Global Journals LaTeX JournalKaleidoscope™
Artificial Intelligence formulated this projection for compatibility purposes from the original article published at Global Journals. However, this technology is currently in beta. *Therefore, kindly ignore odd layouts, missed formulae, text, tables, or figures.*

A New Approach for Calculating Average Value Including NullWithout Aggregate Function

Mridul Kanti Das[1], Goutam Biswas[2] and Md. Masudur Rahman[3]

1

Received: 2 December 2011 Accepted: 27 December 2011 Published: 6 January 2012

Abstract

An evaluation of aggregate functions in relational database system considerable impact on performance in many application areas like geographic information systems and statistical and scientific databases. The problem with existing systems is inefficient execution of aggregate functions with large data volumes and lack of flexibility. It is not possible to extend the systems with new aggregate (average) functions. We show how this could be implemented into a database. We also describe how support for special kinds of aggregate queries and data structures can help in designing future high performance systems.

Index terms— Aggregate Function, NULL, SQL, Boolean Algebra, Sub Queries.

1 INTRODUCTION

ggregate functions perform a calculation on a set of values and return a single value. Aggregate function avg( ) only calculate average without null values. It provides average result, eliminating null values. Null does not have a value (and is not a member of any data domain) but it is a placeholder or "mark" for missing information. Comparisons with Null can never result in either True or False but always in the third logical result is Unknown. So comparing two null is difficult. We discus about

(1) review of the research for handling null values in database system using aggregate function (2) problem structure with null value with respect to database (3) describes existing solution and proposed solution and its algorithm as well as how it works (4) details the experimental work that has been carried out. The experimental evaluation has been performed using a large amount of datasets.

2 COMPARISON TABLE BETWEEN EXISTING AND PROPOSED SOLUTION

From comparison table we see that our propose system takes less times than existing system. By proposed system can reduce time and reduce the problem of existing system. To understand easily a graph chart is given below.

3 VII. GRAPH OF EXISTING SOLUTION VS. PROPOSED SOLUTION

Our propose solution is efficient to calculate average value with Null values from large amount of data.

From the above graph Green bar indicates Existing solution time and Red Bar indicates proposed solution time. We see that in proposed system needs execution time less than existing system. VIII.

4 CONCLUSION

At the age of globalization most of all bank already has been computerized. They store their customer information, balance, transaction etc. in database. And they need to calculate average number of transaction after a certain

39 period of time. Even stock exchange Ltd. Hospital, Airlines etc. need to calculate average number of transaction frequently. So our proposed system will be best for them which can save their times. [1] [2] [3]

bank

VI.

Amount of Data	Existing solution Execution Time (sec)	Proposed Solution Execution Time (sec)
40000	0.2840	0.2460
80000	0.5720	0.5040
120000	0.8440	0.8250
160000	1.1841	1.1591
180000	1.3301	1.3011
200000	1.5511	1.5011
220000	1.7511	1.5781
240000	1.9371	1.7641
260000	2.0601	1.9671
280000	2.1851	2.0762

Figure 1: Table bank

40

[1] © 2011 Global Journals Inc. (US)

[2] December

[3] Global I) 2011 December (A New Approach for Calculating Average Value (Including Null) Without Aggregate Function

V.

.1 PERFORMANCE MEASURE TABLE OF PROPOSED SOLUTION

In this solution we see that if the number of data in database gradually increased then the execution time is increased. **??**uilt

[// d is a data table which like two dimensional //array, size is maximum data row] *// d is a data table which like two dimensional //array, size is maximum data row*,

[Codd (1970)] 'A Relational Model of Data for Large Shared Data Banks'. E F Codd . *Communications of the ACM* June 1970. 13 (6) p. .

[Algorithm Average_WN (d, Avg, size)] *Algorithm Average_WN (d, Avg, size)*,

[Analytic Functions from Oracle® Database SQL Reference 10g Release 1 (10.1) Part Number] *Analytic Functions from Oracle® Database SQL Reference 10g Release 1 (10.1) Part Number*, B10759-01.

[Avg:=sum/size ; 25] *Avg:=sum/size ; 25*, 23. } 24. (Print AVG)

[Comparing Tables By Bill Graziano on 07 (2002)] *Comparing Tables By Bill Graziano on 07*, January 2002.

[December A New Approach for Calculating Average Value (Including Null) Without Aggregate Function 3. Jeff Smith is softwar*December A New Approach for Calculating Average Value (Including Null) Without Aggregate Function 3. Jeff Smith is software developer, he using UNION operator comparing NULL values to other NULLs*, www.weblogs.sqlteam.com