

## New Solution for the Problem of the Non-Barcoded Products Using Database's & Programming's Technologies

Dr. Ammar Suhail Balouch \*

(Received 23 / 8 / 2010. Accepted 11 / 4 / 2011)

### □ ABSTRACT □

The problem has been addressed in this article is general and common problem in the area of the wholesale trade and retail trade in particular. This problem comes through non-barcoded products or products that cannot be coded before they are sold, such as milk, rice, ...etc. Many studies have been found to solve this problem by using one of two principles. The first principle depends on the abstract technical method using devices to generate the barcode (or special programs to perform a task to generate the barcode), and then re-entered the new data from generated barcode into the using accounting system. The second principle depends on a dynamic way to create a list of non-barcoded products with these barcodes, which are previously stored in the using program. Then the user must enter this information manually to the user interface program. Both principles possess disadvantages; have been processed in our own way put forward in this article.

Nowadays, no one can make wonderful works for the practice without database technologies. We use in this article some database techniques to build database system with its necessary relationships. Then we have supported this system with views (interfaces) to avoid the disadvantages of both mentioned principles and use these positive points. Through these interfaces we can determine the non-barcoded product, and to enter the quantities with prices showing a view where we can enter the quantity, price or force it to take together. We can understand by the application of these interfaces to address the problem of lost or quantities of products dedicated to the customer. We addressed and processed all these points through the algorithm XOMA, defined by us and is not supported by any use of accounting system.

**Keyword:** Barcode, database system, algorithm, Delphi PL.

---

\* Lecturer in MEF, University of Tishreen, Lattakia, Syria. Competence: Database's and Information's systems.

## حل جديد لمشكلة المنتجات غير المرزمة باستخدام تقنيات قواعد المعطيات والبرمجة

الدكتور عمار سهيل بالوش\*

(تاريخ الإيداع 23 / 8 / 2010. قُبل للنشر في 11 / 4 / 2011)

### □ ملخص □

إن المشكلة التي تم تناولها في هذا المقال هي مشكلة عامة وشائعة في مجال تجارة التجزئة بشكل خاص وتجارة الجملة بشكل عام. وتنشأ هذه المشكلة من خلال المنتجات الغير مرزمة أو التي لا يمكن ترميزها قبل بيعها مثل الحليب، الرز، ... الخ. وُجدت دراسات كثيرة لحل هذه المشكلة باستخدام احد مبدئين. يعتمد المبدأ الأول على طريقة تقنية مجردة باستخدام أجهزة لتوليد الباركود (أو برامج خاصة لتنفيذ مهمة توليد الباركود)، ويتم إعادة قراءته من جديد ليتم إدخال بياناته لنظام المحاسبة المستخدم. أما المبدأ الثاني فيعتمد على طريقة ديناميكية بإنشاء قائمة تحوي على المواد غير المرزمة لدى التاجر مع ترميزاتها المخزنة مسبقاً في البرنامج المستخدم للمحاسبة، وعلى المستخدم إدخال هذه المعلومات بشكل يدوي إلى واجهة البرنامج المستخدم. كلا المبدئين يملكان سلبيات تمت معالجتها بطريقتنا المطروحة في هذا المقال.

في هذه الأيام لا يمكن لأحد أن ينجز أعمالاً رائعة في الجانب التطبيقي العملي بدون تقنيات قواعد المعطيات. تم في هذا العمل استخدام بعض تقنيات قواعد المعطيات لبناء نظام قواعد معطيات وعلاقاته اللازمة. ومن ثم تم دعم هذا النظام بعدة واجهات لتجاوز السلبيات في المبدئين السابقين الذكر والاستفادة من النقاط الإيجابية فيهما. من خلال هذه الواجهات يمكننا أن نحدد المنتج غير المرزم، ولتحديد الكميات مع أسعارها تظهر نافذة نستطيع فيها إدخال الكمية أو السعر أو إجباره على المدخلتين معاً. مما يمكن فهمه من تطبيق هذه الواجهات أننا نستطيع معالجة مشكلة كميات المنتجات المفقودة أو المهداة للمستهلك.

تمت معالجة جميع هذه النقاط من خلال خوارزمية XOMA المعرفة من قبلنا و الغير مدعومة من قبل أي نظام محاسبة مستخدم.

