Database design and implementation

Lectures 22: Parallel Databases and MapReduce
What is a parallel database?
Parallel v.s. distributed databases

- **Parallel database system:**
  - Improve performance through parallel implementation

- **Distributed database system:**
  - Data is stored across several sites, each site managed by a DBMS capable of running independently
Parallel DBMSs

- **Goal**
  - Improve performance by executing multiple operations in parallel

- **Key benefit**
  - Cheaper to scale than relying on a single increasingly more powerful processor

- **Key challenge**
  - Ensure overhead and contention do not kill performance
Performance metrics for parallel DBMSs

- **Speedup**
  - More processors → higher speed
  - Individual queries should run faster
  - Should do more transactions per second (TPS)
  - Fixed problem size overall, vary # of processors ("strong scaling")

- **Scaleup**
  - More processors → can process more data
  - Fixed problem size per processor, vary # of processors ("weak scaling")
  - **Batch scaleup**
    - Same query on larger input data should take the same time
  - **Transaction scaleup**
    - N-times as many TPS on N-times larger database
    - But each transaction typically remains small
Linear v.s. non-linear speedup

Speedup

# processors (=P)
Linear v.s. non-linear scaleup

Batch Scaleup

# processors (=P) AND data size

×1 ×5 ×10 ×15
Challenges to linear speedup and scaleup

- **Startup cost**
  - Cost of starting an operation on many processors

- **Interference**
  - Contention for resources between processors

- **Skew**
  - Slowest processor becomes the bottleneck
Architectures for parallel databases

- Shared memory
- Shared disk
- Shared nothing
Shared memory

Interconnection Network

Global Shared Memory

P

P

P

D

D

D
Shared disk

Interconnection Network

M

P

D

M

P

D

M

P

D
Shared nothing
Shared nothing

- Most scalable architecture
  - Minimizes interference by minimizing resource sharing
  - Can use commodity hardware
- Also most difficult to program and manage

We will focus on shared nothing

Important question: what exactly can we actually parallelize in a parallel database?
Taxonomy for parallel query evaluation

- **Inter-query parallelism**
  - Each query runs on one processor

- **Inter-operator parallelism**
  - A query runs on multiple processors
  - An operator runs on one processor

- **Intra-operator parallelism**
  - An operator runs on multiple processors
Different types of parallelism

- **Partitioned parallelism**
  - Partition data over all nodes, get the nodes working to compute a given operation (scan, sort, join)

- **Pipelined parallelism**
  - A chain of operators $O_1, O_2, ..., O_k$ run in parallel, with $O_1$ working on tuple $t_n$, $O_2$ on $t_{n-1}$, ..., $O_k$ on $t_{n-k+1}$
  - Can run these operators on different nodes
  - Some operators break pipelining, e.g. sort, hash

- **Independent operators**
  - Consider bushy query plans
  - A join B, C join D are independent
Data partitioning schemes

Partitioning a table:

Range

Hash

Round Robin
Data partitioning

What are the pros and cons?

- Round robin
  - Good load balance but always needs to read all the data

- Hash based partitioning
  - Good load balance but works only for equality predicates and full scans

- Range based partitioning
  - Works well for range predicates but can suffer from data skew
Parallel evaluation of operators

- Selection?
- Aggregates?
- Joins?
Parallel join: R join S on attribute x

Hash on x

Join each hash bucket
MapReduce
A Declaration By the Representatives of the United States of America, in General Congress Assembled. When in the course of human events it becomes necessary for a people to advance from that subordination in which they have hitherto remained, and to assume among powers of the earth the equal and independent station to which the laws of nature and of nature's god entitle them, a decent respect to the opinions of mankind requires that they should declare the causes which impel them to the change.

