Bank Reconciliation Statement Validation Using Quadratically Probed Hashing

A desktop application tailored for account clerks in a companies which use BRS to maintain their policy books of policy holders with bank transactions.

03 April 2020

Contents

BRSV

Abstract

What is this software?

What does it solve?

BRS Validation Using Quadratically Probed Hashing (or *BRSV*), themes the enterprise software designed by the author for insurance companies which do not have an automated solution for the BRS Validation, except for doing it manually.

- The bank might not encash those cheques submitted as premiums directly, but might do those in bundles.
- This is a problem for which the banks prepare a BRS in Excel and hand it over to the employees of the insurance companies where they **spend first week of every month** to match more **170k+ records** between **4 different spreadsheets** manually deleting them when a match is found and **parallel calculations**.
- This tedious task was semi-automated by **BRSV** which implements **Quadratic Probed Hashed** searching, being load-factor 0.8 behind **ReactJS-Flask Client-Server** application setup.

BRSV

BRSV

Introduction of Bank Reconciliation Statement Validation System

How does BRS validation procedure occurs ?

How does the software help ?

Introduction to Bank Reconciliation Statement Validation System

- Let's start by explaining what BRS or Bank Reconciliation Statement is. The BRS sheets in its core looks somewhat like the one on your left (left) and CPNC sheet (The highlighted tabs are the 4 other sheets similar to this one, all totalling to five) (right).

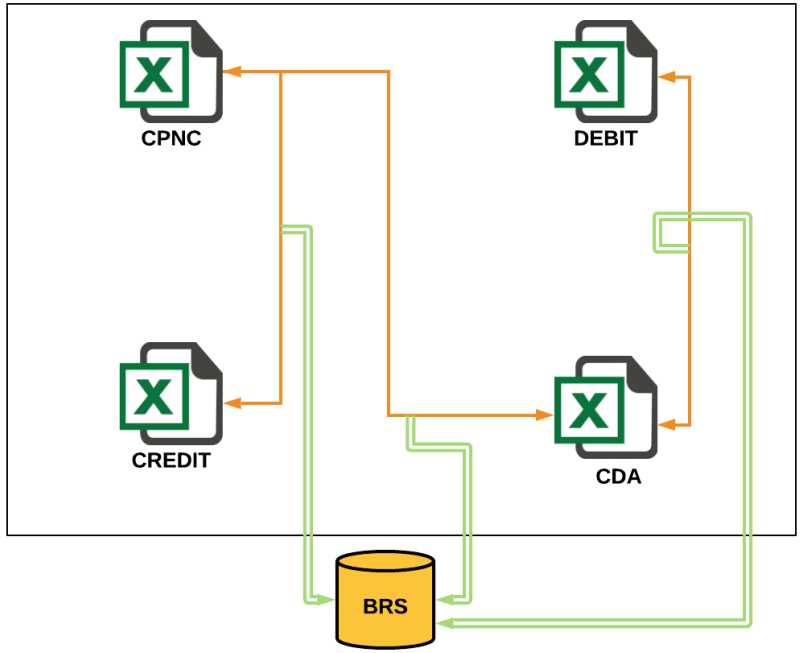
5	BALANCE AS PER PL						14794342.30	DR
6								
7	ADD ITEMS :-							
8								
9		1. CDA				885789.00		
10		2.EX CR (LIST ENCLOSED)				3009388.01		
11								
12		4.SORT REMITTANCE				70.00		
13		TOTAL				3895247.01	18689589.31	DR
14								
15								
16	LESS ITEMS :-							
17		1. CD / REJECTED CURRENT MONTH						
18								
19		2. CHEQUES PAID IN BUT NOT CREDITED				26446204.24		
20		3.EXCESS DEBITS (LIST ENCLOSED)				1602826.48		
21		LESS CREDIT BY BANK				5943.86		
22						49151.90		
23		TOTAL				28104126.48		
24								
25		BALANCE AS PER BRS					-9414537.17	DR
26		BALANCE AS PER BANK STATEMENT					7231248.24	CR
27								
28		DIFF					-16645785.41	
29							16645784.41	
30							-1.00	
31								
32						AO (Cash & Acs)		

