



**HAL**  
open science

## OLAPing Reflexive Multidimensional Fact

Maha Ben Kraiem, Jamel Feki, Ahmed Alghamdi, Franck Ravat

► **To cite this version:**

Maha Ben Kraiem, Jamel Feki, Ahmed Alghamdi, Franck Ravat. OLAPing Reflexive Multidimensional Fact. 12th Conference on Advances of Decisional Systems (ADS 2018), May 2018, Marrakech, Morocco. pp.466-479. hal-02283156

**HAL Id: hal-02283156**

**<https://hal.science/hal-02283156>**

Submitted on 10 Sep 2019

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.



## Open Archive Toulouse Archive Ouverte

OATAO is an open access repository that collects the work of Toulouse researchers and makes it freely available over the web where possible

This is an author's version published in:  
<http://oatao.univ-toulouse.fr/22468>

**To cite this version:** Ben Kraiem, Maha and Feki, Jamel and Alghamdi, Ahmed and Ravat, Franck *OLAPing Reflexive Multidimensional Fact.* (2018) In: 12th Conference on Advances of Decisional Systems (ADS 2018), 2 May 2018 - 3 May 2018 (Marrakech, Morocco).

Any correspondence concerning this service should be sent to the repository administrator: [tech-oatao@listes-diff.inp-toulouse.fr](mailto:tech-oatao@listes-diff.inp-toulouse.fr)

# OLAPing Reflexive Multidimensional Fact

Maha BEN KRAIEM\*, Jamel FEKI\*, \*\*, Ahmed ALGHAMDI\*\*, Franck RAVAT \*\*\*

\* University of Sfax, MIRACL Laboratory,  
Airport Road Km 4, P.O.Box. 1088; 3018 Sfax, Tunisia  
Maha.BenKraiem@yahoo.com

\*\* University of Jeddah, FCIT, IS dept  
Saudi Arabia

{Jfeki, ahmedg}@uj.edu.sa

\*\*\* IRIT, University of Toulouse,  
118, Route de Narbonne, 31069 Toulouse Cedex 9, France  
Franck.Ravat@irit.fr

**Abstract.** The multidimensional data model and implementations of social networks come with a set of specific constraints, such as missing data, reflexive relationship on fact instance. However, the conventional OLAP operators and existing models do not provide solutions for handling those specificities. Therefore, further efforts should be invested to extend these operators to take into consideration the specificities of multidimensional modeling of tweets and their manipulation. Face to this issue, we propose, in this paper, two new OLAP operators that enhance existing solutions for OLAP analyses involving a reflexive relationship on the fact instances. For each OLAP operator, we suggest a user-oriented definition as an algebraic formalization, along with an implementation algorithmic.

## 1 Introduction

The data warehouse has been the backbone of decision support systems for more than two decades and widely accepted and used across the globe in a variety of applications. Contributions of the research community in the data warehousing field, complimented by advancement in the relevant hardware technology, have matured these systems in managing huge volumes of data and providing their access with matchless efficiency to applications and decision-makers. On-Line Analytical Processing (OLAP) is at the core of data warehouse systems enabling multidimensional analysis of warehoused data.

Social media is yet another interesting area producing large data volumes that fascinate the attention of research and business communities. There is growing interests in gaining insights to the way social networks operate, their users behave, engage in conversations, express their opinions and influence others. This involves performing aggregations across conventional and unconventional dimensions in social media data.

Furthermore, businesses can largely benefit from this new resource and market of social media, provided that the underlying technology and systems of data warehousing can deal

with the challenges of heterogeneous data (i.e., semi-structured data) and the speed at which the data originate from social media.

In previous work, we have applied the data warehousing technology to enable comprehensive analysis of massive data volumes generated by the *Twitter* social network. More accurately, we have proposed a multidimensional model dedicated to the OLAP of data exchanged through tweets (Ben Kraiem et al. (2015)). This model takes into account the specificities of data issued from tweets. Among these specificities, we can find links between tweets and their answers' tweets. Regarding this new issue, we have extended the concept of fact by the proposal of a new relationship between fact instances called *reflexive relationship*. This fact-to-fact reflexive relationship allows connecting an instance of the fact to one or several instances of the same fact. Based on this relationship, the fact instances are linked at many successive levels.

Naturally, the concept of levels between fact instances is a novel proposal for which the conventional analysis tools are not designed for. Therefore, we need new OLAP operators to manipulate such a reflexive relationship.