الكلمات المفتاحية: باركود، نظام قواعد المعطيات، خوارزميات، لغة البرمجة دلفي.

\* مدرس في كلية الهندسة الميكانيكية والكهربائية - اختصاص أنظمة قواعد المعطيات وأنظمة المعلومات - جامعة تشرين - اللاذقية - سورية.

## Introduction:

The informatics world and accounting programs have found many works (programs or articles) study and give solutions for non-barcode products in different methods. The first method is adopted to develop programs to help in the design of various forms of barcode, in accordance with the using accounting systems. [1], [2], [3], [4]. The second method focuses its attention on rendering techniques to generate the appropriate barcode for the non-barcode product. This barcode contains information about the type of the product. This barcode has been printed by using special printer and re-read by barcode-reader to enter its information into using accounting program. The user must continue to enter the sold quantity through the used view to get the desired result.[3], [4].

The third method is the simplest one, but it is not practical, because it depends on creating a list of non-barcode products with these predefined and pre-stored barcodes. If the user must enter non-barcode product in this method, enter the information for this product from the created list. [5], [6]. The last method is supported by using electronic and programmatic scale, generated barcode, contains information about type, quantity, and price of product. [7], [8]. In our method, we can get all advantages of previously mentioned methods and avoid these disadvantages, by developing new hyper method.

## Importance of Research and Its Goals:

In this article, we do not try to review a new method to design or generate barcodes, but to use the advantages of these methods in the new technology. This technology depends on the design of user-interface. This interface is used in case the product you want to enter its data is non-barcode. In our program, this interface appears by click on “NON-Barcode”-button in main sale interface. Using this technique, we can avoid implementing the expensive electronic and programmatic scale, and we can calculate the exact sold quantity, and know the lost or dedicated amount for the customer.

With this method, we can see that the software solutions can be more practical and economical than by devices attached to the program or than the non-practical routine of manual entry of data by the user (than the user’s mental capacity (abilities) to save a list of coding). This method solves famous problem in field of general trade and retail. Note, that the failure to resolve this problem creates a complex problem, and causes significant errors in the accounts and commercial inventory.

## Research Methods and Materials:

At first, we build a database with some relevant tables. Then design a view to solve the mentioned problem with intelligent and cost effective method. In this research, we have defined the tables with Oracle Database system ([9], [10], [11]). The diagram of the relationships between the tables presented by UML-Notes ([12], [13], [14]). The implementation of the defined algorithm ([15], [16], [17]) for the process of this case is written by Delphi-Language ([18], [19], [20]).

We don’t want to go, as far as our research is concerned, into the theory of database systems, UML-Diagrams and programming language. We start by defining our resource data for this problem. The important tables in our work are represented as following:

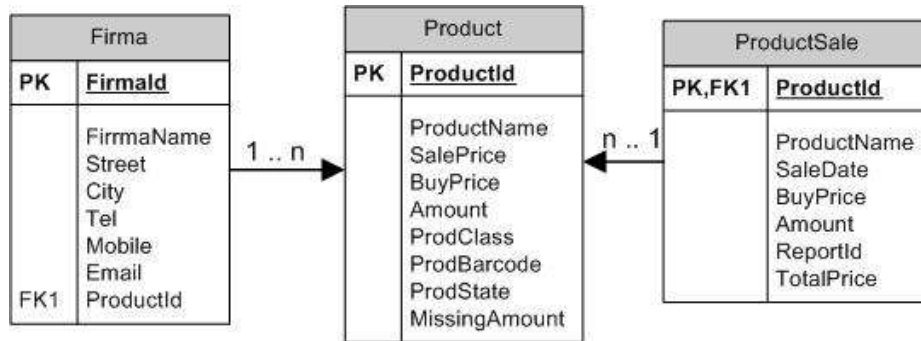


Figure (1): Database tables for product, firma, productSale

These tables contain information about the product like as ProductName, ProductId, SalePrice, BuyPrice,... etc. and about the Production-Firma of them. The third table has information about the Sale’s report. The Major Problem in our work is not to build a database system with views and queries but to give solutions for the problem with products that don’t having barcode. This Problem doesn’t allow the user to know about any exactly reset amount of a product.