We hold these truths to be self-evident; that all men are created equal and independent; that from that equal creation they derive rights inherent and inalienable, among which are the preservation of life, and liberty, and the pursuit of happiness; that to secure these ends, governments are instituted among men, deriving their just power from the consent of the governed; that whenever any form of government shall become destructive of these ends, it is the right of the people to alter or to abolish it, and to institute new government, laying it's foundation on such principles and organizing it's power in such form, as to them shall seem most likely to effect their safety and happiness. Prudence indeed will dictate that governments long established should not be changed for light and transient causes: and accordingly all experience hath shewn that mankind are more disposed to suffer while evils are sufferable, than to right themselves by abolishing the forms to which they are accustomed. But when a long train of abuses and usurpations, begun at a distinguished period, and pursuing invariably the same object, evinces a design to reduce them to arbitrary power, it is their right, it is their duty, to throw off such government and to provide new guards for future security. Such has been the patient sufferings of the colonies; and such is now the necessity which constrains them to expunge their former systems of government. the history of his present majesty is a history of unremitting injuries and usurpations, among which no one fact stands single or solitary to contradict the uniform tenor of the rest, all of which have in direct object the establishment of an absolute tyranny over these states. To prove this, let facts be submitted to a candid world, for the truth of which we pledge a faith yet unsullied by falsehood.

How many “big”, “medium”, and “small” words are used?
Example: word length histogram

Big = Yellow = 10+ letters

Medium = Red = 5..9 letters

Small = Blue = 2..4 letters

Tiny = Pink = 1 letter

A Declaration By the Representatives of the United States of America, in General Congress Assembled.

When in the course of human events it becomes necessary for the people to alter their political system, and to assume among the powers of the earth the equal and independent station to which they are entitled, a decent respect to the opinions of mankind requires that they should declare the causes which impel them to the change.

We hold these truths to be self-evident: that all men are created equal and independent; that they are endowed by their creator with certain inalienable rights, among which are life, liberty, and the pursuit of happiness; that, to secure these rights, governments are instituted among men, deriving their just powers from the consent of the governed; that whenever any form of government becomes destructive of these ends, it is the right of the people to alter or to abolish it, and to institute new government, laying its foundation on such principles and organizing its powers in such form as to them shall seem most likely to effect their safety and happiness. Prudence indeed will dictate that governments long established should not be changed for light and transient causes; and accordingly, it has been shewn that mankind are more disposed to suffer while evils are sufferable than to right themselves by abolishing the forms to which they are accustomed. But when a long train of abuses and usurpations, pursuing invariably the same object, evinces a design to reduce them under absolute power, it is the right of the people to throw off such government, and to provide new guards for future security. Such has been the patient sufferings of the colonies; and such is now the necessity which constrains them to expunge their former systems of government. The history of the present majesty is the history of unremitting injuries and usurpations, among which are so numerous and so conclusive proofs of an intentional design to undo the object, to reduce them under absolute dominion, and to subject them to allegiance only; such is the nature of the case that there is no possibility of redress through the ordinary channels of a legitimate government.

For the truth of which we pledge a faith unsullied by falsehood.
A Declaration By the Representatives of the United States of America, in General Congress Assembled.

When in the course of human events it becomes necessary for a people to advance from that subordination in which they have hitherto remained, and to assume among powers of the earth the equal and independent station to which the laws of nature and of nature's god entitle them, a decent respect to the opinions of mankind requires that they should declare the causes which impel them to the change.

We hold these truths to be self-evident; that all men are created equal and independent; that from that equal creation they derive rights inherent and inalienable, among which are the preservation of life, and liberty, and the pursuit of happiness; that to secure these ends, governments are instituted among men, deriving their just power from the consent of the governed; that whenever any form of government shall become destructive of these ends, it is the right of the people to alter or to abolish it, and to institute new government, laying it's foundation on such principles and organizing it's power in such form, as to them shall seem most likely to effect their safety and happiness. Prudence indeed will dictate that governments long established should not be changed for light and transient causes; and accordingly all experience hath shewn that mankind are more disposed to suffer while evils are sufferable, than to right themselves by abolishing the forms to which they are accustomed. But when a long train of abuses and usurpations, begun at a distinguished period, and pursuing invariably the same object, evinces a design to reduce them to arbitrary power, it is their right, it is their duty, to throw off such government and to provide new guards for future security. Such has been the patient sufferings of the colonies; and such is now the necessity which constrains them to expunge their former systems of government. the history of his present majesty is a history of unremitting injuries and usurpations, among which no one fact stands single or solitary to contradict the uniform tenor of the rest, all of which have in direct object the establishment of an absolute tyranny over these states. To prove this, let facts be submitted to a candid world, for the truth of which we pledge a faith yet unsullied by falsehood.
Example: word length histogram

