Journal of Multidisciplinary Engineering Science and Technology (JMEST)
ISSN: 3159-0040
Vol. 3 Issue 1, January - 2016

Big Data Solution: Industry Trends

Jayant Dani
TATA Consultancy Services, Mumbai
Jayant.dani@tcs.com

***Abstract*—Welcome In this paper authors will take through the evolution of Big Data solution and its application in various industry keeping the view towards this technology. Especially as technology evolved and proved about itself, adoption has started from social sites to core industry verticals. Firstly the authors will analyze and describe some of model solutions in this technology for various problem area where industry looking forwards. First and foremost is the Data management especially document management which is now growing in leaps and bound. Secondly another largest set of unstructured data in Telecom industry – Call Data Records (CDR) and time energy which can be saved for serving legal request for information. Thirdly authors like to draw attention towards need to Analytics with grown data as new technology architecture for analytics.**

I. INTRODUCTION

'Enterprise Information, is the amalgamation of all the data worth created from disparate systems of an organization with the sole objective of making informed decisions within defined timeframe.'

Today the information heterogeneity is not just confined to four walls of an organization. There is wealth of information generated from social media, peripheral mobile devices and other external entities. Enterprise thus seeks to utilize such information with positive intent to enhance their decisions making mechanisms. Big Data in turn helps an enterprise to quantify the 'Emotional Quotient' [1].

With introduction C++ based distributed file sharing framework for data storage and querying. Structured, semi-structured and/or unstructured data is stored and distributed across multiple servers in 2000 , era of Big Data is really started. It has been really come to play first time with Google publishing its first paper on GFS and Map Reduce frame work in 2004.
Now in 2014, Big data technology has been now getting stabilized and various solution getting developed around it.
Following are the area around which Big data solution is getting developed.

Big Data Problem Area	% project
Customer analysis/segmentation	55%
Historical/archived data analysis	46%
Production systems log monitoring/analysis	39%
Website monitoring/log analysis	36%
IT systems log monitoring/analysis	37%
Competitive/market analysis	36%
Content management (photos, images,Douments)	32%
Social media analysis	25%
Testing/analyzing new products/R&D	
Other	5%
Source: 2013 BIG DATA OPPORTUNITIES SURVEY -SAP	

II. SOLUTION TRENDS AND ARCHITURE EVLUTION

In this section , I will elobrate architecture evolution.

III. RETAIL MARKET BASKET ANALYSIS

One of the leading retail chains from North America approached TCS with an interesting problem statement. The retailer had over 10 TB of transactional data for past 8 years and wanted to identify the patterns in the transactional dataset.
Figure 1 illustrates the mechanics of using Mahout [6] for mining data. System (300) is explained in detail below:

- Data Extraction and Loading:

  Extract transactional data from legacy system using Sqoop and load data into HDFS (302) for further processing and analysis.

- Application Logic:

  Transformation of data is done using Pig-Latin (303) to prepare dataset input for 'Mahout Frequent Pattern Mining Algorithm'. The output of pig job is Mahout readable format, which is a flattened dataset with one line for each transaction. Each line consists of the transaction information in the following format

  CAKE, FROZEN BAKED, PET, Jan

- Data feed is provided to FP algorithm for finding pattern and their frequency (304). Frequency is configurable and set as 100000 based on retail chain's feedback and TCS experience. This is set for every run along with top K patterns. For this run setting of K=10. On the top there is a custom MapReduce [7] module (304) to find Support, Lift and Confidence for the particular pattern which is explained below:

  Frequency: The frequency is defined as the number of times a pair of products is purchased together.

  Support: It is defined as the probability of a pair of products being purchased together (The level of support is how frequently the combination occurs in the market basket (database))

  Confidence: Compares the number of times the pair was purchased to the number of times one of the items in the pair was purchased (which is the percentage of cases in which a consequent appears given that the antecedent has occurred)

  Lift: (Improvement) tells us how much better a rule is at predicting the result than just assuming the result in the first place (Lift is equal to the confidence factor divided by the expected confidence. Lift is a factor by which the likelihood of consequent increases given an antecedent)

- Using Sqoop, export data to RDBMS/Hive for BI reporting (305). Orchestration of process is done using Oozie.