In this paper, we define two new OLAP operators called *FDrilldown* and *FRollup*. They allow navigating down and up through the implicit hierarchical levels of the *fact*; this represents the first step to detect strong connections between fact instances and, therefore discover interesting or amazing topics and then conduct much more deep analysis of such data sets.

We have opted for the following organization of this paper. Section 2 studies representative works related to the OLAP operators that addressed the analysis of facts. Section 3 introduces our motivation example and context. Section 4 proposes two new operators called *FDrilldown* and *FRollup* for fact drilling. For each operator, we formalize it as an algebraic definition and develop an algorithm to implement it. Section 5 provides experimental results and assessments on the efficiency of our proposed OLAP operators.

## 2 Related works

To the best of our knowledge, no solution for OLAP analysis is proposed for *Drilling down* and *up* on the fact on the multidimensional schema. Only few *querying* operators on fact (Drill-Across, FRotate) are formally proposed in Abelló et al. (2002) and Ravat et al. (2008).

Drill-Across operator relates information contained in two multidimensional facts having the same dimensions. According to Kimball and Ross. (2002), Drill-Across can only be applied when both cubes have the same schema dimensions and the same instances. Other authors relax this restriction. Abelló et al. (2002, 2003) define the Drill-Across as changing a currently analyzed subject F1 (fact) with another fact F2 while keeping the same analysis space (current dimensions). The authors have identified semantic relationships between dimensions and facts: Derivation, Generalization, Association and Flow to extend possibilities to drill across. These relationships between dimensions and facts improve the conformity between attributes and could be used to navigate or Drill-across between Star schemas, even when dimensions are not shared. Cabbibo and Torlone (2004) define drill across as an extension to the natural join where the intersection of the two dimensions is aggregated at the finest grain of the dimensions. Furthermore, Riazati et al. (2008) propose extending the navigation operation drill across to include the non-conformed dimensions.

The FRotate operator in Ravat et al. (2008) consists in using a new fact in the multidimensional table while preserving the characteristics of the current analysis axes. The new fact must share at least the two current (i.e., displayed) dimensions with the current fact. Note that the fact rotation operation, noted FROTATE, is equivalent to the Drill-Across operation Abelló et al. (2003).

According to this study, we may conclude that none of these works offers tools for the decision-maker to navigate (Drilling -down, and -up) through the fact. So far, the OLAP frameworks lack the ability to cope with this problem.

To alleviate this drawback, our proposed OLAP operators namely *FDrilldown* and *FRollup* go further according to a new *Reflexive* relationship on the fact instances. These new operators allow modifying the analysis level in a fact while keeping the same analysis context( i.e., without changing the dimensions for the currently analyzed fact). Hence, data analysts would benefit greatly from the ability to navigate and view combined multidimensional data from multiple levels of fact.

### 3 Motivation example

Referring to our multidimensional model dedicated to the On-Line Analytical Processing (OLAP) of data exchanged through tweets, our motivation example is built upon the ‘*Tweet Constellation*’ proposed in Ben Kraiem et al. (2014). This model mainly consists of a set of two facts namely *FACTIVITY-TWITTOS* which corresponds to an observation on user accounts and allows the analysis of the user activity over time, and *FACTIVITY-TWEET* fact, which is a reflexive fact. It models links between a tweet and the person concerned by the answer (answered person) and then allows participants and other readers to easily follow the exchange of tweets. Being reflexive, *FACTIVITY-TWEET* allows interconnecting instances of the same fact hierarchically. In practice, if a tweet *tr* is a reply to tweet *t*, *tr* refers *t* (it contains the ID of tweet *t*). This reflexive relationship between tweets will guarantee that every tweet response inserted to the data warehouse corresponds to an existing tweet so that the analysis of a set of linked tweets becomes possible. Our ‘*Tweet Constellation*’ multidimensional model is composed of five dimensions namely *DTime*, *DSource*, *DTweet-Metadata*, *DPlace* and *DUser*.

Fig. 1 shows a fragment modeling the reflexive fact *FACTIVITY-TWEET* and its dimensions. Further details on this model (i.e. *Tweet Constellation*) are in Ben Kraiem et al. (2014), Ben Kraiem et al. (2015).

