A New Viewpoint to Database and Data Using Physics Concepts

Rasoul Kiani[1] and Salman Zivari[2]

[1] Department of Computer Engineering, Islamic Azad University, Fars Science & Research Branch
Shiraz, Fars, Iran

[2] ProSoftGen Company
Torbat-e- Heydarieh, Khorasan Razavi, Iran

Abstract

A modern tool has been introduced in this paper to researchers by definition of new concepts and approaches in order to optimize performance of databases. Although this is not going to argue that optimization of concepts like a better accessibility is impossible by the available tools, the main purpose is rather to conform physics concepts based on relational databases, and also to present some other concepts, definitions and formulations for making the data further accessible in all databases including centralized and distributed ones.

Keywords: *Accessibility, Optimizing, Data, Database, Physics Concepts.*

1. Introduction

The main purpose of physics is studying and analysis of the nature. It has always been trying to understand and predict the nature's behavior in different conditions. Since the physics rules and mathematical relationships make each other perfect, the physics has become able to describe various problems.

Taking into account the ever increasing application of the databases and importance of the data concept in these current days, optimization of the data storage and recovery seems to be one of the substantial features which are very interesting for computer scientists. After first introduction of the database concept in 1960s for solution of the growing problems of design, construction and maintenance of the information systems, different models including hierarchical and network models were proposed for designing the database. A new model called "relational model" was developed in 1970 by a mathematician named Codd.

However, there are two approaches about the future: a number of researchers believe that the above mentioned era will finally end and some other models will such as object-oriented models be replaced instead. On the other hand, some other researchers believe that the relational model is based on propositions, thus as the proposal calculus has lasted since thousands of years ago, the relational model will remain as the database model for ever as well.

This paper aims to use the physics concepts to take a look at the databases and data based on the relational model and develop new concepts or features for them to acquire a correct understanding of the resultant knowledge. Thereby, it can be used to improve and optimize the problems of this field. For example, the concept of temperature will be defined for the databases and data.

2. Related Works

2.1 Constrain Dynamic Physical Database Design

As discussed by Hannes Voigt, Wolfgang Lehner and Kenneth Salem, who investigated to Constrain Dynamic Physical Database Design, "Physical design has always been an important part of database administration. Such advisors generally view database physical design as a static problem. Given a set of queries and updates describing the database workload, plus a storage capacity constraint, a design advisor recommends a set of physical database structures, such as indexes and materialized views, that will minimize the cost of executing the workload.

Some researchers have proposed dynamic, on-line approaches to the physical design problem. For example, Bruno and Chaudhuri [1] model the database workload as a sequence of queries and updates, and they propose a mechanism that monitors the workload and continuously adjusts the database physical design based on the queries and updates that it has observed so far. This is appealing because it attempts to automatically adjust the database physical design to account for changes in the workload over time.

Voigt, Lehner, Salem consider a dynamic, off-line version of the physical design problem. They are given, in

advance, a description of the database system workload consisting of a sequence of queries and updates, as well as a storage capacity constraint. Essentially, Their goal is to recommend a series of physical designs which will result in efficient execution of the workload. The input workload is a sequence of n queries and updates, and the output is a sequence of n physical designs, one for each statement in the workload. This is an ideal formulation for situations in which the given query sequence represents an exact characterization of the expected database system workload.

They are given as input a sequence $[S_1, S_2, \ldots, S_n]$ of SQL statements. Their goal is to choose a sequence of physical designs $[C_1, C_2, \ldots, C_2]$, where C_i is the physical design that will be used for the execution of S_i. A physical design consists of a set of structures (e.g., indexes or materialized views) chosen from a set of candidate structures. They use EXEC(S_i,C_i) to denote the cost of executing statement S_i under physical design C_i, and they use TRANS(C_i,C_j) to denote the cost of changing the physical design from C_i to C_j ."[2]

$$\sum_{i=1}^{n} EXEC(S_i, C_i) + TRANS(C_{i-1}, C_i) \quad (1)$$

2.2 Shortest Path Computing in Relation DBMSs

As discussed by Jun Gao, jiashuai Zhou, Jeffry Xu Yu and Tengjiao Wang, who investigated to Shortest Path Computing in Relation DBMSs, "they make substantial extensions to further improve the scalability and performance of the relational approach. They introduce a weight aware edge table partitioning schema, and design a restrictive BFS strategy over partitioned tables. The strategy can improve both the scalability and performance significantly without needing extra index construction and space overhead.