Map Task 1
(204 words)

A Declaration By the Representatives of the United States of America, in General Congress Assembled,

When in the course of human events it becomes necessary for a people to advance from that subordination under which they have hitherto lived, to assume among the powers of the earth the equal and independent station to which the laws of nature and of nature's God entitle them, a decent respect to the opinions of mankind requires that they should declare the causes which impel them to the change.

We hold these truths to be self-evident; that all men are created equal and independent; that they are endowed by their Creator with certain inalienable Rights; that among these are Life, Liberty and the pursuit of Happiness.

That to secure these rights, governments are instituted among men, deriving their just powers from the consent of the governed: that whenever any form of government becomes destructive of these ends, it is the right of the people to alter or to abolish it, and to institute new government, laying its foundation on such principles and organizing its powers in such form, as to them shall seem most likely to effect their safety and happiness. Prudence indeed will dictate that governments long established should not be changed for light and transient causes; and accordingly all experience hath shewn that mankind are more disposed to suffer, while evils are sufferable, than to right themselves by abolishing the forms to which they are accustomed. But when a long train of abuses and usurpations, pursuing invariably the same object, evinces a design to reduce them to slavery; it is their right, it is their duty, to throw off such government and to provide new guards for future security. Such has been the patience of the Colonies; and such is now the necessity which constrains them to expunge their former systems of government, the history of the present majesty is a history of repeated injuries and usurpations, among which are no less apparent than the establishment of an absolute tyranny over these states. To prove that, facts are submitted as a candid world, for the truth of which we pledge a faith yet unsullied by falsehood.
A Declaration By the Representatives of the United States of America, in General Congress Assembled.

When in the course of human events it becomes necessary for a people to advance from that subordination in which they have hitherto remained, and to assume among powers of the earth the equal and independent station to which the laws of nature and of nature's god entitle them, a decent respect to the opinions of mankind requires that they should declare the causes which impel them to the change.

We hold these truths to be self-evident; that all men are created equal and independent; that from that equal creation they derive rights inherent and inalienable, among which are the preservation of life, and liberty, and the pursuit of happiness; that to secure these ends, governments are instituted among men, deriving their just power from the consent of the governed; that whenever any form of government shall become destructive of these ends, it is the right of the people to alter or to abolish it, and to institute new government, laying its foundation on such principles and organizing its power in such form, as to them shall seem most likely to effect their safety and happiness. Prudence indeed will dictate that governments long established should not be changed for light and transient causes: and accordingly all experience hath shewn that mankind are more disposed to suffer while evils are sufferable, than to right themselves by abolishing the forms to which they are accustomed. But when a long train of abuses and usurpations, begun at a distinguished period, and pursuing invariably the same object, evinces a design to reduce them to arbitrary power, it is their right, it is their duty, to throw off such government and to provide new guards for future security. Such has been the patient sufferings of the colonies; and such is now the necessity which constrains them to expunge their former systems of government. the history of his present majesty is a history of unremitting injuries and usurpations, among which no one fact stands single or solitary to contradict the uniform tenor of the rest, all of which have in direct object the establishment of an absolute tyranny over these states. To prove this, let facts be submitted to a candid world, for the truth of which we pledge a faith yet unsullied by falsehood.
MapReduce programming model

- Input & Output: each a set of key/value pairs
- Programmer specifies two functions:

  map \((\text{in\_key}, \text{in\_value}) \rightarrow \text{list(\text{out\_key}, \text{intermediate\_value})})\)
  
  - Processes input key/value pair
  - Produces set of intermediate pairs

  reduce \((\text{out\_key}, \text{list(\text{intermediate\_value})}) \rightarrow \text{list(\text{out\_value})})\)
  
  - Combines all intermediate values for a particular key
  - Produces a set of merged output values (usually just one)

Inspired by primitives from functional programming languages such as Lisp, Scheme, and Haskell
MapReduce

