International Journal of Future Generation Communication and Networking
Vol. 14, No. 1, (2021), pp. 2852-2858

Design Of Dynamic Ware House By Using Simulated Annealing Algorithm With On-Line Analytical Processing System

Ch. Suresh Kumar[1], Dr.Dhyan Chandra Yadav[2], Dr.D.Kiran Kumar [3]

[1]*research Scholar, Js University, Shikohabad, Up.*

[2]*Professor & Supervisor, JS University, Shikohabad, UP.*

[3] *Professor &Co- Supervisor, JS University, Shikohabad, UP.*

Abstract

The Quantity Of Data Accessible To Large-Scale Businesses Is Increasing At A Breakneck Pace. Continuously, Operating Systems Produce New Data. In A Warehouse, Decision Support Services Such As Online Analytical Processing (Olap) May Include Hundreds Or Thousands Of Sophisticated Aggregate Queries Across A Huge Quantity Of Data. A Data Warehouse May Be Thought Of As A Collection Of Materialized Views That Are Specified Over A Set Of Relationships. When A Question Is Asked To Be Answered In This Document, The Appropriate Materialized Views And Tables Will Be Utilized To Generate The Finest Views And Tables For Building Any New Query. To Accomplish And Implement The Dynamic Warehouse Design, Three Complex Olap Queries With Join And Aggregation Operations Were Created, Views Were Created And Updated Via Windows Task Scheduler And Batch Files Based On Base Table Updates, Lattices Of Views Were Created Via Multiple View Processing Plant Operations, And The Simulated Annealing (Sa) Algorithm Was Developed And Introduced For Query Resolution. The Primary Objectives Of This Study Are To Demonstrate How The Usage Of Derived Data, Such As Materialized Views, For Run-Time Re-Optimization Of Aggregate Queries (Fast Response Time), Is Critical To The Success Of Any Data Warehouse.

Keywords: On-Line Analytical Processing (Olap), Materialized View, Dynamic Warehouse, Simulated Annealing, Data Warehouse.

Introduction

A Data Warehouse (Dw) Is A Relational Database That Is Used To Store Data That Has Been Gathered From Various Operational Databases In Order To Facilitate Sophisticated Business Analytical Queries Such As Summarization Aggregates And So On. The Aim Of A Data Warehouse Is To Rapidly Replicate New Data Across All Relevant Tables And Displays [1]. The Characteristics Of Dw Are That They Are "Subject-Oriented, Integrated, Non-Volatile, And Time-Variant"[2]. When A Business Environment Changes Or Is Updated, The Content And Structure Of The Underlying Data Sources May Change. Thus, Dw Is Always Evolving And Maturing In Its Surroundings. This Process Requires Considerable Attention, And Many Scholars Have Tackled These Problems From A Variety Of Perspectives [3]. [4]. A Data Warehouse May Be Thought Of As A Collection Of Materialised Views (Mvs) Defined Over A Collection Of Basic Relations; Materialised Views Are The Query Results Or Some Intermediate Results That Are Physically Stored In Order To Prevent Repeated Calculation Operations [5]. Complex Grouping/Aggregation Queries Were Utilised In On-Line Analytical Process (Olap) And Judgment Support System (Dss) Applications To Make A Decision And Respond To The Query. When Base Relations Are Changed, The Mv At The Dw Views Must Be Updated As Well [1]. Numerous Developed Methods For Query Optimization Have Been Suggested To Accelerate Data Warehouse Performance, Including Multiple View Processing Plan (Mvpp) And Traditional Algorithms For Materialised View Selection, Such As Simulated Annealing, Genetic Algorithm, And A* Algorithm.

ISSN: 2233-7857 IJFGCN

Background

The Selection Of A Materialized View In Data Warehouse Architecture Is An Essential Design Choice. Figure 1 Shows The Fundamental Data Warehouse Architecture [10]. Historical Information Is Collected From The Sources. Each Source Of Data Link To A Data Translation And Change Detection Wrapper/Monitor. The Wrapper Converts The Source Scheme To The Scheme Used By The Data Warehouse, And The Monitor Detects The Source Modifications That Are Subsequently Sent To The Integrator. The Integrate Shall Insert And Modify The Data Within The Data Storage Facility, Dissipate Changes To The Source Relation And, If Necessary, Maintain Tables In The Storage Facility.