Thiers paper study weighted (directed or undirected) graphs. Let G = (V,E) be a graph, where V is a node set and E is an edge set. Each node v ∈ V has a unique node identifier. Each edge e ∈ E is represented by e = (u,v), u,v ∈ V. W(e) is the weight for the edge e. The logical relational schema for a graph is illustrated in Fig. 1. Let G = (V, E) be a graph. TN table is to represent nodes V . They can record nid for the node's identifier and other attributes in TN table. TE table is to store edges E. For an edge (u,v) ∈ E, the identifiers of node u and v, as well as the weight of the edge, are recorded by fid, tid and cost field respectively. As this paper studies the shortest path discovery, only TE table is used in the following operations.

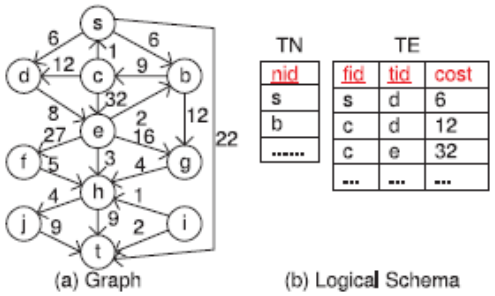

Fig. 1 Relational representation of graph.

They abstract three key operators, namely F, E and M-operator, and then provide a generic graph searching framework FEM. They find new features, such as window function and merge statement introduced by recent SQL standards, can simplify the expression and improve the performance."[3]

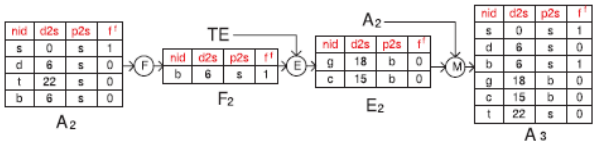

Fig. 2 F, E and M-operator in the second iteration.

Definition 1 (F-operator). *$F(A_i) \leftarrow \sigma_{nid=mid} A_i$ returns frontier nodes F_i from visited nodes A_i in the ith expansion.*

Fig. 3 Definition F-operator.

Definition 2 (E-operator). *$(F_i)E(\mathrm{TE})$ returns the expanded nodes E_i based on the frontier nodes $F_i(r, d2s, p2s, f^f)$ and $\mathrm{TE}(r, x, w)$ table in the ith expansion.*

1. $minCost(x, c) \leftarrow_x \mathcal{G}_{min(d2s+w)}\Pi_{(x,d2s,w)}(F_i(r, d2s, p2s, f^f) \bowtie TE(r, x, w))$;
2. $E_i \leftarrow \Pi_{(x,d2s+w,r,0)}\sigma_{c=d2s+w} minCost(x, c) \bowtie F_i(r, d2s, p2s, f^f) \bowtie TE(r, x, w)$;

(r, x, w) in TE *table is for an edge from r to x with its weight w.*

Fig. 4 Definition E-operator.

Definition 3 (M-operator). *$(E_i)M(A_i)$ returns visited nodes A_{i+1} based on expanded nodes $E_i(x, d2s_e, p2s_e, f_e^f)$ and visited nodes $A_i(x, d2s_a, p2s_a, f_a^f)$ as follows:*

1. $A_i \leftarrow A_i - \Pi_{x,d2s_a,p2s_a,f_a^f}(\sigma_{d2s_e < d2s_a} (E_i(x, d2s_e, p2s_e, f_e^f) \bowtie A_i(x, d2s_a, p2s_a, f_a^f))$;
2. $E_i \leftarrow E_i - \Pi_{x,d2s_e,p2s_e,f_e^f}(\sigma_{d2s_e > d2s_a} (E_i(x, d2s_e, p2s_e, f_e^f) \bowtie A_i(x, d2s_a, p2s_a, f_a^f))$;
3. $A_{i+1} \leftarrow A_i \cup E_i$;

Fig. 5 Definition M-operator.

2.3 Automate Schema Design for Large Scientific Databases Using Data Partitioning

As discussed by Stratos Papadomanolakis and Anastassia Ailamaki, who investigated to Automate Schema Design for Large Scientific Databases Using Data Partitioning, "To optimize performance, database researchers have proposed data placement and partitioning schemes [4][5]. Vertical partitioning is known to optimize I/O performance since the early days of relational databases [6].