- Reporting is done in Tableau (306) to provide visualization of data for better insight.

As evident from Table 1 for year 2003 top selling products are determined by frequency. Table 2 and Table 3 provide the detail of computing the numbers on the hardware setup specified below. Products that have highest affinity are the ones which are most likely sold together, like 'other cheese' and 'shelf juice'. Table 4 shows that they were sold together 9,156,950 times and their corresponding market basket analysis parameters. Finding affinity using FP (Frequency Pattern) growth algorithm on distributed environment is much faster than other approaches.

Fields	Value
Shelf Juice	27659106
Other Cheese	24786645
Paper Towels	19783102
Cereal	19712965
Alcohol	16190391

Table 1: Year 2003 - Top Selling Product Frequency

Fields	Value
Input Data for Year - 2003(Transaction)	182,693,777
Time Taken to find all analytics (Frequency, Support, Lift and Confidence)	27min 09 sec
Input Data Size	3.4 GB

Table 2: Analysis Statistics

Fields	Value
Data Node configuration	4 node (32 GB RAM, 10 TB Storage and Quad-core processor)
Name Node Configuration	32 GB RAM, 7 TB storage, Quad Core processor

Table 3: Hardware Details

Category Combination	Other Cheese, Shelf Juice	Cereal, Other Cheese	Cereal, Frozen Ice Cream, Other Cheese, Shelf Juice
Frequency	9156950	7029310	1060850
Support	5.01	3.85	0.58
Dominant Category	Other Cheese	Cereal	Frozen Ice Cream
Confidence	36.94	35.65	7.25
Lift	2.44	2.62	3.36

Table 1 : Market Basket Analysis

Calculating frequency of patterns involves combinations of each transaction which causes data to explode and may consume like multiple times of the actual data. TCS has classified the frequency of patterns as SSR [8] (Scalable with respect to System Resources) type. The implementation of which uses LLBD [8] (List, List, Brainstorm and Design) technique. This data along with the processing logic is distributed over nodes to tackle scalability problem.

Each of these techniques works best when provided with a large amount of good input data. In some cases, these techniques must work not only on large amounts of input, but must produce results quickly. These factors quickly make scalability a major issue. Due to availability of a scalable framework like MapReduce it is now possible to find the affinity of N x N items in the basket. The result is discovery of new affinities as the entire dataset is now used for processing instead of using just the subset of information.

IV. VIGILANCE, TRACKING & MONITORING IN TELECOM

As per the government norms all telecom operators have to maintain 2 years of CDR (Call Detail Records) for investigations, vigilance etc. For Pan India on average there are 900 million CDRs per day

sizing up to 150GB/Day. The problem statement is to handle and process this stream of data and come up with meaningful information. TCS customer's current tracking and monitoring system for the CDR maxed out after processing few weeks' data. Tracking system was not able to process more than 1 year's data at a time. After processing, 1 year's data is flushed out from the system. Problems associated with this approach is that data analyzing window is relatively small and user's usage pattern for cell id (Based on a given time range, the user would also like to track all the call made to a given cell tower) and switch id (Based on a given time range, the user would like to track all the call that traversed via the given switch) couldn't be analyzed.

This gave birth to the idea of 'Vigilance Tracking and Monitoring System'. System is built keeping in mind the need for processing voluminous data. Wherein the data is pulled, transformed and loaded from a singular or a plurality of source systems to a big data storage system. Further, a query engine is configured to execute one or more query in real-time for retrieving the data from the target big data storage system. A processor maps the executed query with the data thus stored by generating a key value in a preset format with respect to each query, such that the query results are retrieved by scanning the target big data storage system in accordance with the key value thus formed.