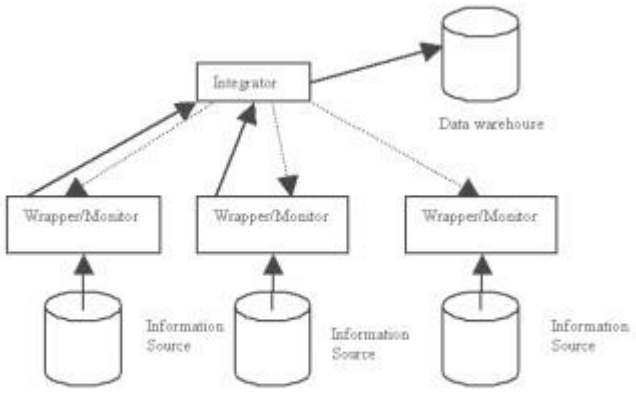

Figure 1. Basic Architecture of Data Warehousing

Dynamic Data Warehouse

Dws Are Dynamic Entities Which May Be Constantly Developed Or Updated Over Time. If Time Goes On, Dw Must Answer New Questions. Some Of The New Questions May Be Addressed Using Dw-Related Views. However, Additional Fresh Questions To Respond By Dw Require New Perspectives To Materialize. In Every Instance, A Full Rewrite Of The Query Over (Old And New) Materialised Views Must Be Done For A Question To Be Answered By Dw. Rewrite Queries May Be Made Either Through Old Views Or New Views, Or Partly By New Views And Partly Via Old Views. For The Materialisation Of New Views, More Room Must Then Be Provided In Order To Respond To New Requests[7]. Many Methods Were Utilized For Selecting A Range Of Views For Genetic, A*, Greedy And Simulation Algorithms.

Simulated Annealing Algorithm

Applied To Various Combinatory Optimization Problems Like A Travel Agent Issue, Simulated Annealing (Sa) Is A One-Solving Meta Heuristic. Sa Technique Is Helpful For Dealing With A Problem With A Very Big Discrete Arrangement Area With An Excessive Search Space, Whereby Minimize (Or Maximize) An Objective Cost Function [9]. This Article Introduces And Applies The Simulated Annealing (Sa) Method. The Reason For Adopting Sa Algorithm Is To Resolve A Problem Of Materialised Visual Selection Based On The Fact That The Data Store May Have Several Views And That The Queries Often Change. The Realised Selection Of Views Is Thus Seen As A Difficult Issue With A Very Wide State Area.

Answering Queries Using Views Or Rewriting Queries Using Views

ISSN: 2233-7857 IJFGCN

It Is An Operation To Identify Effective Ways For Rewriting The Query Using A Collection Of Previously Created Materialised Views. Query Rewriting Is The Rewrite Of The Query, Depending On Information From The Query Containment Control, Query Rewriting In Different Query Languages, Such As (Selection, Screening And Joining) Queries And Aggregated Queries [5]. This Is Very Helpful In A Data Storage Context As It Provides A Method For Seamlessly Enhancing End User Or Database Applications By Rewriting The Sql Query Using A Materialised View Instead Of Querying The Table [10]. The Mvpp Tree Is Taken As The Input And The Output Is Generated As A Sequence Tree Of Views And Tables To Answer Queries [8].

The Proposed And Designed System

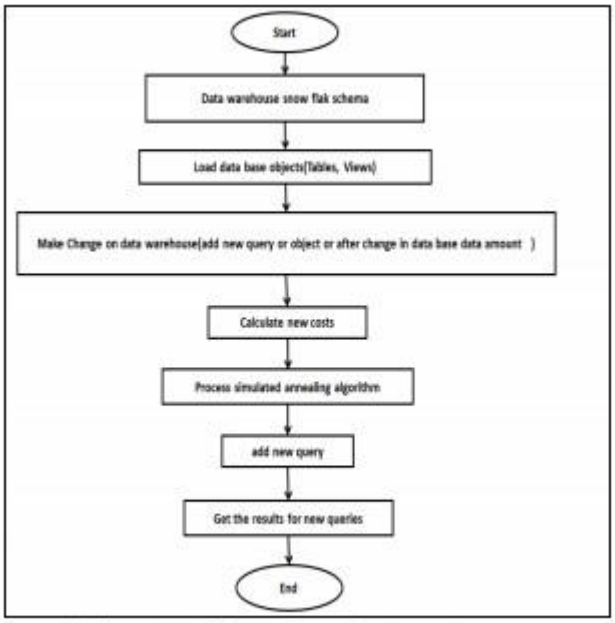

Figure 1.The proposed dynamic warehouse system.

