

A programming model in Cloud: MapReduce

- Programming model and implementation developed by Google for processing large data sets
- Users specify
 - ❑ a map function to generate a set of intermediate key/value pairs
 - ❑ a reduce function that merges all intermediate values associated with the same intermediate key
- Programs can be automatically parallelized and executed on a large cluster of machines
 - ❑ Partitioning input data
 - ❑ Scheduling execution
 - ❑ Handling machine failure
 - ❑ Managing communication
- Allow users without much experience to utilize the resources of a large distributed system

An example: Count word frequency

```
map(String key, String value):  
    // key: document name  
    // value: document contents  
    for each word w in value:  
        EmitIntermediate(w, "1");  
  
reduce(String key, Iterator values):  
    // key: a word  
    // values: a list of counts  
    int result = 0;  
    for each v in values:  
        result += ParseInt(v);  
    Emit(AsString(result));
```

MapReduce operations

→ When the user program calls the MapReduce function, the following sequence of operations occurs

- ❑ **Partition input data**

- Split the input files into M pieces;

- ❑ **Assign tasks**

- starts up many copies of the program on a cluster of machines: one is master and others are workers
- Master assigns map tasks and reduce tasks to workers

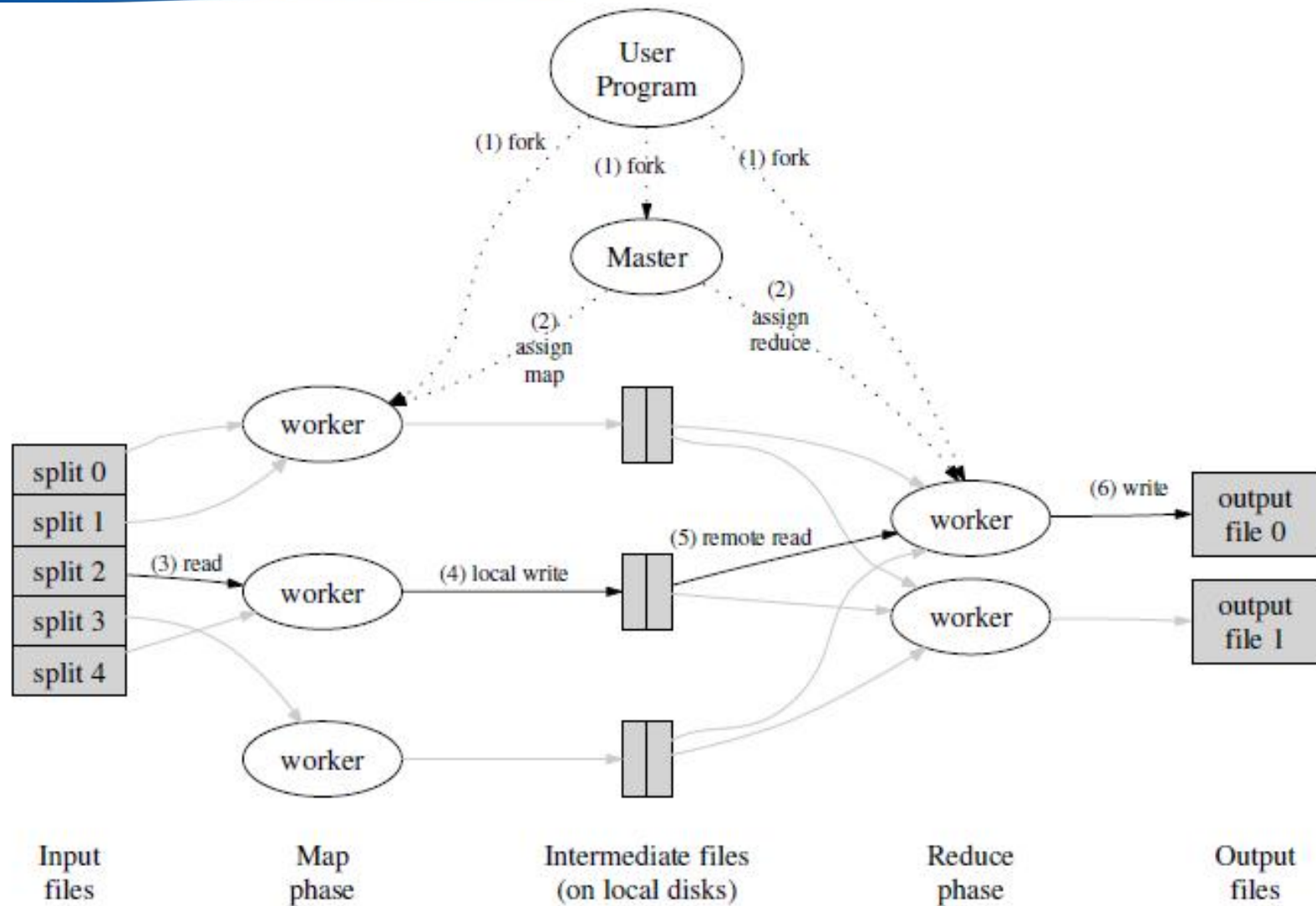
- ❑ **A Map worker does:**

- reads input split,
- abstracts key/value pairs from the input data,
- passes each pair to the Map function,
- the Map function produces the intermediate key/value pairs
- Buffers the intermediate key/value pairs to the local disks; the disk locations of the data are passed back to the master; the master forwards the locations to the reduce workers

MapReduce operations

- ❑ **The reduce workers does**
 - **uses remote procedure calls to read the buffered intermediate key/value pairs from the local disks of the map workers.**
