

# Role of Candidate Key in Metadata for Data Analysis

Padma Lochan Pradhan, Nalla Malla Reddy Engineering College, India\*

 <https://orcid.org/0000-0003-2284-6819>

## ABSTRACT

This proposed research paper focuses on the candidate key to generate a frequent pattern from the large dataset. The functional dependency and candidate key rolling are major activities for creating and collecting the exact pattern to support the decision support system. The functional dependency helps a decision support system to assemble metadata to resolve the uncertainty, unstructured, and unordered data. The large data sets (BigData) can be reassembled through Java programming faster and better by applying combinational algebra. The candidate key is directly proportional to the set of metadata. The large data sets are lively connected to the customer, manufacturer, vendor, order, and product key correctly at the right time. Therefore, the computational mechanism has to develop through the candidate key for better and faster data analysis. The knowledge and decision pattern can be acquired through mapping and integration of candidate key management.

## KEYWORDS

Candidate Key, Decision Support System, Frequent Pattern, Management Information System, Metadata, Online Transactions Processing System (OLTP), Primary Key, Relation Function Operation, Services

## INTRODUCTION

The data set is logically interconnected and satisfies several users simultaneously over a network. The database is well organized and created, stored, processed, generated, and analysis of data for a better information system that is stored and accessed electronically anywhere at any time. Database designers and developers typically organize the data to model to maintain real-life applications as per business requirements. The RDBMS defines database relationship on the form of tables, also known as relations, association, and dependency. Relational DBMS usually have pre-defined data types that can support metadata. This is a most popular RDBMS type in the current market. The examples of relational database management systems include Sybase, MySQL, Oracle, and Microsoft SQL Server Database. The Object-Oriented Relation DBMS, this type supports the storage of new data types. The data to be stored is in form of objects that make metadata. The objects to be stored in the database have many more attributes (i.e. male, female, gender, age) and methods that define what to do with the metadata (Abraham, 2013; Elmasri, 2007). Postgres-SQL is an example of an object-oriented relational DBMS.

DOI: 10.4018/IJBDA.318411

\*Corresponding Author

This article published as an Open Access article distributed under the terms of the Creative Commons Attribution License (<http://creativecommons.org/licenses/by/4.0/>) which permits unrestricted use, distribution, and production in any medium, provided the author of the original work and original publication source are properly credited.

The metadata and its importance in a data-driven world is going to be a great future of data science. In most information technology applications and usages, the prefix meta are used as a primary keys, which is described as the details of identifications and classification of the particular tuple. The understanding of the data structure, limitations, definition, and description of data providers protects against misinterpretation support ensures database integrity(Alex, 2013; Gary 2009; Jiawei, 2011). The metadata is described by AIIM as the structured data and information about a document, Examples of metadata are writer, title, subject keywords, date of publication, classification, access number, rights information, and so on.

Metadata is the data about data. the set of data items makes a record or tuple. The set of tuples makes a table. The table consists of several rows and columns systematically. The table contains the primary key, foreign key, candidate key, alternate key, non-key attributes, and composite key. Each table has keys and candidate key attributes, which help identifications and classifications of data in the object orientation system(Alex, 2013; Gary 2009; Jiawei, 2011). The table that provides information about other tables is called metadata. Many distinct types of tables make a group of information that makes a classification. One table is associated with another table through the primary & foreign keys. One or more tables are associated with others, which makes metadata. There are five types of metadata available today around the globe. The following types of metadata are identified and classified as descriptive metadata, structural metadata, administrative metadata, reference metadata, and statistical metadata (Alex, 2013; Gary 2009; Jiawei, 2011).

The metadata is generated in frequent patterns and collected because it enables and improves the use of that data or information. Good metadata can make up for human fallibilities. The metadata also makes text documents easier to find because it explains exactly what the document is about to relate to others.

The metadata is data that describes other data. The meta is a prefix that in most information systems usages the primary and foreign key, that terminology helps to decision support system. The metadata summarizes basic information about data, which can build, search and work with particular instances or objects of data easier(Alex, 2013; Gary 2009; Jiawei, 2011).

## **RELATED THEORETICAL WORK**

The metadata can be easily defined as data about data, the key about key, and information about information. This terminology of metadata which is used to represent other data set is known as metadata(Alex, 2013; Gary 2009; Jiawei, 2011). The metadata in data mining defines from the warehouse objects (OLAP). The metadata acts as a directory. This directory helps the decision support system to locate the contents of a data mining.

The keys are the attributes of a particular table and the set of attributes are used to access tuples from the table, they are also used to build a relationship between the tables. In this research paper, we are going to discuss the primary candidate key that difference between them is a great demand for information management on both primary and candidate keys uniquely identify a tuple in relation or table(Abraham, 2013; Elmasri, 2007). but the most important point that differentiates them is that there can be only one primary key in a relation. however, there can be more than one candidate key non-key attribute in a relation.

The above mentioned keys are the essential elements of any relational database. These keys are identified and classified each tuple in a relation uniquely. The keys are also used to generate and establish the relationship among the tables in a schema. In this article, we have to focus, take care and emphasize these two basic keys of any relational database that is super key and candidate key(non-keys attributes). Every candidate key is a superkey, but every super key may or may not be a candidate key(Abraham, 2013; Elmasri, 2007). There are many other distinguishing factors between the super key and candidate key, which we have briefly highlighted in Figures 1 & 2. The candidate's key is the subset of the primary as well as the super key. Therefore we have to prove key about key and build the link about various tables for a better referential key. The candidate key is the subset of the Super key.