	A	B	C	D
1	CPNC AS ON -02-2017			
2	TRAN DATE	Pay in slip	CHQ NO	amount
3			Total	26446204.24
4				
5	15-03-2008		72647	4197.20
6	09-04-2008		3363593	796.00
7	16-04-2008		86529	3757.00
8	08-05-2008		377269	9191.00
9	31-05-2008		107176	886.00
10	31-05-2008		216429	963.50
11	31-05-2008		137053	2826.00
12	31-05-2008		704493	2838.00
13	03-02-2009		764959	448.40
14	06-02-2009		939235	5419.00
15	09-02-2009		45177	110.00
16	09-02-2009		175782	409.00
17	09-02-2009		13356	499.00
18	09-02-2009		13795	499.00
19	09-02-2009		512314	557.00
20	09-02-2009		37463000	776.00
21	09-02-2009		135973	2255.00
22	09-02-2009		428644	2591.00

cpnc | Dr | Cr | BRS | BS | PIS | CDA | Sheet2 | Sheet1

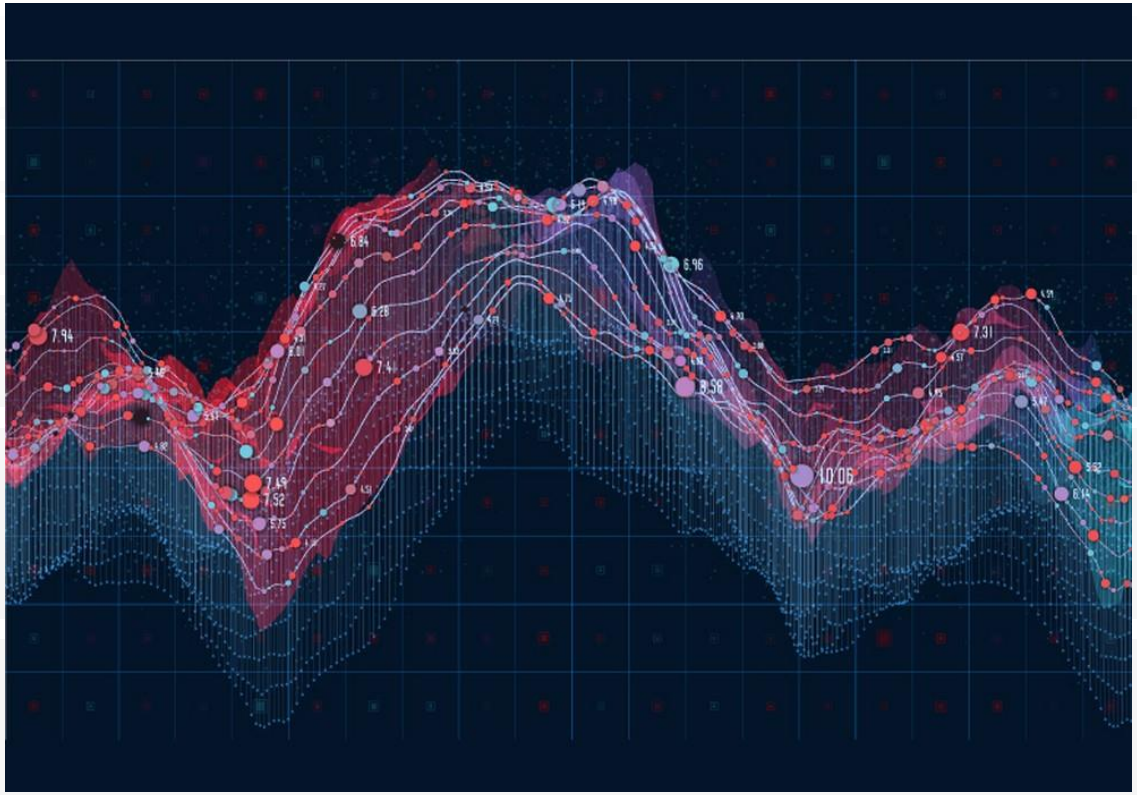
Introduction to Bank Reconciliation Statement Validation System