In This Paper, We Proposed A Dynamic Warehouse Design And It Consists Of Several Phases As Shown In The Figure 1. These Phases Are: 1. Using Data Warehouse Snow Flake Schema. 2. Create Views. 3. Loading Data Base Objects Views And Tables. 4. Make Change On Data Warehouse (Add New Query Or Object After Change In Data Amount). 5. Calculate New Costs. 6. Process Simulated Annealing Algorithm. 7. Add New Query. 8. Update Views Data. 9. Getting Result For New Query.

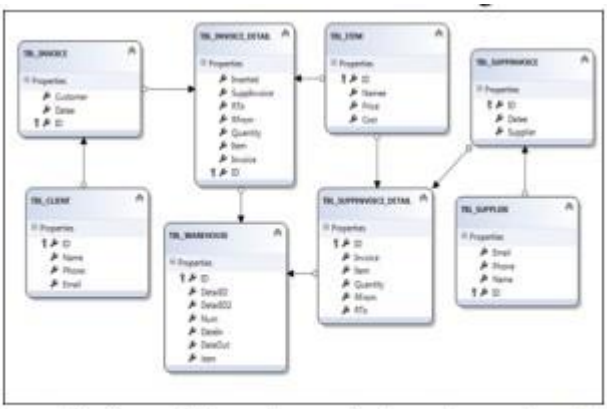

Figure 2. Snow flake schema of the sales system data warehouse.

ISSN: 2233-7857 IJFGCN

As A Case Study For This Stage Of Sales, The First Step Of The Proposed System, Which Employs A Data Warehouse Snowflake System, Is 4.5 Gb In Size With One Fact Table And Seven Dimensional Tables (Posts, Customers, Client Factors, Invoice Details, Suppliers, Supplier Invoices, Supplier Invoices), The Relationship Between Fact And Data

Firstly, Get A List Of Customers Who Purchased A Certain Item (Directly From Base Tables Or By Materialised Views). 2. The Second Question Is To Get A List Of Vendors That Have Sold More Items (Directly From Base Tables Or By Materialised Views). 3. The Third Question Is To Get A List Of Seasonally Bestselling Goods (Directly From Base Tables Or By Materialised Views).

If The Query Is Performed Using The Base Table Alone, The Query Is Lengthy Instead Of Running Utilising Materialised Views As Shown In Table (1).

Table1. Execution time of query.

No. of query	Using Materialized Views	Using Base Table in data warehouse
1	1.512	10.875
2	2.751	14.325
3	0.881	12.920

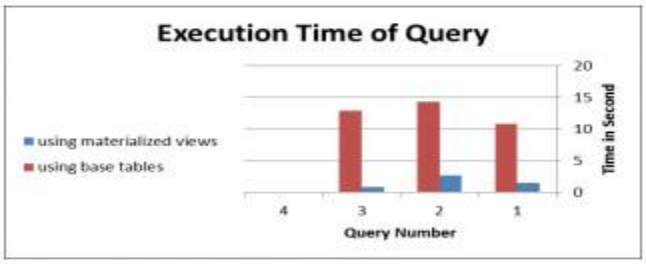

Figure 3. Difference query execution time.

Mvpp, As Illustrated In Figure 4, Was Utilised To Create Query Space, As In [9]. In This Image, You Create A Query Execution Plan Based On Database Table Selection, While In Figure 5, You Create A Query Execution Plan Based On Materialised Visual Data Selection, Which Was Replaced Instead Of Basic Tables.

For Each Individual Query, There Is A Local Access Plan, Which Is Fused Into Common Data Sets Based On Shared Processes. The Search Results Are Collected From The Search Space Levels, Beginning At The Base Level To The Top Level, Through Operations Between The Hierarchies (Box Down Or Roll Up). Views Are Manually Constructed By Choosing The Active Lines From The Base Table And Using Active Methods For Updating Tables And Obtaining Tables Of Views. The Aim Is To Constantly Maintain Materialised Views In Order To Manage Jobs From The Windows Scheduler By Running Batch Files. The Programmer Thus Retains Time And Completes The Tables.

ISSN: 2233-7857 IJFGCN

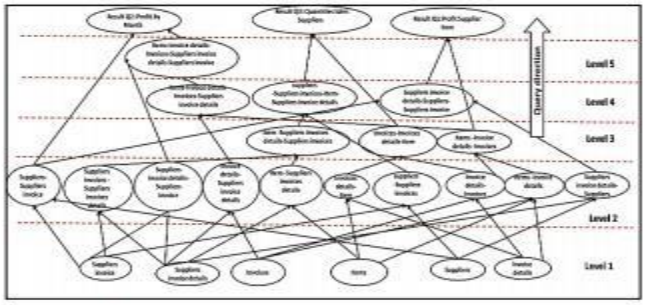

Figure 4. Lattice illustrates the levels of query optimization before using MV.