 - **The reduce workers passes the intermediate key/value pairs to the Reduce function**
- ❑ **After all map and reduce tasks have been completed, the call of MapReduce returns to the user code**

MapReduce execution flow



MapReduce library

→ Partition the input data

→ Startup and schedule execution

→ Scheduling map and reduce workers

→ Manage communication

→ Reduce workers retrieve the intermediate results from Map workers

→ Handle machine failure

















- ❑ Ping every worker periodically
- ❑ If no response in a certain amount of time, mark the worker as failed
- ❑ Any map or reduce tasks in progress on a failed worker will be rescheduled
- ❑ The map tasks completed by the failed worker will be rescheduled
- ❑ The reduce tasks completed by the failed worker do not need to be rescheduled

Popular Cloud Systems




- ⑩ **Google Cloud**
- ⑩ **Amazon Cloud**

Google provides numerous online services













Search

-  [Alerts](#)
Get email updates on the topics of your choice
-  [Blog Search](#)
Find blogs on your favorite topics
-  [Book Search](#)
Search the full text of books
-  [Checkout](#)
Complete online purchases more quickly and securely
-  [Desktop](#)
Search and personalize your computer
-  [Earth](#)
Explore the world from your computer
-  [Finance](#)
Business info, news, and interactive charts
-  [GOOG-411](#)
Find and connect with businesses from your phone, for free
-  [iGoogle](#)
Add news, games and more to the Google homepage
-  [Images](#)
Search for images on the web
-  [Maps](#)
View maps and directions
-  [News](#) - now with [archive search](#)
Search thousands of news stories
-  [Notebook](#)
Clip and collect information as you surf the web
-  [Patent Search](#)
Search the full text of US Patents
-  [Product Search](#)
Search for stuff to buy
-  [Scholar](#)
Search scholarly papers

Explore and innovate

-  [Code](#)
Download APIs and open source code
-  [Custom Search](#) ^{New!}
Create a customized search experience for your community
-  [Labs](#)
Explore Google's technology playground

