com)
(full,www.cnn.com)
(happens,www.cnn.com)
(haunts,www.cnn.com)
(have,www.cnn.com)
(hear,www.cnn.com)
(her,www.cnn.com)
(husband,www.cnn.com)
(if,www.cnn.com)
(it,www.cnn.com)
(killing,www.cnn.com)
(letter,www.cnn.com)
(nothing,www.cnn.com)
(now,www.cnn.com)
(of,www.cnn.com)
(pray,www.cnn.com)
(read,,www.cnn.com)
(saved,www.cnn.com)
(sister,www.cnn.com)
(stands,www.cnn.com)
(story,www.cnn.com)
(the,www.cnn.com)
(they,www.cnn.com)
(time,www.cnn.com)
(trial,www.cnn.com)
(wonder,www.cnn.com)
(wrong,www.cnn.com)
(wrote,www.cnn.com)

- Core indexing step is to sort by terms

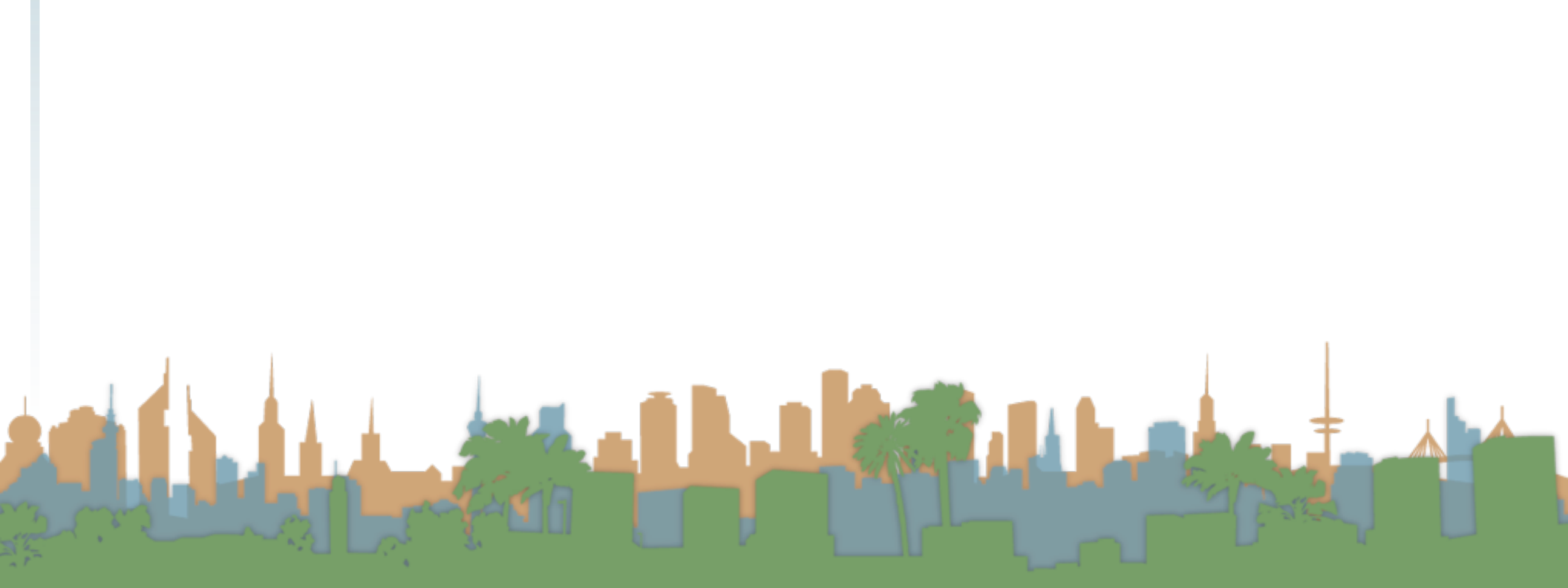
Term-document pairs make lists of postings

