

Achieving the Efficiency for Information Systems - Possibilities and Prospects

D. Litan, M. Velicanu, L. Copcea (Teohari), M. Teohari, A. M. Mocanu (Virgolici),
I. Surugiu, and O. Raduta

Abstract—Nowadays companies cannot cope with competition successfully unless they have performing information systems. Therefore, the need for efficient information systems is very high. The efficiency can be achieved by using various methods/ techniques and technologies “to build” the information system. In this paper we are going to present and analyze to what extent the new technologies and methods/techniques: methods of integrating the information systems; techniques of databases optimization; “special” types of databases (mobile, multimedia and spatial databases); data warehouses, have increased or could increase the efficiency of information systems.

Keywords— database optimization, data warehouse, Enterprise Application Integration, mobile databases, multimedia databases, spatial databases, Oracle Warehouse Builder.

I. INTRODUCTION

FRACTIONS of seconds can change a person’s life or a company’s destiny as we live in a dynamic world, in a permanent move. Thus, the need of being informed, regardless of place or time, is very great.

However, things have not always been like this: in time, as technology advanced, the companies have developed separate information systems in order to simplify some activities which used to be carried out “manually” and thus a medium-sized company started to hold maybe tens of information systems which were not interconnected, thus having difficulties in updating and administering them. On the other hand, as company grew and extended their activities in different parts

D. Litan is Ph.D. Candidate in Economic Informatics, the Bucharest Academy of Economic Studies, Romania (corresponding author to provide phone: +40 0747-113-795; e-mail: litan_daniela@yahoo.com).

M. Velicanu is with Economic Informatics Department at the Faculty of Cybernetics, Statistics and Economic Informatics from the Academy of Economic Studies of Bucharest, Romania (e-mail: manole.velicanu@ie.ase.ro).

L. Copcea (Teohari) is Ph.D. Candidate in Economic Informatics, the Bucharest Academy of Economic Studies, Romania (e-mail: larisa.copcea@yahoo.com).

M. Teohari is Ph.D. Candidate in Economic Informatics, the Bucharest Academy of Economic Studies, Romania (e-mail: mteohari@yahoo.com).

A.-M. Mocanu (Virgolici) is Ph.D. Candidate in Economic Informatics, the Bucharest Academy of Economic Studies, Romania (e-mail: mocanuaura@yahoo.com).

I. Surugiu is Ph.D. Candidate in Economic Informatics, the Bucharest Academy of Economic Studies, Romania (e-mail: isurugiu@totalsoft.ro).

O. Raduta is Ph.D. Candidate in Economic Informatics, the Bucharest Academy of Economic Studies, Romania (e-mail: ovidiu.raduta@gmail.com).

of the world, other needs emerged: the need for keeping their employees or at least the strategic management permanently informed and connected to the company’s systems, even when they are not at their workplace; the need for the strategic management to have easy and very fast access to the company’s reports as „today’s managers rely more than ever before on management information systems to provide them with the business data needed to make critical decisions.”[1]; the need to process the information in a very short time, from various operators simultaneously and the list can go on. Therefore, we will present and analyze in our paper a number of methods, techniques and technologies that can help increase the efficiency of information systems, such as:

- *methods of integrating the information systems: Enterprise Application Integration (EAI)*: through software integration we manage to get heterogeneous systems to behave as a whole, connected and complete, prepared to interact and communicate in order to exchange and consume information whenever the need arises. The purpose of EAI strategies is to create an integrated, complex, coherent software solution, beyond any geographical, social, national or heterogeneous business types limits;
- *techniques of databases optimization*: in order to improve the performance of database processing can be applied several optimization techniques (stored procedures, indexes, efficient queries);
- and “in terms” of *technology*, we will consider: “*special*” *types of databases* (mobile, multimedia and spatial databases): extensions of normal/old databases, and *data warehouses* (using Oracle Warehouse Builder).

II. METHODS OF ENTERPRISE APPLICATION INTEGRATION (EAI)

Enterprise Application Integration (EAI) is a strong and important issue when discussing about information systems efficiency. Software products are heterogeneous and cover all aspects of businesses, crossing all types of areas, domains, activities, business flows and work procedures. The need of software integration comes as a must for disconnected businesses that need to interact; EAI is the tendency to create complex, robust, efficient, and integrated and, finally, complete software solutions.

A. EAI - Overview

Especially designed for both managerial and operational

purposes, integrated software solutions offer the best support tool for any type of business because they combine efficiently all necessary input to achieve any target.

There are several methods to integrate software products. EAI principles subscribe to architectural styles like Service Oriented Architecture (SOA), Web Oriented Architecture (WOA) and Cloud Computing.

