# Column-Family Stores

## Introduction

# Basics

- AKA (Also Know As) Wide-column, columnar
- Data model
  - rows that have many columns, all associated with the same key
- Columns family
  - is group of related columns
  - often accessed together

# Data Model: Column

- column is the basic data item
- represented as 3-tuple
  - column name
  - value
  - timestamp

# Data Model: Row

- Row is a collection of columns attached to the same row key
  - columns can be added to any row at anytime without having to add for all rows

```
// row
   "martin-fowler" : {
        firstName: "Martin",
        lastName: "Fowler",
        location: "Boston"
    }
```

```
// row
   "Jack-fowler" : {
        firstName: "Jack",
        lastName: "Fowler",
   }
```

# Data Model: Column Family (CF)

- CF is a set of columns containing related data
- For example, UserInfo in table User is a column family

User

| | UserInfo | | Likes | | |
|---|---|---|---|---|---|
| 123 | Name | Email | 111 | 222 | ... |
| | Jay | jp@ebay.com | iphone | ipad | |
| ⋮ | | | | | |

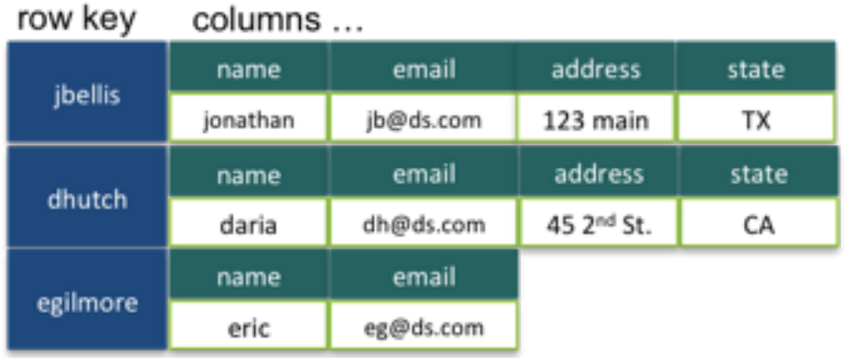
Item

| | ItemInfo | | LikedBy | | |
|---|---|---|---|---|---|
| 111 | Title | Desc | 123 | 4556 | ... |
| | iphone | It's a phone | Jay | John | |
| ⋮ | | | | | |