Figure 1. Primary Key and Candidate Key

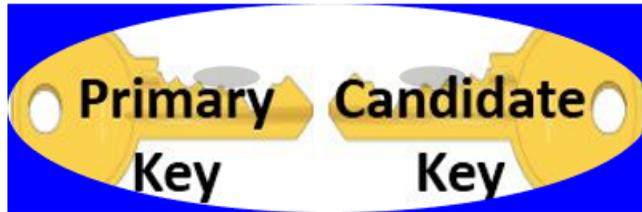
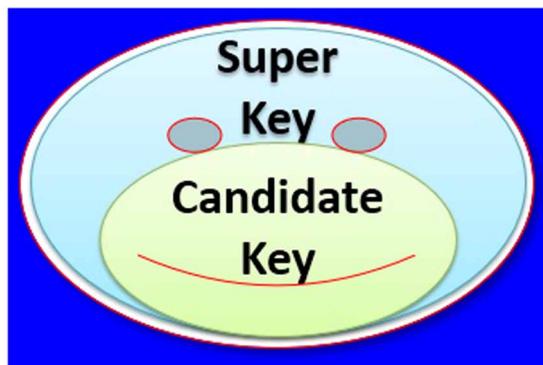


Figure 2. Super Key and Candidate Key



As we know candidate key is a unique key that can be used as a primary key. but not necessarily used as one. A composite key is a key of two or more attributes that uniquely identify the tuple. A key is a set of columns that can be used to uniquely identify each row within a table. A star schema is designed to enforce the referential integrity of loaded data. Referential integrity is enforced by the use of primary and foreign keys. Primary keys in dimension tables become foreign keys in fact tables to link each record across dimension and fact tables(Abraham, 2013; Elmasri, 2007)

## RELATED PRACTICAL WORK (FUNCTIONAL DEPENDENCY)

Functional dependency (FD) is a set of constraints between two attributes in a relation. Functional dependency says that if two tuples have the same values for attributes  $A_1, A_2, \dots, A_n$ , then those two tuples must have to have the same values for attributes  $B_1, B_2, \dots, B_n$ .

The functional dependency is represented by an arrow sign ( $\rightarrow$ ) that is,  $X \rightarrow Y$ , where  $X$  functionally determines  $Y$ . The (LHS) left-hand side attributes to determine the values of attributes on the (RHS) right-hand side(Bernard, 2007; Mohapatra, 2008; Seymour, 2010).

Transitivity rule – Same as transitive rule in algebra, if  $a \rightarrow b$  holds and  $b \rightarrow c$  holds, then  $a \rightarrow c$  also holds.  $a \rightarrow b$  is called as a functionally that determines  $b$ (Bernard, 2007; Mohapatra, 2008; Seymour, 2010).

- **Union:** If  $X \rightarrow Y$  and  $X \rightarrow Z$ , then  $X \rightarrow YZ$
- **Transitivity:** If  $X \rightarrow Y$  and  $Y \rightarrow Z$ , then  $X \rightarrow Z$ . e.g.; Let  $X$  represents  $\{C-ID\}$ ,  $Y$  represents  $\{E-CITY\}$  and  $Z$  represents  $\{C-STATE\}$ . As  $\{C-ID\} \rightarrow \{C-CITY\}$  and  $\{C-CITY\} \rightarrow \{C-STATE\}$  is true for the relation, so  $\{C-ID\} \rightarrow \{C-STATE\}$  will also be true.

- **Transitivity:** If  $P \rightarrow C$  and  $C \rightarrow F$ , then  $P \rightarrow F$ . e.g.; Let P represents {C-ID}, C represents {C-CITY} and F represents {C-STATE}. As {C-ID}  $\rightarrow$  {C-CITY} and {C-CITY}  $\rightarrow$  {C-STATE} is true for the relation, so {C-ID}  $\rightarrow$  {C-STATE} will also be true. (Ref Table 1)
- **Note:** P- Primary Key, C- Candidate key, F- Foreign Key.

CUSTOMER relation given in Table 1:

- FD C-ID  $\rightarrow$  C-NAME holds because for each E-ID, there is a unique value of C-NAME.
- FD C-ID  $\rightarrow$  C-CITY and C-CITY  $\rightarrow$  C-STATE also holds.
- FD C-NAME  $\rightarrow$  C-ID does not hold because C-NAME 'John' is not uniquely determining C-ID. There are 2 C-IDs corresponding to John (C001 and C003).

