International Research Journal of Engineering Science, Technology and Innovation Vol. 8(6) pp. 1-6, December, 2022
Available online http://www.interesjournals.org/IRJESTI

Research Article

Modeling for Learning the SQL Select Statement Internal Sequence Processing to Tracing Optimization and Anticipation the Final Result to Meet the User's Requirements

Hashan Idhaim[1]*, Ayoub Alsarhan[2], Bashar Igried Deb Alkhawaldeh[3] and Asem Hasan Idhaim[4]

[1]Computer Information System Prince Al-Hussein Bin Abdullah II for Information Technology Hashemite University Zarqa Jordan

[2]Computer Information System Prince Al-Hussein Bin Abdullah II for Information Technology Hashemite University Zarqa Jordan

[3]Department of Computer Science and Applications (CSA) Prince Al-Hussein Bin Abdullah II for Information Technology Hashemite University Zarqa Jordan

[4]Software and Information Technology Engineering Department ETS University Montreal CA

*Corresponding Author's E-mail: Robert_a@55gmail.com

Received: 08-Dec-2022; Manuscript No: irjesti-22-82750; **Editor assigned:** 10-Dec-2022; Pre-QC No: irjesti-22-82750 (PQ); **Reviewed:** 15-Dec-2022; QC No: irjesti-22-82750; **Revised:** 24-Dec-2022; Manuscript No: irjesti-22-82750 (R); **Published:** 30-Dec-2022, DOI: 10.14303/2315-5663.2022.85

Abstract

Structure Query Language (SQL) command used for data retrieval and the most common usage used is the SELECT statement. A lot of papers presenting valuable information for SQL commands writing, tuning, usage, and optimization but comparatively little effort has been spent systematically focusing on the hidden internal processes of the SELECT statement and based on the axiom a better understanding of how things work, better results can be obtained, so this paper purposes the repost Internal Hidden Processing Modeling (IHPM) that decomposes the internal processing of the SELECT statement into five sequences phases to learn significantly the SELECT statement internal processing to help researchers, novice, and senior programmers, database practitioners, and students in a rewriting and modelling a robust SELECT statement that meeting a user requirements and open a chance to the researchers to develop each phase independently.

Keywords: Structured Query Language (SQL), Metadata, Relational Database Management System (RDBMS), Internal Processing Modeling (IHPM).

INTRODUCTION

A significant amount of the world's information and knowledge is stored in relational databases. However, the ability for users to retrieve facts from a database is limited due to a lack of understanding of query languages such as Structure Query language (SQL) (Zhong V, 2017). A database is a collection of data organized into a structured format defined by metadata. Metadata are data about the data being stored: they define how the data are stored within the database (Sheldon R, 2003). SQL (pronounced es-cue-el) is the relational database language standard that initially was developed by IBM in the early 1970s, published jointly by ISO (International Organization for Standardization) and IEC (International Electro technical Commission), and the latest edition SQL: 2011 was published by ISO/IEC in December 2011 (Kulkarni K, 2012). SELECT is the most important SQL statement type. Even when we're changing data, most of the logic will go in the SELECT and WHERE clauses of the statement. Before we can insert, update, or delete a set, we

must be able to choose a set (Heller J, 2019). Database query languages provide access to information in a database. Such queries may be composed via menus, command languages, or direct manipulation, but at last appear as (SQL) queries (Smelcer JB, 1995). Michael Stonebreaker has referred to SQL as “intergalactic data speak” (a fact that served as the inspiration for our paper). Indeed, like, and, SQL may be among the most widely used and understood computer languages (Jim M, 1999). SQL is today the de facto standard language for relational and object-relational databases (Brass S, 2006). The physical execution of SQL SELECT statement has the implicitly internal processing and showing only the final result which hardening the detection, tracing semantic and logical error. A little effort has been spent systematically focusing on the internal processing of SQL SELECT statement, which are the most SQL statements used, longest in written, and the most difficult in detection and tracing. The proposed IHPM is modelling the SELECT statement into five sequenced phases to illustrate each independent phase data customized processing till predicting the SELECT statement final result. By implementing IHPM the result of SELECT statement physical execution is breaking down to five phases showing how the data processed in reverse direction till the beginning of the extracting data from the raw input tables. This paper intends to present IHPM for the SQL SELECT statement to provide a full brief reference guide with knowledge needed to understand and learn the SQL SELECT statement to help the researchers in developing and enhancement each phase of the SELECT statement independently in addition to make the detection, tracing, semantic and logical error to be simple and easy to follow up. The paper will be presented in the following order: Testing data description, background and related work, the proposed IHPM methodology, IHPM model testing experimentally, tips of writing the SELECT statement, and conclusion.