SOA is an architectural style that offers the ability to provide functionality in the form of services. These services communicate by messages and contracts, and implement business functionalities, policies and hold states.

SOA refer to a set of architectural principles for building autonomous services that is based on four main concepts:

1. Explicit service boundaries – both access and implementation must obey service contracts.
2. Services are autonomous – services are independently deployed, versioned, secure and available, responsible for the integrity of the business data they manage.
3. Service interaction contracts are based on message formats and message exchange patterns – the only knowledge a service consumer needs in order to use a service is what messages to send and what messages to receive.
4. Services compatibility is based on policies and common standards such as those of the Web Services Interoperability (WS-I).

The main technical advantages of using SOA, once software products are split into independent services, are the usage of Enterprise Service Bus (ESB) for integration, XML for data exchange, different repositories of data sources and code/service reuse facility. Based on SOA approach resulted WOA concepts that enriched integrated services with Internet capabilities and cloud computing.

Referring to software architecture and design principles, virtualization and modularization will be the main purpose of integration: integration at the data source level, deployment of independent services, loosely coupled and being able to compound themselves into new complex service structures with minimum effort, binary decoupling using versioning, including business entities and business functionalities.

EAI standards are commonly used to fasten development of complex software projects, they improve development and deployment agility and offer conceptually appropriate solutions, open standards at any level, a complete independence between functionalities exposed as services. Some disadvantages and aspects to be taken care with software integration will be related to a detailed management of dependencies, deployment configurations or development efforts and expected big business and technical analysis. Anyway, EAI projects benefits exceed costs in case of complex implementations, so efficiency is guaranteed.

By definition, EAI refers to *“the process of integrating multiple applications that were independently developed, may use incompatible technology, and remain independently managed”*. Within its two basic components, like Business Process Integration (BPI) and Enterprise Information

Integration (EII), EAI it's the most commonly and efficient approach in complex software development.

B. Benefits of EAI solutions

Among the features that standards usage offers in the process of software information systems integration, the following can be included: increased productivity, allowing developers to reduce the demands and time of writing new code for the integration of two or more software products; ability to access real time information and support, regardless of platform used, including mainframe environments, different operating systems, hardware variety, target users, heterogeneous businesses having each of them distinct specific activities; optimization based on costs, allowing interconnected enterprises to reduce time spent to unify search operations, reporting, integration between different information systems. Time spent on writing standards and unifying guidelines and rules leads to work quality, performance and efficiency, it allows customers to accelerate business integration processes, and developers to increase productivity in terms of supporting integration projects. Software integration standards and distributed capabilities also allow enterprises to access and integrate information in real time. Besides software creation, EAI standards also fasten the process of software products delivery, through methodologies and practices used in software development lifecycle stages.

EAI software solutions usually consists of a collection of technologies and services that form a middleware which enables integration of existing information systems and software products across multiple enterprises, areas of activity, interconnected industries. These disparate software solutions are enhanced through the integration process; they become productive and efficient as EAI solutions provide flexibility to create desktop, web and RIA (Rich Internet Application) applications. Anyway, EAI projects benefits exceed costs in case of complex implementations, so efficiency is guaranteed.

Software integration solutions provide tools to fasten software development at least in the following areas:

- Data entry and reporting tools – in order to create, update and display information based on data from the data sources;
- Scripting support – allow specific applications' behavior when events occur;
- Complex data access, data manipulation and data presentation tools – using easier, codeless, graphical experience, this enables enterprises to more easily visualize, analyze and manipulate data;
- Presentation styles for enhanced user experience;
- Data and process modeling - integration architectures facilitates greater synergy between business application descriptions and information systems;
- Data and process mapping tools to assure continuity of data and business process functionalities across multiple applications.

EAI solutions are efficient no matter which development

integration solution is chosen: web services, SOA, WOA, process integration. The main advantages of integrating applications instead of using them independently, given the variety of integration technologies, lead to efficient and complete software solutions. At least from these reasons, EAI solutions are:

- “Efficient and fast: As mentioned in the previous point, Web Services allow applications to be broken down into smaller logical components, which makes the implementation easier as it is done on a granular basis.
- Dynamic: services and processes provide a dynamic approach to integration for EAI by offering dynamic interfaces.
- Widely supported and deployed: The active supporters of Web Services include significant industry participants such as Microsoft, IBM, Intel, Sun Microsystems, Oracle, SAP, and so on. That means the EAI solution is easy to integrate applications across platforms, operating systems, and programming languages.” [5]

Other EAI benefits would be the abilities to access real-time information and to sustain data integrity across multiple systems, automation and composition of business processes from different information systems that need to interact and communicate on both level of input and output.