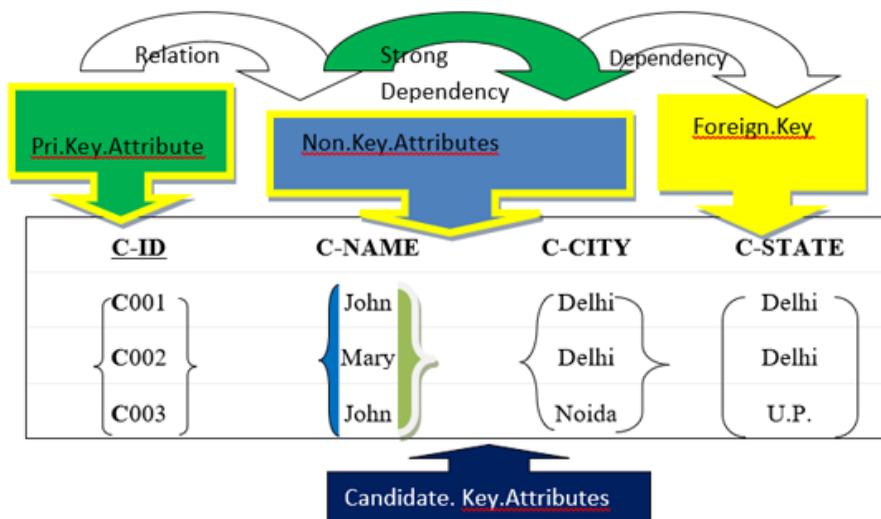
The FD set for CUSTOMER relation given in Table 1 are:

- **Candidate Key:** The candidate Key is a **minimal set of attributes** of a relation which can be used to identify a tuple uniquely. For Example, each tuple of **CUSTOMER** relation given in Table 1 can be uniquely identified by **C-ID** and it is minimal as well. So it will be the candidate key of that relation. This candidate key may or may not be a primary key.
- **Super Key:** The super Key is **set of attributes** of a relation which can be used to identify a tuple uniquely. For example, each tuple of CUSTOMER Relation given in Table 1 can be uniquely identified by **C-ID or (C-ID, C-NAME) or (C-ID, C-CITY) or (C-ID, C-STATE) or (C\_ID, C-NAME, C-STATE)** etc. So all of these are super keys of **CUSTOMER** relation. This candidate key is always a super key but vice versa is not true.

## RELATED LITERATURE SURVEY

Every organization grows with the demand for business, technology, and resources. Now a day, Information and services are a top priority for each and every organization. Data analysis is the most

Table 1. Customer Table



important and helpful scenario for any organization. The management information system and decision support is a set of processes, procedures, policy, personnel, business, and technology applied to the characteristics of assets (technology). The necessity of an information system in any organization has increased because of the changes in logic, structure, programming, and type of technology applied to services that create data. Finally, the business increases along with technology, which creates metadata and spread over the business, technology, and resources.

**History About the Development of DBMS**

Edgar F. Codd, Edgar Frank “Ted” Codd (19 August 1921 – 18 April 2003) was an American computer scientist who, while working for IBM, invented the relational model for database management as well as, the theoretical RDBMS (table 2).

- **Primary Key:** The primary is a column or set of columns in a table that uniquely identifies tuples (rows) in that table (3).
- **Foreign Key:** The foreign keys are the columns of a table that points to the primary key of another table. They act as a cross-referential integrity. Therefore, key about key is a meta key which is called metadata and meta information.

The keys are the attribute, or a set of attributes that are used to access tuples from a table or they are also used to construct a relationship between two tables (table 4). In this article, we are going to discuss Primary and Candidate Key and the differences between them. Both Primary and Candidate Key uniquely identifies a tuple in a relation or table. But, the most important point that differentiates them is that there can be only one primary key in a relation. However, there can be more than one candidate key (non-key attributes) in a relation (table 5).

**PROBLEM STATEMENTS**

We can conclude that from the existing literature survey and analysis, there are many more problems that are reflected and unsolvable now. Therefore, the MIS, OLTP, OLAP, and current decision support system are problematic with the following points. The OLTP and MIS data quality are poor, uncertain, disintegrated, unstructured, unclassification, and unordered, and the static system is not uniquely identified and classified as per top management’s desired manner:

Table 2. History of DBMS Development

Year	Developed the concept	Remarks
1960	CODASYL	Data System R
1970	E.F Cood’s	Database
1976	Rdbms-ER Model	DB
1978	SQL/Oracle	SQL
1980	Wayne	Dbase
1980	Object Oriented	XML DB
1990	OLTP & Clients Server	OLTP & Clients Server
2000	Metadata	MIT USA
2000	XML DB	USA