- The match flow of the BRS Validation procedure looks somewhat like the one on your (right).
- As it is visible from the diagram, there are four Excel sheets in BRS –
  i. **Cheques Paid but Not Credited (CPNC)**
  ii. **Credits**
  iii. **Debits**
  iv. **Cheques Dishonour Action (CDA)**



Introduction to Bank Reconciliation Statement Validation System

- The software displays its frontend and makes up data frames by inputting the various columns of the excel sheet.
- These data-frames is sent to the Flask api-server for processing. After the processing of the data-frames occurs, fresh Excel sheets are created under the names of the respective sheets and spatial vector graphs and reports are generated from the response (right).



BRSV

Hashing and Quadratic Probing

Why was it chosen ?

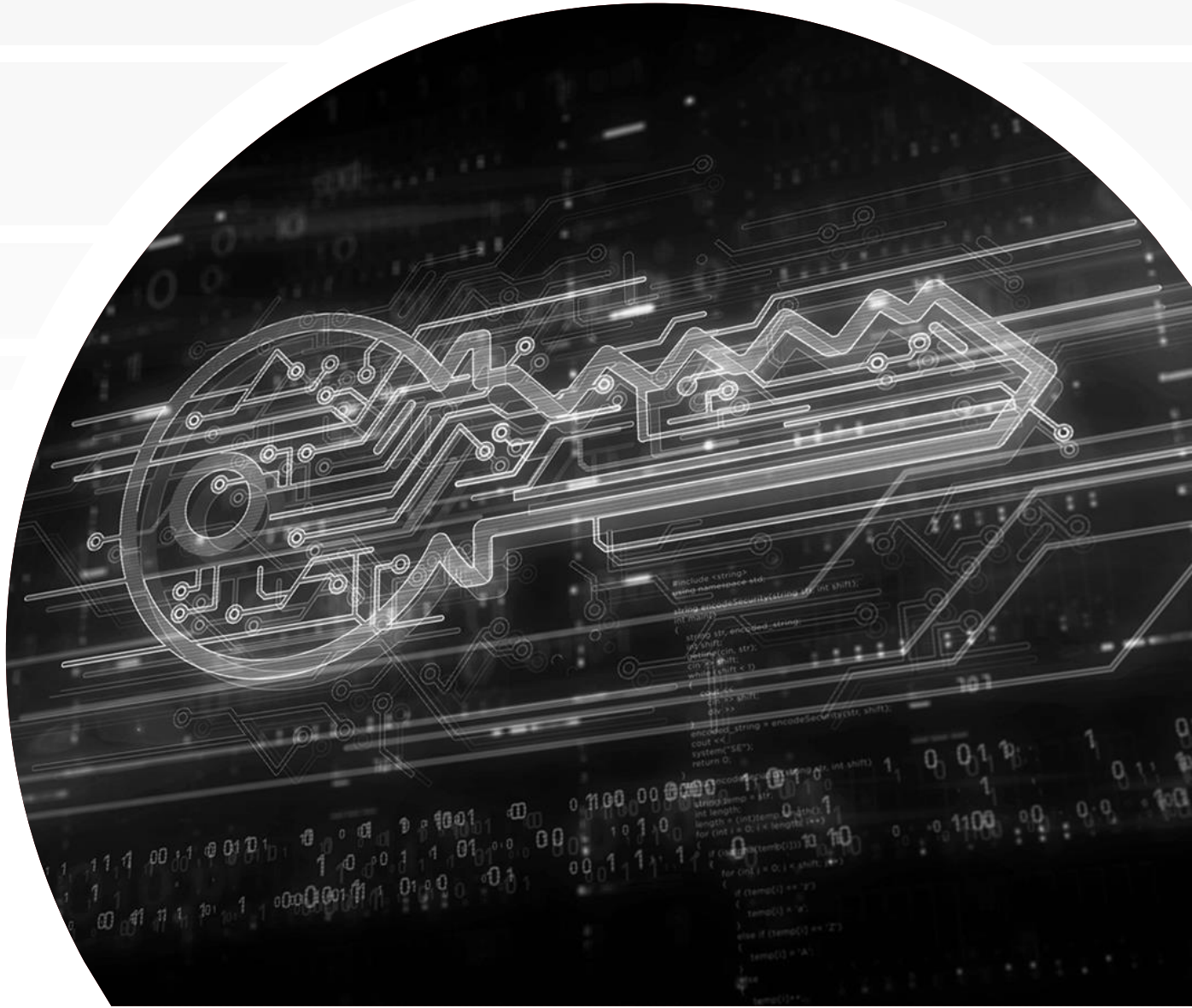
BRSV

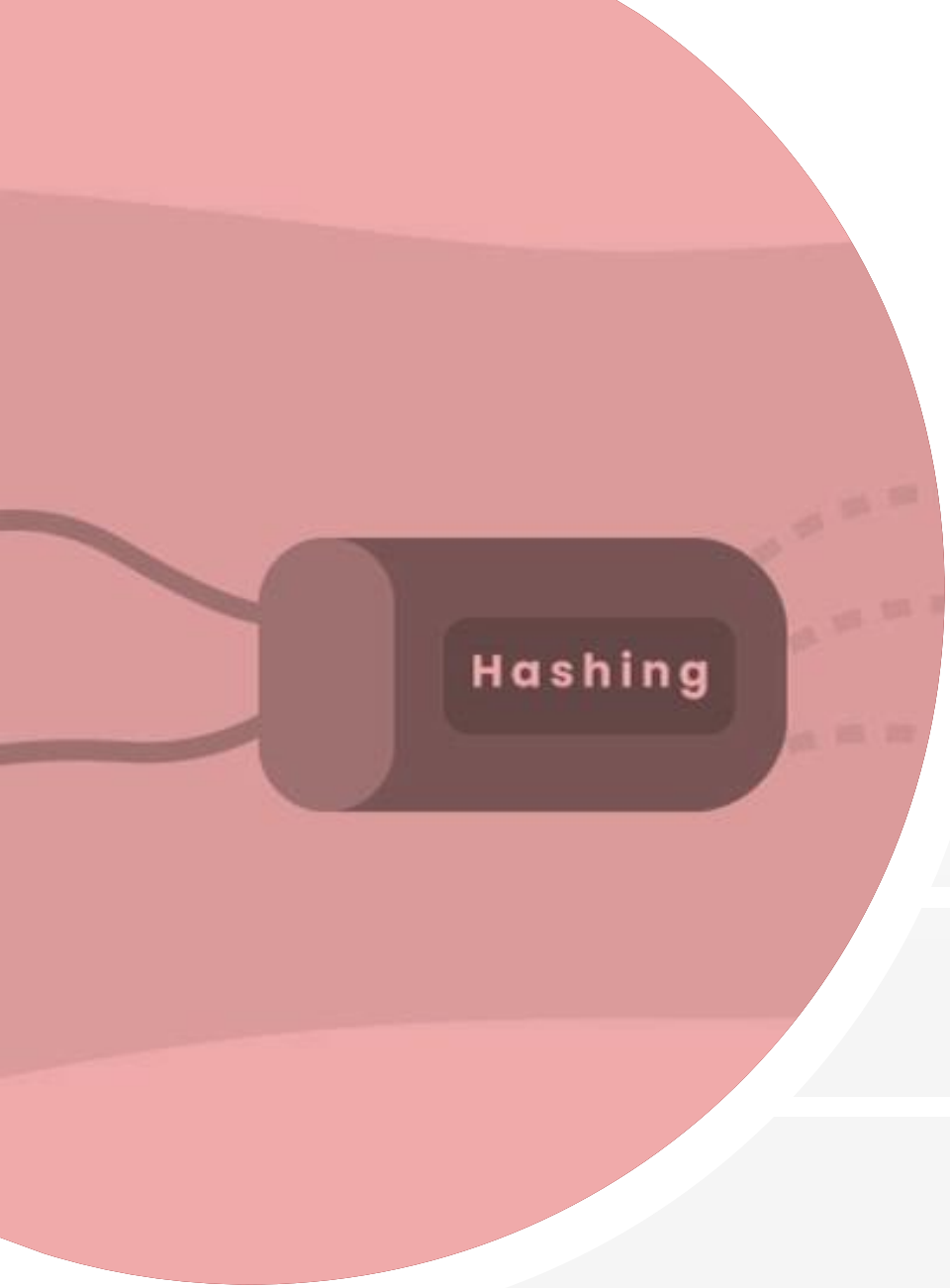
Why Hashing ?