Figure 2 as a block diagram illustrates a VTM system in accordance with one or more embodiments of the technique. VTM system (400) comprises of a user interface (402) configured to provide to one or more user, an access to the big data storage system in a network, a loading engine (404) which is configured to pull the data from one or more source system (406) and push the data in order to populate one or more target big data storage system (408), a query engine (410) configured to execute one or more query in a real-time and a processor (412) to map the executed query with the stored data by generating one or more key values for a particular query. The loading engine (404) pushes the data in batches. The data pushed by the loading engine (404) is transformed and is stored in a master table (414). This master table stores the original data. The system designs the big data storage system (408) in a manner such that it is provided with a query layer, wherein the query engine (410) is used for executing one or more queries.

In general, while querying the big data storage system (408) like HBase, scanning of more than billions of items is to be done which increases the response time of a query. In order to address this issue TCS has developed the 'Mapping the Map' methodology which will reduce the query retrieval time. The processor (412) in communication with the loading engine (404) then processes the data with respect to the query executed by the user for retrieving the results. Referring to Figure 1 and 2 the query engine (410) and the processor (412) along with the generating module (416) is configured to prepare a key value for each query. The master table (500) stores the original data. The further tables are created for the particular type of query (Q1_map_table (502), Q2_map_table (504) etc.). For each query type, the generation module (416) generates a key value (Q1key, Q2 key etc.).
In the method of 'Mapping the Map', when the query is executed, based on the key value, the data from the respective tables (502, 504, etc.) is mapped to the master table (500) for retrieving the results.

For each query, rather than scanning the entire big data storage system (408), the key value (prepared for the particular type of query data) fetches the results from the master table for the executed query in a much lesser time. The key value further comprises a start key and a stop key coupled with a time range. This process of obtaining query results by scanning a particular portion of big data storage system (408) by using the related key value is claimed to be 'Mapping the Map' methodology.
Since the system (400) is further horizontally scalable (because of the transformation thus performed), it implies that the storage will not be a constraint which in turn makes the system (400) more effective in analyzing the data.

Here, the source system comprises of a CDR system and the target big data storage system comprises an HBase.
There may be below listed scenarios where one or more user would like to track the CDR's:
Based on a given time range, a user would like to track all the incoming and outgoing calls made from a given phone number. It can also include but is not limited to a list of phone number

Based on a given time range, a user would like to track the CDRs for a given IMEI (International Mobile Station Equipment Identity) number. It can also include but is not limited to a list of IMEI numbers

Based on a given time range, the user would also like to track all the call made to a given cell tower. It can also include but is not limited to a list of cell tower identification numbers.

Based on a given time range and switch ID, the user would like to track all the call that traversed via the given switch. It can also include but is not limited to a list of switch ID.

V. 'MAPPING THE MAP' METHODOLOGY

Figure 4 illustrates the execution of 'Mapping the Map' methodology, as per the above listed query scenarios. For each query executed by the query engine (410) the data from CDR is stored in the corresponding master table like, switch_map_table (604), imei_map_table (606), cell_map_table (608)

and ph_map_table (610). The data is processed by the processor (412). All these tables store the related key value which is generated by the generation module (416). For switch_map_table (604), the key value is a combination of switch ID, call date and time. For imei_map_table (606), the key value is a combination of IMEI, call date and time. For cell_map_table (608), the key value is a combination of first cell ID, call date and time or last cell ID, call date and time. For ph_map_table (610), the key value is a combination of calling number, call date and time or called number, call date and time. Based on these query types, when the query is executed by the user, the key value from the corresponding table is mapped with the master table (602) rather than scanning the entire target big data storage system (408) for retrieving the results.

The system (400) is quick in key based retrieval. The system (400) can quickly jump on these key ranges and scan for retrieving for the (408) query thus executed. The data for a key value is fetched from the master table stored in the big data storage system.

VI. CONCLUSION

Big data solution will be get more and more towards various industry main track business process.

Ha doop is a very promising distributed platform for performing analytics at low cost which has been demonstrated in above case studies.

TCS has shown how Retail affinity analysis on large structured datasets can be done efficiently using distributed framework. The results of which are shown in section 3.0.

Unstructured data processing can be done with ease as proved in VTM application used for Telecom CDR Analytics in section 4.0.

Case studies mentioned in this paper (Retail and Telecom domain) along with other thousands of use cases ranging from social analytics to genome mapping is just the tip of the iceberg.

