# STORAGE SERVER FOR APARTMENT BUILDINGS

**TEAM 37** 

**ARYAMMAN JAIN** 

PRANAV MEHNDIRATTA

VAIBHAV VIJAY

### **OVERVIEW**

- Data management at apartment buildings
- Need efficient means of storage and methods for maintaining this data
- Usage of hash tables and C string libraries to implement system
- Performance of aforementioned approaches
- Importance of organizing data



## RESIDENTIAL DATA MANAGEMENT

- Client: management, superintendent, security
- Problem
  - Current system is fully paper based and unorganized
  - Access is slow and inaccurate
- Solution
  - Data to be stored in hash tables and centralized at one location
  - Chosen data structure is built for quick access and consistent performance





## WHAT DATA STRUCTURE ARE WE USING?

- Requirements:
  - · Direct access of a tenant record
  - Capable of storing large amounts of data
  - Easy to maintain
- Alternatives: array,
  linked list, binary
  search tree, hash table

Our pick: Hash Table

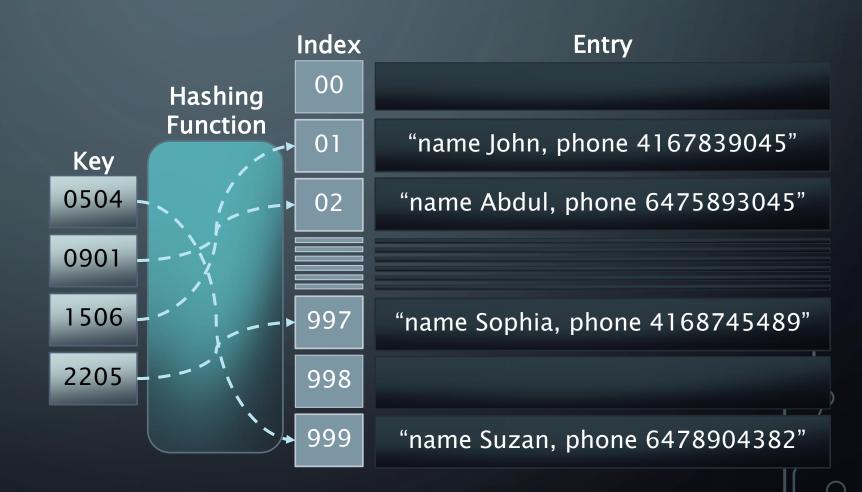
Why?

- Possibility of collisions
- · Clustering of data
- Known number of entries

- Fast access
- Consistent performance
- Easy to perform operations

## THE IMPLEMENTATION OF OUR STRUCTURE

- **Key**: Apartment number
- Hashing function: Function to convert keys to addressable indices in array
- Index: Position of data in array
- Entry: Data to be recorded



## HOW ARE WE READING DATA?

- Requirements:
  - Robustness
  - Handling different characters
  - Identifying errors
- Types: User commands, storage data, configuration
- Alternatives: string
   manipulation, Lex/Yacc (or similar tools)

Our pick: String Manipulation

Why?

- Manually done
- Not easily scalable

- Simplicity
- No learning curve
- Close control of functionality

## THE IMPLEMENTATION OF PARSING

- strtok() and sscanf() are C string library functions
- Each string token is parsed using variations of sscanf()

Value to read from user

name John Smith, phone 6472930375, vehicles 1



Run strtok() with ','

name John Smith phone 6472930375 vehicles 1



Individual strings obtained from running strtok() name John Smith

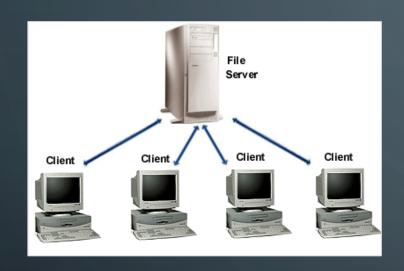
phone 6472930375

vehicles 1



Run a form of sscanf() to parse string token For the database: name = "John Smith", phone = 6472930375, vehicles = 1

## ADDITIONAL SERVER FEATURES Request



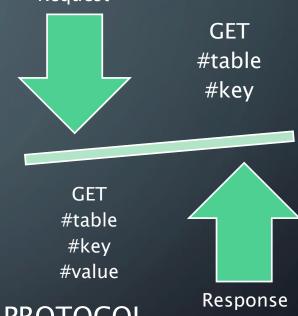
#### **CONCURRENCY**

Server capable of handling up to 10 clients simultaneously



#### LOGGING

Thread-safe log files created on user's computer



**PROTOCOL** 

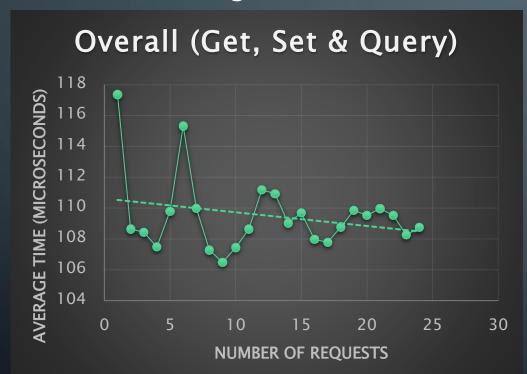
Minimize data needed to be transmitted between client library and server

## USE CASE SCENARIO: MANAGEMENT OFFICE UPDATING A TENANT RECORD

⊗ ⊜ □ ECE297 Storage Server Demo	
jainarya@Aryamman-Dell:~/src\$ ./client	
<ul><li>1) Connect</li><li>2) Authenticate</li></ul>	
3) Get	
4) Set	
5) Query 6) Disconnect	
7) Exit	
Please enter your selection: _	

## END-TO-END EVALUATION RESULTS

Single Client



**Multiple Clients** 



## CHAOS WAS THE LAW OF NATURE; ORDER WAS THE DREAM OF MAN

Henry Adams

## OTHER PERFORMANCE RESULTS

#### **Test Cases**

- Constructed using check framework
- Tested Get, Set and Query individually

**Transaction**: A pair of Get and Set requests to update a record

