A Parametric Approach in Real-time Datawarehousing

Imane LEBDAOUI, Said EL HAJJI

***Abstract*— Since decision making processes become more demanding of the freshest information, decision support systems have to handle and deliver information, when needed, as fast as possible. Unfortunately, even in real-time datawarehouse environment, many constraints and parameters, still, are tackling decision making processes causing latency. This paper presents a parametric approach that dynamically balances between system resources and real-time datawarehousing requirements.**

***Index Terms*— Datawarehouse, Real-time datawarehousing, latency, parametric approach.**

I. Introduction

A datawarehouse (DW) integrates data that come from independent heterogeneous operational data-sources (ODS) and creates a single view of the organization. Once in the DW, data is turned into structured information that is easily handled by decision making processes.

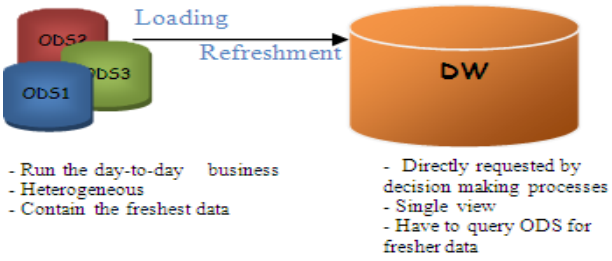

Fig. 1. Macro view of datawarehouse solution

The main components of a DW solution are: Operational data-sources, the DW and Extraction-Transformation-Loading (ETL) tools. Besides, techniques of Change Data Capture (CDC) catch just changed data from ODS. Then, these changed data move inside the next DW components and undergo, inter alia, extraction transformation, integration, cleansing and loading. This movement generates, in each step, additional delays: Delays in capturing real-world events and delays in loading and integrating data into the DW [2]. Moreover, if systems resources are limited, delays and performance are influenced accordingly.

Because of the new business requirements for ever up-to-second updated information and since timely data ensures better-informed decisions [4], real-time datawarehousing (RTDWg) is one of the savior trends that provides an access to accurate, integrated, consolidated view of the organization's information [1] in real-time.

Our bibliographic study reveals that there still is no published work on identifying the best balancing between system resources and RTDWg requirements. This is the basis of this paper that presents a new parametric approach insuring to get the maximum of both freshness and accuracy.

The remainder of this paper is organized as follows: Section 2 presents the concepts of RTDWg. Section 3 gives problem statement. In section 4, we introduce the new parametric approach. Section 5 discusses preliminary experiments. Finally, we give conclusion and future work in section 6.

II. Real-time Datawarehousing Concepts

A. Notion of Real-time Data Warehousing

Unlike traditional DW, Real-time DW (RTDW) must be continuously updated whenever a data change is made at ODS. RTDWg aims decreasing the time it take to make decisions and try to attain the zero latency [1], low or no latency [4], between cause of change and its effect. It can be defined as "online DW", "active DW" [1] or "dynamic DW" [2].

The just-in-time datawarehousing consists on delivering information to decision making systems just before they are needed. In this sense, it can be considered like RTDWg when information is needed in real-time.

A RTDW has to process a huge amount of data in real-time basing on real-time systems (Real-time ETL and Real-time CDC [4]) and requires continuous activities, like continuous data integration, that enables the DW to cope with real-time requirements, in order to deal with the most recent business data [1].

B. Categories of Data in RTDWg

RTDWg environment, not every data is important to handle in real-time, because there is important data and not very important data.

In relation to this importance, we can distinguish between static and dynamic data.

- *Static data* keep one value and are rarely changed;

Manuscript received March 16, 2013; revised April 06, 2013.

Imane LEBDAOUI, Laboratory of Mathematics, Computing and Applications, Faculty of Sciences, University of Mohammed V–Agdal, BP.1014 RP. Rabat, Morocco (e-mail: imane.lebdaoui@gmail.com).

Said EL HAJJI, Laboratory of Mathematics, Computing and Applications, Faculty of Sciences, University of Mohammed V–Agdal, BP.1014 RP. Rabat, Morocco (e-mail: elhajji@fsr.ac.ma).

ISBN: 978-988-19252-9-9
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)

- *Dynamic data* get many values and are changed many times. We identify:
  - *Real-time (RT) data* whose changed value must be replicated into the DW in real-time;
  - *Non real-time (NRT) data* are data that even changed, it is not necessary to replicate their value into the DW in real-time.