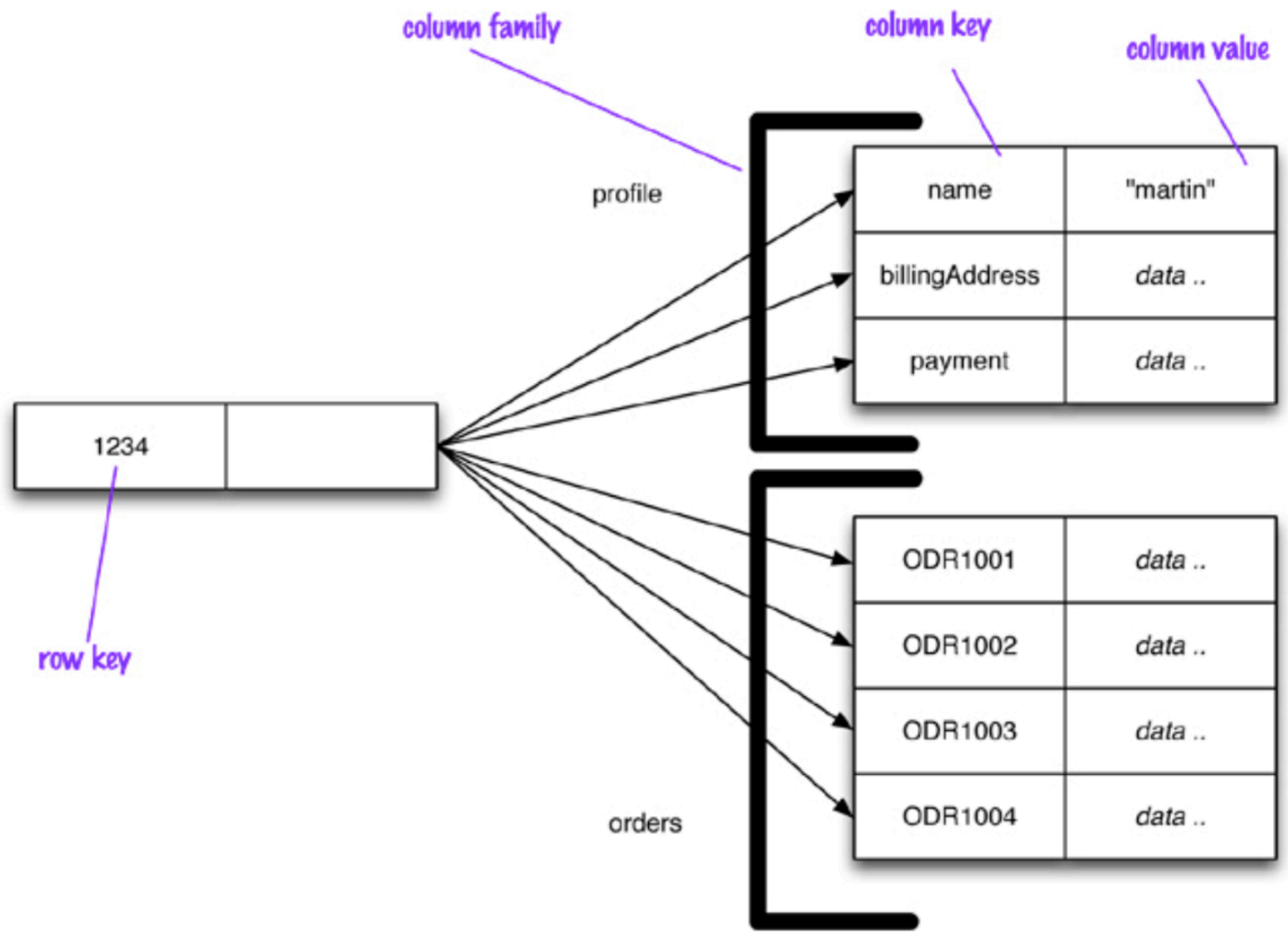
# Data Model: Interpretation 1

- Each column family is equivalent to a relational table

- Column family is considered a map of map
  - Map<rowKey, Map<columnKey, columnValue>>



# Data Model: Interpretation (Visual)



# Column-family Stores

- Representative
  - Cassandra
  - BigTable
  - HBase
  - Hypertable
  - Accumulo

Ranked list: http://db-engines.com/en/ranking/wide+column+store

# BigTable

- Google's paper:
  Chang, F. et al. (2008). Bigtable: A Distributed Storage System for Structured Data. ACM TOCS, 26(2), pp 1–26.

- Data model: column family

- Multi-dimensional map

  - (row:string, column:string, timestamp:int64) —> value

# HBase

- An open source implementation of Google BigTable
- Initial release: 2008
- Implementation: Java
- Runs on top of Hadoop Distributed File System
- Operating systems: linux, windows (you need Cygwin)
- can handle billions of rows with millions of columns

# Cassandra

- Developed at Facebook
- Initial release in 2008, stable release in 2013
- Written in Java
- Operations
  - Cassandra Query Language (CQL)
  - MapReduce support (can cooperate with Hadoop)

# BigTable & HBase in Details

# BigTable: Overview

- Highly available distributed storage
- One big table distributed on multiple nodes
- Built with semi-structured data in mind
  - billion of URLs with their content over time (versions)
  - users data: profiles, preferences, queries
  - Geographical data: road map, satellite images

# BigTable: Large Scale

- Petabytes of data across thousands of computers
- Billions of users
- thousands of queries per second

# BigTable: Uses

- At Google used for
  - Google analytic
    - a web service tool that track and analyze web traffic
  - Google Finance
    - stock information
- Personalized Search
    - based on users preferences show personalized search
- Youtube
- Google Earth & Map
- and more

# A big table

- Appears as one table
- characteristics:
  - sparse
  - distributed
  - multi dimensional map

# Characteristics

- A map
- Is an associative array
- values can be quickly looked up for a given key
- key identifies the row
- value identifies set of columns

# Characteristics

- Persistent
  - Data stays on disk after operations finish
  - Data is stored persistently on disk
- Distributed
  - BigTable data is distributed across multiple machines
  - BigTable runs on top of Google File System (GFS)
  - The table is split based on the rows

# Characteristics

- Sparse
  - different rows have different columns
  - some columns may be empty for some rows
- Sorted
  - BigTable uses associative array (which is not sorted)
  - but BigTable sort based on rows
  - related data will be adjacent
  - For example if we want to store all pages of the same domain, then we reverse the url and use it as row key
    - edu.ptuk.ps all pages from edu domain will be adjacent