VII. ABOUT AUTHORS

Jayant Dani is Masters in Engineering, BITS Pliani having 20 years of industry experience. Currently heads ‘Technology Center of Excellence’ for TCS product group. He was a main contributor in IEEE Indicon 2012. He is also life member of ISTE and CSI. Currently his research area is application of Big Data in various industry domains.

VIII. REFERENCES

1) Dr. Santosh Mohanty, Jayant Dani, Srikar C, “A Tutorial on Big Data Management”, IEEE INDICON, 2011.
2) Beyer, Mark. "Gartner Says Solving 'Big Data' Challenge Involves More Than Just Managing Volumes of Data", Gartner, Jul. 2011.
3) Lustig, Irv, Dietric, Brenda, Johnson, Christer, and Dziekan, Christopher, "The Analytics Journey", www.analyticsmagazine.com, Analytics, Nov. / Dec. 2010.
4) Tyler, M. E. and Ledford, J. L. “Google Analytics” Third Edition, Wiley Publishing, Inc., 2010.
5) Julia Layton, “How Amazon Works”, HowStuffWorks.com, Jan. 2006.
6) Sean Owen, Robin Anil, Ted Dunning, Ellen Friedman, “Mahout in Action”, Manning Publications, 2012.
7) Chuck Lam, “Hadoop in Action”, Manning Publications, 2011.
8) Jayant Dani, “Designing Scalable Applications”, TCS TACTiCS Publication, 2005.