Table 3. Multi-Key Relationship

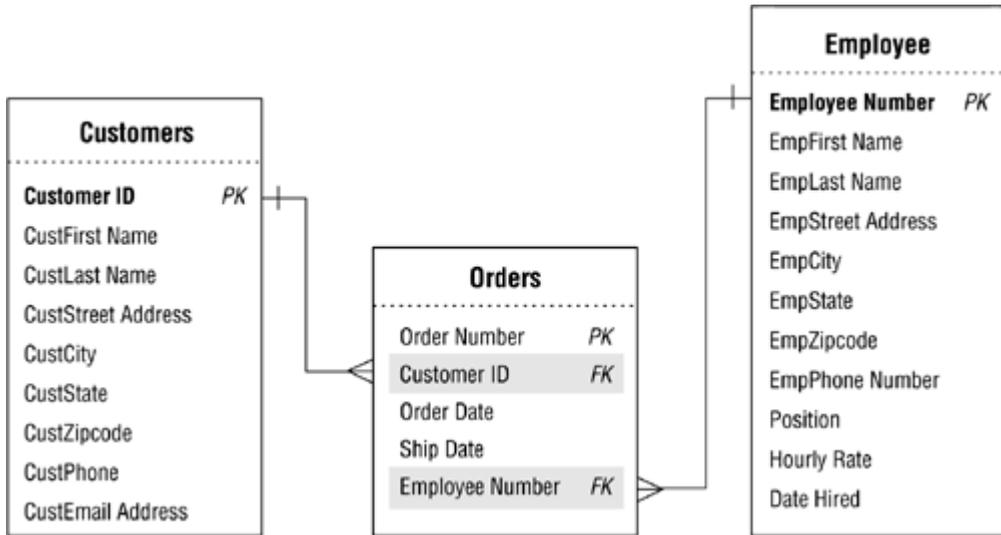


Table 4. Multi key Relationship on DDL

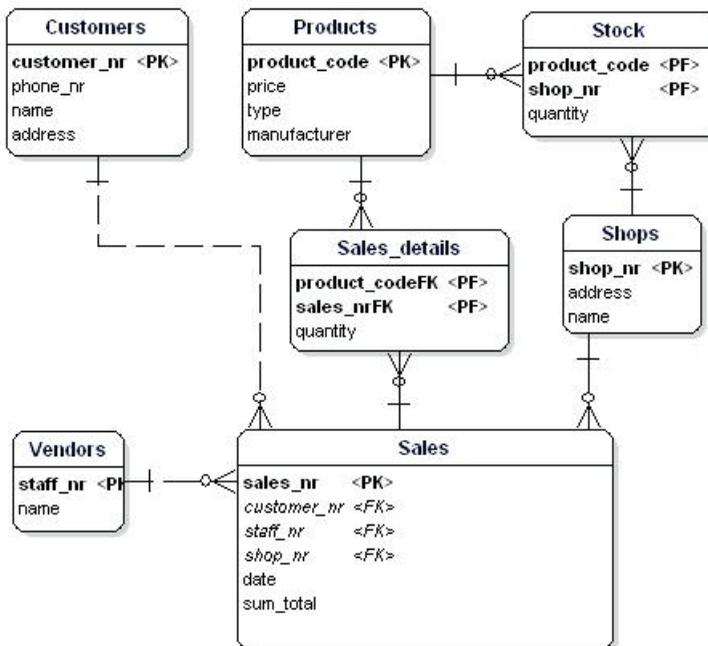


Table 5. Cross Relation of Table for Pattern Matching

The diagram shows a table with 5 columns and 5 rows. The columns are labeled 'attributes' and 'column'. The rows are labeled 'tuple'. The entire table is labeled 'table (relation)'. The table data is as follows:

SID	SName	SAge	SClass	SSection
1101	Alex	14	9	A
1102	Maria	15	9	A
1103	Maya	14	10	B
1104	Bob	14	9	A
1105	Newton	15	10	B

- The large reference Integrity is one of the key dependencies, but theoretically, association and dependency have no life but, composition and aggression have life.
- OLTP, EDP, MIS, EIS, and DSS are not supporting frequently multi-pattern data analysis.
- The scalability, integrity, mapping, and reliabilities issues in data and information.
- The problem in Identification and Classification of Objects and Devices which are available in a traditional database.
- Inconsistency data and information degrade the performance of the large network as well as the operating system.
- RDBMS does not support many to many relationships.
- According to database theory, a multivalued dependency is a full constraint between two sets of attributes in a relation. In connection with the functional dependency, the multivalued dependency requires that certain tuples be present in a relation.
- Issue on efficiency and effectiveness of data as well as information.
- Dirty data over an OLTP (Unstructured and Inconsistencies issue).
- OLTP Running the business, but OLAP Analysis the Business (Both operation and analysis are not resolved by OLTP and OLAP).
- Data is rich and information is poor.
- SQL Support batch processing, generate MIS and OLTP only, therefore, there is a big problem in the decision support system.

### Research Statement

We have to focus on data, information, operation, and services and should be perform the following services all the time and every time. The candidate key generates attributes-based information, that provides dynamic decision support to the top management. We have to emphasizing, taking care of the generation of candidate key frequent patterns that should be a built-in anti-fragile mechanism:

- Service Oriented (Relation, Function, Operation, and Services)
- Object Oriented (Association, collaboration, interaction, communication, dependency)
- Distributed (support to IIOT)
- Dynamic decision support system
- Structure, Order, identification, group, subgroup, and classified
- Maintain Integrated, complex, and integrity
- Privacy and Security

## RESEARCH METHODOLOGY

### Define

We have to define, design, develop and deployment the large candidate key pattern, mechanism, operation & services to fix up dynamic, integration, structure, classification and distributed object-oriented system. Meanwhile, we can maintain the organizational DSS by applying automated method, model, mechanism (M<sup>3</sup>) & tools on Real Time system level to optimize the risk and maximize the decision management to achieve the highest business objective.

This research paper contributes to the define, design, development of an optimization and demoralization model that aims and objective to determine the optimal cost, time and maximize the quality of service to be implemented into the dynamic DSS model & mechanisms deciding on the measure components of MIS, DSS, EIS, and OLTP as follows:

C = Customer, O=Order, M= Manufacturer, V = Vendor, P=Product

Let us take a relation R (C, O, M, V, P); we have following dependencies for a relation R, and we have checked each for being candidate key. Where C = Customer, O=Order, M= Manufacturer, V = Vendor, P=Product and Where Pk= Primary Key, Ck= Candidate Key, Fk= Foreign key & Sk=Super key.

