

Indexing TREC

A WUME Lab Search Engine Effort

David Deschenes
WWW Search Engines
October 15, 2002

Output of the Parser

- Dictionaries
 - Map useful values (document names, terms, etc.) to dictionary keys
- Relationship Maps
 - Map useful relationships (document \square term, etc.)
 - Made up of dictionary keys (written in ASCII)
 - May run over several files
 - Used to generate the indices

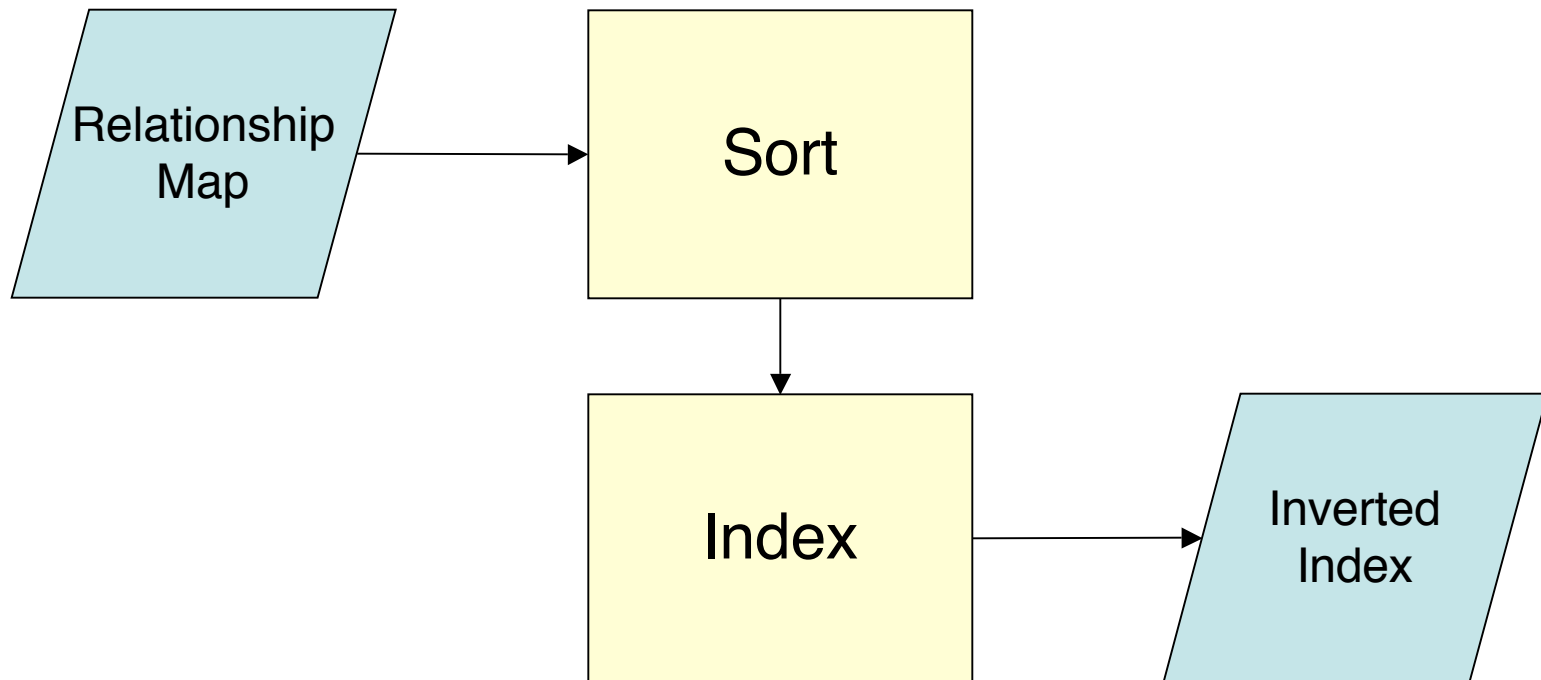
Important Relationships

- Document □ Term
- Term □ Document
- Document □ Page Description
- Document □ Forward Link
- Document □ Reverse Link
- Link □ Link Text
- Many others