- Google: [Dean 2004]
- Open source implementation: Hadoop

- Map-reduce = high-level programming model and implementation for large-scale parallel data processing
Motivation: large-scale data processing

- Want to process lots of data, unstructured or structured

- Want to parallelize across hundreds/thousands of commodity computers
  - New definition of cluster computing: *large numbers of low-end processors working in parallel to solve a computing problem.*
  - Parallel DB: *a small number of high-end servers.*

- Want to make this easy
Implementation

- There is one master node
- Master partitions input file into $M$ splits, by key
- Master assigns workers (=servers) to the $M$ map tasks, keeps track of their progress
- Workers write their output to local disk, partition into $R$ regions
- Master assigns workers to the $R$ reduce tasks
- Reduce workers read regions from the map workers’ local disks
Why is MapReduce successful?

- **Easy**
  - Democratization of parallel computing
  - Just two *serial* functions
  - Time to first query: a few hours (contrast with parallel DB...)

- **Flexible**
  - Schema-free, “In situ” processing
  - “First, load your data into the database...”
  - “First, convert your images to bitmaps...”
  - “First, encode your 3D mesh as triangle soup...”

- **Fault-tolerance**
Example: Count Word Occurrences

map(String input_key, String input_value):
    // input_key: document name
    // input_value: document contents
    for each word w in input_value:
        EmitIntermediate(w, "1");

reduce(String output_key, Iterator intermediate_values):
    // output_key: a word
    // output_values: a list of counts
    int result = 0;
    for each v in intermediate_values:
        result += ParseInt(v);
    Emit(AsString(result));

How do we implement this using a relational DBMS? Customized data loading (data may be used only once), then Group By.
Click Stream Analysis: Page Frequencies

Clicks(time, url, referral_url, user_id, geo_info...)

map(String tuple_id, String tuple):
    EmitIntermediate(url, "1");

reduce(String url, Iterator list_tuples):
    int result = 0;
    for each t in list_tuples:
        result += ParseInt(t);
    Emit(AsString(result));

Select count(*)
From Clicks
Group By url;
Parallelism

- The map() function is stateless, so many instances can run in parallel on different splits (chunks) of input data.
- The reduce() function is stateful, but works on an output key at a time, so many copies can run in parallel on different keys (groups).

- Performance bottleneck: reduce phase can’t start until map phase is completely finished.
MapReduce summary

- Hides scheduling and parallelization details
- However, very limited queries
  - Difficult to write more complex tasks
  - Need multiple map-reduce operations
- Solution:
  - Use MapReduce as a runtime for higher level languages
  - Pig (Yahoo!, now apache project): RA-like operators
  - Hive (apache project): SQL
  - Scope (MS): SQL ! But proprietary...
  - DryadLINQ (MS): LINQ ! But also proprietary...
MapReduce: A major step backwards?

- Seminal debate in Jan 2008
  - David DeWitt, Michael Stonebraker

- Five points
  - MapReduce is a step backwards in database access
  - MapReduce is a poor implementation
  - MapReduce is not novel
  - MapReduce is missing features
  - MapReduce is incompatible with the DBMS tools
MapReduce is
A step backwards in database access

- No schema or schema free
- Separation of the schema from the application is good
- High-level access languages are good
MapReduce is
A poor implementation

- No index. Only offers brute force access.
- Poor handling of skew
- Shuffle phase incurs a huge random access on disks
MapReduce is Not novel

- User-defined functions have been around in database for decades
- Many of the parallel distributed processing techniques have been extensively researched in database literature
MapReduce is Missing features

- Bulk loader
- Indexing
- Updates
- Transactions
- Integrity constraints
- Referential integrity
- Views
MapReduce is Incompatible with the DBMS tools

- Report writers
- Business intelligence tools
- Data mining tools
- Replication tools
- Database design tools
Do you agree or disagree?
Making parallelism simple

- Sequential reads = good read speeds
- In large cluster failures are guaranteed; MapReduce handles retries
- Good fit for batch processing applications that need to touch all your data:
  - data mining
  - model tuning
- Bad fit for applications that need to find one particular record
- Bad fit for applications that need to communicate between processes; oriented around independent units of work
MapReduce vs RDBMS

