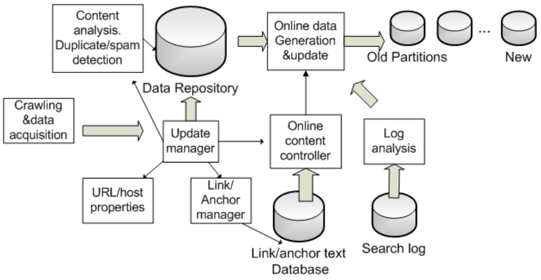
Offline Data Processing: Tasks and Infrastructure Support

T. Yang, UCSB 290N

Table of Content

- Offline incremental data processing: case study
- Example of content analysis
- Data store support

Offline Architecture at Ask



Content analysis. Duplicate/spam detection

Data Repository

Online data Generation &update

Old Partitions

...

New

Crawling &data acquisition

Update manager

Online content controller

Log analysis

URL/host properties

Link/ Anchor manager

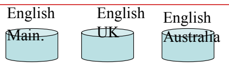
Link/anchor text Database

Search log

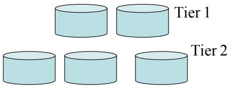
Content Management

- **Organize the vast amount of pages crawled to facilitate online search.**
  - Data preprocessing
  - Inverted index
  - Compression
  - Classify and partition data
- **Collect additional content and ranking signals.**
  - Link, anchor text, log data
- **Extract and structure content**
- **Duplicate detection**

Classifying and Partitioning data



- **Classify**
  - Content quality. Language/country etc
- **Partition**
  - Based on languages and countries. Geographical distribution based on data center locations
  - Partition based on quality
    - First tier --- high chance that users will access
      - Quality indicator
      - Click feedback
    - Second tier – lower chance

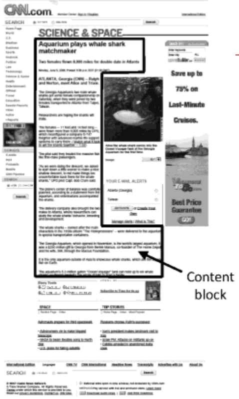


Examples of Context Extraction/Analysis

- **Identify key phrases that captures the meaning of this document.**
  - For example, title, section title, highlighted words.
    - HTML vs PDF
- **Identify parts of a document representing the meaning of this document.**
  - Many documents on the web contain a side-menu which do not contain primary material.
- **Capture page content through Javascript analysis.**
  - Page rendering and Javascript evaluation within a page

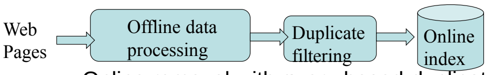
Example of Content Analysis

- **Detect content block related to the main content of a page**
  - Non-content text/link material is ignored or reduced in importance during indexing process



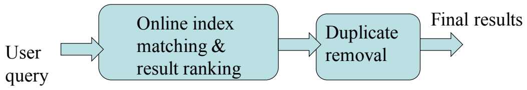
Redundant Content Removal in Search Engines

- **Over 1/3 of Web pages crawled are near duplicates**
- **When to remove near duplicates?**
  - Offline removal



Web Pages → Offline data processing → Duplicate filtering → Online index

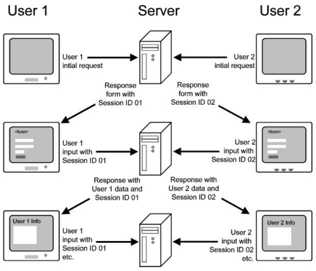
  - Online removal with query-based duplicate removal



User query → Online index matching & result ranking → Duplicate removal → Final results

Why there are so many duplicates?

- **Same content, different URLs, often with different session IDs.**
- **Crawling time difference**

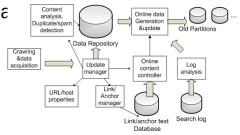

User 1
Server
User 2
User 1 intial request
User 2 intial request
Response form with Session ID 01
Response form with Session ID 02
<form>
<form>
User 1 input with Session ID 01
User 2 input with Session ID 02
Response with User 1 data and Session ID 01
Response with User 2 data and Session ID 02
User 1 Info
User 2 Info
User 1 input with Session ID 01 etc.
User 2 input with Session ID 02 etc.

Tradeoff of online vs. offline removal

	Online-dominating approach	Offline-dominating approach
Impact to offline	High precision Low recall Remove fewer duplicates	High precision High recall Remove most of duplicates Higher offline burden
Impact to online	More burden to online deduplication	Less burden to online deduplication
Impact to overall cost	Higher serving cost	Lower serving cost

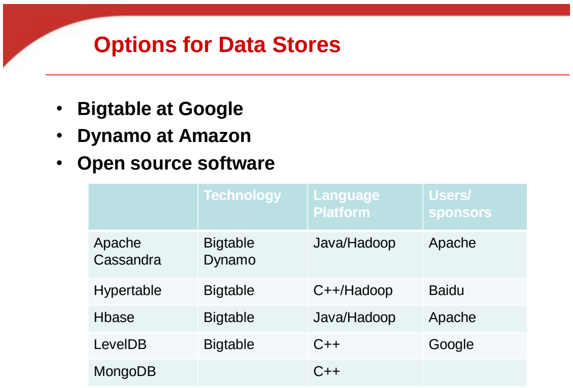
Software Infrastructure Support at Ask

- **Programming support (multi-threading/exception Handling, Hadoop MapReduce)**
- **Data stores for managing billions of objects**
  - Distributed hash tables, queues etc
- **Hadoop Communication and data exchange among machines/services**
- **Execution environment**
  - Controllable (stop, pause, restart).
  - Service registration and invoca
  - service monitoring
  - Logging and test framework.


Content analysis. Duplicate/spam detection
Data Repository
Online data Generation &update
Old Partitions
...
Crawling &data acquisition
Update manager
Online content controller
Log analysis
URL/host properties
Link/ Anchor manager
Link/anchor text Database
Search log

Requirements for Data Repository Support in Offline Systems

- **Update**
  - handling large volumes of modified documents
  - adding new content
- **Random access**
  - request the content of a document based on its URL
- **Compression and large files**
  - reducing storage requirements and efficient access
- **Scan**
  - Scan documents for text mining.



Options for Data Stores

- **Bigtable at Google**
- **Dynamo at Amazon**
- **Open source software**

	Technology	Language Platform	Users/ sponsors
Apache Cassandra	Bigtable Dynamo	Java/Hadoop	Apache
Hypertable	Bigtable	C++/Hadoop	Baidu
Hbase	Bigtable	Java/Hadoop	Apache
LevelDB	Bigtable	C++	Google
MongoDB		C++	