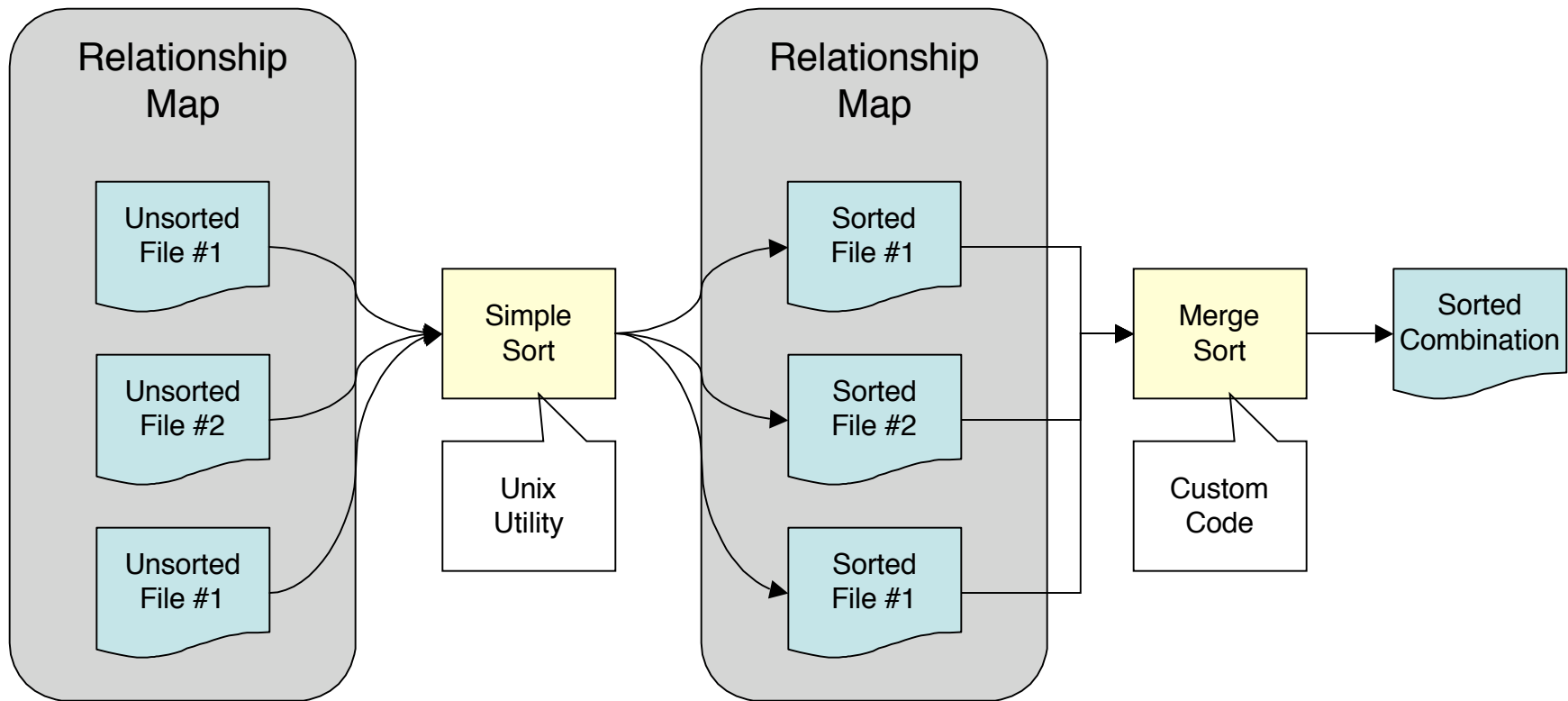
Indexing Decisions

- Use inverted indices
- Use binary index files
 - Conserve disk space
- Depend on sorted data
 - Provide for efficient sorting
- Do not load indices into memory
 - Reduce memory usage
 - Implications for file structure

Indexing Process



Sorting in Depth



Inverted Index Generation

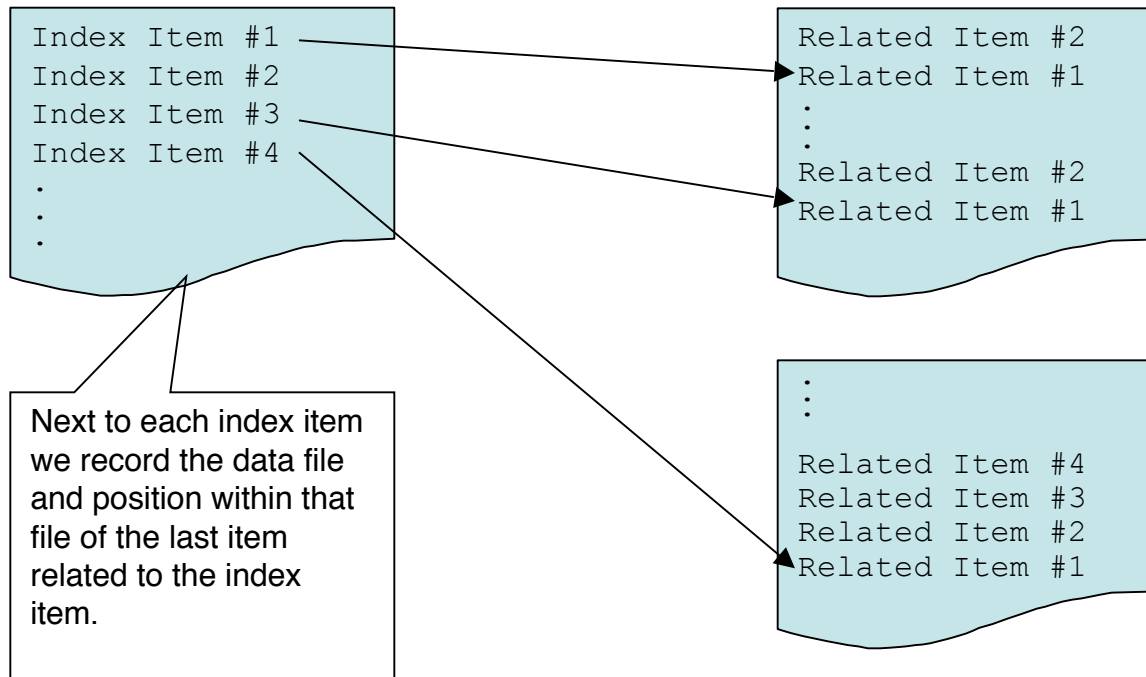
- Details
 - One index file
 - Many data files
 - Limited to 1GB in size
- Algorithm

For each index item record each related item in the appropriate data file and then record the data file # and data file position alongside the index item in the index file.

Inverted Index Visual

Index File

Data Files



Inverted Index Search Algorithm

- Binary search through the index file
- Retrieve file # and file position of last related item from the index file
- Retrieve all related items until the last item related to the previous index item is encountered

Problems

- Heterogeneous nature of items requires multiple index generation implementations
- Data files proliferate
- Do you see any others?

Comments or Questions