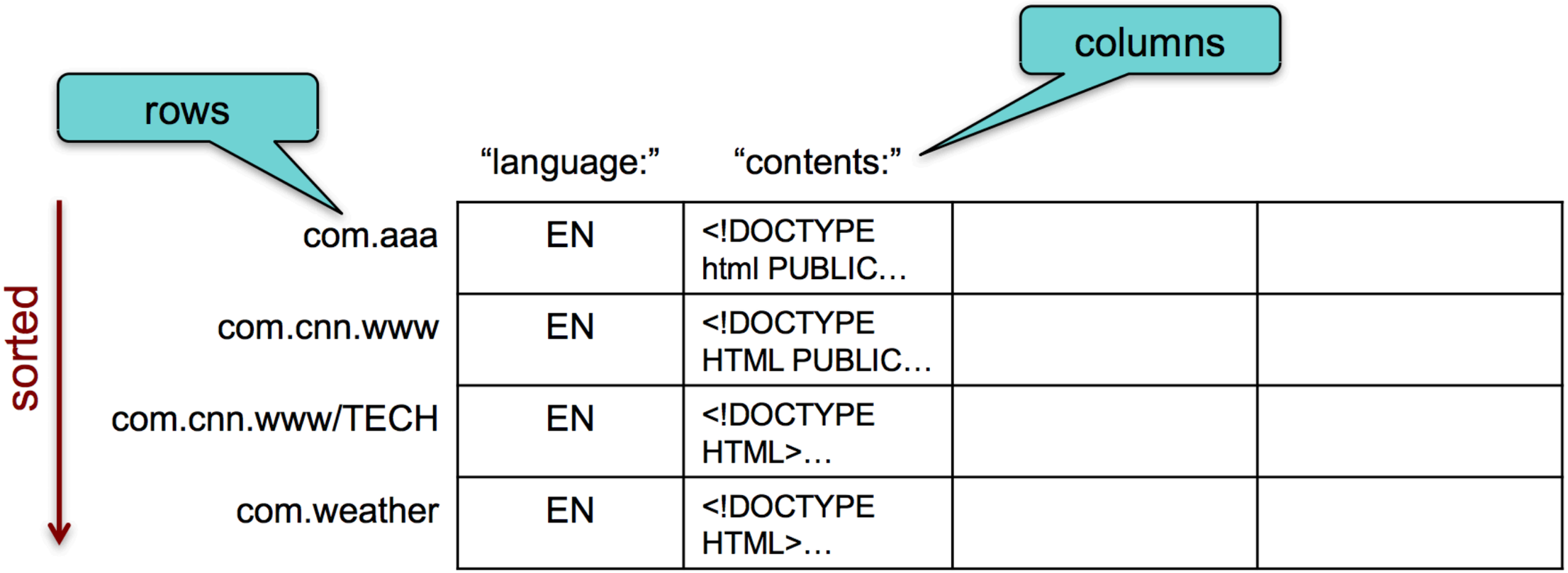
# Characteristics

- Multidimensional
  - Table is indexed by row
  - a Table contains one or more column families
    - at least one column family is created when the table is created
    - a column family can contain various number of columns which are usually related
    - columns in the column family can be created on the fly
- three level of hierarchy; row, column family, and column