However, to gain the benefits of EAI solutions, an organization must implement technologies that deal with the problems presented by this architecture:

- Interoperability: the various components of the infrastructure may use different operating systems, data formats and languages
- Data integration: in order for a modular, distributed system to be functional, a standard method of handling the flow of data between applications and systems to enforce consistency across the data sources is crucial
- Robustness, stability and scalability: because they are the components that hold together a modular infrastructure, integration solutions must be highly robust, stable, and scalable.

Given all these, organizational efficiency is no longer a distant goal with EAI technologies, but an immediate one, especially for complex information systems.

III. TECHNIQUES OF DATABASES OPTIMIZATION

Namely, talking about how to improve processing running time, the way to decide which keys to be used in the query, is called the query optimizer.

“The query optimizer is the component of a database management system that attempts to determine the most efficient way to execute a query. The optimizer considers the possible query plans for a given input query, and attempts to determine which of those plans will be the most efficient. Cost-based query optimizers assign an estimated <<cost>> to each possible query plan, and choose the plan with the smallest cost”.[8]

For analyzing and optimizing the performance of a query

the measurement and collection of statistical data is inevitable. Indexes should be kept as slight as possible, to reduce the amount of processing related with each query. The statistical performance data about the index keys are maintained in DB server. If it’s configured, non-indexed keys can also be measured statistically.

The main statistical concepts, to be understand regarding performance optimization, are: *cardinality* (how many unique values exist in the data set), *density* (is determined by dividing the number of rows that correspond to a given key by the number of rows in the table and measures the uniqueness of values within a data set) *selectivity* (is determined by dividing the number of keys requested by the number of rows they access and measures the number of rows that will be returned by a particular query). To calculate the relative cost of a query plan, the optimizer needs a valid selectivity measure. DB server automatically updates the statistical information periodically as the data in the tables change. Sampling the data, rather than analyzing all of it minimizes the cost of this automatic statistical update.

Optimal performance: designing and choose indexes in a complex database table is normally an inefficient task. DB server has a built-in Index Tuning Wizard to establish an optimal set of statistics and indexes. Thus, running it, a list of suggestions based on a scripted workload will be provided in order to improve the query performance of the database. Generally it is enough to implement only the recommendations of the wizard, applied to a particular situation. But for a dynamic system, it is necessary to update periodically our analysis on optimal database performance. Therefore, next, we will present, some techniques of databases optimization.

A. Stored procedures

Often, quick access to a database is observed when running stored procedures. These play a vital role in enhancing the performance of the database.

One way to optimize table access is using “*nolock*” option; considering that transaction safety is not mandatory for most of the database access, this is apparent in database product that it does not contribute to any record locking capability. Another hint is to use “*nocount*” option. For each operation, stored procedures returns information concerning the number of rows affected. If we turn on nocount option, stored procedures will block row count information so that it will reduce the network overhead occupied with information communication to the user.

Using return values – a query is used to validate information or to return a single value. The returned value of a stored procedure can be used to retrieve a single value from a query and mainly useful when inserting a new record. The information necessary for retrieval is the primary key value. Place “(*\$result*)” like argument (e.g.: *myDataBase_free_result(\$result);*) at the end of the query to utilize the return value in stored.

B. Indexes

Another advantageous method for increasing the performance of queries is the indexing (efficient indexes). Thus, it will be avoided the necessity of scanning the entire table for results. The optimizer module of all database management tools focused on the idea that “*the INDEX hint explicitly chooses an index scan for the specified table. Optimizer chooses the most cost profitable index in case of using more indexes*”[9].

If you have created an index, database server (DB server) will automatically measures and stores the equivalent statistical information regarding the distribution of values in the indexed column. The optimizer to determine the optimal strategy for evaluating a query uses this statistical information.

There are two types of indexes: clustered and non-clustered, each with some unique advantages depending on the data set. A clustered index dictates the storage order of the data in a table. When searched for range of values the clustered indexes are more efficient on columns of data as the data is sorted. These index types also shine at finding a specific row when the indexed value is unique. Using a non-cluster index, the data is stored in one place and the data value in another place. When a query search for a data value, first the query searches the non-cluster index for the corresponding data value in the table to find the location and then retrieves the data from the location. Most often non-cluster indexes are used for the queries resulting in exact matches.

In most cases in practice, it is necessary to analyze which indexes are the best to use. For example, supposing that we are searching for a book (Author X, Title Y), where there are 2 indexes, one alphabetical by X, and the other by Y. At a glance, you see that there are 30000 unique authors, and 500000 unique books, so you decide to search by author. But if you knew that there were 60 books written by author X, and that book (Y) is the only one whose name begin with letter “T”, your search criteria would change. You can provide similar information for the Optimizer by running.

```
ANALYZE TABLE tablename;
```

This stores the key distribution for the table (running ANALYZE is equivalent to running `myisamchk -a` or `myismachk --analyze`).