- **RDBMS**
  - Declarative query languages
  - Schemas
  - Logical Data Independence
  - Indexing
  - Algebraic Optimization
  - Caching/Materialized Views
  - ACID/Transactions
  - DryadLINQ, Pig, HIVE
  - HIVE, Pig
  - Hbase
  - Pig, (Dryad, HIVE)

- **MapReduce**
  - High Scalability
  - Fault-tolerance
  - “One-person deployment”
Is it silly for MapReduce to try and be like a parallel database?
- SQL on Hadoop: Hive

Is it silly for a parallel database to try and be like MapReduce?
- MapReduce in SQL: Greenplum, Aster Data, ...
Things people like about Hadoop

- $0 to get started
  - No viable open source parallel database
- Scalability and fault tolerance on commodity hardware
  - Easy to set up
  - Works ok without tuning
- Freedom from the “Warehouse Priesthood”
  - Analysts like to load data in HDFS and experiment with it
  - A rigid warehouse schema is often not what they want
- Open source brings innovation and choices
  - Languages, ML libs, …
- Extensibility & Programmability of the platform
  - Able to do stuff they probably couldn’t do in a parallel database
Shared-nothing parallel databases

- Teradata
- Greenplum
- Netezza
- Aster Data Systems
- Datallegro
- Vertica
- MonetDB

- EMC (July 2010)
- IBM (Sep 2010)
- Teradata (March 2011)
- Microsoft (July 2008)
- HP (Feb 2011)
- Commercialized as Vectorwise
What is Pig?

- An engine for executing programs on top of Hadoop
- It provides a language, Pig Latin, to specify these programs
- An Apache open source project
Why use Pig?

Suppose you have user data in one file, website data in another, and you need to find the top 5 most visited sites by users aged 18 - 25.
In MapReduce

public class MRExample {
  public static class LoadPages extends MapReduceBase
      implements Mapper<LongWritable, Text, Text, Text> {
    public void map(LongWritable key, Text val,
        OutputCollector<Text, Text> output, Reporter reporter)
        throws IOException {
      // Pull the key out
      String line = val.toString();
      int firstComma = line.indexOf(","); // First comma
      int secondComma = line.indexOf(",", firstComma + 1); // Second comma
      Text url = new Text(line.substring(firstComma + 1, secondComma));
      Text page = new Text(line.substring(secondComma + 1));
      // Prepares an index to the value so we know which file it's from.
      long asso = 0;
      while (secondComma > 0) {
        asso = asso * 10 + (line.charAt(secondComma) - '0');
        secondComma = secondComma - 1;
      }
      output.collect(url, page);
    }
  }
  public static class LoadFilterUsers extends MapReduceBase
      implements Mapper<Text, Text, Text, Text> {
    public void map(Text key, Text val,
        OutputCollector<Text, Text> output, Reporter reporter)
        throws IOException {
      // Pull the key out
      String line = val.toString();
      int secondComma = line.indexOf(",", 1); // Second comma
      Text user = new Text(line.substring(secondComma + 1));
      Text value = new Text(line.substring(1, secondComma));
      // Prepares an index to the value so we know which file it's from.
      long asso = 0;
      while (secondComma > 0) {
        asso = asso * 10 + (line.charAt(secondComma) - '0');
        secondComma = secondComma - 1;
      }
      output.collect(user, value);
    }
  }
  public static class Join extends MapReduceBase
      implements Reducer<Text, Text, Text, Text> {
    public void reduce(Text key, Text[] values,
        OutputCollector<Text, Text> output, Reporter reporter)
        throws IOException {
      // Pull the key out
      String line = new String(values[0].toString());
      int firstComma = line.indexOf(",", 1); // First comma
      int secondComma = line.indexOf(",", firstComma + 1); // Second comma
      Text page = new Text(line.substring(firstComma + 1, secondComma));
      Text value = new Text(line.substring(secondComma + 1));
      // Prepares an index to the value so we know which file it's from.
      long asso = 0;
      while (secondComma > 0) {
        asso = asso * 10 + (line.charAt(secondComma) - '0');
        secondComma = secondComma - 1;
      }
      output.collect(page, value);
    }
  }
}