DATA DESCRIPTION

Two tables are creating for implementation, testing and experimenting IHPM mode the students and courses. The table’s data shown below is an arbitrarily selected data sample to be used in the experimental usage implementation the IHPM model (Table 1 and 2).

Table 1. Students data.

Id	S_Name
1	Ali
2	Muna
3	Sami

Table 2. Courses data.

Id	C_Name
1	Java
2	C++
3	VB

BACKGROUND AND RELATED WORK

Database Management System (DBMS) has multiple kinds, such as ORACLE, SQL Server, MySQL, PostgreSQL, DB2, Sybase, Teradata and SQLite; SQL command can be adopted to inquire about the inquiry of database. SQL was one of the first commercial languages to utilize Edgar F. Codd’s relational model. The model was described in his influential 1970 paper, "A Relational Model of Data for Large Shared Data Banks (SQL, 2003). SQL is a database language designed for managing data held in a relational database management system. SQL was initially developed by IBM in the early (Codd EF, 1970). The initial design of SQL (then called SEQUEL) was performed by Chamberlin and Boyce (1974) at IBM in the early 1970s (Codd EF, 1970). SQL is a domain-specific programming language designed to store and access data in relational databases (Kirby M, 2014). Despite not entirely adhering to the relational model as described by Codd, SQL became the most widely used database language (Kirby M, 2013) and since SQL-92, the major revisions of the SQL standards have been SQL: 1999, SQL: 2003, SQL: 2008, and now SQL: 2011, and there are a unique SQL versions dedicated to a special RDBMS like SQL: 2016, and SQL: 2019 that is used only for Microsoft SQL server. SQL commands is grouped into 3 groups: Data definition language (DDL), Data Manipulation language (DML), and Data Control language (DCL) (Prasad A, 2020). SELECT statement belongs to DML in conventional database consists of these six parts: Select, From, Where, Group by, having, and Order by (Chapple M, 2009).

METHODOLOGY

The proposed IHPM model of the SELECT statement composed of five sequenced processing phases (Silva YN, 2016). It comprises all the SELECT statement keyword to help researchers and database practitioners learning the detailed hidden internal processing, input, and output of each individual phase.IHPM model aiming to clarify and understanding how the "SELECT statement" producing final result in five sequenced phases which is shown only in one phase in the physical execution. Below shows IHPM phases (Figure 1).

Sql select statement

The SELECT statement is used to select data from a database.

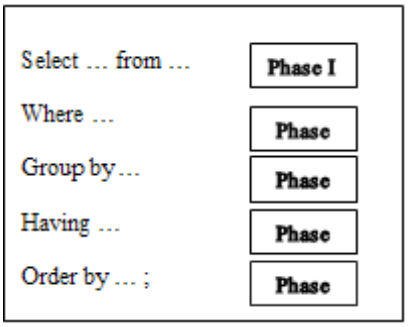

Figure 1. IHPM Phases.

The data returned is stored in a result table, called the result-set; it composed from the following clauses.

Select clause: contains select key word followed by column name, equation, or group function.

From clause: contains the from key word followed by table or view.

Filter clause: contains the where, group by, having, or order by hey word.

IHPM Phases: IHPM model is consisting all the SELECT statement clauses. Each phase output is the input of the next phase, and the output of the final phase (phase-V) is the SELECT statement final result. Every IHPM model phase usages, clauses, components, keyword's functionality, inputs, and outputs, is briefly explained in below (Kvet M, 2015). The IHPM model phases are (Table 3).

Phase-I: The initial main phase: this phase in primary composed of the following initial, main, and basic SELECT statement clauses.

Select clause: it's the first clause in the SELECT statement, uses for identifying the columns name, alias (columns headings), output formatting, function, mathematical expression, and group function. Group functions calculations are deferred into phase III as will be explained later.

