



# Journal of Applied Sciences

ISSN 1812-5654

**science**  
alert

**ANSI***net*  
an open access publisher  
<http://ansinet.com>

## How to Select the Best Method For Database Applications, Database Independent Platform or Not?

<sup>1</sup>Halil Ibrahim Bulbul and <sup>2</sup>Yasar Guneri Sahin

<sup>1</sup>Department of Computer Education, Gazi University, Ankara, Turkey

<sup>2</sup>Department of Computer Engineering, Yasar University, Izmir, Turkey

---

**Abstract:** Software developers generally want to create software that could run with many different DBMSs (database independent platform). But, this idea sometimes may not be implemented to real world applications. The key point is that, it is too hard to decide which method (dependent to specific DBMS or independent from DBMS) should be selected. Each method has particular advantages and disadvantages. Actually, the decision is depended on some restrictions that are related to money, human resources, duration of project, etc. In addition, one of the most important factors is the choice of CASE tool that will be used for the software project. This study addresses how a decision should be made as to which method will be convenient for a specific application. Furthermore, possible development scenarios, that may be used in the project has been presented and some coefficient factors were found which help the developers to see which method may be more appropriate for the application.

**Key words:** Software project, decision, database independent platform

---

### INTRODUCTION

Using a software basically has three special purposes which may be summarized like; reducing the redundant paper works, reducing allocated time and saving money. At the same time, there is one more important point of using software that is the aspect of the software developer companies. Since the software developer companies want to sell their products as much as possible, they should develop their products as flexible and compact as possible and the products also must be able to be used on many platforms (Yu, 2003).

Database independent application software development method is not a new idea and it is known that it provides advantages to developers. But, of course, this method always may not be implemented to every database application, because, the method has some restrictions and limits which can change according to the project and vendor type. In order to decide to use the method, typically, one should know what database independent software development is.

Database independency basically means that, an application could work together with a DBMS without requiring any features of DBMS. Database management systems have a lot of features belonging to data and some DBMS have specific and extreme functions in order to speed up the transactions and data processes. The

method of database independent application software development is based on the principle that some of these features (standard functions like SQL that can be used on every DBMS) can be done in the application layer as well. Figure 1 shows the database dependent application software. In this method, tables and indexes are part of the database. Other relational functions for data: Views, triggers, constraints, stored procedures, users and some other features can be managed by the DBMS (Sahin, 2005).

Figure 2 shows an indefinite database application software graph. The point here is all the further features like views, stored procedures, users, triggers and constraints in DBMS shown Fig. 1, are inside the data module in application layer for the independent application. These relational features in data module can be managed in the application layer. This method uses only database part of DBMS and, database contains just tables and indexes.

In the method of database independent application development, any relational database including tables and indexes can be used in the application group of management system. This way, both the selection of database and the situation requiring database management would be excluded (Sahin, 2005; Hardman *et al.*, 2003; Schlegel, 2004).

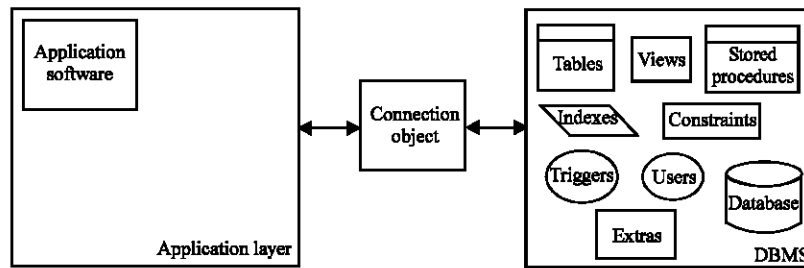


Fig. 1: Database dependent application infrastructure

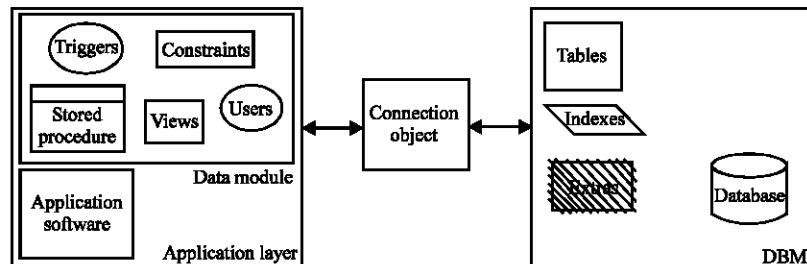


Fig. 2: Database independent application infrastructure

One important point must be emphasized that, extra features which are not common functions (such as DSQL, Query optimizations features, etc.) for most of DBMS is not allowed for the database independent applications. Because, this method must support the application software to run with most of the DBMSs. Therefore, along all developing stages of the application, this basic rule must be obeyed.