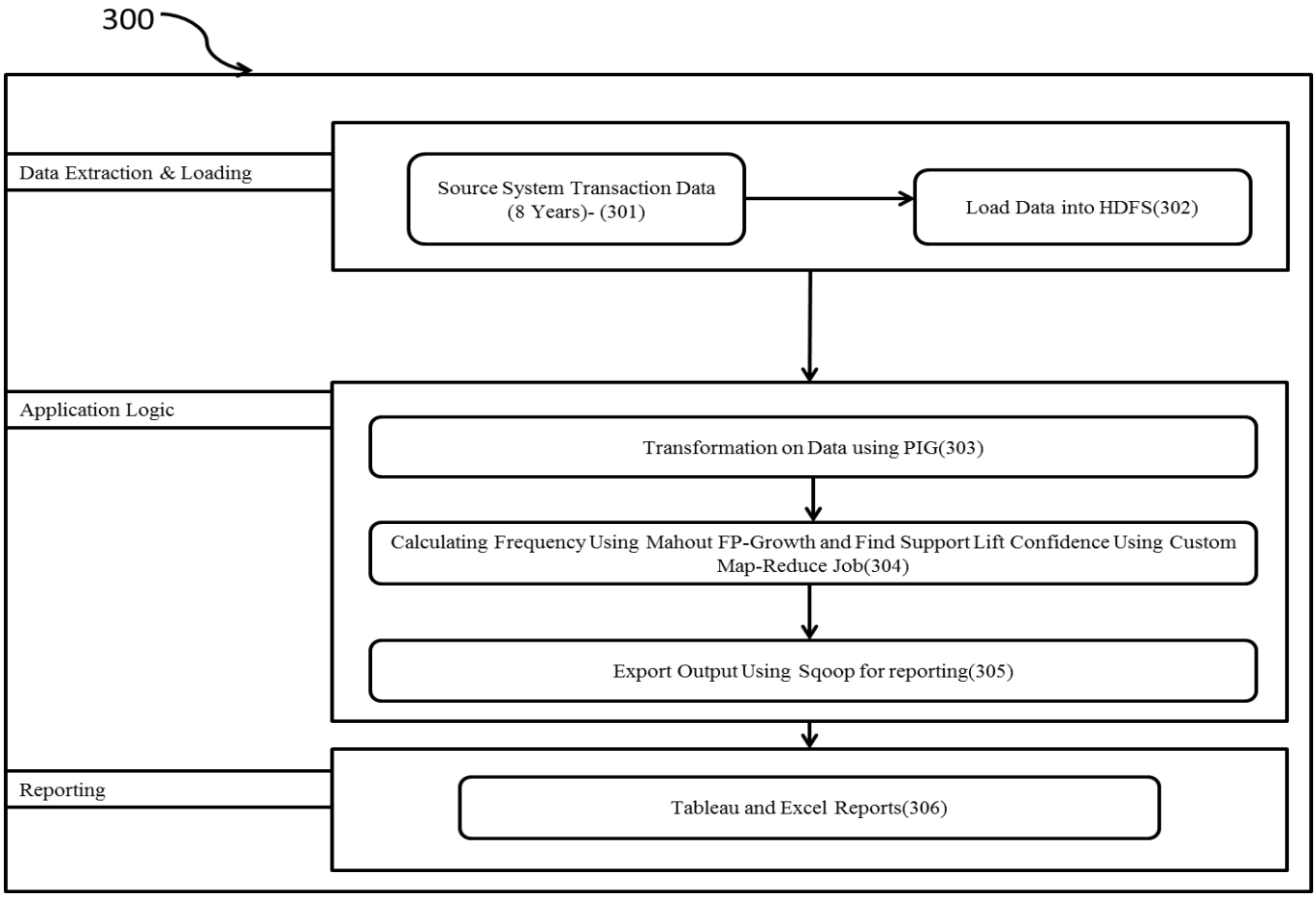

Figure 1 : Mechanics of using Mahout for Mining Data