They propose AutoPart, an algorithm that automatically partitions database tables utilizing prior knowledge of a representative workload. AutoPart suggests an alternative, high-performance schema that executes queries faster than the original one and can be indexed using a fraction of the space required for indexing the original schema. To evaluate AutoPart, they build an automated schema design tool that interfaces to commercial database systems."[7]

```
invoke categorical_partitioning(R,Q,N)
/* Schema PS is the best partial solution so far */
/* Compute atomic fragments */
1. schema PS := AF
/*Composite fragment generation*/
2. for each composite fragment F ∈ SF(k-1)
   2.a E(f) := {composite_fragments (F,A∈AF) ∪
               composite_fragments (F, A ∈ AF) having
               query extent > X }
   2.b CF(k) := CF(k)∪ E(f)
/*Composite fragment selection */
3. for each composite fragment F ∈ CF(k)
   3.schema S_F := add_fragment (F,PS)
   3.b if size(S_F) > B then continue with the next F
   3.c compute cost(S_F, Q)
4.select F_min = arg_max (cost (S_F, Q))
        with cost (S_Fmin, Q) < cost (PS, Q)
5. if no solution was found then goto 9 /* exit */
6. PS := S_Fmin
7. SF(k) := SF(k) ∪ F_min
8. remove F_min from CF(k)
9. repeat steps 3-8
/* proceed with next iteration*/
10. k++
```

Fig. 6 The AutoPart algorithm.

2.4 Schema Transformation – A Quality Perspective

As discussed by TauqeerHussain, ShafayShamail and Mian M. Awais, who investigated to Schema Transformation – A Quality Perspective, "Conceptual modeling is one of the most demanding and challenging steps in database design methodology. The success of a software system depends upon the quality of a conceptual model that can be determined by how close and how accurate it represents the problem domain.

A schema transformation is a function that maps a conceptual schema to another schema. The other schema can be at the same conceptual level (expected to be with improved quality) or at a lower level typically at logical level where it usually gets transformed to a relational schema. They have proposed two quality metrics, Completeness Index and Normalization index, to measure the quality of the conceptual model that results after applying the proposed rules."[8]

$R = \{R_i \mid$ Every R_i is a relation obtained by transforming a given ERD, and $i = 1,2,3, \ldots, n\}$

w_j = weight arbitrarily assigned to j^{th} normal form where $w_k > w_l \; \forall \; k > l$, that is,

$$w_{BCNF} > w_{3NF} > w_{2NF} > w_{1NF}$$

$NV(R_i) = w_j$ = Normalization Value of Relation R_i which is in j^{th} normal form

then, $$NI = \sum_{i=1}^{n} NV(R_i)$$

Fig. 7 Normalization Index.

$\Im$ be a set of functional dependencies identified from a problem domain,

ξ be a given ERD , and

f be the projection set of $\Im$ on ξ, i.e. $f = \pi_{\Im}(\xi)$

$n(\Im)$ = total number of functional dependencies in $\Im$

$n(f)$ = total number of functional dependencies in f

then $CI = n(f) / n(\Im)$

Fig. 8 Completeness Index.

2.5 Transform Relational Model to Source Ontology for Data Integration

As discussed by Kiran Sonia and Sharifullah Khan, who investigated to Transform Relational Model to Source Ontology for Data Integration, "Data integration provides a uniform access to a set of data sources, through a unified representation of data called global schema. Generating global schema requires exact understanding of data held in sources to resolve the heterogeneity among them [9].

Conceptual modeling of the sources can provide semantic understanding of different sources because a conceptual model is at a higher level of abstraction [10], [11], [12]. The examples of conceptual models are entity relationship diagram (ERD), extended entity relationship diagram (EERD), and ontology [10, 11, 12, 13]. A conceptual model can represent the sources at the same level of understanding. One way to integrate the sources into a scalable and flexible system is to represent source descriptions, i.e., metadata of the sources, in a conceptual model in order to resolve their heterogeneity [11,14].