Communicate, show & share

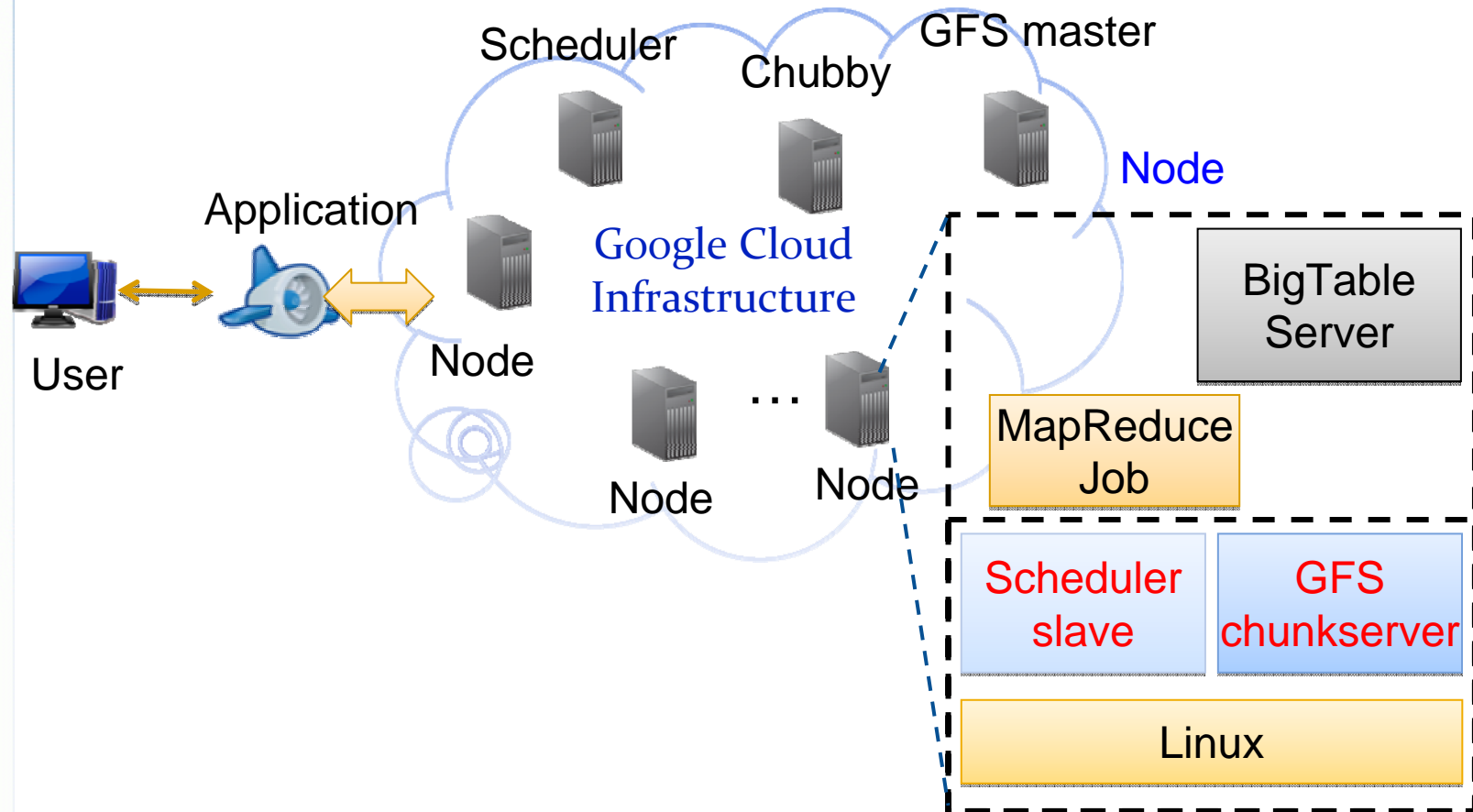
-  [Blogger](#)
Share your life online with a blog – it's fast, easy, and free
-  [Calendar](#)
Organize your schedule and share events with friends
-  [Docs](#)
Create and share your projects online and access them from anywhere
-  [Gmail](#)
Fast, searchable email with less spam
-  [Groups](#)
Create mailing lists and discussion groups
-  [orkut](#) [Orkut](#)
Meet new people and stay in touch with friends
-  [Picasa](#)
Find, edit and share your photos
-  [Reader](#)
Get all your blogs and news feeds fast
-  [SketchUp](#)
Build 3D models quickly and easily
-  [Talk](#)
IM and call your friends through your computer
-  [Translate](#)
View web pages in other languages
-  [YouTube](#)
Watch, upload and share videos

Google Cloud Computing: Focusing on User Data



- User data is stored in “Cloud”
- Data access is not constrained by geographic locations
- Data can be conveniently shared