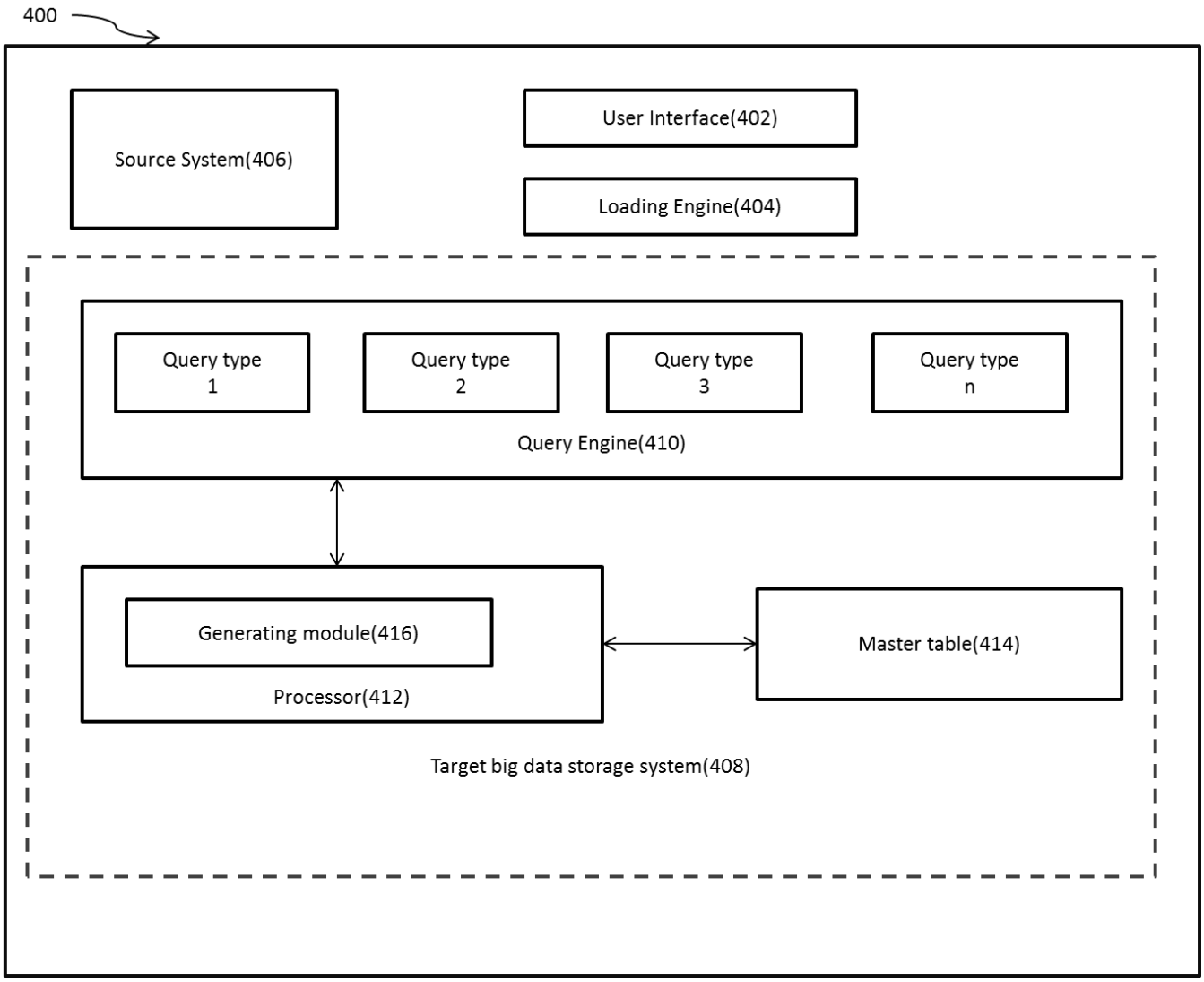

Figure 2 : VTM System Block Diagram

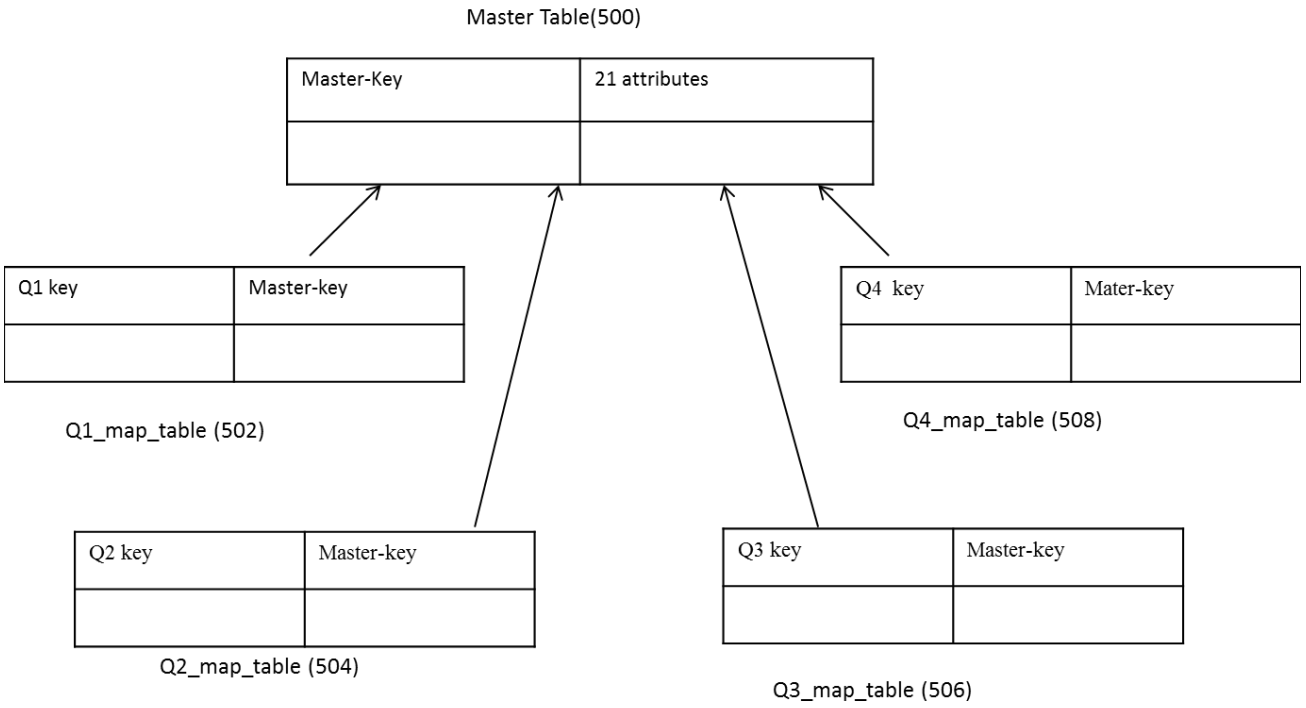