From clause: this phase is used to identify the data resources (database tables or views) of the columns identified in phase I. The input of this phase is the projection of the select clause columns over the from clause data resources. The output is the same input if the from clause has only a single data resource, while the output is the Cartesian product if the from clause has more than one data resource (Nunu K, 2021).

Phase-II: Column's constraints: its secondary phase composes of the SELECT statement where clause only, contains of the columns Boolean conditions. The output of this phase is the input rows that meet and satisfy where clause conditions, and it's called processed data by column constraints.

Phase-III: Grouping and the group function calculations: its secondary phase unless the select clause has column and group function like the example in figure-2, in this case it will be considered as primary phase. In this phase the entire scope of the input rows is divided into sub groups based on the group by clause columns, and the output is called the processed data by grouping (Figure 2).

Phase-IV: Group function constraints: secondary phase, compose of the SELECT statement having clause only, contains group function Boolean conditions. The HAVING

```
Select x, count(x), max(y), sum (z), min(r), avg (w)…..
```

Figure 2. Column & Group functions

Table 3. IHPM dodul phases.

Phase	Key word	Mandatory	Component	Remark	Input	Output
I	Select	X	*,Column, Function, Group function, Values, alias ,Mathematical expression	(1) *: used to display all column in tables & views in from clause, and with group function count: count (*).	Projection of the select clause component over the from clause components	(1) Same input.
	From	X	Table, View	(2) Output type identification based on the number of the "from clause" components.		(2) Cartesian rows set produced by the from clause component.
II	Where		Column Boolean expression, logical operator, sub query		Phase I output	The Processed Data by column conditions
III	Group by	X*	Column, aggregate/ group function	X*: mandatory in case having column & group function in select clause.	Phase II output	The Processed Data by grouping
				Group function calculation in this phase		
IV	Having		Group function Boolean expression, logical operator, sub query		Phase III output	The Processed Data. by Group function conditions
V	Order by		Column, Group function, Values ordering, type ascending or descending	The output is the SELECT statement final result	Phase IV output	The Processed Data by sorting

clause acts similar to a WHERE clause, but on the group by function (count (), max (), min (), sum (), avg () … ets). The input rows that meet and satisfy having clause constraints are the output of this phase. The phase output is called processed data by group function constraints.

Phase-V: Result sorting: it's the final, secondary phase, composes of the order by clause only, and used to rearrange the output based on the identified order by clause. The ordering is ascending (asc), or descending (desc), the default is ascending (asc). The output is called the processed data by sorting. The five IHPM model phases are summarized in below table.

IHPM MODEL TESTING

IHPM model testing hypothesis: Testing IHPM model empirically by making sure that the final result of implementing any SELECT statement by the IHPM model which it is processed by five sequenced phase must be equal and fully matched with the output result of SELECT statement physical execution. IHPM model testing experiment will be in the following order: setting up the DBMS software for physical execution, building and preparing empirical data for testing, chosen a SELECT statement example for testing, implementing the IHPM model testing, physical execution of the SELECT statement, comparison the result set of IHPM model with the SELECT statement physical execution, and obtaining the hypothesis testing result. The processes of IHPM testing as follows:

1. Setting up the DBMS software for physical execution: ORACLE 18c XE (open source) is installed, SQL plus command line is a command-line interface for accessing Oracle Database XE that enables operating SQL, PL / SQL, and SQL*Plus commands.

2. Building and prepare empirical data for testing: the empirical data for testing is explained in section 2 above.

3. Chosen and written a SELECT statement sample example for testing: written a testing SELECT statement sample example that contains all phases of IHPM model for testing purpose the IHPM model. The testing SELECT statement sample example is shown in (Figure 3).

4. Implementing the IHPM model testing: the chosen testing SELECT statement example shown in figure 3 will be used in IHPM model testing implementation, figure 4 will show the data processing in all IHPM phases testing till showing the final selected testing SELECT statement output. IHPM model testing phases explained in section 5.2 above will be implementing in the following sequence order.

```
Select S_Name, count (C_Name) from
students, courses

    Where id=s_id

    Group by S_Name

    Having count (C_Name)>1

    order by count(c_name);
```

Figure 3. SELECT statement testing example.