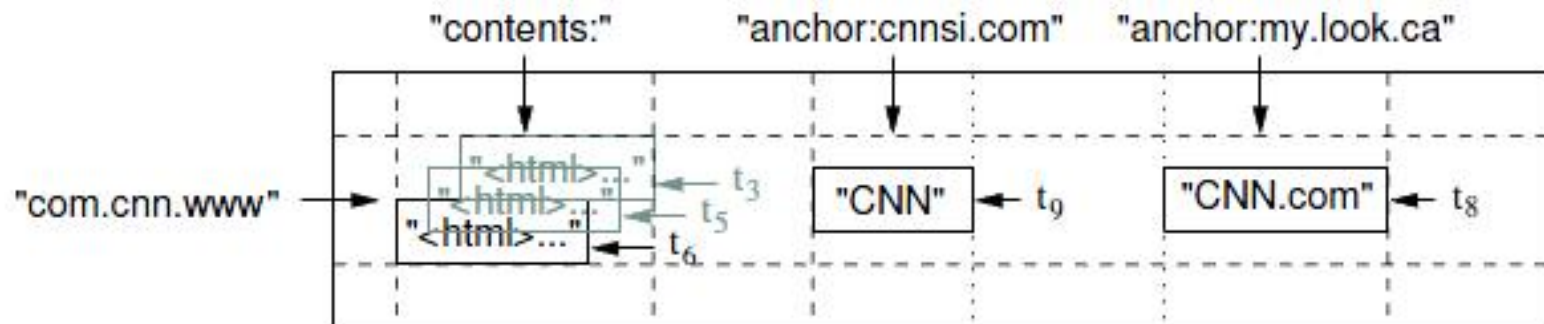
Google Cloud Infrastructure



Bigtable: a distributed storage system in Google Cloud

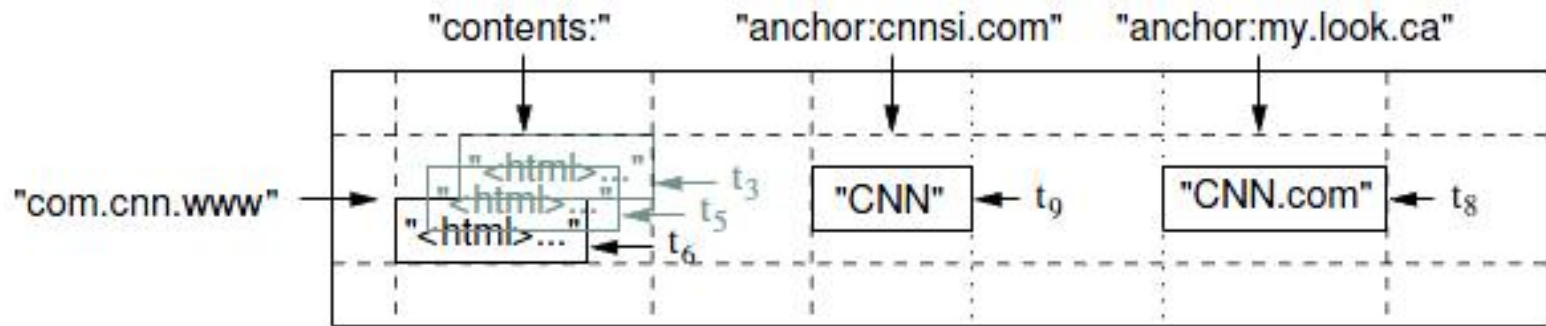
→ Resembles a database

- ❑ Aims to scale to a very large size: petabyte of data across thousands of nodes
- ❑ Provides a simple data model that
 - Data is indexed using row and column names
 - supports dynamic partition of the table
 - Data is treated as strings



Bigtable: a distributed storage system in Google Cloud

- A row key in a table is an arbitrary string
- Bigtable maintains data in lexicographic order by row key
 - Web pages in the same domain are grouped together into contiguous rows by reversing the hostname components of the URLs
- A table is dynamically partitioned into row ranges
- Each row range is called a tablet
 - The unit of distribution and load balancing
 - Tablets are distributed and stored in multiple machines