Their research focuses on generating the source model of a data source and representing them into a conceptual model that is ontology to enable sources to integrate into a scalable, flexible and interoperable integration system. The methodology has two Steps; (Step 1) to extract metadata from database relations and (Step 2) to provide transformation rules for transforming metadata to ontology. The technique will be evaluated for correct identification of metadata."[15]

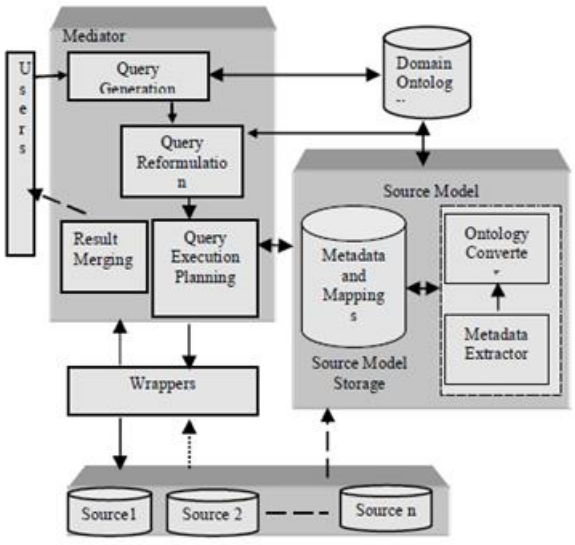

Fig. 9 Ontology building Architecture.

3. Concepts

This section will investigate a set of definitions and concept formulations in order to develop a tool for fulfillment of the objectives of this paper.

3.1 Material

Definition 1 – Material of Database: Any database can be a material, the constituents of which are data.
Definition 2 – Material of Data: Any data can act as a material, with its constituents being data contents.

3.2 Kinetic Energy

Definition 3 - Kinetic energy of data: Count of requests to the certain data.
$$Kdata = Access\ To\ Data \quad (2)$$

Proposal: Since requests of the database is examined and translated by the Query Processor, the initial data of Kdata for each newly created data can be considered as zero and each request adds one unit to the related Kdata after translation.
Definition 4 - Kinetic Energy of Database: Average kinetic energy of the content data of that database.
$$Kdatabase = (\sum Kdata) / n\ records \quad (3)$$

3.3 Potential Energy

Definition 5 - Potential Energy of Data: Probability of record request due to access to the content of this data.
$$PEdata = P(Access\ To\ Record) \quad (4)$$

The initial potential energy of each data is considered as zero. In T period of time after the last access, m units are deducted from the Kdata and the same is added to the PEdata.
Definition 6 - Potential Energy of Database: Probability for requests which are received by a database.
$$PEdatabase = P(Access\ To\ this\ Database) \quad (5)$$

The initial potential energy is considered zero in the centralized database.
$$PEcentralized\ Database = 0 \quad (6)$$

The initial potential energy in the distributed database within T period of time after the last access is deducted for m units from the Kdatabase and added to the PEdatabase.
NOTE (1): The values of T and m can both be simulated.

3.4 Temperature

Definition 7 - Temperature of Data: The temperature of a data which is never requested is defined as zero and when the temperature is increased followed by each request, the temperature of data will be decreased within time periods if not used.

$$Tempdata \propto Num\ of\ Request\ in\ Time \quad (7)$$

Definition 8 - Temperature of Database: Average temperature of the content data of that database.

$$Tempdatabase = (\sum Tempdata) / nField\ of\ Record \quad (8)$$

NOTE (2): The temperature is directly related to the kinetic energy.
NOTE (3): Raising the temperature will increase the kinetic energy of the material particles.
NOTE (4): The location for storage of the temperatures of data can be ant place of the memory.

Challenge: For conversion of Temp $\propto$ K to Temp = hK, it is necessary to calculate the h correlation coefficient.

3.5 Heat

Definition 9 – Heat of Data: The variation in the temperature of data within Δt period of time is called the transferred heat of data in that certain period of time.

$$Qdata = \int Tempdata\ dt \quad (9)$$

Definition 10 – Heat of Database: Total heat of data which is transferred via the data.

$$Qdatabase = \sum Qdata \quad (10)$$

3.6 . Latent Heat of Fusion

Definition 11 - Latent Heat of Fusion: The amount of positive heat transferred by data in order to decide to make it further accessible. For instance, it may be located in the cache memory. Calculation of the relative or accurate values for this section needs simulation as the exposed challenges in the way of the newly developed systems.

$$QHdata = ? \quad (11)$$

Definition 12 - Latent Heat of Database: Average latent heat of all data.
NOTE (5): This node must be more accessible in these conditions for the distributed systems.

$$QHcentralized\ Database = 0 \quad (12)$$

$$QHdistributed\ Database = (\sum QHdata) / n\ Record \quad (13)$$

3.7 Latent Heat of Solidification

Definition 13 - Latent Heat of Solidification: The definition of this section is similar to the latent heat of fusion but this time with a negative heat. In this case, the system has the possibility to quickly cancel the data access. Although this is not mandatory, but a suitable candidate data for transition and input of the other data for being accessible, is the one which has already reached the latent heat of solidification.

