Ali Taheri. 989294669

1. Relational databases have some problems same to locking, recovery log, two phase commits, complicated for scaling (Vertical Scaling), relations between data, normalization to reduce redundancy and defined schema that made RDBMS not suitable for some type of applications. Because of these issues, NoSql databases are popular at these days. One example that NoSql is very better than relational databases is saving resume of people. Resume has a lot of unrelated information about a person. In other view, it is a document for each person that we could implement it by document model (same to JSON). Other examples could be social applications, online advertisements or archiving data.
2. Hash is a function for conversion of variable length sequence of bytes to a fixed length sequence of bytes. Input of function could be a string, integer or other data structures. Then we could save data in an address that is based on hashing value, so our read time would improve a lot. In Key- Value stores, we would apply hashing function to Key and based on output of hash(Key), we would save Value in an address that is proportional to hash(Key). For example, we could have 100 buckets and save values on a bucket with number of hash(key) modulo 100.
3. Key-value: (Memcached, Redis) Document: MongoDB Column Family: Cassandra

Graph: Neo4j

Laptop = “ { “Processor” : [ “1.2GHz dual-core Intel Core m3, Turbo Boost up to 3.0GHz, with 4MB L3 cache” , “1.3GHz dual-core Intel Core i5, Turbo Boost up to 3.2GHz, with 4MB L3 cache”, “1.4GHz dual-core Intel Core i7, Turbo Boost up to 3.6GHz, with 4MB shared L3 cache” ] ,

“Memory” : [“ 8GB of 1866MHz LPDDR3 onboard memory”, “8GB of 1866MHz LPDDR3 onboard memory” ] ,

“Hard Disk”: “256GB PCIe-based onboard SSD1storage” ,

“Dimensions” : { “Height”: “0.14–0.52 inch (0.35–1.31 cm)”, “Width”: “11.04 inches (28.05 cm)”, “Depth”: “7.74 inches (19.65 cm)”, “Weight: 2.03 pounds (0.92 kg)”}

}”

1. A lot of failures could happen in data centers. Some of them could be related to power supply, damaged server, problem in core network and external network of data center, accidents, fire, natural disasters or even strikes. All of them could destroy our data or we would not have access to them for a while. For resolving of these issues, we need replication of data. At first stage we could make some redundancies inside data center. For example we could have double power supply, double server, redundant external link as backup and … . In second phase we could make availability zones by building new datacenters with replicated data near to mentioned datacenter. Finally, we could make availability zone in other regions (other side of country or even other continent).
2. a. population\_documents = population.find({'city': $eq :'Cupertino’}) mongodb

b. Select count(\*) from city\_population where city = 'Cupertino’; -🡪 cassandra

c. match (zip:Zip)-[:in\_city]->(city:City{name:'Cupertino'})-[:in\_county]->(county:County)-[:in\_state]->(state:State) return zip.zipcode,county.name,state.name; ---🡪 Neo4j