→ Timestamps

- 64-bit integers
- Can be used to specify only new enough versions are kept

→ API

- creating and deleting tables and columns

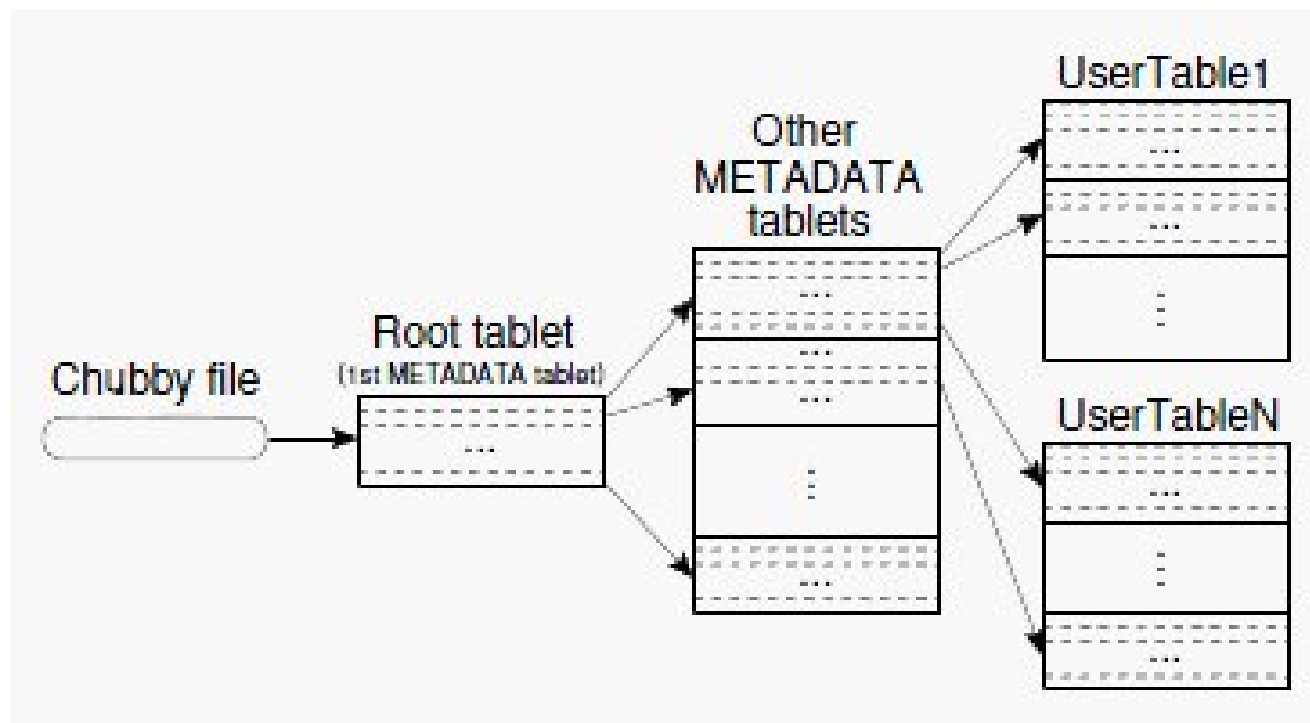
Big picture of BigTable

- **A Bigtable cluster stores a number of tables.**
- **Each table consists of a set of tablets**
- **Each tablet contains all data associated with a row range.**
- **Initially, each table consists of just one tablet.**
- **As a table grows, it is automatically split into multiple tablets, each approximately 100-200 MB in size by default**

Infrastructure of BigTable

- **If a number of machines are needed to store the a BigTable**
 - **One machine will be elected as the master;**
 - **Other machines are called tablet servers**
- **The master is responsible for assigning tablets to tablet servers, balancing tablet-server load**
- **Each tablet server manages a set of tablets (typically we have somewhere between ten to a thousand tablets per tablet server)**

Infrastructure of BigTable



Chubby

- **A lock service**

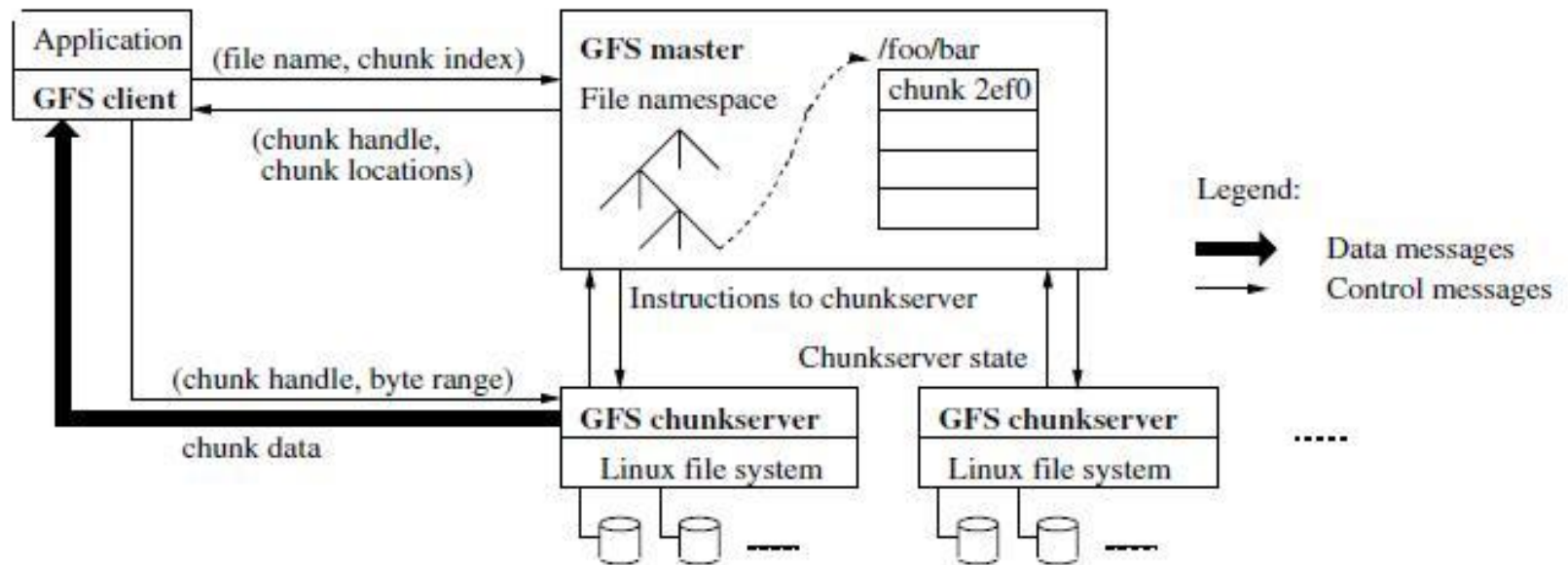
- **synchronize activities: e.g. two clients cannot update a cell at the same time, data consistency among different tablet servers**
- **Distributed consensus: e.g., electing a leader from a set of equivalent servers**