Figure 3 : Master References

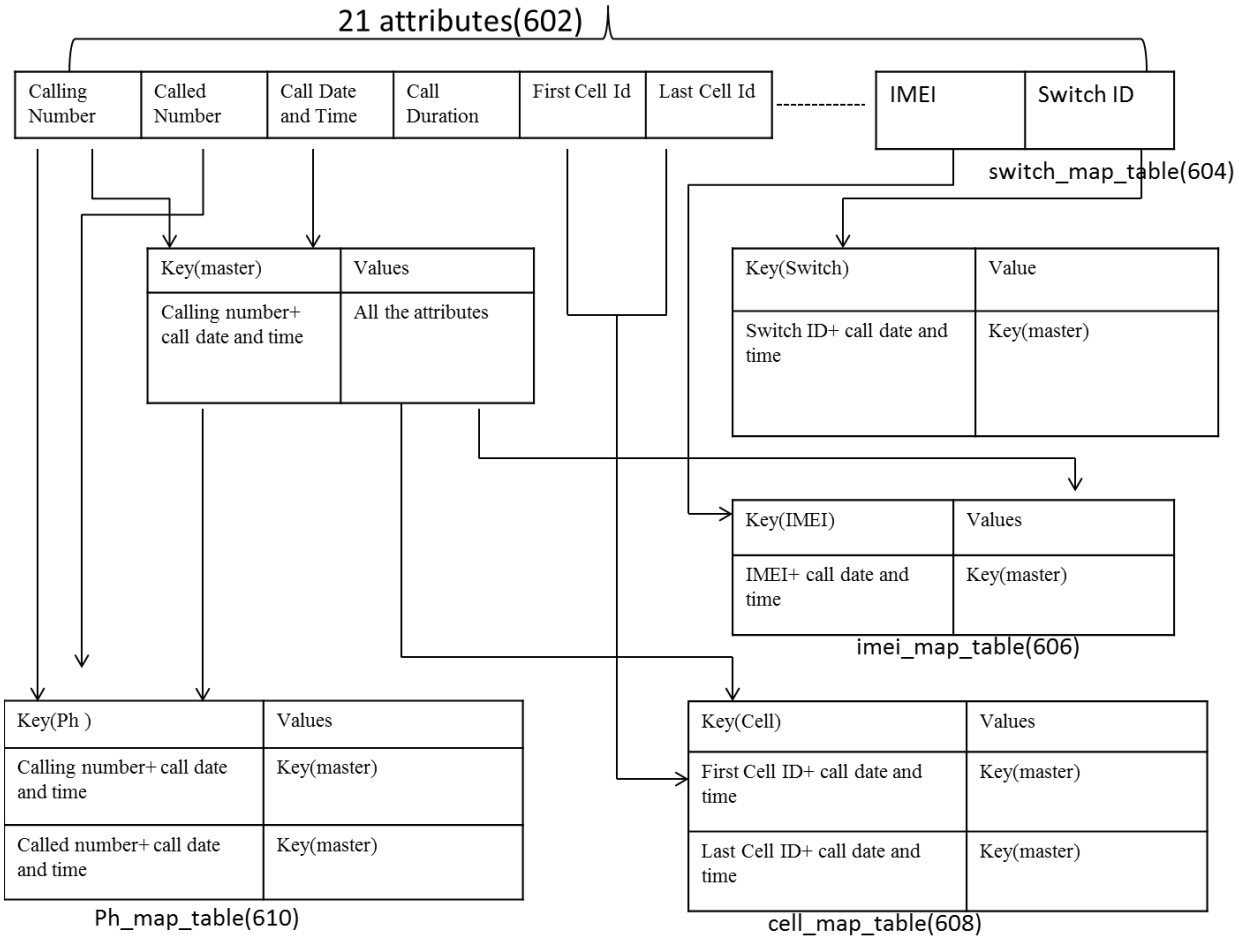

Figure 4 : ‘Mapping the Map’ Methodology