As Shown In The Preceding Model, The Number Of Nodes Decreases When Mv Is Used Instead, So That The Search Space Will Be Reduced, Leading To A Reduction Of Query Time. The Manual Procedures For Substituting View Instead Of Tables Will Take A Long Time And The Query May Not Be Prepared In Precise Form To Introduce And Apply The Query Writeback Or Query Optimization Operation.

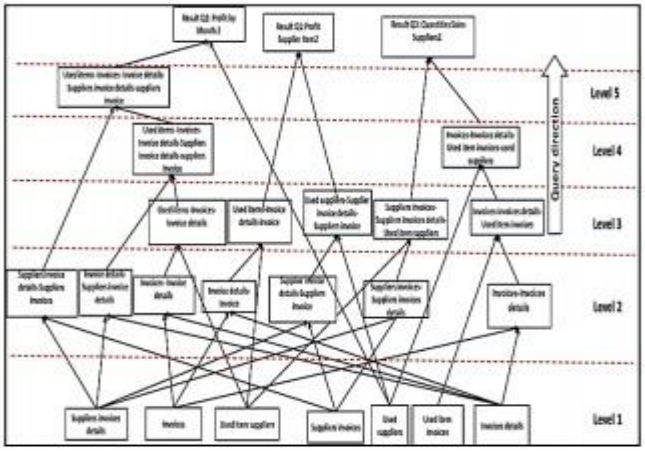

Figure 5. Lattice illustrates the levels of query optimization after using MV.

In Some Situations The Reporting Methods In The Database Must Be Updated To Alter Query Writings, Thereby Modifying The Query Execution Plan And Improving The Query Performance. In Certain Reports, Speed Is Not Even Accessible Without The Improved Query. The Situation In Which The Query Writing Technique Has To Be Upgraded Is As Follows: • Adding To The Database New Materialised Views. • Collect Additional Records Affecting The Operating Speed Of The Existing Database.

An Application Must Thus Have The Capacity, For The Sake Of Determining The Tables For Which The Query Or The Charges For Calling Data From Materialised Views, To Access The Information Required To Provide New Queries. • The Cost Of Obtaining All Rows Off The Table Is The Information That The Warehouse Upgrade Programme Acquires. • The Materialised Views Representing Tables And Tables That May Be Replaced. • The Tables' Connections And The Materialised Views. • The Required Objects For Each Query. All Of This Information Is Therefore Included In The Application Classes And Interfaces That The User Who Should Be A Database Manager Or Person Responsible For Managing The Database Management System Would Obtain. The Data Was Also Directly Obtained From The Database.

Classes (Query, Join, Table) Provide The Characteristics Necessary To Determine The Optimal Layout Of Objects On The Query. The Primary Difficulty In The Study Has Been To Identify The Calculable Form For Tables Inside The Query So That Artificial Intelligence Algorithms May Be Applied To That Query.

ISSN: 2233-7857 IJFGCN

Annealing Is Simulated In Order To Determine The Optimum Formulation Of Queries In Each Query. Each Query Is Entered As Tables That Are Essential For Working With The Query. However, Tables Which Can Be Replaced By Tabulations May Be Introduced To The System As Materialised Views. It Will Therefore Look As The Same Table In The Computations, But With Improved Performance. Simulated Rinsing Is A Technique Based On Which Unrestricted And Boundary Optimization Issues May Be Resolved. The Technique Simulates The Physical Process Of Heating A Material And Reduces The Temperature Gently, To Reduce Flaws, Reducing The Energy Of The System. Therefore, The Present Performance Counter Requires Two Factors, While The Other Is The Factor That The Performance Counter Decreases. This Is A Looping Form For A Big Quantity That Allows Systems With Uncertain Probabilities To Achieve Stability Or The Optimal State. In Our Instance, We Evaluate The Cost Of The Question, Which Is Generated Every Time, To Determine If The Case Is Appropriate Or Not. A Swap Operation Is Carried Out In Every Loop On A Random Table Of The Query And The Query Is Verified To Make It Adequate And Compliant With The Original Query Requirements And With Or Without All The Tables Within It. And Then The Costs Are Checked. The Query Will Reach The Optimal State After Hundreds Or Thousands Of Possibilities. Table 2 Shows The Cost Of Each Table, Expressed (Rpt) By View (Tbl) By Table.