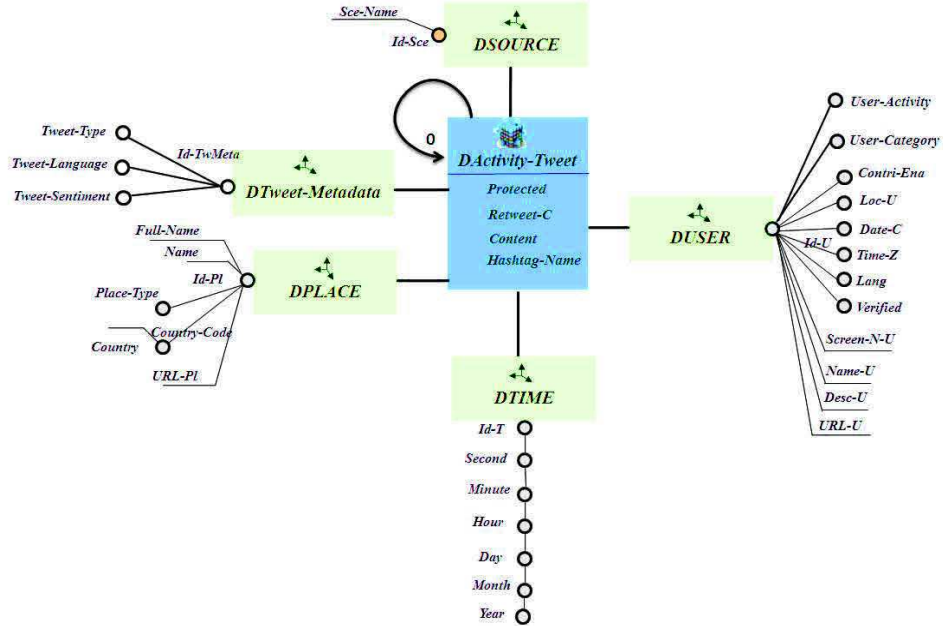


FIG 1. *FACTIVITY-TWEET* fact and its dimensions

Table 1 shows a set of seven reflexive tweets from the fact *FACTIVITY-TWEET*.

N	ID-Twt	Content	Id-Twt-Response	LEVEL
1	946077853262778373	Tu sais que tu n'as rien foutus de ta journée quand ton AppleTV te demande si tu es encore là. #BingeWatching #NoLife	-	1
2	946078190027661312	"@lolfr C'est bien aussi :) T'as vu quoi ?"	946077853262778373	2
3	946084024375750657	@cegron On my list. J'ai jusqu'au 31 pour rattraper mon retard.	946078190027661312	2
4	946078475923935232	@cegron The Punisher. J'étais en retard. ¿	946078190027661312	3
5	946078699283206145	"@lolfr Moins que moi, alors. Je n'en suis qu'au 8".	946078475923935232	4
6	946079260711768064	@cegron Ah mais j'ai pas fini. Épisode 6 seulement. ¿	946078699283206145	5
7	946080193910853633	"@lolfr I.N.E.X.C.U.S.A.B.L.E ¿ Tu as le spécial noël de Doctor Who aussi"	946079260711768064	6

TAB. 1. *Sample of seven interconnected instances of tweets from FACTIVITY-TWEET*

In example in TAB. 1, we distinguish six hierarchical levels. The first level corresponds to the tweet at line 1. The second level corresponds to tweets at lines 2 and 3, which are

responses to the same tweet in line 1. Finally, tweets from lines 4 to 7 correspond, respectively, to levels 3, 4, 5 and 6 (cf. Fig. 2). Hence, we may notice that due to the reflexive relationship on fact instances, the fact is composed of hierarchical data at multiple levels and allows a decision-maker to navigate between levels. Using levels in OLAP offers further alternatives analyses since it provides users with the flexibility to view data from different perspectives.