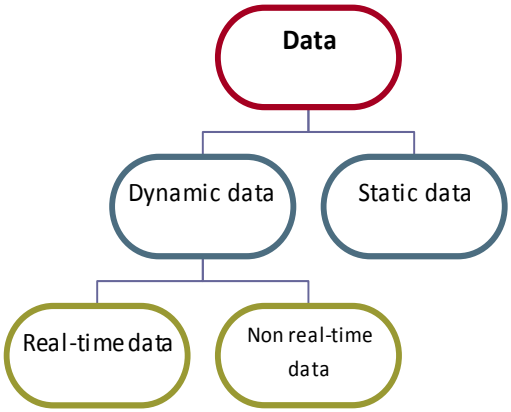

Fig. 2. Categories of data in RTDWg

C. *Real-time Datawarehousing requirements*

Requirements are things we must live with and adapt our system to them. In RTDWg, the main requirements consist on processing data changes in real-time fashion while insuring two principal goals: Freshness and Accuracy.

- **Freshness** is related to data latency; data is fresh when its latency equals zero otherwise; it is stale and serves just for historical purpose. One other definition of fresh involves how quickly the data is delivered to end users. Data freshness should be driven by organization requirement not by technologies power.
- **Accuracy** is one component of quality. Sometimes, it means integrity. Data is accurate if it correctly reflects the real world object or an event being described [3]. Since data quality in any database varies over time and business rules also change, the level of requiring accuracy may vary as well [5].

Freshness implies that the DW must integrate change data with no or low delay.

Accuracy, however, supposes several operations to guarantee that the integrated change is correct and was appropriately cleansed and checked.

Consequently, the issue is how to get the maximum availability of fresh data in the DW while guaranteeing data accuracy.

III. PROBLEM STATEMENT

In our model, we consider parameters like: Data size, data type, ODS type, ETL configuration, storage space, CPU…

Problem statement. Given a set of n-parameters $\{p_1, p_2, \ldots, p_n\}$ which belongs to DW solution components, our model must be able to:

- distinguish between real-time parameters and non-real-time parameters;
- identify real-time and critical parameters ($RTCp_i$);
- measure, regularly, the rate $r = 1-t_1/t_2$ where t_1 symbolizes the time when data change occurs in the operational source side and t_2 is the time when this change is loaded into the suitable table in the DW. This rate is only measured for real-time parameters;
- identify the best value (threshold t_h) of each $RTCp_i$ for which r is *minimal* (according to organization's requirements on RTDWg) and data accuracy is *maximal*.

We denote requirements that we have mentioned in the section 2, by the couple (F, A) where F is Freshness, A is Accuracy. Thus, our model must be able to balance between (F, A) in one side and system resources (parameters pi) in the other side.

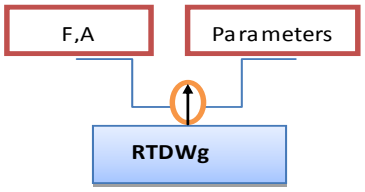

Fig. 3. Balancing between requirements and resources in RTDWg

For example, in case of two parameters, our model must be able to solve the following equations:

$$x_1.p_1 + x_2.p_2 = F \quad (1)$$
$$x'_1.p_1 + x'_2.p_2 = A \quad (2)$$

Depending on the components of the action time of Richard Hackathorn, the majority of latency is due to the initial data acquisition and its delivery to the warehouse.

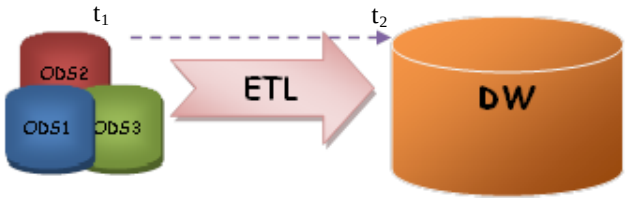

Fig. 4. Perimeter of the new model

For this reason, we limit the scope of our work inside the interval $[t_1, t_2]$; t_1 is the instant when a change occurs in the ODS and t_2 is the moment when the change incorporates DW. Our model considers data movement in the direction indicated by the arrow in Figure 4.

IV. PARAMETRIC AND ALGORITHMIC APPROACH IN RTDWG

A. *Parameters of the new model*

At the current stage of this work, our model uses two analytical dimensions; each dimension contains one potential parameter:

- P_1 - the Change Data size: because once the amount of transferred information is reduced, resources requirements are minimized while both speed and

ISBN: 978-988-19252-9-9
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)

efficiency are maximized [2].

- P_2 – the DW size: the basic principle consists of the fact that new row insertion procedures in tables with few (or without) contents, are performed much faster than in big size tables [1]. Furthermore, with few or without integrity constraints, data are loaded faster into the target tables [1].