Implementation And Results

The System Suggested And Developed Was Implemented Using (Visual Basic.Net 2013 And Sql 2008 R2 Express Programming Languages). The System Is Thus Implemented In Stages. The System Provides An Interface For Easy Execution And Supports The Management Or Decision Maker To Create Queries. Tables And Views Are Entered In The Query Optimizer System While The Output Is The Optimal Tables And Views For Building A Query, E.G.: Table1, Table2, Table3 Input Table1 Output, View1, Table3.

Table 2: Execution time of tables and views

Tables and views	Average work time (mille Sec)	Average elapsed time (mille Sec)
RPT_CLIENTS	227010	243490
RPT_ITEM_SUPPINVOICE	29723	56250
RPT_ITEMS_CLIENTS	12581	12582
RPT_ITEMS_INVOICES	17498	23395
RPT_ProfitSuppItemDate	9164	24874
RPT_VENDOR	11134	18100
RPT_VENDORS_ITEMS	253	253
RPT_WAREHOUSE_INSERTED	346667	465582
TBL_CLIENTS	191193	302015
TBL_INVOICE_DETAILS	330294	488234
TBL_INVOICE	13084	234097
TBL_ITEMS	18995	44850
TBL_RANGES	266858	410311
TBL_SUPPINVOICE_DETAILS	447351	628555
TBL_SUPPINVOICE	66769	91763
TBL_SUPPIERS	33443	81064
TBL_WAREHOUSE	33443	81064

Conclusions

Following Installation Of The Suggested Design Dynamic Data Warehouse System And The Performance Of The Proposed Algorithm, The Simulated Warehouse Upgrade Annealing Method Has Been Used To Choose The Best Solution Of Many Options And Results Obtained:

ISSN: 2233-7857 IJFGCN

- Data Base Used In Different Field Of Works, Due To The Multiple And Different Using Of Data Bases, It Needs To Enhance Operations To Change Its Structure In Terms Of Queries Or Indexes To Speed Up And Increase The Efficiency Of Paging Operations.
- The Most Common Way To Improve Database Performance Is To Change The Way Of Writing The Query.
- An Artificial Intelligence Algorithm On One Hand Is Needed And On The Other Hand Is A Simple Way Of Writing The Query And Delivers Them With A Meaningful Manner, So That The Proposed System Can Find A Better Way To Rewrite The Queries.
- Simulated Annealing Is An Artificial Intelligence Algorithm That Is Highly Efficient To Find A Stable Solution From Among Multiple Options For Solutions.
- When Using Materialized Views For Answer Queries, Time Will Decrease In 7 Percent When Using Base Tables.

Future Work

In The Future, We Plan To Expand Our Approach To Include Parallel Simulated Annealing In Order To Deal With Even Larger Sets Of Views And Gain Further Improvements In Solution Quality.

References

[1] Payal Pahwa, Rashmi Chhabra, “An Object Oriented Data Warehouse Design”, International Journal Of Soft Computing And Engineering (Ijsce), Iv (2), Pp. 2231- 2307, 2014.

[2] Jin-Hyuk Yang, In-Jeong Chung, “Asvmrt: Materialized View Selection Algorithm In Data Warehouse”, International Journal Of Information Processing Systems, Ii (2), Pp. 57-75, 2005.

[3] Jing Hu, “Optimizing Queries Using A Materialized View In A Data Warehouse”, College Of The Oklahoma State University, Oklahoma, Usa, 2005.

[4] William Inmon, Derek Strauss, Genia Neushloss, The Architecture For The Next Generation Of Dw, Genia Neushloss, 2008.

[5] Songting Chen, “Efficient Incremental View Maintenance For Data Warehousing”, Thesis Of A Doctor Degree In Computer Science, 2005.

[6] Sergio Lujan, Juan Trujillo, "Acomprehensive Method For Data Warehouse Design", In Proceedings Of 5th International Workshop On Design And Management Of Data Warehouse(Dmdw’03) , Pp. 1-14, 2003.

[7] Phan Quoc Nghia, “Building Olap Application To Exploit Database Of Rice Pests”, International Journal Of Emerging Technology And Advanced Engineering, Iii (11), Pp. 130-137, 2013.

[8] Mukesh Mohania, Guozh Dong, “Algorithms For Adapting Materialised Views In Data Warehouses", International Symposium On Cooperative Database Systems For Advanced Applications, 2000.

[9] Roozbeh Derakhshan, Frank Dehne, Othmar Korn And Bela Stantic “Simulated Annealing For Materialized View Selection In Data Warehousing Environment ", Springer, Pp.89-94, 2005.

[10] Bilal Adil Mahdi," Design And Implementation Of Materialized Views Tool For Data Warehouse Structure", Thesis Of A Master Degree In Computer Science, 2013.

ISSN: 2233-7857 IJFGCN