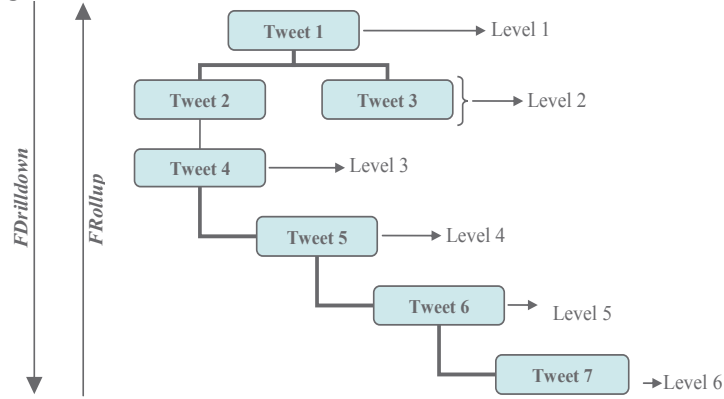


FIG 2. Hierarchy of levels of tweets listed in Table 1

However, classical OLAP algebra does not provide solutions for handling navigation between levels of the fact's instances since the multidimensional conventional models do not offer the reflexive relationship. Solving this issue requires appropriate operators. Hereafter, we define two new OLAP operators called *FDrilldown* and *FRollup*. They allow navigating through the hierarchical levels of the *fact*, in order to analyze a measure with more or less precision. The proposed operators are well suited to decision making applications since they can produce an output that leads to many different kinds of analyses. Basically, they allow identifying topics that have elicited a significant number of responses; these topics can be more investigated/explored later using sophisticated techniques as those used in "Text Mining" tools. Thus, we can extract knowledge from tweets and strengthen more semantics.

## 4 OLAP operators for reflexive fact

The result of an OLAP analysis is usually presented as a *Multidimensional Table* (Gyssens and Lakshmanan, 1997). A multidimensional table is a visualization structure that displays, from a single fact, data calculated according to two of the fact dimensions.

A multidimensional table, denoted *MT*, is defined by  $(F, MES, Dim, Hier, Pred)$  where:

- *F* is the name of the fact (subject) analyzed,
- $MES = \{f_1(m_1), \dots, f_p(m_p)\}$  is a set of *p* measures  $m_1, \dots, m_p$ , each measure is associated with an aggregation function  $f_1, \dots, f_p, f \subseteq \{SUM, AVG, MAX \dots\}$ ,
- $Dim = \{D_1, D_2\}$  is the set of two dimensions currently displayed in *MT*,
- $Hier = \{H^{D1}, H^{D2}\}$  is a set of two hierarchies currently displayed in *MT* and belonging to dimensions  $D_1$  and  $D_2$  respectively.
- $Pred = \{pred_1 \wedge \dots \wedge pred_s\}$  is a normalized conjunction of predicates (restrictions of dimensions data and fact data).

## 4.1 FDrilldown Operator

The *FDrilldown* operator applies to a reflexive fact; it consists in moving from coarser-level data to finer level data within the same analyzed fact. This drilling is possible due to the presence of the reflexive relationship on fact instances. Next, we give the *FDrilldown* algebraic formalization.

### 4.1.1 Conceptual definition

$MT \leftarrow FDrilldown (MT_k, F, Lvl_{inf})$	
Input	<ul style="list-style-type: none"> <li>- <math>MT_k</math>: A multidimensional table currently displayed</li> <li>- <math>F</math>: is the <b>reflexive</b> fact analyzed in <math>MT_k</math> and on which the drilling operation is applied.</li> <li>- <math>Lvl_{inf}</math> is a level lower than the displayed level in the current fact <math>F</math>.</li> </ul>
Output	$MT$ is the result multidimensional table.

TAB 2. Algebraic Formalization of the *FDrilldown* operator

### 4.1.2 Logical definition

The algorithm *FDrilldown* develops the logic of the *FDrilldown* operator. Note that “For each row  $r$  in the result set, the keyword LEVEL returns the depth in the hierarchy (hierarchical level) of the node represented by row  $r$ . The LEVEL of the root node is 1, the LEVEL of an immediate child of the root node is 2, and so on”<sup>1</sup>.

<p><b>Algorithm FDrilldown:</b> <math>MT \leftarrow FDrilldown (MT_k, F, Lvl_{inf})</math></p> <p><b>Input</b></p> <p><math>MT_k</math>: Multidimensional table</p> <p><math>F</math>: is the reflexive fact analyzed in <math>MT_k</math>, on which the drilling down operation is applied.</p> <p><math>Lvl_{inf}</math> is a level of <math>F</math>, to be reached by the <i>FDrilldown</i></p> <p><b>Output</b></p> <p>New multidimensional table <math>MT</math>, with the same structure as <math>MT_k</math></p> <p><b>Begin</b></p> <ol style="list-style-type: none"> <li>1. Let Levels = <math>\{Lvl_n, Lvl_{n-1}, \dots, Lvl_c\}</math> be the set of displayed levels of <math>F</math> with <math>Lvl_c</math> is the finest level, and <math>Lvl_n</math> is the highest level (<math>c \leq n</math>)</li> <li>2. Let <math>NB</math> be the number of levels in the reflexive fact <math>F</math> in <math>MT_k</math></li> <li>3. Query-Level = ‘ <b>SELECT MAX (LEVEL)</b> ’    ‘ <b>FROM</b> ’    <math>F</math>    ‘ <b>CONNECT BY PRIOR</b> ’    child_expr = parent_expr (i.e., <math>Id-Twt = Id-Twt-Response</math>);</li> <li>4. <math>NB = \text{Result of Query-Level}</math></li> <li>5. If <math>Lvl_c \leq Lvl_{inf}</math> OR <math>Lvl_{inf} \geq NB</math> then</li> <li>6. // Impossible Drilling operation</li> <li>7. Else</li> </ol>
--