3.8 Heat Transfer or Convection

Convection is one of the ways of heat transfer.
Definition 14 - Heat Transfer in Database: The data with a higher temperature gradually approach a better accessibility and they are replaced with the data of a lower temperature.

3.9 Effect of Heat in Adjacent Data

Definition 15 - Effect of Heat in Adjacent Data: In a database, the related data (related tables) receive some of the heat from the primary data based on a conceptual design (logical layer). Therefore, a better accessibility of the primary data will make even the secondary data further accessible (with a smaller coefficient). The same can be obtained vice versa: the secondary data can render the primary data accessible.

4. Conclusions

This paper proposed a new approach to data and database using the physics concepts, in addition to studying the mutual effects of physical and conceptual designs on each other.
One of the advantages of this modern approach is using the science of physics and conformation of it based on the concepts of data and database, beside definition and formulation of them.
However, it should be noticed that all these attempts have been made to provide the researchers with a different tool in order to use it for optimization of the related problems in this field.

References

[1] N.Bruno, and S.Chaudhuri, "An Online Approach to Physical Design Tuning ", In Proceedings of the 23nd International Conference on Data Engineering, Istanbul, Turkey, 2007.
[2] H.Voigt, W.Lehner, and K.Salem, "Constrained Dynamic Physical Database Design",IEEE,2008.

[3] J.Gao, J.Zhou, J.Xu Yu, and T.Wang, "Shortest Path Computing in Relational DBMSs", IEEE Transactions on Knowledge and Data Engineering,2014.
[4] Copeland G.P, and Khoshafian S.F., "A Decomposition Storage Model", SIGMOD, 1985.
[5] Zhou J., and Ross K.A, "A Multi-Resolution Block Storage Model for Database Design", Proceedings of the 2003 IDEAS Conference, 2003.
[6] Teorey T, and Fry P.J., "The Logical Access Record Approach to Database Design", ACM Computing Surv. 12(2):179-211(1980).
[7] S.Papadomanolakis, and A.Ailamaki, "Automating Schema Design for Large Scientific Databases Using Data Partitioning", International Conference on Scientific and Statistical Database Management, IEEE,2004.
[8] T.Hussain, and S.Shamail, "Schema Transformation - A Quality Perspective ",IEEE,2004.
[9] Schuette, and Rotthowe, "The Guidelines of Modeling - An Approach to Enhance the Quality in Information Models", Proceedings of the 17th hernationalConference on Conceptual Modeling , 1998.
[10] D. Yeh, and Y. Li, "Extracting Entity Relationship Diagram from Table Based Legacy Database", Proceedings of the ninth conference on software maintenance and reengineering (CSMR'05) , 2005.
[11] P. Johannesson, "A Method for Translating Relational Schema to Conceptual Schemas", In Proc. Of the international Conference on Data Engineering, 1994.
[12] H. E. Ghalayini, M. Odeh, and R. McClatchey, "Engineering Conceptual Data models from Domain Ontologies: A Critical Evaluation", 4th International Conference on Computer Science and Information Technology (CSIT'06), Amman Jordan, 2006.
[13] R. Alhajj, " Extracting The Extended Entity Relationship Model From a Legacy Relational Database", Information systems Conference, 2003.
[14] S. Khan, and F. Morvan,"Data Integration in Distributed Biomedical Sources", In proceedings of ISCA 19th International Conference on Parallel and Distributed Computing System, California, USA, 2006.
[15] K.Sonia, and S.Khan," Transforming Relational Model to Source Ontology for Data Integration ", International IEEE Conference "Intelligent Systems",2008.

Rasoul Kiani received the B.S degree from Torbat-e- Heydarieh University, Iran, in 2012. Currently, he is a student of Software Engineering, M.S degree, in Islamic Azad University of Fars science and research branch, Iran. His major research interests include data mining and database system implementation.

Salman Zivari received the B.S degree from Torbat-e- Heydarieh University, Iran, in 2013. Currently, he is a developer and analyzer at ProSoftGen company, Iran. His major research interests include games engine programming, database system implementation and web data management.