### COMPARING METHODS

There are so many differences such as human resources and time and cost for operation and maintenance period, between with database independent or database dependent application software development methods. It is very important to examine these changes with their pros and cons to figure out which method is more useful in which environment.

The application development life cycle changes according to the sector planned and its scale beginning with the planning and going on with the analysis, development, design and maintenance phases in turn. With in this process, human resources, time and cost required for them, should be calculated according to the size of the sector and the works that will be done and the development duration should run by determining the method first.

**Economical evaluation of the methods:** Before assessing and comparing two methods from the point of view of

cost, some critical factors that will be used in assessments of methods must be determined. The most important factors are the size of the market to be served and possible clients' budgets. These factors are the key point and very important, because if the target clients are Small-Medium-Enterprises (SME), then, both the software and DBMS license should have convenient price, otherwise, there is a big risk factor that the software may never be sold. In addition, If the software to be served to complex and specific market especially manufacturing sector, that time, the project needs extra time and effort to make its requirement specification and analysis. Therefore, firstly, the target sector must be considered.

When a software project planned, all possible and necessary products and resources that will be used with the software should be defined. Main necessary product is of course DBMS for a database application. So, the license price of DBMS is very important for the clients. Some of the SME says that there is no extra money for DBMS, or, some of them declare that we already have a DBMS, if it possible to use it for the software, then we can save too much money for both installation and maintenance. This is another key point of the assessment. Table 1 shows prices of some example DBMS. There are a couple of DBMS other than listed in table and that can be downloaded and used freely. At this point, the main subject is that DBMS brings additional economical burden to the clients. If a client is able to use free DBMS or choose the best DBMS according to their work-load, then, some additional payments may be reduced and the application may sold more easily to them.

**Table 1: Some example DBMS expenditures**

DBMS	Version	License <sup>1</sup> for server+10 client	Annual maintenance price <sup>1</sup>	Total <sup>1</sup>
DB2	OLAP S.	20.610	5.000	25.610
Oracle	9i Std. Edition	13.000	4.660	17.660
Ms.SQL Server	2000	2.249	1.000	3.249
Informix	IDS Int. Edition	12.065	4.300	16.365
SyBase	ASE v8.0	8.999	3.000	11.999
Microsoft Access	2003	247	None	247
MySQL	Pro.	Free	Free	Free
Paradox	7.0	Free	Free	Free
PostgreSQL	7.4.3	Free	Free	Free

<sup>1</sup>: Prices (in USD) is gathered form (Oracle, 2005; IBM DB2, 2005a, b; IBM, 2005a, b; MySQL, 2005; Sybase-Aberdeen, 2005; Borland, 2005; PostgreSQL, 2005) at 10/03/2005

The clients may choose any of the DMBS listed in Table 1 freely, if the application developed for database independent platform. This factor should be marked as a first coefficient factor of decision with highest priority.

Another cost factor that is human resources has to be clarified. In this factor, there are two main human resources discussion. One of them is members of developer team in the project, another is maintenance people of DBMS after software is sold. The first human resources have less importance as well as the maintenance members. Since maintenance is the longest phase of the project life cycle, it requires the people to have many skills over the project and other environment products.

If an application software is being developed for database independent platform, there is no need to employ additional people who are DBA or specialist on specific DBMS, since, most of the data processing will be done in application layer. In addition, there is also no need to employ a DBA for maintenance phase. This is another important coefficient factor of decision with high priority.

There is a big disadvantage of independent platform that is project duration. It must be noted that, no part of the database application project can be done concurrently. Because, all the data processing must be held in application layer, so it requires to be worked sequentially. Therefore, using the method of database independent application development, most of the projects take much more time than dependent application projects. This is also another coefficient factor for decision with high priority. In the meanwhile, duration of the project is also depends on the CASE tool that is used in project. Since, most of the CASE tools have no ability to create source code according to independent platforms, coding must be done by project code writers. This is also another burden on sellers and companies. So, this should be taken as a new coefficient with middle-high priority.

Another important factor is failing risk of the project at any coding stage because of not employing of a DBA.

Sometimes, software engineers and coders may not have the good database knowledge and that situation always is a risk for the project. In order to prevent the project form this risk factor, well-educated people (developers, coders, etc.) must be employed in the project. This is another cost for the project, because, employment of well-educated people means additional expenditures and burdens on sellers and companies. Therefore, this situation should be determined as a coefficient with middle-high priority.