<sup>1</sup>[https://www.enterprisedb.com/docs/en/9.5/eeguide/EDB\\_Postgres\\_Enterprise\\_Guide.1.036.html](https://www.enterprisedb.com/docs/en/9.5/eeguide/EDB_Postgres_Enterprise_Guide.1.036.html)



8. Translate *FDrilldown* ( $MT_k; F, Lvl_{inf}$ ) into query  $Q$  such as  
 $Q = \text{' SELECT LEVEL ' } \parallel p_n, p_{n-1}, \dots, p_1 \parallel f_1(m_1), f_2(m_2), \dots \parallel \text{' FROM ' } \parallel$   
 $D_1, D_2, F \parallel \text{' WHERE ' } \parallel \text{Pred} \parallel \text{' AND ' } \parallel D_1.\text{primary key} = F.\text{foreign key-}D_1 \parallel \text{' AND ' } \parallel$   
 $D_2.\text{primary key} = F.\text{foreign key-}D_2 \parallel \text{' AND LEVEL = ' } \parallel Lvl_{inf} \parallel \text{' CONNECT BY$   
 $\text{PRIOR ' } \parallel \text{child\_expr} = \text{parent\_expr (i.e., Id-Twt} = \text{Id-Twt-Response)} \parallel \text{' GROUP BY ' } \parallel$   
 $\text{LEVEL, } p_n, p_{n-1}, \dots, p_1;$   
 9.  $MT = \text{Result of query } Q.$   
 10. Display  $MT$   
 11. End if  
**End**

**Example 1.** To explain how the *FDrilldown* executes, we provide an example of analysis. Suppose that the decision-maker wishes to count the number of tweets (*Count (Id-Twt)*) by *Tweet-Sentiment* of the *DTWEET-METADATA* dimension and by *Country* of the *PLACE* dimension. As a result for this requirement, (s)he obtains the multidimensional table  $MT_1$  shown in Fig. 3. Each cell in  $MT_1$  gives the number of tweets for each combination of *Country* and *Tweet-Sentiment*.

FACTIVITY-TWEET   Count(Id-Twt)		DPLACE   H_Geo				
		Country	Belgium	Canada	France	Spain
DTWEET- METADATA   H_Sen	Tweet-Sentiment					
	Positive		167	46	10885	172
	Negative		41	28	6279	97
	Neutral		85	17	7658	43
<b>R = Ø</b>						

FIG 3.  $MT_1$ : Result multidimensional table for Example 1

To the extent that this sample is representative, most conversations that occur in Twitter appear to be dyadic exchanges of three to six messages. For this reason, based on the results presented in  $MT_1$  (cf. FIG 3), the decision-maker intends to restrict the analysis to tweets that tie in *level 6*. In fact, his aim is to move deeper into a chain of data, from high-level information to more detailed information. Hence, data pertaining to fact can then be pre-summarized and then be available for more analyses (number of intense conversation...). This OLAP analysis is calculated using the following algebraic expression:

$$MT_2 \leftarrow FDrilldown (MT_1, FACTIVITY-TWEET, 6) \quad (1)$$

After execution, the decision-maker obtains the multidimensional table presented in Fig. 4.

FACTIVITY-TWEET   Count(Id-Twt)		DPLACE   H_Geo		
		Country	Belgium	France
DTWEET- METADATA   H_Sen	Tweet-Sentiment			
	Positive		37	985
	Negative		9	392
<b>R = O</b>				

FIG 4.  $MT_2$ : Result of the expression 1

This result will allow the analyst to get interesting values from topics that have elicited more responses, performing specialization if more details are needed and, finally gleaning valuable insights about the way of propagation of data within each level. It also allows identifying where relevant tweets originate from.