- **A simple file system**

- **performs whole-file reads and writes,**
- **Both GFS and Bigtable use Chubby to store a small amount of meta-data (e.g., root of their data structure)**

Google File System

- Provide the traditional file system interface
- Files are divided into chunks, stored in chunkservers
- Consists of a single Master and multiple chunkservers
- Master stores the locations of the file chunks



Google App Engine

- Complex Server structure: Apache, SQL, etc
- Endless monitoring and debugging
- Exhausting System upgrading and fault tolerance

Concise development and running platform for web applications

Running on the data centre provided by Google

Managing the entire life cycle of web applications



EASY TO USE

EASY TO SCALE

EASY TO START

Google App Engine

- Enable clients to run web applications on Google's infrastructure
- Clients do not have to maintain servers
 - Users just upload the applications
 - The applications can then serve the users of the web applications
- Supports Java and Python language
- You pay for what you use
 - There are no set-up and maintenance costs

Google App Engine



Run your web applications on Google's infrastructure. Build apps on the same scalable systems that power Google applications.

Try it now

This is **preview release** of App Engine -- for now applications are restricted to the free account limits.

Featured: Serving BuddyPoke's 12 Million users with App Engine

App Engine has been a godsend in terms of our scalability...we saw a **8x jump in installs** overnight and we are now serving **12 million users!**

Dave Westwood, BuddyPoke!

[Who's using App Engine?](#) | [Why App Engine?](#)



Getting Started

1. [Sign up](#) for an App Engine account.
2. [Download](#) the App Engine SDK.
3. Read the [Getting Started Guide](#).
4. Check out the [App Gallery](#) to see sample applications.

Demonstration



Developing and deploying an app on Google App Engine. [Watch Now](#)

Recent Articles

[Running App Engine Applications on Django](#)

In this article we explain how to port applications that were originally designed for App Engine with Django to any environment that supports Django. The helper application provides a Django based implementation of the App Engine APIs...

[Building an OpenSocial App with Google App Engine](#)

Welcome to the Cloud

Amazon Web Services makes cloud computing a reality for hundreds of thousands of customers looking for a cost-effective infrastructure to deploy highly scalable and dependable solutions.

[Learn how you can benefit from cloud computing](#)



1 2 3

Get Started

Sign up for a free AWS account.

[Sign Up Now](#)

Explore Products

Infrastructure Services

- Amazon Elastic Compute Cloud (Amazon EC2)
- Amazon SimpleDB
- Amazon Simple Storage Service (Amazon S3)
- Amazon Simple Queue Service (Amazon SQS)
- AWS Premium Support

Payments & Billing

On-Demand Workforce

Web Search & Information

Amazon Fulfillment & Associates

News & Events

What's New?

Media Coverage

Events

- | | |
|--------------|--|
| Nov 06, 2008 | Announcing the Finalists in the AWS Start-Up Challenge |
| Nov 03, 2008 | Salesforce.com Announces Force.com Toolkit for AWS |
| Oct 23, 2008 | Amazon EC2 Running Windows Server Now Available |
| Oct 23, 2008 | Amazon EC2 Exits Beta and Now Offers a Service Level Agreement |
| Oct 23, 2008 | AWS Provides Glimpse at New Amazon EC2 Features |

[view all](#)

Developers

Simply sign up & start developing in the cloud with these resources and tools:

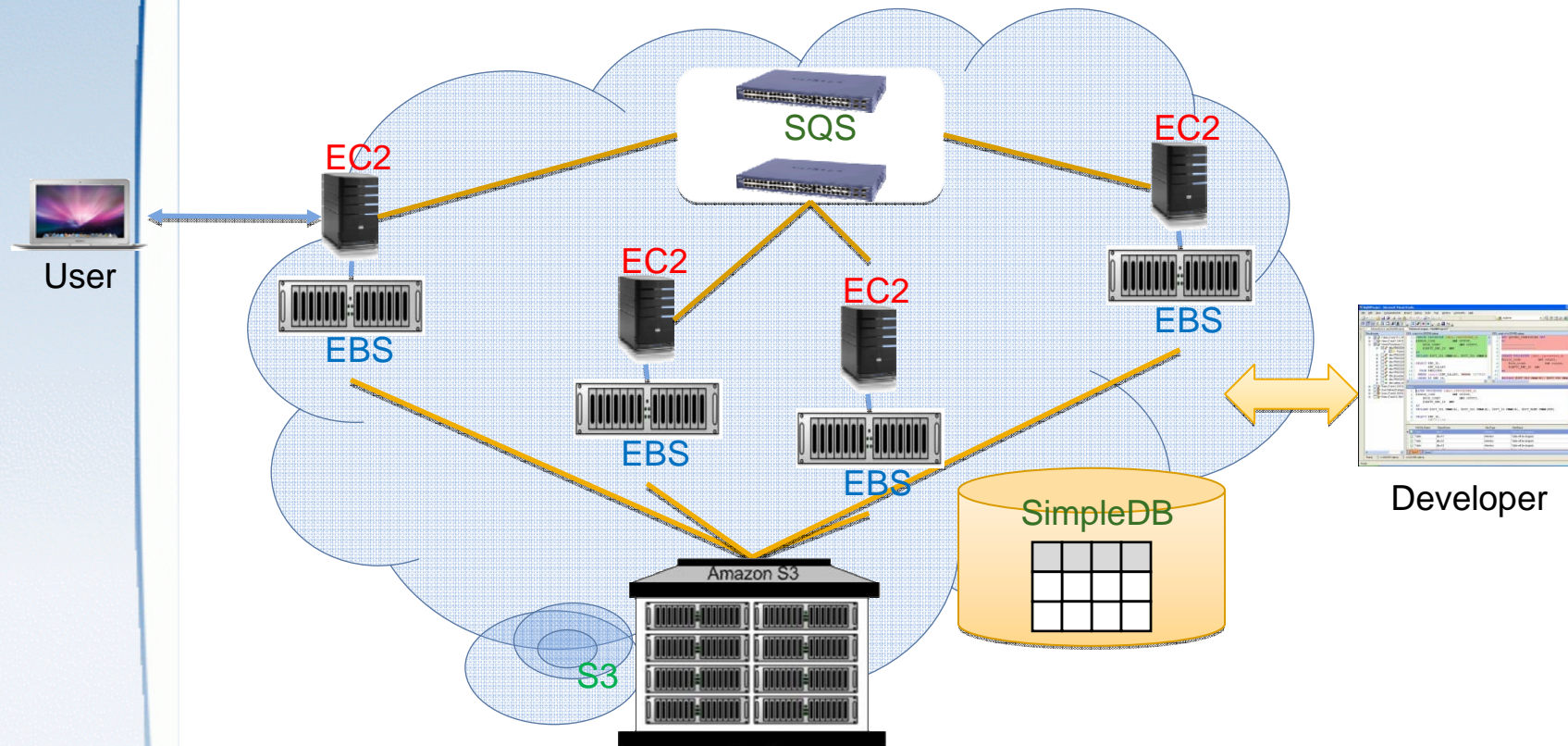
- Technical Documentation
- Cloud Architectures Whitepaper (pdf)
- Amazon Machine Images
- AWS Community Forums

Business Managers

Learn how Amazon Web Services enables you to reach business goals faster:

- AWS Solutions for Enterprise Customers
- Security Whitepaper (pdf)
- Case Studies & Customer Testimonials
- AWS Blog 

Amazon Elastic Computing Cloud (EC2)



EC2: Running Instance of Virtual Machines

S3: Simple Storage Service, SOAP, Object Interface

SimpleDB: Simplified Database

EBS: Elastic Block Service, Providing the Block Interface, Storing Virtual Machine Images