We have to do some research on functional dependency issue.

R = (C, O, M, V, P)

The functional dependencies are:

C => O  
PV => C  
OM => P

It then lists the candidate keys as:CMV, OMV, MVP.

How are these candidate keys derives from the above FDs?

Similarly, where R = (C, O, M, V):The functional dependencies are:

V => O  
CO => V  
CO => M  
M => C

It then lists the candidate keys as:

CO, OM, MV, CV

## Research Design

We have to design the high reliable, scalable, integration and highly available architecture to run the complex business on MIS, DSS, EIS, and OLTP organizational infrastructure to meet over the multi Relation, Function, Operation & Service Level in around the globe for all the time on every time to meet the need of management . These candidates keys set of elements are equally satisfying to the transitive, union of, associative, composition, dependency, association and sum of all for the DSS. We are proposing this idea based on combinational algebra as follows (Figure 3-4). We have to design and develop the integration, reliability and availability model for our complex management infrastructure. We have to find out the best solution to applying step by step of methods, model & mechanism to achieve our objective.

We have to design and develop the Frequent patterns are patterns which appear frequently within a dataset. A frequent item sets is one which is made up of one of these patterns, which is why frequent pattern mining is often alternately referred to as frequent item sets mining. Candidate key about candidate key, Patterns about pattern, Relation about relation, association about association and dependency about dependency working all together as a single operation and services for top management (Bernard, 2007; Mohapatra, 2008; Seymour, 2010).

Associative Laws:  $(Pk + Ck) + k = Pk + (Ck + Fk)$

$(Pk \times Ck) \times Fk = Pk \times (Ck \times Fk)$

Distributive Law:  $Pk \times (Ck + Fk) = Pk \times Ck + Pk \times Fk$

- **Transitivity rule:** Same as transitive rule in algebra, if  $Pk \rightarrow Ck$  holds and  $Ck \rightarrow Fk$  holds, then  $Pk \rightarrow Fk$  also holds.  $Pk \rightarrow Ck$  is called as a functionally that determines  $Ck$ .
- **Union:** If  $Pk \rightarrow Ck$  and  $X \rightarrow Fk$ , then  $Pk \rightarrow CkFk$
- **Transitivity:** If  $Pk \rightarrow Ck$  and  $Ck \rightarrow Fk$ , then  $Pk \rightarrow Fk$ . e.g.; Let  $Pk$  represents {C-ID},  $Ck$  represents {C-CITY} and  $Fk$  represents {C-STATE}. As {C-ID}  $\rightarrow$  {C-CITY} and {C-CITY}  $\rightarrow$  {C-STATE} is true for the relation, so {C-ID}  $\rightarrow$  {C-STATE} will also be true.
- **Transitivity:** Alternatively  $(Pk, Ck) @ R, (Ck, Fk) @ R, (Fk, Pk) @ R$  (Bernard, 2007; Mohapatra, 2008; Seymour, 2010).

The star schema is defined and designed to enforce referential integrity ( $Pk-Fk$ ) of loaded data. This referential integrity is enforced by the use of primary and foreign keys. The primary keys in dimension tables become foreign keys in fact tables to link each record across dimension and fact tables. That's build-in tables about tables and key about key of this proposed design mechanisms are mentioned below.,

The well-organised and designed star schema allows you to quickly understand, navigate and analyze large multidimensional data sets. The main advantages of star schemas in a decision support environment are defined and designed in Figure 3.