## 4.2 FRollup Operator

The *FRollup* operator is the reverse of *FDrilldown*, it consists in moving from a finer level to a coarser level on a currently displayed fact based on fact instances linked through the reflexive relationship (Tweet Response – Tweet). Each tweet may be connected to  $n$  ( $n \geq 0$ ) tweets responses within the same fact.

### 4.2.1 Conceptual definition

$MT \leftarrow FRollup (MT_k, F, Lvl_{sup})$	
Input	<ul style="list-style-type: none"> <li>- <math>MT_k</math>: A multidimensional table currently displayed</li> <li>- <math>F</math>: is a reflexive fact, on which the FRollup operation is applied.</li> <li><math>Lvl_{sup}</math> is a coarser-graduation level on the current fact.</li> </ul>
Output	$MT$ is the resulting multidimensional table.

TAB 3. Algebraic Formalization of the *FRollup* operator

### 4.2.2 Logical definition

The algorithm *FRollup* develops the logic of the *FRollup* operator.

<p><b>Algorithm FRollup:</b> <math>MT \leftarrow FRollup ((MT_k, F, Lvl_{sup})</math></p> <p><b>Input</b></p> <p><math>MT_k</math>: Multidimensional table</p> <p><math>F</math>: is the fact actually analyzed in <math>MT_k</math>, on which the rolling up operation will apply. The relationship between the instances of the analyzed fact must be reflexive.</p> <p><math>Lvl_{sup}</math> is a level of F, to be reached by the <i>FRollup</i>.</p>
--

**Output**

Result multidimensional table  $MT$ , with the same structure as  $MT_k$

**Begin**

1. Let Levels =  $\{Lvl_n, Lvl_{n-1}, \dots, Lvl_c\}$  be the set of levels displayed for F;  $n$  and  $c$  are respectively the lowest and highest levels ( $c \geq n$ )
2. Let  $NB$  be the number of levels in the fact F analyzed in  $MT_k$
3. Query-Level = ‘ SELECT ’ || MAX (LEVEL) || ‘ FROM ’ || F || ‘ CONNECT BY PRIOR ’ || child\_expr = parent\_expr (i.e.,  $Id-Twt = Id-Twt-Response$ ;
4.  $NB$  = Result of Query-Level
5. If  $Lvl_c \geq Lvl_{sup}$  OR  $Lvl_{sup} \geq NB$  then
6. // Impossible FRollup operation
7. Else
8. Translate FRollup ( $MT_k: F, Lvl_{sup}$ ) into query  $Q$  such as  
 $Q = \text{‘ SELECT LEVEL, ’} || p_n, p_{n-1}, \dots, p_1 || f_1(m_1), f_2(m_2), \dots || \text{‘ FROM ’}$   
 $|| D_1, D_2, F || \text{‘ WHERE ’} || \text{Pred} || \text{‘ AND ’} || D_1.\text{primary key} = F.\text{foreign key-D}_1 || \text{‘ AND ’}$   
 $|| D_2.\text{primary key} = F.\text{foreign key-D}_2 || \text{‘ AND LEVEL = ’} || Lvl_{sup} || \text{‘ CONNECT BY PRIOR ’}$   
 $|| \text{child\_expr} = \text{parent\_expr (i.e., } Id-Twt = Id-Twt-Response || \text{‘ GROUP BY ’} || LEVEL, p_n, p_{n-1}, \dots, p_1;$
9.  $MT$  = Result of query  $Q$ .
10. Display  $MT$
11. End if

**End**

**Example 2:** Assume that the decision-maker starts his analysis by displaying the number of tweets at level 3 by *Sce-Name* (source Name) on the *DSource* dimension and by *User-Category* on *DUSER* dimension. He obtains a multidimensional table as in Fig. 5. Each cell represents the number of tweets of level 3 for a given *Source Name* and a given *User Category*.