# Characteristics

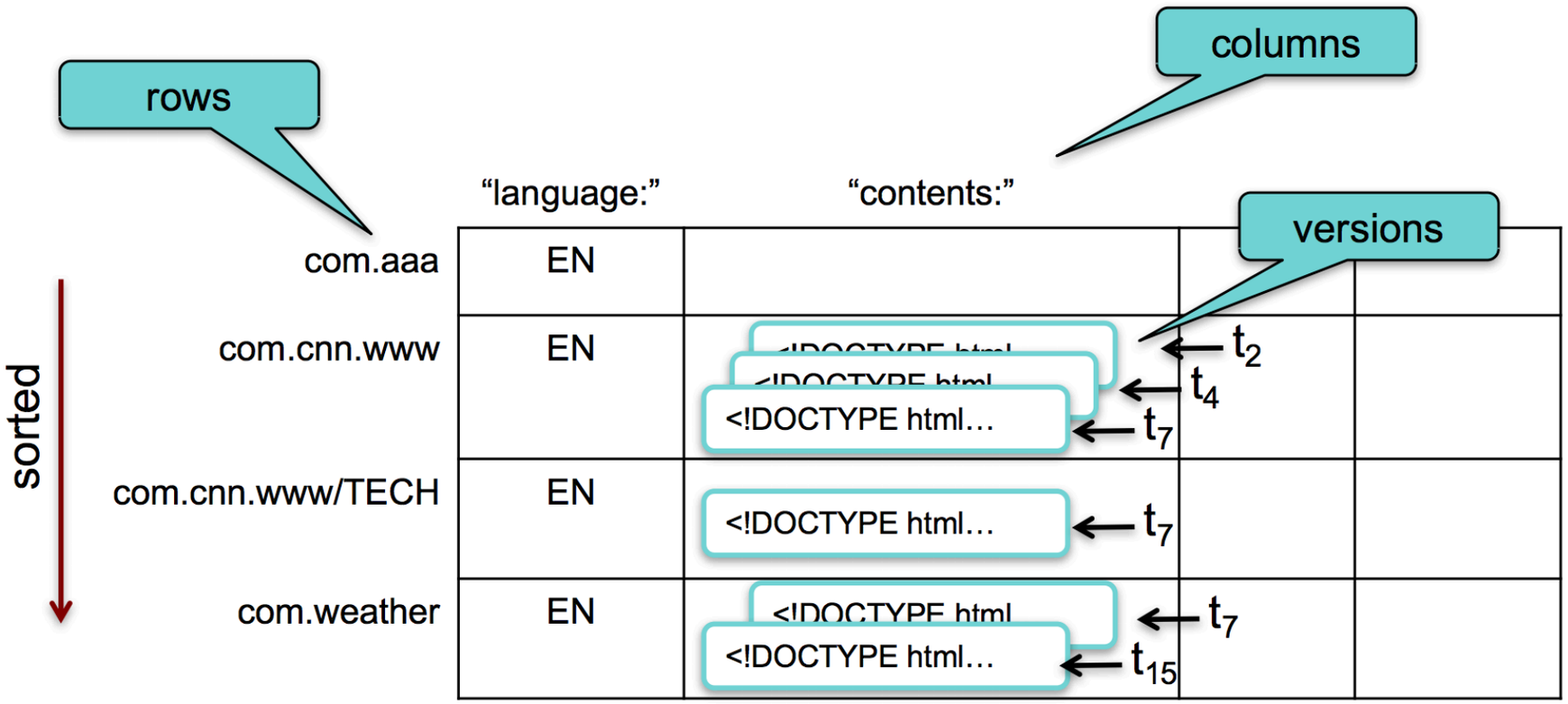
- Time based
  - another dimension in BigTable
  - keep multiple versions of the same data
  - at retrieval time (search) if the time is not specified then the latest version will be returned

# Example


columns
rows
“language:”
“contents:”
sorted
com.aaa
EN
<!DOCTYPE html PUBLIC…
com.cnn.www
EN
<!DOCTYPE HTML PUBLIC…
com.cnn.www/TECH
EN
<!DOCTYPE HTML>…
com.weather
EN
<!DOCTYPE HTML>…

# Table Model

- (row, column, timestamp) -> cell content
- Multiple versions of the same cell



# Rows & Partitioning

- A table is split among rows into sub tables called (**Tablets**)
- A tablet
  - consist of a set of consecutive rows
  - it is the unit of data distribution & load balancing
    - stored on one server
- rows are sorted by key
  - reading is efficient
    - single row or scan range of rows
- designing the row is very important as it determines how data is stored
  - influence read performance, and data distribution

# Table splitting

- A table start as one tablet
- As it grows it will be split into tablets (100 - 200 MB each)
  - can be configured

| | "language:" | "contents:" | | |
|---|---|---|---|---|
| com.aaa | EN | <!DOCTYPE html PUBLIC… | | |
| com.cnn.www | EN | <!DOCTYPE HTML PUBLIC… | | |
| com.cnn.www/TECH | EN | <!DOCTYPE HTML>… | | |
| com.weather | EN | <!DOCTYPE HTML>… | | |

*tablet*

# Splitting a Tablet

| | "language:" | "contents:" | | |
|---|---|---|---|---|
| com.aaa | EN | <!DOCTYPE html PUBLIC… | | |
| com.cnn.www | EN | <!DOCTYPE HTML PUBLIC… | | |
| com.cnn.www/TECH | EN | <!DOCTYPE HTML>… | | |