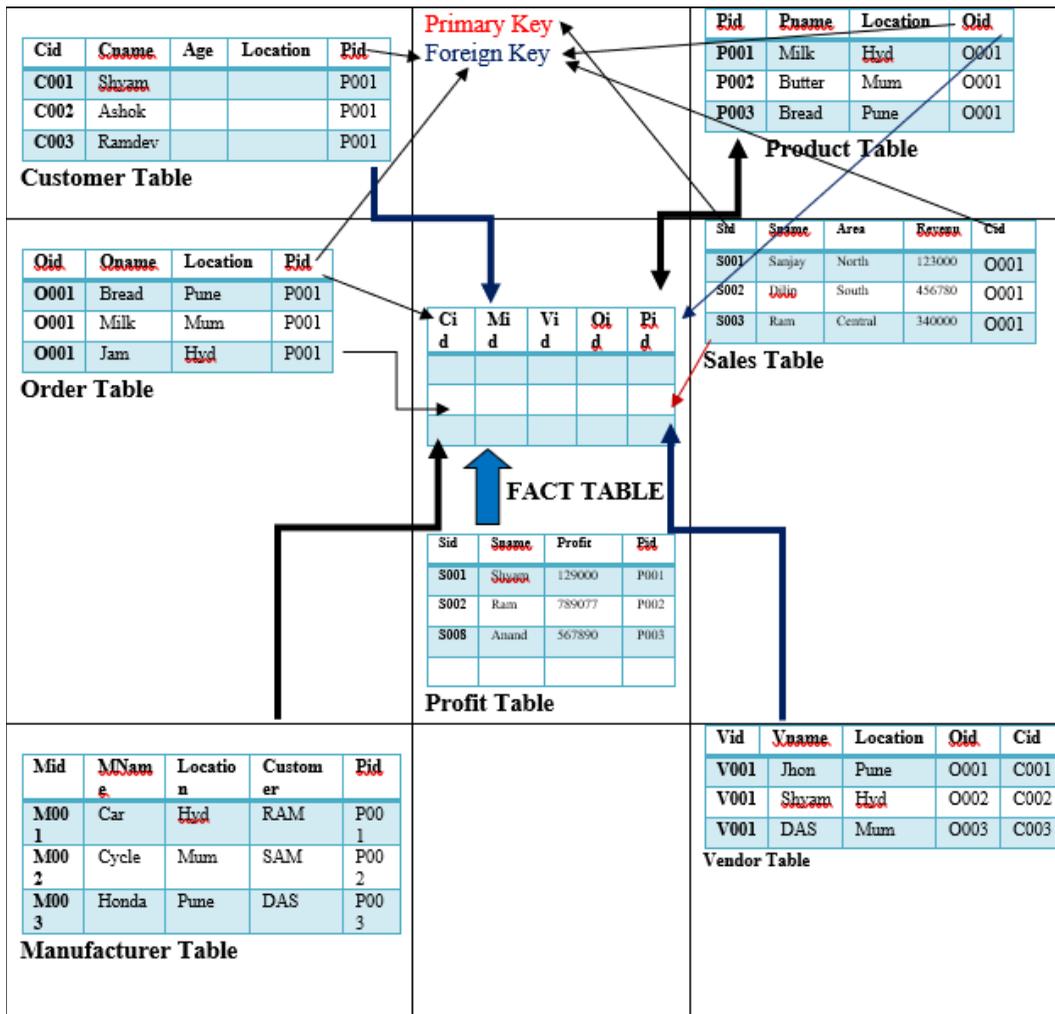
## Related Development (Theoretical Approach)

### Combinations

Sometimes, we want to count all of the possible ways that a single set of objects can be selected - without regard to the order in which they are selected.

In general,  $n$  objects can be arranged in  $n(n-1)(n-2) \dots (3)(2)(1)$  ways. This product is represented by the symbol  $n!$ , which is called **n factorial**. (By convention,  $0! = 1$ ) (Bernard, 2007; Mohapatra, 2008; Seymour, 2010). A **combination** is a selection of all or part of a set of objects, *without* regard to the order in which they were selected. This means that (Customer, Order & Vendor) COV is considered the same combination as COV.

Figure 3. Functional dependency Table (Star-Schema)



- The number of combinations of  $n$  objects taken  $r$  at a time is denoted by  ${}_n C_r$ .

**Rule 1:** The number of combinations of  $n$  objects taken  $r$  at a time is:

$${}_n C_r = n(n-1)(n-2) \dots (n-r+1)/r! = n! / r!(n-r)!$$

Therefore, this proposed Combination is the right things to evaluate the performance of data & information.

Combination Rule:  $n! / r!(n-r)!$

$C = 4! / 2!(4-2)! = 6$  Sets of metadata, Where  $n$  no of objects = 4 &  $r=2$ ,  $r$  ways combination.

$C = 4! / 3!(4-3)! = 4$  Sets of metadata, Where  $n$  no of objects = 5 &  $r=3$ ,  $r$  ways combination.

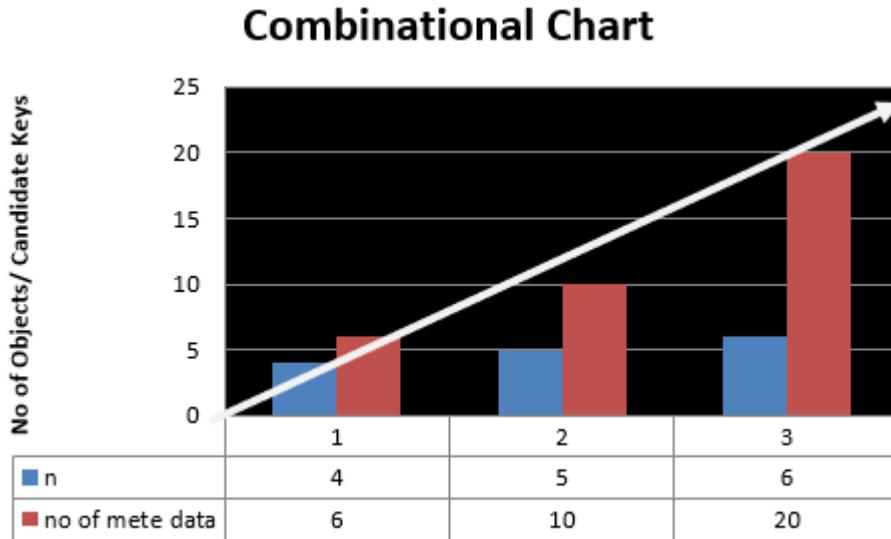
$C = 5!/2!(5-2)! = 10$  Sets of metadata, Where n no of objects = 5 & r=2 ways combination.  
 $C = 5!/3!(5-3)! = 10$  Sets ways of metadata, Where n no of objects = 5 & r=3 ways combination.  
 $C = 5!/4!(5-4)! = 5$  Sets of metadata, Where n no of objects = 5 & r=4 ways combination.  
 $C = 6!/2!(6-2)! = 15$  Sets of metadata, Where n no of objects = 15 & r=2 ways combination.  
 $C = 6!/3!(6-3)! = 20$  Sets of metadata, Where n no of objects = 5 & r=3 ways combination.  
 **$C = 6!/4!(6-4)! = 30$  Sets of metadata, Where n no of objects = 5 & r=2ways combination.**