FACTIVITY-TWEET   Count(Id-Twt)		DUSER   H_Cat			
		User-Category	Friendship relationship	Information Seeker	Information Sharing
DSOURCE   H_Sce	Sce-Name				
	TendancesFrance		-	-	58
	Tweetbot for iOS		2	-	4
	Tweetbot for Mac		-	2	1
	Twitter for Android		37	33	65
	Twitter for iPad		-	-	2
	Twitter for iPhone		-	75	172
Twitter Web Client		25	13	36	

$R = \emptyset$

FIG 5.  $MT_3$ : Result multidimensional table for Example 2

Suppose the decision-maker carries out the same analysis described in the example 2, but he puts less emphasis on the depth of involved level (Level 3) (cf. Fig. 5), the decision-maker continues by rolling up analysis level. This time he expects to get the number of tweets at level 2. The corresponding analysis expression is:

$$MT_4 \leftarrow FRollup(MT_3, FACTIVITY-TWEET, 2) \quad (2)$$

Fig. 6 shows the obtained result within level 2.

FACTIVITY-TWEET   Count(Id-Twt)		DUSER   H_Cat			
		User-Category	Friendship relationship	Information Seeker	Information Sharing
DSOURCE   H_Sce	Sce-Name				
	TendancesFrance		3	17	103
	Tweetbot for iOS		4	2	12
	Tweetbot for Mac		5	3	3
	Twitter for Android		157	176	208
	Twitter for iPad		1	4	8
	Twitter for iPhone		256	250	481
	Twitter Web Client		93	66	97
<b>R = Ø</b>					

FIG 6.  $MT_4$ : Result of expression 2

According to this result, we may conclude that the number of tweets for Information Sharing category and Twitter for iPhone as well as Twitter for Android is important. In fact, information sharing users post news and tend to have a large base of “followers” and answers about that news.

## 5 Experimental results

In order to evaluate the drilling Up and Down operators using the reflexive relationship between tweets, we have integrated these operators into our software prototype called *OLAP4Tweet* (Ben Kraiem et al., (2015)), developed using JAVA and ORACLE 10g database.

The *OLAP4Tweet* framework is composed of two modules, namely *Analysis Engine* and *Interactive Restitution*. Each module has specific roles and interacts with the other. The *Analysis Engine* module is designed for R-OLAP environment. It is composed of a set of algebraic operators and one parser:

- The set of algebraic operators defines elementary operations that decision-makers can carry out while analyzing. The definition of algebraic operators is independent of tools and implementation languages.
- The operator parser (a) translates algebraic operators into queries, (b) generates corresponding SQL queries and executes them.

The *Interactive Restitution* module contains (a) a graphical implementation of analysis operators in order to facilitate decision-makers’ tasks and (b) a graphical interface showing analysis results.

We have loaded a dataset containing 71,739 tweets collected by crawling two hours of public tweets (from Fri Dec 22 10:48:50 UTC 2017 to Fri Dec 22 12:48:50 UTC 2017). These tweets are written in different languages. Once we load the “Tweet Constellation” multidimensional model with data, we can express and execute OLAP queries. For this purpose, we include a user-friendly decision-making process in our analysis framework. A

decision-maker starts an analysis by exploring the proposed model through an interface. (S)he selects measures and attributes related to their analysis needs by clicking and then an SQL code is generated. Queries involved in the experimental assessments aggregate the measure through the *COUNT* aggregation function. Finally, the interface provides the decision-maker with a dashboard interface representing the analysis result in tabular and graphical forms.

To illustrate how the *FDrilldown* and *FRollup* operators perform, we provide an example of analysis. We assume that the decision-maker wants to analyze the *number of tweets* by *Tweet-Sentiment* (of the *TWEET-METADATA* dimension) and by the parameter *Country* (of the *PLACE* dimension). A bar chart is required by the decision-maker ((s)he just click on Bar chart icon) to display the analysis result (cf. Fig. 7).



FIG 7. Number of tweets by Tweet-Sentiment and by Country.

Based on the analysis result presented in FIG 7 the decision-maker intends to restrict the analysis to tweets that correspond to *level 6*. This level can represent a valuable source of information that could help obtain a full picture of topics. Hence, statistical data pertaining to fact can then be pre-summarized, and then be available for large analyses. A slider (on the right top of the interface in Fig. 7) allows navigating along a set of levels on the fact. Fig. 8 shows the obtained result.



FIG 8. Interface after the execution of the FDrilldown operator

## 6 Conclusion

In order to exploit the reflexive relationship on fact instances, we have proposed two specific OLAP operators namely *FDrilldown* and *FRollup*. They provide solutions for handling an intuitive navigation between different levels within the fact. The proposed operators are well suited to decision making applications since they can produce an output that leads to many different kinds of analyses. They highlight the importance of tweets responses to show how information is propagated through each tweet. Basically, they allow identifying topics that have elicited a significant number of responses; these topics can be more investigated/explored using sophisticated tools based on "Text Mining" techniques; thus, we can extract knowledge from tweets and strengthen more semantics.