The last economical factor is recompilation and reconstruction cost. If any project has been developed with database independent method, that means all the necessary database functions are made in data module in application layer, so, it is enough to compile code one time and, to change database Alias, ADO provider or ODBC deriver for different DBMS. If database dependent method is used for the project, that time, it is required to build all database functions in DBMS for each selected DBMS and, it requires recompilation of the code whenever DBMS is changed. So reconstruction and recompilation can be taken as another coefficient with middle priority.

Finally, the new coefficient factors can be added to the list, in fact, it depends on the project developer company and the team structure of the project. Extension of the factors may help the project leader to select the best way for development.

**Performance evaluation of the methods:** One of the main important factors is system performance of the methods. This factor, of course, is important as much as the other cost factors. In fact, there are many elements that affect the system performance, such as CPU speed, data size, operating system, etc. Although, database independent applications have many advantages, the system performance is lower than other method, because, many of the DBMS products have many additional features that increase the system performance like SQL optimizer, different type of DB engines, etc. But the main problem is, these additional features have not common usage and, all features may differ for each DBMS. Thus, each feature can be used individually according to DBMS used.

Table 2 shows an example of performance results test. This test was applied for three different DBMS. DBMSs were used in the test, are the famous DBMS and they hold the 60% of the world software market. The test includes sample possible data operations that could be in an application.

Test results shows us, speed performance decreases by 14.53%, 14.30%, 15.59 for Ms. Access, Ms. SQL Server and Oracle, respectively. This performance reduction may

Table 2: Performance test results according to methods (Sahin, 2005)

Test type	Tupple count	Ms. Access (sec)		Ms. SQL Server 2000 (sec)		Oracle 9.1.xi (sec)	
		M-1 <sup>1</sup>	M-2 <sup>2</sup>	M-1	M-2	M-1	M-2
DB connection time	0	3.00	3.00	5.00	5.00	6.00	6.00
View test (Small size)	1.151	0.12	0.10	0.22	0.19	0.28	0.23
View test (Big size)	68.308	6.38	5.59	5.98	5.23	4.79	4.12
Fetching	1	0.05	0.05	0.04	0.04	0.04	0.04
Range query (Small size)	1.617	0.18	0.14	0.19	0.16	0.21	0.17
Range query (Big size)	16.297	1.93	1.71	1.43	1.30	1.41	1.29
Insert test	1	0.04	0.04	0.05	0.05	0.05	0.05
Delete test	1	0.08	0.08	0.10	0.10	0.12	0.12
Update test	6.312	0.99	0.82	0.62	0.48	0.59	0.46
Total	93.688	9.77	8.53	8.63	7.55	7.49	6.48

<sup>1</sup>: M-1: Method 1, software developed for independent platform; <sup>2</sup>: M-2: Method 2, software developed for specific DBMS platform

Table 3: Evaluation form that helps the developer to select the best method

Factor	Priority	Possible priority scores	Method-1 (Independent)		Method-2 (Dependent)	
			Coeff.	Total	Coeff.	Total
DBMS freedom (No maintenance cost, license free)	Highest	6-7-8-9-10	1		0.5	
System performance	Highest	6-7-8-9-10	0.5		1	
Less employment	High	4-5-6-7	1		0.5	
Project length	High	4-5-6-7	0.5		1	
CASE tool	Middle-High	2-3-4	0.5		1	
Well-educated employee	Middle-High	2-3-4	0.5		1	
Recompilation and reconstruction	Middle	1-2	1		0.5	
Grand total						

Table 4: An evaluated sample form to select the best method

Factor	Priority	Possible priority scores	Method-1 (Independent)		Method-2 (Dependent)	
			Coeff.	Total	Coeff.	Total
DBMS Freedom (No maintenance cost, license free)	Highest	6-7-8-9-10	1	10	0.5	5
System performance	Highest	6-7-8-9-10	0.5	3	1	6
Less employment	High	4-5-6-7	1	7	0.5	3.5
Project length	High	4-5-6-7	0.5	2	1	4
CASE tool	Middle-High	2-3-4	0.5	1.5	1	3
Well-educated employee	Middle-High	2-3-4	0.5	2	1	4
Recompilation and reconstruction	Middle	1-2	1	2	0.5	1
Grand total				27.5		26.5

be too important for some database applications. This, of course, depends on the company profile that the application will be used in. This performance factor must be taken as another important coefficient with highest priority.

**Overall evaluation of the methods:** Both economical and performance factors (coefficients) can be collected in a simple evaluation form as shown Table 3.