## Practical Approach (Java Programming-Implementation)

### Java Programming (Experiment)

```
public class Testcomb
{
    static void printCombinations(char[] sequence, int N)
    {
        char[] data = new char[N];
        for (int r = 0; r < sequence.length; r++)
            combinations(sequence, data, 0, N - 1, 0, r);
    }
    static void combinations(char[] sequence, char[] data, int
start, int end,
        int index, int r)
    {
        if (index == r)
        {
            for (int j = 0; j < r; j++)
                System.out.print(data[j] + " ");
            System.out.println();
        }
        for (int i = start; i <= end && ((end - i + 1) >= (r -
index)); i++)
        {
            data[index] = sequence[i];
            combinations(sequence, data, i + 1, end, index + 1,
r);
        }
    }
    public static void main(String args[])
    {
        char[] sequence = { 'C', 'M', 'V', 'O', 'P', };
        System.out.print("The combinations are: ");
        printCombinations(sequence, sequence.length);
    }
}
D:\Java\jdk1.7.0_21\bin>java Testcomb.java [run the java program]
Outcome
```

Figure 4. Candidate keys vs. Set of Metadata



## RESULTS ANALYSIS

The number of candidate key (Object) increase, and the set of metadata and frequent pattern data increases, Therefore, the candidate key is directly proportional to the metadata. The relation, association, dependency, collaboration, communication, integration, mapping, and synchronizations of subsystems became generated, established and improve the DSS. Now this proposed research paper improves the dynamic decision support system and normalizes, optimizes, and standardized the resources. The stronger generation of candidate keys and frequent patterns establish the anti-Fragile information system for top management. Therefore, Combination and Permutation are the right things to evaluate the relationship and performance of data and information:

- Improve the patterns of various keys and datasets.
- Improve the data as well as information quality.
- Improve the operation and services.
- Improve the composite referential integrity.
- Faster reporting, analysis, and planning for top management.
- More accurate reporting, and analysis of current and future business forecasting.
- Better as well as good business decisions.
- Improved data and information quality for DSS.
- Improved employee satisfaction.
- Improved operational and services are more efficient.
- Improved customer satisfaction.
- Support better competitive advantage.

## CONCLUSION

Frequent patterns are patterns which appear frequently within a dataset that maintain multiple and multi ways pattern, which is the frequent pattern mining is often alternately referred to frequent item

sets mining. This frequent pattern will be great helpful for decision support system (DSS) under uncertainty, an order for better, faster and secure management of distributed system computing.

This proposed candidates keys pattern mechanism & analysis providing the better and faster decision support system. This mechanism is accomplished through Real Time System that requires accountability, availability, reliability & integrity are in the part of system management. This research paper is more practical idea & less in theoretical approach as well as available in both analytical & graphical methods. The candidates key patterns are providing attributes based DSS that applied to the dynamic unordered, un-setup and uncertainty. The risk assessment will be help to avoid the conflict among the resources. In this way, we can achieve the operational and service goal and finally maintain the better services to top management.

The frequent pattern is very helpful to the present dependable technology to co-op with the Associations, Aggregation, Composition, Dependency & Collaboration satisfying to the unordered, uncertain environment as well as dynamic and self-autonomy system for web services for all the time on every times. This model highly support to top management to improve the high BCP, DRP, Quality of service, scalability, reliability and availability for all the time.

This dynamic frequent pattern approach helps to the any organization the compete more effectively in local, national and international markets at any time and any place around the clock as minimizing costs & maximize DSS, maximize profits, ROI & TCO. Meanwhile improving functionality, quality & decision (TQM), Maximizing Production, Sales, Market, Share and generate the capital and mean while optimizing time & maximize the utilization of Man, Machine, Material, Market, Money, and Method(M<sup>5</sup>) solving multiple problems at right time with optimal cost and utilizing overall resources more effectively at right time and right place.