For each of these operators, we have presented an algebraic formalization, and a pseudo code algorithm.

To the best of our knowledge, this is the first initiative that has tackled OLAP operators for drilling down and up within a fact by exploiting the reflexive relationship.

As perspective work, we intend to integrate more analysis operators that take into consideration the specificities of our multidimensional model, as dynamic Data. These operators will help the interpretation of the results of multidimensional analyses on tweets and their metadata. It is also important to use OLAP mining, which integrates on-line analytical processing (OLAP) with data mining so that mining can be performed in different portions of data warehouses and at different levels of abstraction at user's fingertips.

Moreover, we plan to conduct experiments to measure the quality of the result extracted by our OLAP operators.

## References

- Abelló, A, Samos, J and Saltora, F (2002), *On relationships offering new drill-across possibilities*. In Proceedings of the 5<sup>th</sup> ACM international Workshop on Data Warehousing and OLAP, McLean, Virginia, USA, ACM Press, 2002, pp 7-13.
- Abelló, A, Samos, J and Saltora, F (2003), *Implementing Operations to Navigate Semantic Star Schemas*. In Proceedings of the 6<sup>th</sup> ACM International Workshop on Data Warehousing and OLAP, New Orleans, Louisiana, USA, ACM Press, 2003, pp 56-62.
- Ben Kraiem, M., J. Feki, K. Khrouf, F. Ravat, and O. Teste (2017), *OLAP operators for missing data*. In Revue des Nouvelles Technologies de l'Information (RNTI), Ed. Hermann, vol. B- 13, pages 53-66. 13<sup>èmes</sup> journées francophones sur les Entrepôts de Données et l'Analyse en Ligne, Business Intelligence & Big Data, 2017.
- Ben Kraiem, M., J. Feki, K. Khrouf, F. Ravat, and O. Teste (2015). *Modeling and OLAPing social media: the case of twitter*. In Social Network Analysis and Mining 5(1), 47:1–47:15.
- Ben Kraiem, M., J. Feki, K. Khrouf, F. Ravat, and O. Teste (2014). *OLAP of the tweets from modeling toward exploitation*. In 8th International Conference on Research Challenges in Information Science (IEEE RCIS'2014), 45–55.
- Cabibbo, L. and Torlone, R (2004). *Dimension compatibility for data mart integration*. In Proceedings of the 12<sup>th</sup> Italian Symposium on Advanced Database Systems, Cagliari, Italy, 2004, pp. 6-17.
- Ravat, F., Teste, O, Tournier, R, and Zurfluh, G. (2007), *Graphical Querying of Multidimensional Databases*. In Advances in Databases and Information Systems, Berlin, Heidelberg: Springer Berlin Heidelberg, 2007, Vol. 4690, pp. 298–313.
- Ravat, F., Teste, O, Tournier, R, and Zurfluh, G. (2008), *Algebraic and graphic languages for OLAP manipulations*. In International Journal of Data Warehousing and Mining, IJDWM 4(1), 2008, pp.17–46.
- Riazati, D, Thom, J.A and Zhang X (2008), *Drill Across & Visualization of Cubes with Non-Conformed Dimensions*. In the Nineteenth Australasian Database Conference (ADC 2008), Wollongong, Australia, January 2008. Conferences in Research and Practice in Information Technology, Vol. 75.

## Résumé

Le modèle de données multidimensionnel et les implantations des réseaux sociaux sont assortis d'un ensemble de contraintes, telles que des données manquantes, des relations réflexives sur des instances de fait. Cependant, les opérateurs OLAP classiques et les modèles multidimensionnels ne fournissent pas de solutions pour gérer ces spécificités. Par

conséquent, des efforts méritent d'être déployés pour étendre ces opérateurs afin de prendre en compte la spécificité de la modélisation multidimensionnelle des tweets et de leur manipulation. Face à ce problème, nous proposons de nouveaux opérateurs OLAP qui exploitent l'existence d'une relation réflexive entre les instances d'un fait. Pour chacun de ses opérateurs, nous proposons une définition orientée utilisateur (c'est-à-dire une formalisation algébrique) ainsi qu'une traduction algorithmique pour sa mise en œuvre.