*Split*

| | | | | |
|---|---|---|---|---|
| com.weather | EN | <!DOCTYPE HTML>… | | |
| com.wikipedia | EN | <!DOCTYPE HTML>… | | |
| com.zcorp | EN | <!DOCTYPE HTML>… | | |
| com.zoom | EN | <!DOCTYPE HTML>… | | |

# Columns & Column Families

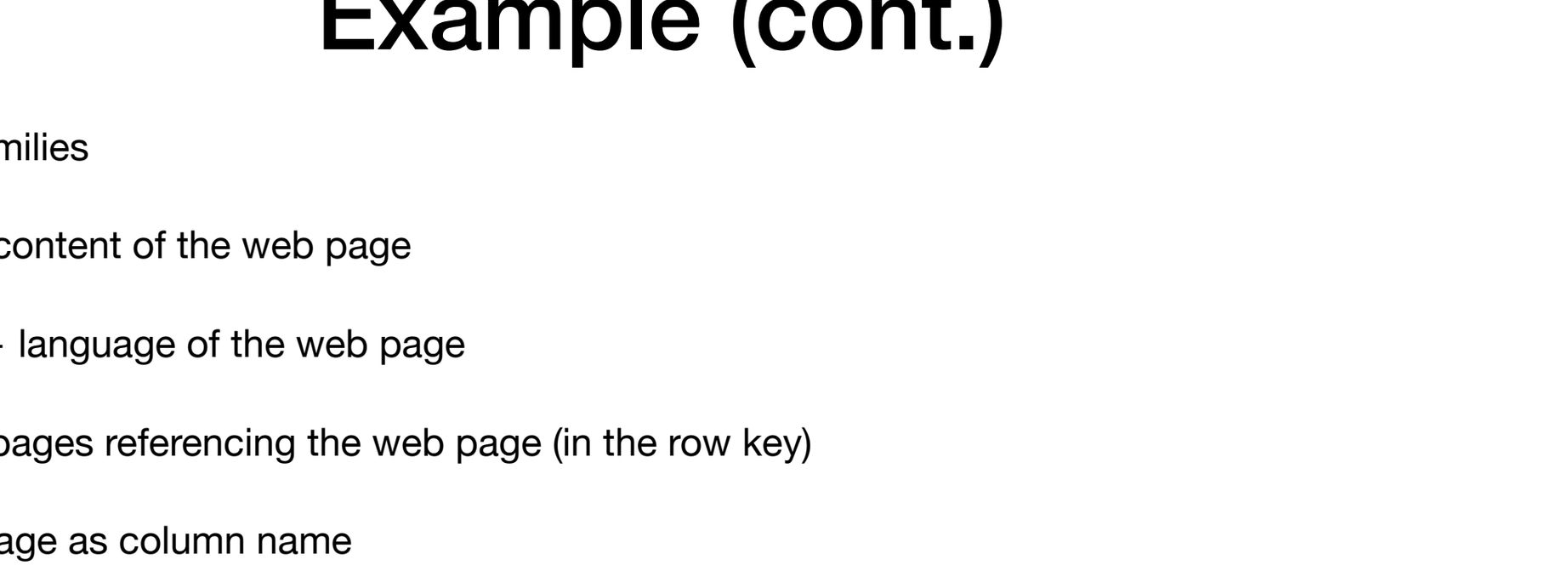
- Column Family
  - group of columns
  - basic unit of data access
  - data is of the same type (related)
  - data in the same column family is compressed together
  - operations:
    - create a column family
    - store data in any column key
- Table can have unlimited number of column families
- Number of columns in the same column family is up to hundreds
- identified by➔ family:qualifier

# Example

- Web pages table
  - store web pages and their information in a table
  - use URL as the row key
  - several column families to store various attributes of Web pages
    - content (multiple content over time)
    - anchors (multiple anchors)
    - language single value

# Example (cont.)

- Three column families
  - “content:” — content of the web page
  - “language:” — language of the web page
  - “anchors” — pages referencing the web page (in the row key)
    - use url of page as column name
    - and the cell value is anchor text
    - ancho is an example of dynamic column family



# Timestamps

- Each column family cell may contain multiple versions of content
- In the previous example, same URL may have different content over time
- Timestamp is identified by 64-bits
    - which represent real time, time when the cell was added
    - or manually specified by the user
- At retrieval time
    - most recent version will be returned if no timestamp is specified
    - if the timestamp is specified
        - if no exact match then the latest version that is earlier than the specified timestamp