The key idea behind building BRSV is fastening the access of records and making searching as fast as possible.

Key Possibilities with their limitations:

Possibilities	Downsides on time complexity or table size
Sorted Array Implementation	Insertion and searching takes **O(log(n))**
Linked List Implementation	Insertion and searching takes **O(n)**
Balanced Binary Search Tree	Insertion and searching takes **O(log(n))**
Direct Access Table Implementation	Size of the table which if exceeds **$m*10^n$**
Hashing	None on time complexity or table size





Hashing and Hash Functions

The idea of hashing is to distribute entries (key/value pairs) uniformly across an array. Each element is assigned a key (converted key).

$$hash = hash\_func(key)$$

$$index = hash \% array\_size$$

A hash function is any function that can be used to map a data set of an arbitrary size to a data set of a fixed size, which falls into the hash table. The values returned by a hash function are called hash values, hash codes, hash sums, or simply hashes.

To maintain the performance of a hash table, it is important to manage collisions through various collision resolution techniques.

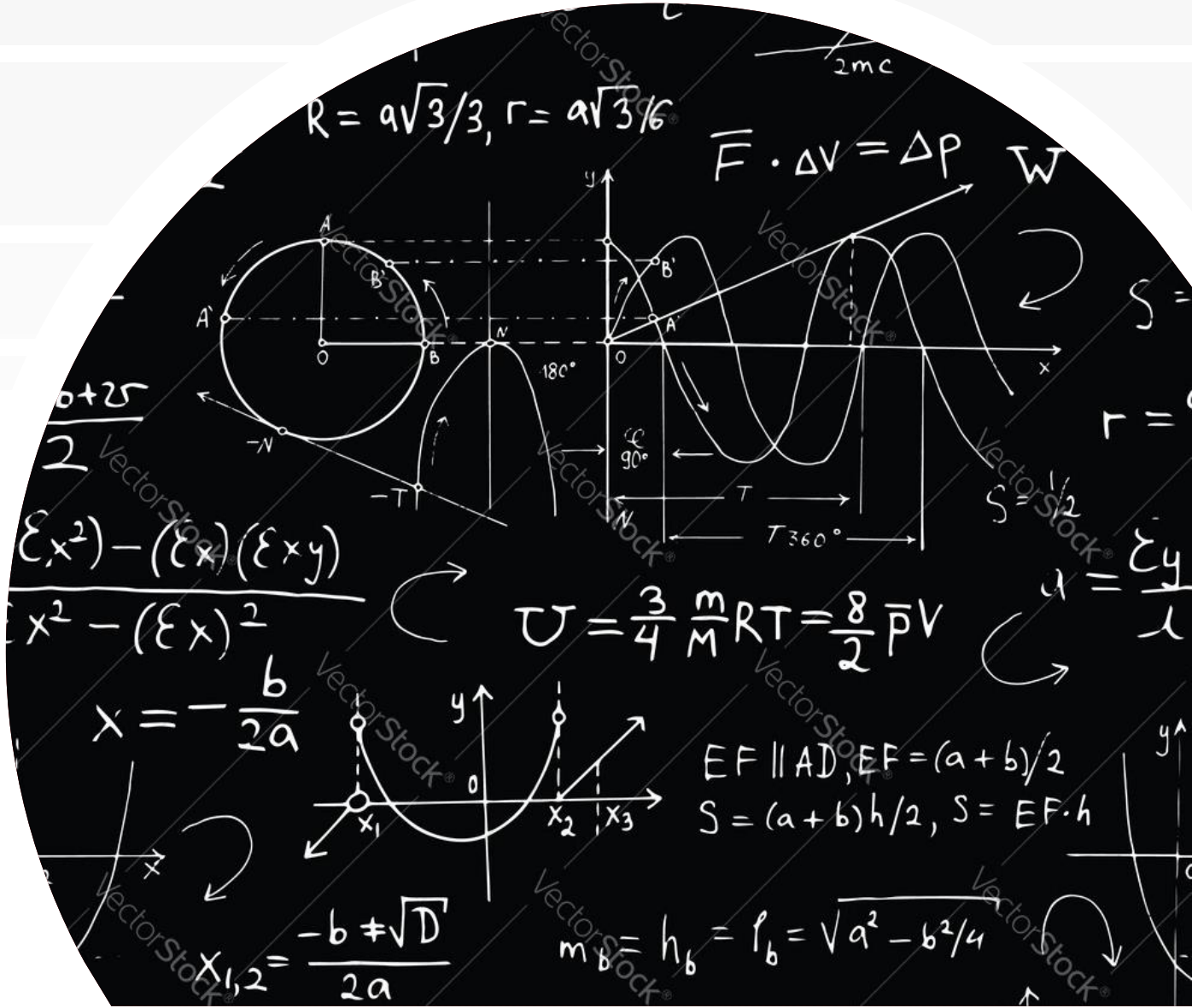
BRSV

BRSV

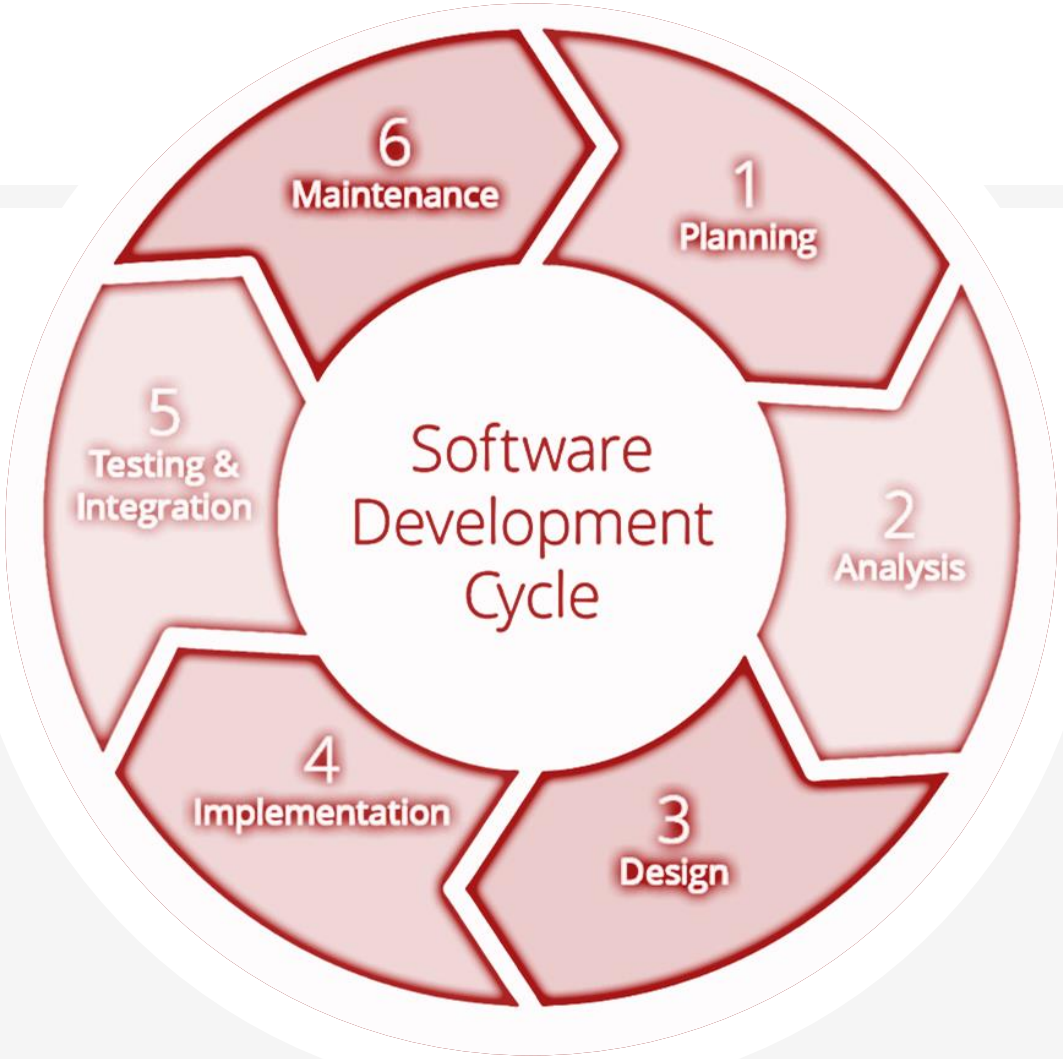
Why Quadratic Probing ?