5. IHPM model Phase-I testing: Since we have two tables' Students and Courses in the form clause, the result will be the Cartesian product shown in (Phase-I: Cartesian product). The number of rows shown in phase I is 18 rows.

6. IHPM model Phase-II testing: Where condition checking, the constraints "id=s_id"of the testing SELECT statement is implemented in the data produces in phase-I, as showed in figure 4, Phase II: Where condition checking, the rows that meet and satisfy the where condition is showed in Phase- II: Output Result. The output contains 6 rows.

7. IHPM model Phase-III testing: Group by, in this phase the group by clause is performed parallel with the calculation of the group by function over the output of phase-II, and the output of processing this phase showed in figure 4 (Phase III: Group by), the output is grouped and summarized into 3 rows only.

8. IHPM model Phase-IV: Having checking, this phase include only the having clause group function condition "count (C_Name)>1"of the testing SELECT statement. The having clause group function processed on the output data of phase-III mentioned in the previous section above , the group function condition checking result showed in (Phase-IV: Output result) (Figure 4).

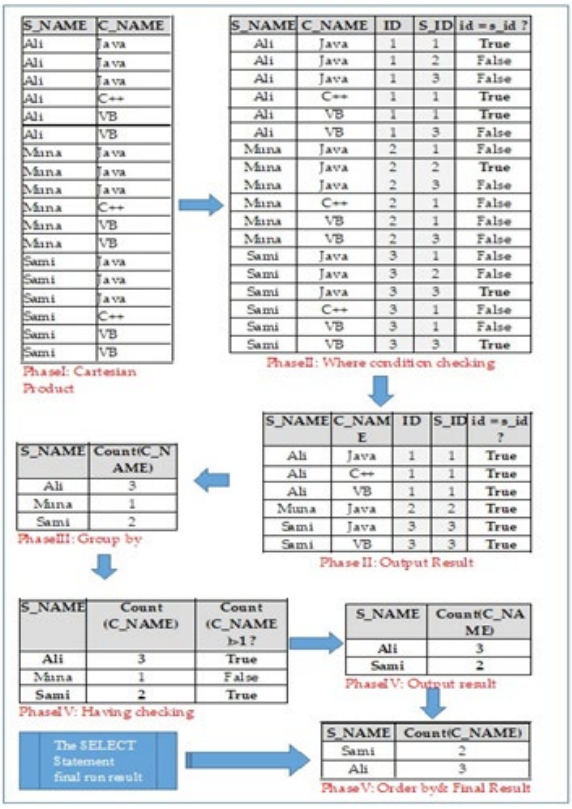

Figure 4. IHPM Data processing phases.

9. IHPM model Phase-V: the last processing phase in the IHPM model, the output of the previous phase (Phase IV) is rearranged by sorting it based on the order by clause of the SELECT statement, the output of this phase will be final result of SELECT statement showed in figure 4(Phase V:Order by and final result).

10. Physical execution of the SELECT statement: the chosen SELECT statement for testing showed in figure 3 above is executed on software testing environment identified and described in step 5.1 above. Below is showing a snapshot of the physical running of the SELECT statement and result denoted in red rectangle (Figure 5).

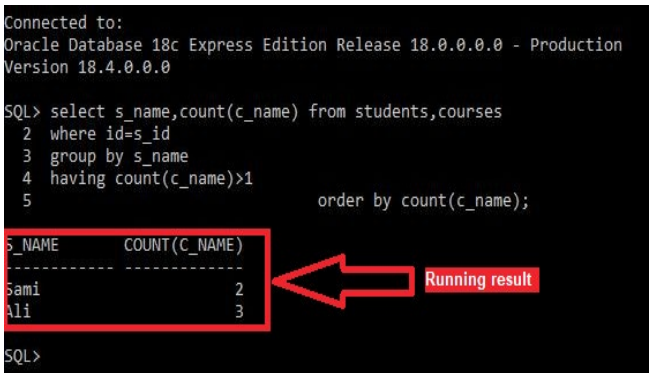

Figure 5. Physical SELECT statement running & result.