**Results and Discussion:**

Our solution begins with the idea of creating an interface that includes logos of all non-barcoded products. By this interface the user can choose the product then write the amount or the price or the both. The data, which the user entered by this interface have been stored in our database. In the following part of discussion, we will explain our idea.

**Views Design:**

The first interface that we see (if the user calls this method) must contain all products, which present as follows:



Figure2: The Interface to choose the aim product

By using the following SQL-Code:

```

select ProductName
from Product
where ProdBarcode= "NULL" or ProdState= 0
    
```

We can receive in this view all the non-barcoded products. The attribute ‘ProdState’ can contain either ‘1’ for barcoded product or ‘0’ for non-barcoded product. When the ‘ProdState’ contains the ‘0’, the ProdBarcode can’t contain any information. This query must be embedded by a programming language to get the requirement. For example, Delphi has Query-tool that allows typing this SQL-statement.

If the user clicked on one of them (ex. Milk), it soon shows the second interface in this process. This interface has two fields to receive information (data) from the user (see figure3).



Figure3: The Interface to enter amount, price or the both

Under these circumstances, we have already defined an algorithm to apply to all cases, which can appear after entering the data. This Algorithm can calculate the exact sale-amount and the missing or donated one. In the following discussion, we will illustrate this algorithm with normal algorithm language:

**The XOMA Algorithm (eXact Or Missing Amount):**

Before clearly explaining our algorithm, we present the architecture of the case-level for our algorithm.



Figure4: Case-Level Architecture for XOMA

The XOMA algorithm is only active when the non-barcoded products are stored in the “basket buy”.

Algorithm XOMA;

Input: Amount ( $Am$ ), BuyPrice ( $Bp$ ) or both of a product

Output: Calculated Amount ( $CAM$ ), Missing Amount ( $MA$ )

1- Begin

2- If at first 'Amount' is entered and only it:

a. get the Buy-price ( $Bp$ ) by the following SQL-Statement:

```
select BuyPrice
from Product
where ProductName= "Milk"
```

b. yet, show the price in the Price-Field by calculating it in the form of " $Bp * Am$ ".

c. Store the Amount and the price in the database.

d. If the user wants to enter other price must be clicking on check box of 'Both Sure' (go to step 4).

3- If at first 'Price' is entered and only it:

a. get the amount for the entered price by calculation it from the database table like as:

$$CAM = \frac{NewBp * Am}{Bp}$$

' $NewBp$ ' is the entered price, ' $CAM$ ' is the calculated Amount

b. If the user wants to enter the real amount must be clicked on check box of 'Both Sure' (go to step 4).

4- If the user clicks on check box of 'Both Sure' at first, we meet the missing amount or the non-fixed price, because the user will enter either price that is not relevant with the amount, or the amount that is not relevant with the price.

a. In this step, the entered data will be stored in our database. Then the algorithm calculates the missing amount by using the following statement:

$$MA = EA * PerMis$$

$MA$ : Missing Amount,  $EA$ : Entered Amount,

$PerMis$ : Percent of missing amount. It is given by the manager. For example, in the most of Trade-report, we find that the percent of missing amount is nearly 1.5%. In this case,  $MA$  "for entered Amount" equal to '3.3' is equal to '3.3 \* 1.5 %'

b. To know the missing or donated amount must be compared to the relevant buy-price in the database.

c. The entered price that is not equal to the fixed price, must be proved by a condition. This condition explains that the price must be belonging to the fixed interval [ $LowPrice$ ,  $HighPrice$ ].

The fixed interval is given by the manager.

d. In the next the  $MA$  will be stored in our database. Finally we summarize all stored  $MA$  to be compared with reset amount in the warehouse.

5- In each step, depending on entered price, test if price smaller than the buy's price, then multiply the percent of the minimum of win. That means:

*if  $NewBp < Bp * PerMin$  Then show error*

$PerMin$ : Percent of the minimum of win.