(TERM,DOCUMENT, DOCUMENT, DOCUMENT, ...)
(**1998**,www.cnn.com,news.google.com,news.bbc.co.uk)
(**Every**,www.cnn.com, news.bbc.co.uk)
(**Her**,www.cnn.com,news.google.com)
(**I**,www.cnn.com,www.weather.com,)
(**I'm**,www.cnn.com,www.wallstreetjournal.com)
(**Jensen's**,www.cnn.com)
(**Julie**,www.cnn.com)
(**Letter**,www.cnn.com)
(**Most**,www.cnn.com)
(**all**,www.cnn.com)
(**allegedly**,www.cnn.com)

- A posting is a list of all documents in which a term occurs.
- This is "inverted" from how documents naturally occur

Terms

- How do we construct an index?



Interactions

- An indexer needs raw text
  - We need crawlers to get the documents
  - We need APIs to get the documents from data stores
  - We need parsers (HTML, PDF, PowerPoint, etc.) to convert the documents
- Indexing the web means this has to be done web-scale

Construction

- Index construction in main memory is simple and fast.
- But:
  - As we build the index we parse docs one at a time
  - Final postings for a term are incomplete until the end.
  - At 10-12 postings per term, large collections demand a lot of space
  - Intermediate results must be stored on disk

Overview

- Introduction
- Hardware
- BSBI - Block sort-based indexing
- SPIMI - Single Pass in-memory indexing
- Distributed indexing
- Dynamic indexing
- Miscellaneous topics

System Parameters

- Disk seek time = 0.005 sec
- Transfer time per byte = 0.00000002 sec
- Processor clock rate = 0.00000001 sec
- Size of main memory = several GB
- Size of disk space = several TB

System Parameters

- Disk seek time = 0.005 sec
- Transfer time per byte = 0.00000002 sec
- Processor clock rate = 0.00000001 sec
- Size of main memory = several GB
- Size of disk space = several TB

System Parameters

- Disk Seek Time
  - The amount of time to get the disk head to the data
  - About 10 times slower than memory access
  - We must utilize caching
  - No data is transferred during seek
- Data is transferred from disk in blocks
  - There is no additional overhead to read in an entire block
    - 0.2 seconds to get 10 MB if it is one block
    - 0.7 seconds to get 10 MB if it is stored in 100 blocks

System Parameters

- Data is transferred from disk in blocks
  - Operating Systems read data in blocks, so
  - Reading one byte and reading one block take the same amount of time
- Wait, is that a contradiction?

System Parameters

- Data transfers are done on the system bus, not by the processor
  - The processor is not used during disk I/O
  - Assuming an efficient decompression algorithm
  - The total time of reading and then decompressing compressed data is usually less than reading uncompressed data.

Overview

- Introduction
- Hardware
- BSBI - Block sort-based indexing
- SPIMI - Single Pass in-memory indexing
- Distributed indexing
- Dynamic indexing
- Miscellaneous topics

Reuters collection example (approximate #'s)

- 800,000 documents from the Reuters news feed
- 200 terms per document
- 400,000 unique terms
- number of postings 100,000,000

REUTERS

You are here: Home > News > Science > Article

Go to a Section: U.S. International Business Markets Politics Entertainment Technology Sports Oddly Enoug

Extreme conditions create rare Antarctic clouds

Tue Aug 1, 2006 3:20am ET

Email This Article | Print This Article | Reprints

[-] Text [+]

SYDNEY (Reuters) - Rare, mother-of-pearl colored clouds caused by extreme weather conditions above Antarctica are a possible indication of global warming, Australian scientists said on Tuesday.

Known as nacreous clouds, the spectacular formations showing delicate wisps of colors were photographed in the sky over an Australian meteorological base at Mawson Station on July 25.

Reuters collection example (approximate #'s)

- Sorting 100,000,000 records on disk is too slow because of disk seek time.
  - Parse and build posting entries one at a time
  - Sort posting entries by term
    - Then by document in each term
- Doing this with random disk seeks is too slow
- e.g. If every comparison takes 2 disk seeks and N items need to be sorted with N log2(N) comparisons?
  - 306ish days?