170 lines of code, 4 hours to write
In Pig Latin

Users = load 'users' as (name, age);
Fltrd = filter Users by
   age >= 18 and age <= 25;
Pages = load 'pages' as (user, url);
Jnd = join Fltrd by name, Pages by user;
Grpd = group Jnd by url;
Smmrd = foreach Grpd generate group,
   COUNT(Jnd) as clicks;
Srtd = order Smmrd by clicks desc;
Top5 = limit Srtd 5;
store Top5 into 'top5sites';

9 lines of code, 15 minutes to write
Essence of Pig

- Map-Reduce is too low a level to program, SQL too high
- Pig Latin, a language intended to sit between the two:
  - Imperative
  - Provides standard relational transforms (join, sort, etc.)
  - Schemas are optional, used when available, can be defined at runtime
  - User Defined Functions are first class citizens
  - Opportunities for advanced optimizer but optimizations by programmer also possible
How It Works

Script
A = load
B = filter
C = group
D = foreach

Parser
Logical Plan ≈ relational algebra
Semantic Checks
Logical Plan
Logical Optimizer
Plan standard optimizations
Logical Plan
Logical to Physical Translator
Physical Plan
MapReduce Launcher
Map-Reduce Plan
Map-Reduce Plan = physical operators broken into Map, Combine, and Reduce stages
Physical To MR Translator
Physical Plan
Jar to hadoop
hadoop
Physical Plan = physical operators to be executed
Fragment replicate join

Users = load 'users' as (name, age);
Pages = load 'pages' as (user, url);
Jnd = join Pages by user, Users by name using "replicated";
Hash join

Users = load 'users' as (name, age);
Pages = load 'pages' as (user, url);
Jnd = join Users by name, Pages by user;

- Map 1: Users block n
- Map 2: Pages block m
- Reducer 1: (1, user) → (1, fred), (2, fred), (2, fred)
- Reducer 2: (2, name) → (1, jane), (2, jane), (2, jane)
Skew join

Users = load 'users' as (name, age);
Pages = load 'pages' as (user, url);
Jnd = join Pages by user, Users by name using "skewed";

Map
  1 Pages block n
  2 Users block m
Reducer
  1  (1, user)
  2  (1, fred, p1)
  3  (1, fred, p2)
  4  (1, fred, p3)
  5  (1, fred, p4)
  6  (2, name)
  7  (2, fred)
  8  (2, fred)
Merge join

Users = load 'users' as (name, age);
Pages = load 'pages' as (user, url);
Jnd = join Pages by user, Users by name using "merge";

<table>
<thead>
<tr>
<th>Pages</th>
<th>Users</th>
</tr>
</thead>
<tbody>
<tr>
<td>aaron</td>
<td>aaron</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>1</th>
<th>2</th>
</tr>
</thead>
<tbody>
<tr>
<td>Users</td>
<td>Pages</td>
</tr>
<tr>
<td>aaron</td>
<td>amr</td>
</tr>
<tr>
<td>amr</td>
<td>barb</td>
</tr>
</tbody>
</table>

Map 1

<table>
<thead>
<tr>
<th>Map 2</th>
</tr>
</thead>
<tbody>
<tr>
<td>Users</td>
</tr>
<tr>
<td>amy</td>
</tr>
<tr>
<td>amr</td>
</tr>
<tr>
<td>barb</td>
</tr>
</tbody>
</table>
What are people doing with Pig

- At Yahoo ~70% of Hadoop jobs are Pig jobs
- Being used at Twitter, LinkedIn, and other companies
- Available as part of Amazon EMR web service and Cloudera Hadoop distribution

What users use Pig for:
- Search infrastructure
- Ad relevance
- Model training
- User intent analysis
- Web log processing
- Image processing
- Incremental processing of large data sets
Questions

- Is it silly for MapReduce to try and be like a parallel database?
  - SQL on Hadoop: Hive

- Is it silly for a parallel database to try and be like MapReduce?
  - MapReduce in SQL: Greenplum, Aster Data, ...