11. Hypothesis testing result: By comparison the two output result sets of the chosen SELECT statement for experiment that was showed in figure 3; first, the result set which was showed in figure 4 above, and it produced from implementing IHPM model, second, the result set which was showed in figure 5 above, and it produced from the physical execution, we got the same identical and equals result sets in the both comparisons sides. So the result of the testing hypothesis is true.

SELECT STATEMENT WRITING TIPS

From the implementation and the experiment of IHPM model we obtained following obtained useful tips in the writing the SELECT statement:

1. The written SELECT statement must have at least the IHPM phase I, in other word, must have the first two main/ mandatory clauses, the “select” and “from” clauses.

2. All the processing phases from (Phase-II) to (Phase-V) are optional except the Phase-III special case that mentioned in table-3 above.

3. If the SELECT statement have more than one object in from clause (Table or View), and joining condition is not specified in phase II, the SELECT statement result is the Cartesian product, and this is considered a semantic or logical error.

4. Customizing the retrieved data based on conditions: if a condition on the columns, it will write on IHPM phase II, and if it’s on the group functions, it will write on IHPM phase IV.

CONCLUSION

The only and most popular SQL command used for data retrieval and the most usage used is the SELECT statement. Based on the axiom a better understanding of how things work, better results can be obtained, and for having fundamental understandings of how a SELECT statement is executed, the invented IHPM model in this paper divided the SELECT statement into five process phases based on its clauses functionality, and it’s a robust choice for users from different aspects to SELECT statement modeling, development, error detection and tracing to help practically and theoretically in the development modern computer application systems that meet a conceived user’s requirements. This work opens a plethora of a new wide range researches in different levels from the IHPM phase individually to the integral of all phases that explained how the SELECT statement final result is produced theoretically and meets identically with the practical SELECT statement physical execution.

REFERENCES

1. Zhong V, Xiong C, Socher R (2017) Seq2SQL: Generating Structured Queries from Natural Language Using Reinforcement Learning.
2. Sheldon R (2003). SQL: A beginner’s guide (2nd Ed.). Osborne New York McGraw-Hill.
3. Kulkarni K, Michels JE. (2012) Temporal features in SQL: 2011. IBM Corporation, SIGMOD Record. 41: 34-43.
4. Heller J (2019) Query the Database with Advanced SELECT Features In: Pro Oracle SQL Development. Apress Berkeley CA. 127-190.
5. Smelcer JB (1995). User error in database query composition. Int J Hum Comput. 42: 353-381.
6. Jim M, Alan RS (1999) SQL: 1999 Understanding Relational Language Components Morgan Kaufmann Publishers San Francisco, CA 94104-3205, USA ISBN: 1-55860-456-1.
7. Brass S, Goldberg C (2006). Semantic errors in SQL query A quite complete list. J Syst Softw. 79: 630-644.
8. SQL (Structural Query Language) Command Compiling Method and SQL Command Compiling Device. (2013).
9. Codd EF (1970) A Relational Model of Data for Large Shared Data Banks. Commun ACM. 13: 377–387.
10. Watt AN (2014). Eng Database Design – 2nd Editions. Victoria, B.C.: Campus. Retrieved from.
11. Kirby M, Sambasivam S, Hadfield S, Wolthuis S (2013). Relational Algebra and SQL: Better Together. J Inf Syst Educ.11: 4-13.
12. Prasad A, Badhya SS, Yashwanth YS, Shetty R, Shobha G, Deepamala N (2020). Enhancement of Natural Language to SQL Query Conversion using Machine Learning Techniques. Int J Adv Comput Sci Appl. 11: 495-503.
13. Chapple M (2009). SQL Fundamentals Databases. About com Retrieved. 28

14. "Structured Query Language (SQL)".(2006). International Business Machines. Retrieved 2007-06-10.
15. Silva YN, Almeida I, Queiroz M (2016). From Traditional Databases to Big Data. Conference the 47th ACM Technical Symposium. 143-418.
16. Kvet M, Matiaško K (2015). Temporal Extension of the Select Statement.
17. Nunu K, Julaeha S, Parulian D, Selvia N, Ambarsari EW (2021). Venn versus Relation Diagram Models for Database Relation in SQL Command Line. J Phys Conf Ser. 17831: 12050.
18. Oracle. “A Using SQL Command Line”. Database Express Edition 2 Day Developer Guide.