SQS: Simple Queue Service

AWS is Production-level Cloud System

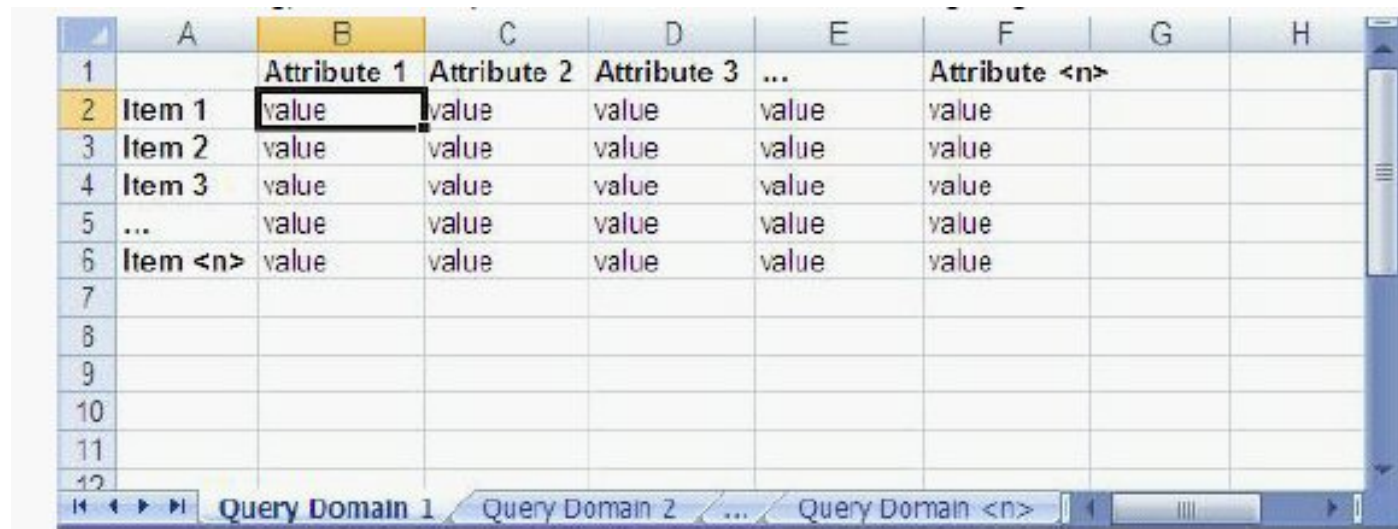


Amazon SimpleDB

→ A web service for running queries on structured data in real time

- Data model

- Domain, Items, Attributes, Values



The screenshot shows a data table with columns labeled A through H and rows numbered 1 through 12. The table structure is as follows:

	A	B	C	D	E	F	G	H
1		Attribute 1	Attribute 2	Attribute 3	...	Attribute <n>		
2	Item 1	value	value	value	value	value		
3	Item 2	value	value	value	value	value		
4	Item 3	value	value	value	value	value		
5	...	value	value	value	value	value		
6	Item <n>	value	value	value	value	value		
7								
8								
9								
10								
11								
12								

The table is displayed in a window titled "Query Domain 1" with a navigation bar at the bottom showing "Query Domain 1", "Query Domain 2", "...", and "Query Domain <n>".

Amazon SimpleDB

- **API**

- **CreateDomain, DeleteDomain, PutAttributes, Select,**

- **Consistency**

- **Keeps multiple copies of each domain**

Amazon Simple Storage Service

→ S3 has a simple web services interface that can be used to store and retrieve any amount of data, at any time, from anywhere on the web

- ❑ Buckets: a container for objects
- ❑ Objects
 - Objects consist of object data and metadata
 - The metadata is a set of name-value pairs that describe the object
- ❑ Keys
- ❑ Versioning
 - An object consists of two components: a key and a version ID
- ❑ Operations
 - Create a Bucket, write an object, listing keys
- ❑ provides a REST and a SOAP interface
 - The REST API: use web browser to access S3
 - The SOAP API: write web service client to access S3
- ❑ Regions: choose the region where S3 will store the bucket user create
 - US, US-West, EU

Amazon Simple Storage Service

→ Amazon S3 data consistency model

- ❑ High availability is achieved by replicating data across multiple servers within Amazon data centre
- ❑ A process write a new object to S3 and immediately attempts to read it. Until the change is fully propagated, S3 might return “key does not exist”

Reference papers

- **“Xen and the Art of Virtualization”**
- **“MapReduce: Simplified Data Processing on Large Clusters”**
- **“Bigtable: A Distributed Storage System for Structured Data”**
- **“The Chubby lock service for loosely-coupled distributed systems”**
- **“The Google File System”**

Summary

- ⑩ **Cloud computing Concepts**
- ⑩ **Virtualization technology**
- ⑩ **Programming model in Cloud**
- ⑩ **Google Cloud**
- ⑩ **Amazon Cloud**