Table 3 can be used to decide which method is proper for the project. In order to use this as a form, firstly the priority scores must be determined in accordance with its importance in the project. Next, the sub-totals are calculated by using the multiplication of the scores selected from priority scores and appropriate coefficient. Next, grand totals are calculated using sub-totals. Finally,

decision may be made observing the grand totals (Greater total shows the better). In the form, no coefficient is given 0, because, every factor has specific importance and must be considered that they are necessary for all projects.

**Example:** If a company plans to create a software that will service SME and it is thought, that the price is the first restriction. At this point, company may fill the form as follows (Table 4).

The totals are 27.5 and 26.5 for independent and dependent methods, respectively. Independent method's grand total is greater than dependent method's total, therefore, independent method may be chosen for the application as a better way.

This form presents flexibility in decision, because, priority scores are determined by producers.

## CONCLUSIONS

Each software project has specific goals and initial parameters such as fund, target clients, profit expectations, etc. These parameters affect the project life cycle and product (application) very much. One of the goals of developers (application) is to create application for database independent platform. This facility supports the product to be sold and distributed more easily. Furthermore, the product may be maintained cheaply. But, the project life cycle for independent platform is not easy, unlike for dependent platform. Both methods have some advantages and disadvantages and all advantages and disadvantages must be assessed and evaluated in order to choose better method. Although, database independent application software development has a little bit advantage than dependent method, this paper does not recommend any method, because the decision should be made by developers according to restrictions and conditions of goals. Proposed form presented in this study can be used to make a decision on which method may be better for a specific application project.

## REFERENCES

- Borland, 2005. Products Shop. Retrieved: <http://amos.shop.com/cc.class/cc?sy=products> and main = ccn-search and okp = 1 and search-form = and ccsyn = 260 and ost = Software 20% Borland and st = BorlandSoftware and ccsid = 376132130-26429 and ccn test = 1.
- Hardman, S., D. Crichton and T. Fouser, 2003. Reusable Software Services For Science Data Systems. 5th International Symposium on Reducing the Cost of Spacecraft Ground Systems and Operations, Pasadena, CA, July, 8-11.
- IBM, 2005a. Informix Dynamic Server Enterprise and WorkGroup Edition prices. Retrieved: <https://www-12.ibm.com/software/howtobuy/buyingtools/paexpress/Express?P0=E1> and part-number=D6DCRLL and catalogLocale=en-US and Locale=en-US and country=USA.
- IBM, 2005b. Informix Extended Parallel Server prices. Retrieved: <https://www-112.ibm.com/software/howtobuy/buyingtools/paexpress/Express?P0=E1> and part-number = D6DAXLL, D5282LL, D6DB5LL, D6DBDLL, D53KKLL and catalogLocale=en-US and Locale=en-US and country=USA.
- IBM DB2, 2005a. Personal edition prices. Retrieved: [https://www-12.ibm.com/software/howtobuy/buyingtools/paexpress/Express?P0 = E1](https://www-12.ibm.com/software/howtobuy/buyingtools/paexpress/Express?P0=E1) and part-number = D5B69LL and catalogLocale = en-US and Locale = en-US and country = USA.
- IBM DB2, 2005b. DB2 UDB Enterprise Server Edition Processor prices. Retrieved: <https://www112.ibm.com/software/howtobuy/buyingtools/paexpress/Express?P0=E1> & part-number = D518GLL, D518JLL, D53Y2LL and catalogLocale=en-US and Locale = en-US and country= USA.
- MYSQL, 2005. Prices. Retrieved: <http://www.globalink.com.sg/mysql/prices/>.
- Oracle, 2005. Oracle for small and midsize businesses products. Retrieved: <http://www.oracle.com/solutions/mid/index.html>.
- PostgreSQL, 2005. Support and Maintenance Service. Retrieved: <http://osb.sra.co.jp/PostgreSQL/Service/service-en.php>.
- Sahin, Y.G., 2005. The Method of Database Independent Application Software Development. Doctorate Thesis, Gazi University.
- Schlegel, K., 2004. Siebel: The Analytics Company?. Enterprise Analytics Strategies. Meta Delta Report. Retrieved: <http://www.siebel.com/downloads/analyst-reports/meta-siebel-the-analytics-company.pdf>.
- Sybase-Aberdeen, 2005. Mass-Deployment Database Embedded in Workgroup Environment Cost-Of-Ownership Study. Retrieved: [www.sybase.com/content/1020326/Sybase2.pdf](http://www.sybase.com/content/1020326/Sybase2.pdf).
- Yu, E.K., 2003. Database Independence! Myth or Reality?. Senior Solutions Architect, Advanced Solutions Group, University of South Carolina, 3-6, Retrieved: [www.asg.sc.edu/pdf/DatabaseIndependence.pdf](http://www.asg.sc.edu/pdf/DatabaseIndependence.pdf).