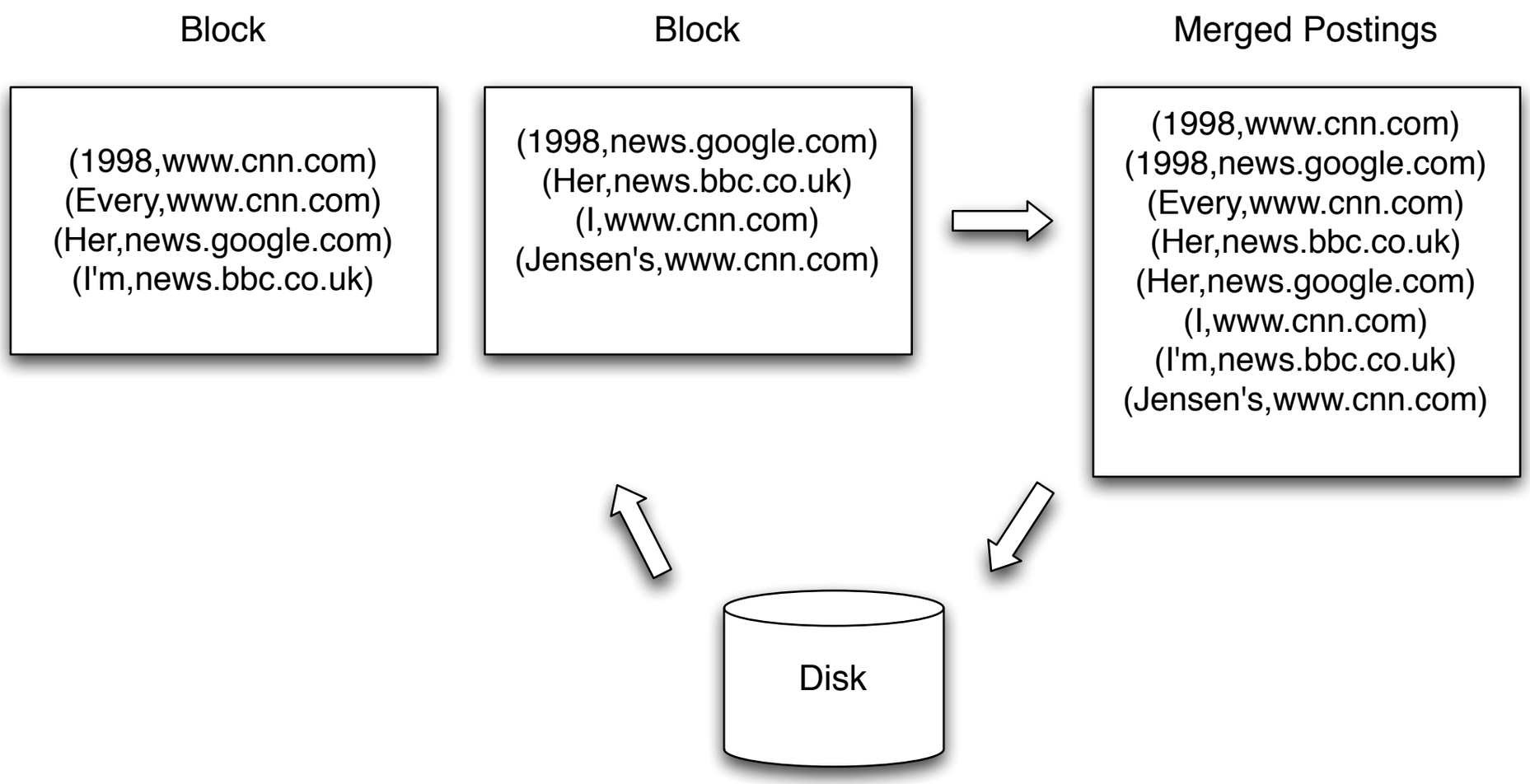
Different way to sort index

- 12-byte records (term, doc, meta-data)
- Need to sort T= 100,000,000 such 12-byte records by term
- Define a block to have 1,600,000 such records
  - can easily fit a couple blocks in memory
  - we will be working with 64 such blocks
- Accumulate postings for each block (real blocks are bigger)
- Sort each block
- Write to disk
- Then merge