## REFERENCES

- Abbas, N., Riaz, M., & Does, R. J. M. M. (2014). Memory-Type Control Charts for Monitoring the Process Dispersion. *Quality and Reliability Engineering International*, *30*(5), 623–632. doi:10.1002/qre.1514
- Abraham & Sudarshan. (2013). *Database System Concept*. McGraw Hill.
- Alex & Stephen. (2013). *Dataware housing Data Mining and OLAP*. McGraw Hill.
- Anowar, F., & Sadaoui, S. (2020). Detection of auction fraud in commercial sites. *Journal of Theoretical and Applied Electronic Commerce Research*.
- Bernard, K. (2007). *Discrete Mathematical structures*. Person Education India (PHI).
- Boerger, J. (2021). *Rethinking Information Technology Asset Management*. Import.
- Chopra, R. (2022). *Data Science with Artificial Intelligence, Machine Learning and Deep Learning - Simplified Q & A*. Khanna Book Publishing.
- Coleman, T. (2011). *A practical guide to risk management*. Research foundation of CFA Institute.
- den Boer, N. (1995, January). A Risk Control Strategy Corporate Computer Operations. *Computer Audit*, *4*, 18–30. doi:10.1016/S0960-2593(00)80068-3
- Elmasri & Navathe. (2007). *Fundamental of Database System*. Person Education India (PHI).
- Gary & James. (2009). *Database Managent & Design*. Person Education India (PHI).
- Grus, J. (2015). *Data Science from Scratch*. O’Reilly
- Gupta, K. (2012). *Management Information System*. S Chand Publishing.
- Haigh, A. (2011). *Object Oriented Analysis & Design*. McGraw Hill.
- Jain, V. K. (2018). *Data Science and Analytics* Paperback Khanna Publishing, India
- Jiawei & Kamber. (2011). *Data Mining Concept & Technology*. Elsevier.
- Kobs, D. (2021). *Data Science for Beginners: Comprehensive Guide to Most Important Basics in Data Science*. Alex Published.
- Liu, L., & Mohapatra, D. P. (2008). *Elements of Discrete Mathematics*. McGraw Hill.
- Maheshwari, A. (2017). *Data Analytics Paperback*. McGraw Hill Education.
- Maital, S., & Seshadri, D. V. R. (2008). *Innovation Management*. Response Books.
- Meenakshi, N., Rajakumari, K. E., & Hariharasitaraman, S. (2021). *Data Science and Machine Learning*. Notion Press.
- Michael & Rumbaugh. (2011). *Object Oriented Methodology & Design with UML*. Person Education India (PHI).
- Minelli, M. (2018). *Big Data, Big Analytics: Emerging Business Intelligence and Analytic Trends for Today’s Business*. Gildan Media.
- Neha & Manoj. (2018). Analysing the impact of online Brand trust on Sales Promotions and Online Buying Decision. *The IUP journal of Marketing Management*, *17*(3).
- Nilsson. (2002). *Principle of Artificial Intelligence*. Narosa Publication House.
- Provost, F., & Fawcett, T. (2013). *Data Science for Business Publisher(s)*. O’Reilly Media, Inc.
- Raz, T., & Hillson, D. (2005). A Comparative Review of Risk Management Standards. *Risk Management*, *7*(4), 53–66. doi:10.1057/palgrave.rm.8240227
- Rich, E. (2012). *Artificial Intelligence*. McGraw Hill.
- Richard, B. Chase, R., & Nicholas, N. (2006). *Operations Management*. Tata McGraw Hill.

- Lipschutz, S. & Patil, V. (2010). Discrete Mathematics, New Delhi, India. McGraw Hill.
- Sahil, R. (2017). Management Information System | Second Edition | By Pearson Paperback. .
- Sales Rodrigues, K. A. (2022). Book Review: An Introduction to Nonparametric Statistics. *Journal of Behavioral Data Science*, 2(1), 124–127. doi:10.35566/jbds/v2n1/p8
- Shats, S. M., Dong, F., & Xu, H. (2010). Reasoning under uncertainty for skill detection in online auctions using Dempster-Shafer theory. *International Journal of Software Engineering and Knowledge Engineering*.
- Shon, H. (2002). *Security Management Practices*. Wiley Publishing Inc.
- Shu, Q. L. (2014). Research on the quality stability evaluation and monitoring based on the pre-control chart. *International Journal of Quality & Reliability Management*, 31(9), 966–982. doi:10.1108/IJQRM-02-2013-0032
- Sudarshan.(2019). *Database System Concepts*, (7th ed). McGraw-Hill.
- Trivedi. (2009). *Artificial Intelligence*. Khanna Book Publishing.
- Turban, A., & Liang, S. (2009). *Decision Support and Business Intelligence Systems*. Person Education India (PHI).
- Vohra, N. D. (2007). *Quantitative Techniques in Management*. Wiley Publishing Inc.
- Waggoner, P., & Kennedy, R. (2022). The Role of Personality in Trust in Public Policy Automation. *Journal of Behavioral Data Science*, 2(1), 106–123. doi:10.35566/v2n1/p4/
- Weber, R. (2014). Information System Control and Audit. New Delhi, India: Person Education India (PHI).
- Xin, T. & Xiaofang, B. (2014). A Hierarchical Information System Risk Evaluation Method Based on Asset Dependence Chain. *Intl. Journal of Information & Network Security*, 3(3).

*Padma Lochan Pradhan received M Tech and Ph. D in Computer Science & Engineering in the field of System Security. In addition to this, he is certified in UNIX SUN SOLARIS from Sun Micro System USA. His area of interest is Data Science, ML, Big-Data, System Security, Real-Time system, System Programming, and Optimization. He has around 18 years in IT industries and 15 years in Academic & Research in various capacities at IBM, Sun Micro System, Thomson Scientific (ISI) in the USA, and PwC Singapore. His research interests are System Security, Risk Assessment, Data Science, Decision Science, and Optimization. He is also acting as a Reviewer, Editorial Board Member, and Associate Editor of many more International Research Journals. He published 28 research papers in the area of system security and risk optimization. As a Reviewer, he reviewed 115 research papers in the last three years (IGI-Global, Springer, and Asian Journals).*