There are different ways to resolve collisions while making a hash table:

Possibilities	Downsides on time complexity
Separate Chaining	Cache performance is not so good and there is wastage of space. **O(1/1-α)**
Quadratic Probing $h_i = (hash(x)$	None on time complexity or table size



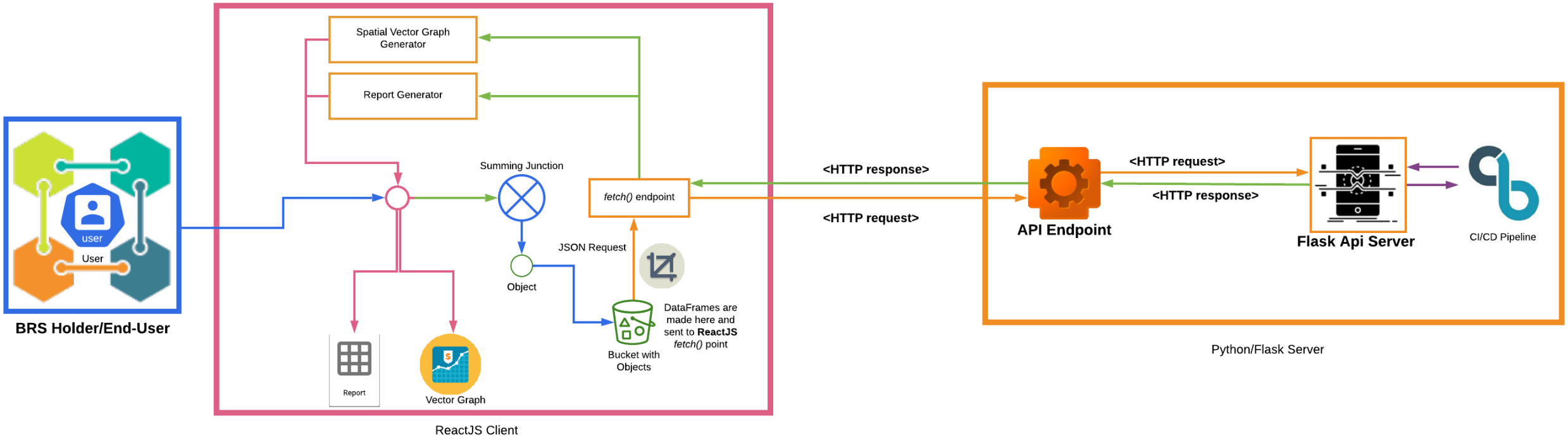
BRSV



Software Workflow

How was it made to work and run smoothly ?

Implementation of Bank Reconciliation Statement Validation System



Conclusion

- This presentation describes how the enterprise software was implemented.
- Also, this presentation covers the evidence supporting how a basic Data Structure Fundamental topic solves a very tedious manual task which a whole bunch of people spending almost an entire week of time into quick and easy job of bare 5-7 minutes(depending on the amount of transactions in data sheets).
- Although this software was deployed on premise, yet there could be further improvements like **Amortized Hashing using Doubled Probing** could be explored for same solution.
- Further, as rolling updates for bugs and enhancements, the server could be made into a **gRPC platform** and similar enhancements.

THANK YOU!

Aditya

1605004@kiit.ac.in