Different way to sort index

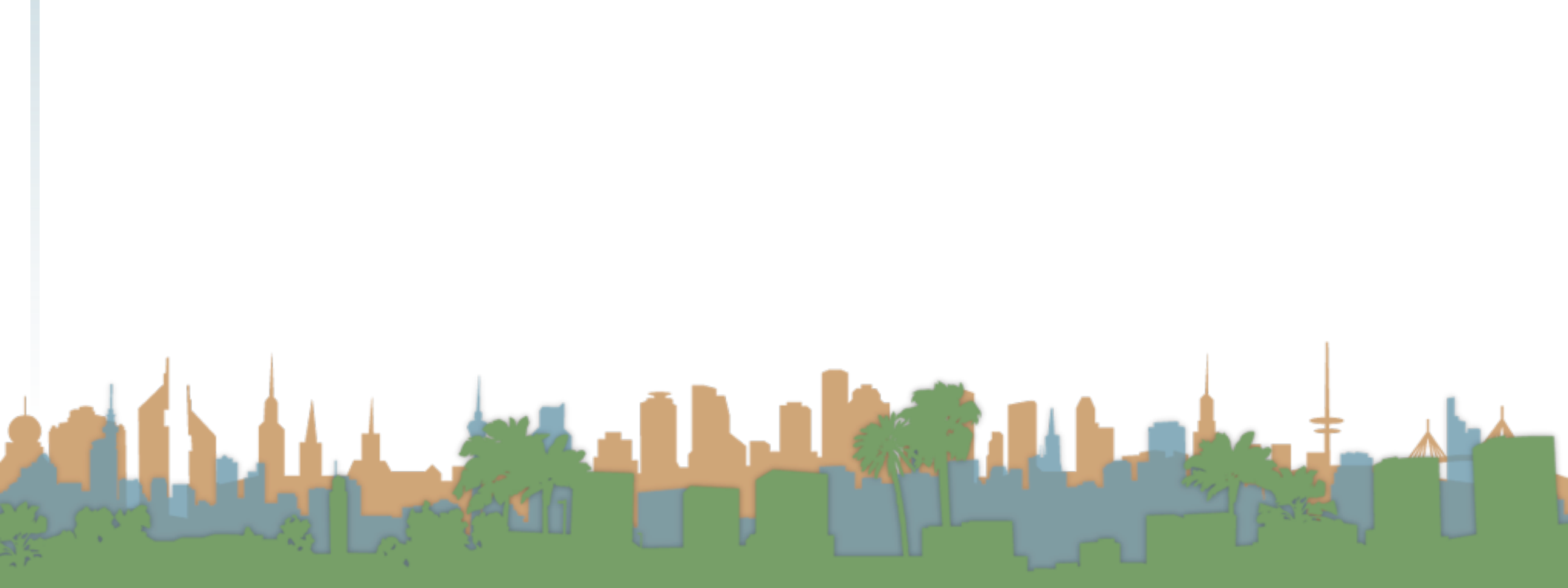
- 12-byte records (term, doc, meta-data)
- Need to sort T= 100,000,000 such 12-byte records by term
- Define a block to have 1,600,000 such records
  - can easily fit a couple blocks in memory
  - we will be working with 64 such blocks
- Accumulate postings for each block (real blocks are bigger)
- Sort each block
- Write to disk
- Then merge

Different way to sort index



Block merge indexing

- Sequentially process documents and write each sorted block to disk
- Then merge all blocks into one large postings file
- Need 2 copies of the data on disk



Block merge indexing

```
BLOCKMERGEINDEXCONSTRUCTION()
 1  for n ← 1 to ∞
 2  do if (all documents have been processed)
 3        then BREAK
 4        block = READNEXTBLOCKOFDOCUMENTS()
 5        INVERT(block)
 6        WRITETODISK(block, f_n)
 7  MERGEBLOCKS(f_1, ..., f_n; f_merged)
```