6- End XOMA;

In this algorithm, we denote each procedure's step in this algorithm and the entered date plays a big role. For this algorithm we designed views to make the process very

simple and perfect. The views have been implemented depending on database system and on a programming language (Delphi) “we can use other programming language like as Java, VC++, C#, ... etc.” The entered data by the previous interface will be transferred into the following view:

Figure5: Interface for entering information of sold product

Then, we go forward with the main interface to complete the entering of sale-products.

### Conclusions and Recommendations:

In this article, we have defined a simple but efficient XOMA algorithm. This Algorithm can be extended to include all cases that we can find in small, middle and big trades. The cost of our algorithm is not discussed in this article but we have proved that this algorithm has more efficient than other defined algorithm for other works (that mentioned in ‘introduction’), and processed more case than the other one, can’t process it. We can give some information about the cost of our algorithm. All operations in XOMA-Algorithm are basic operations and have the cost  $O(1)$ . XOMA-Algorithm is a sub algorithm and can be integrated in all main sale interfaces. This means that this algorithm doesn’t change the cost of total algorithm.

Throughout this article, we have endeavored to prove that the academic ideas can be brought more efficiently for the industry and the associations.

## References:

- [1] HOLLAND N. N., *Death in a Delphi Seminar: A Postmodern Mystery (Suny Series, the Margins of Literature)*, St Univ of New York August 1995. 334.
- [2] BARROW J.; MILLER L.; MALAN K.; OXFORD, *Introducing Delphi Programming: Theory Through Practice*, Oxford Univ Pr, Juli 2005. 544.
- [3] BARROW J.; MILLER L.; MALAN K. *Introducing Delphi Programming: Theory Through Practice*, Oxford Univ Pr, Juli 2004. 560
- [4] ICON GROUP *Barcode: Webster's Timeline History, 1907-2007* International von ICON Group International, Inc. März 2010. 34.
- [5] URL: [www.alameen-aleppo.com](http://www.alameen-aleppo.com) برنامج الأمين للمحاسبة والمستودعات، سوريا، حلب
- [6] URL: [www.alrasheedsoft.com](http://www.alrasheedsoft.com) برنامج الرشيد لمحاسبة الشركات
- [7] URL: [www.bazergi.com](http://www.bazergi.com) برنامج البازار للمحاسبة الصغيرة والمتوسطة
- [8] URL: [www.edarisoft.com](http://www.edarisoft.com) برنامج الاداري للإدارة والمحاسبة
- [9] FEUERSTEIN S. et at. *Oracle PL/SQL Programming* O'Reilly Media, Oktober 2009, 625.
- [10] PRICE J., et at. *Oracle Database 11g SQL: Master SQL and PL/SQL in the Oracle Database (Osborne Oracle Press Series)*, Mcgraw-Hill Professional November 2007, 1100.
- [11] LONEY K., *Oracle Database 11G: The Complete Reference (Oracle Press)*, Mcgraw-Hill Professional Januar 2009, 1233.
- [12] PILONE D. et al. *UML 2.0 in a Nutshell: A Desktop Quick Reference (In a Nutshell (O'Reilly))*, O'Reilly Media Juni 2005. 240.
- [13] MILES R.; HAMILTON K.; MCLAUGHIN B.; OBRIEN M. T. *Learning UML 2.0* O'Reilly Media Mai 2006. 150.
- [14] ROSENBERG D., et al. *Use Case Driven Object Modeling with UML: Theory and Practice* Apress Januar 2007. 171.
- [15] LLC. *Postal Markings: Postmark, Cancellation, First Day of Issue, Franking, Digital Postmarks, Intelligent Mail Barcode, Daguin Machine*, Books LLC of Books Llc, Mai 2010. 96.
- [16] MCXOY N.; MCCOY N. H.; JANUSZ G. *Introduction to Abstract Algebra* Academic Press Inc Dezember 2000. 281.
- [17] THIMAS H.; CORMEN C. E.; LEISERSON R. L.; STEIN R. C. *Introduction to Algorithms*, Mit Press Student Edition. September 2009. 1312.
- [18] CANT M., *Delphi 2010 Handbook*, Createspace Februar 2010.
- [19] ROTERMANN S. *Ekanban and Barcode implementation in an automotive supplier plant* VDM Publisher, April 2009. 132.
- [20] BUSHNELL, *Getting Started With Barcodes*, Cutter Information Corp Dezember 1990.125.