Our goal is to define the thresholds (t_h) of each parameter (p_i)

TABLE I
CLASSIFICATION OF PARAMETERS / DIMENSION

Symbol	Dimension	Parameter
P_1	Data source	Data size (changed data size)
P_2	DW	DW size

that fulfill RTDWg according to end-users requirements; freshness and accuracy (F, A).

B. Algorithm

We have designed the following algorithm that checks whether a parameter is real-time or not and whether it is critical for real-time or not.

Algorithm : Is_critical_RT(P)

```
Function is_RTC(p)
    RT, C: boolean
Given (F,A)
Begin
    Assess if p is RT
        set RT ← value1
    Assess if p is critical for Real-time
        Set C ← value2
    Return p(value1,value2)
End
```

We consider that a parameter is:

- **Real-time (RT)** when its presence or absence affects data freshness. It is considered as RT when its values influence data acquisition latency.

It becomes

- **Critical (C)** when its value exceeds the pre-established threshold (t_h), requirements (F, A) are influenced accordingly.

C. Structure of the new model

The methodology consists on streaming data from sources to destination in a way that respects the fixed threshold (t_h) of each parameters and prioritize data according to their types (Fig. 2).

We have designed instructions that process data as follows:

We have designed the instructions that process data as follows:

- When a change is captured at time t_1, the Dynamic RT data are always the first to be handled, the static RT data are processed after all Dynamic RT data are treated,
- RT data must be processed in compliance with the predetermined threshold (t_h). If the parameter value exceeds the threshold, the program works by using successive loop parallelism and partitioning,
- NRT data are processed after all RT data arrive at their destinations.

These steps are briefly schematized in Fig. 5 which shows that when a data change (Δdata) is captured, an analysis of type of data is triggered. This analysis involves a checking of the volume of data change and the size of the target DW. The overcoming of the DW size and data change size thresholds triggers the creation of temporary DW. Later on, all temporarily created DW are deleted.

V. Experiments and Result Analysis

A. Framework of experiments

Our experiments have been conducted under the following configuration (Table II):

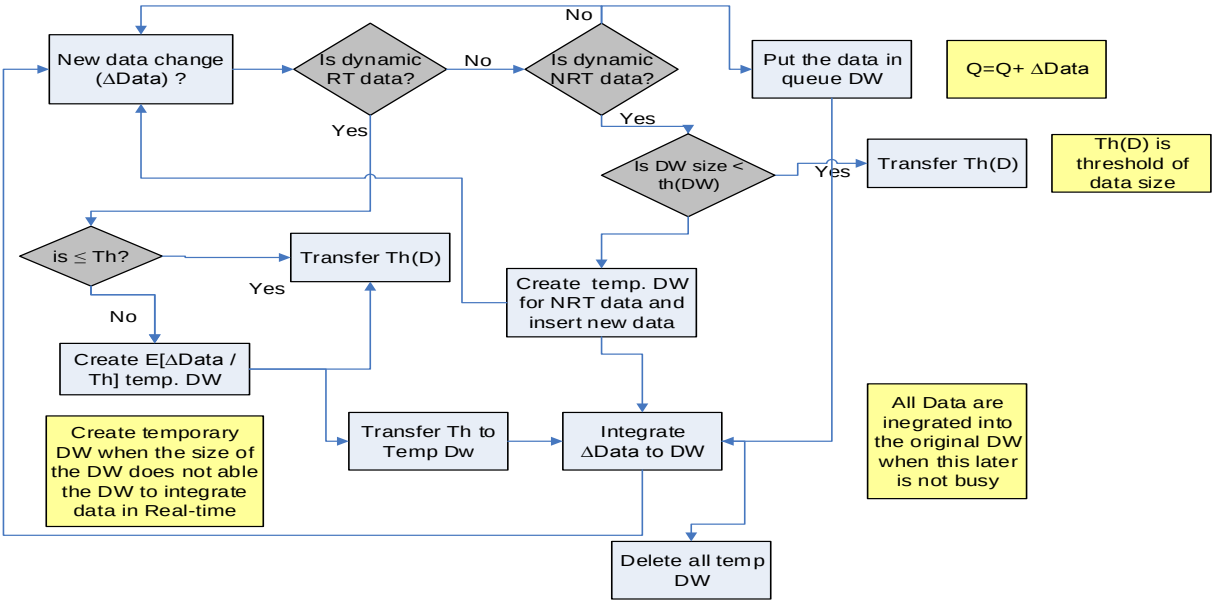

Fig. 5. Global schema of the proposed parametric approach

ISBN: 978-988-19252-9-9
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)

TABLE II
SIMULATION FRAMEWORK

OS	DW/ ETL	Data Source
- XP Windows (2GB RAM, 2Ghz)	- Oracle warehousing Builder 10g [4]	- Oracle 10g - Flat files

The simulation environment is located on the same machine: an Intel Core 2 Duo 2 GHz system with 2 GB of RAM running XP Windows.

B. Hypothetis:

Our approach is progressive and based on increasing the level of difficulty of hypothesis.

First, we examined the first degree following hypothesis:

- All data are dynamic RT
- We have assigned the threshold data value of 66, 4 MB
- ODS is an oracle 10g database;
- No queue DW has been considered.

C. Schema of our test-bed DW

We have chosen two tables in ODS, our simplified ETL consist of one jointure and one expression. The DW is simply constituted of one fact table.

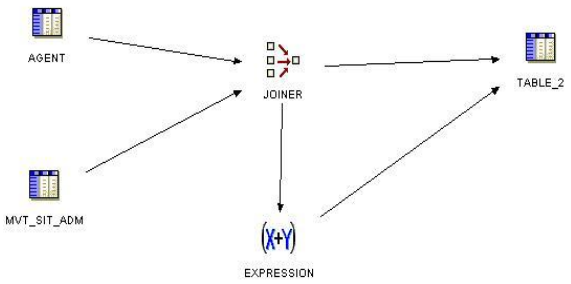

Fig. 6 Simplified mapping of the test-bed

D. Preliminary results

According to the above-mentioned hypothesis, we have obtained the following results:

a. Result n°1:

We have assessed that both parameters (data size and DW size) are RT because when their values change, the performance of the solution is influenced consequently. We have also assessed that they are critical for RTDWg.

b. Result n°2:

Given that data change in our test-bed doesn't exceed the tolerated threshold. The following table shows the elapsed time in each data change (figure 7, Table III).

TABLE III
PRELIMINARY DATA CHANGE RESULTS

line	ΔD (MB)	DW size(MB)	F (s)
1	34	72	1,2
2	42	92	1,2
3	60	96	2,3
4	66,3	104	1,2

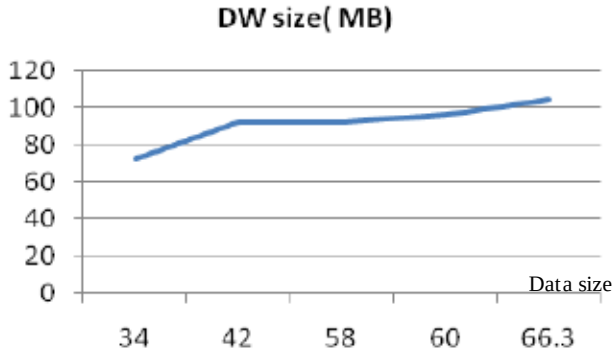

Fig.7 . Preliminary relationship between both paremters (change data size and Dw size)

VI. CONCLUSION AND FUTURE WORK

RTDWg is an advanced stage of data evolution. Although the considerable attention it has received, it still is handcuffed by some parameters whose value can be customized in order to achieve datawarehousing in real-time. We have introduced a new model that balances between RTDWg requirements and some parameters that belong to DW solution. Through a progressive approach, we are conducting testing under a specific framework; advanced simulations are needed to improve the new model.

As future work we intend to formulate advanced hypothesis and conduct our experiments under further frameworks in order to establish a general parametric model in RTDWg. We will assess the need of extra algorithms to balance continuously between requirements (A, F) and parameters (P_1, P_2). We are currently preparing a new model called IA-RTDWg that aims preserving data integrity in RTDWg.

REFERENCES

[1] Santos, R.J., Bernardino, J.: Real-time Data Warehouse Loading Methodology. In: 12th International Database Engineering and Applications Symposium, pp. 49–58. ACM, New York (2008)

[2] C K Bhensdadia, Yogeshwar P Kosta, Empirical Study on Dynamic Warehousing, International Journal of Computer Theory and Engineering, Vol. 2, No. 5, October, 2010, 1793-8201 (2010)

[3] Kamal Kakish, Theresa A. Kraft, ETL Evolution for Real-Time Data Warehousing, Proceedings of the Conference on Information Systems Applied Research ISSN: 2167-1508 New Orleans Louisiana, US(2012)

[4] Best Practices for Real-time Data Warehousing: Oracle White paper (2012)

[5] Muhammad S. Khakwani, Saudi Aramco Exploration & Producing Data Warehouse, A Case Study, proceeding of WCE 2007, july 2-4, London, U.K, ISBN:978-988-98671-5-7 (2007)

ISBN: 978-988-19252-9-9
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)