As we know, many deletes and updates leave gaps in the table (especially when you're using varchar). This means there are more unnecessary disk I/O's, as the head needs to skip over these gaps when reading. Running “*OPTIMIZE TABLE tablename*” solves this problem. Both of these statements should be run fairly frequently in any well looked after system. To take advantage of *short indexes*, is another factor that most people don't use when they try to use indexing. More exactly, it is not necessary to index on the entire field. Applying it on the next example, it is possible to reduce the size of your index

Example: our surname and firstname fields are 40 characters each. That means the index we created above is 80

characters. Inserts to this table then also have to write an additional 80 characters, and selects have 80 character blocks to maneuver around (in addition, it's interesting to remind that the disk I/O is the primary hardware bottleneck).

```
ALTER TABLE employee
ADD INDEX (surname(20), firstname(20));
```

Applying this, our updates write to an index half the size, and selects have a smaller index to search. Both will be faster (unless you make the indexes too short - imagine a book index, instead of giving the full word, only contained the first letter of the word.

Generally, they apply this on the original field definitions. It is true that nowadays we have an ample disk space, we don't often worry about space. But we have to admit that smaller usually means faster, Therefore, defining our surname and firstname fields as CHAR (255) could be a wrong decision if we know from the beginning that the biggest firstname is never more than 20 characters. On the other hand, we don't want to cut names off, but remember that you can ALTER the field later if conditions change, and you need to give permission for more characters. Here, a suggestion is using VARCHAR instead CHAR (it is more advantageous to use variable length characters than fixed length characters).

C. Efficient queries

Most systems need to be highly optimized for selects – for example, a news site which performs millions of queries per day, but where the data arrives in large batches of text files. So, at certain times of the day, inserts need to be optimal, without noticeably affecting the millions trying to access the data.

Assuming a nicely formatted ‘|’ delimited text file that we want to insert into the table above, take this piece of PHP code:

```
if (!$fp = fopen("datafile.txt", "r"))
// open the file for eading
{
print "\nUnable to open file.txt";
// display error
exit();
// end the running of the program
}
while (!feof ($fp))
// loop through the file line by line
{
$line = fgets($fp, 4096);
// put the data into the variable $line

$line = chop($line);
// remove the newline

list($no,$fname,$sname,$age,$job) = split("|",$code);
// split the line on "|", populating
```

```
// the ind. variables

$db->query("insert into
employee(employee_number,firstname,surname,
age, job) values($no,$fname,$sname,$age,$job)");
}
// end while loop
```

This would work, but would be very slow. After every insert, the index buffer would be flushed. Until NOLOCK new option, they did not allow data to be inserted at the same time as being read. Now it is possible, but only if there are no deleted records present (which is unlikely in a heavily used system). Consequently, the entire table is locked for the duration of each insert. There's a solution however - the best way to insert the data is to use "LOAD DATA INFILE" option. This is much faster (20 times according to MySQL). The syntax is quite simple:

```
$db->query("LOAD DATA INFILE 'datafile.txt'
INTO TABLE employee
(employee_number,firstname,surname,tel_no,salary)
FIELDS TERMINATED BY '|')");
```

If nothing will be specified, LOAD DATA INFILE will have defaults of: FIELDS TERMINATED BY '\t' ENCLOSED BY ' ' ESCAPED BY '\\'. And, like an ordinary insert case, you need to specify a field list if the order of the fields is different, or (as in the example), you are not inserting data for every field. Always, to specify a field list is a good practice for all queries anyway (for example, if someone will add later a field to the table, you do not want to review it again in order to fix all your previous INSERT and SELECT statements.

If it is not about inserting from a text file, there are ways to make sure that the users selecting are not badly affected. One suggestion is to use INSERT LOW PRIORITY. This will wait until there are no more reads waiting to happen, or waiting for the gap, and not pushing in as it were. Another one, is to use INSERT DELAYED. In this case, the client is immediately free and the insert put into a queue (with all other INSERT DELAYED's still waiting for the queue to end).

This means that, when the client is freed, there can be no important information passed back to the client, as the INSERT has not been processed. Also, it is important to say that a catastrophe such as an unexpected power failure here will result in the queued INSERT's being lost. So, for all these methods you won't have any idea when the data is inserted.

But getting data it is not the only action that needs to be quick - sometimes you need to get it out quickly too. For example: Rather it is recommended to use "TRUNCATE TABLE *classifieds*" instead of "DELETE FROM *classifieds*". The difference is that DELETE drops records one by one, and that can be too slow if we have million records. Many database management tools made DELETE remove records one by one so as to return the number of records deleted, but

TRUNCATE still did the quick delete.

Often, the best way to improve is to try and rewrite the query in as many different ways as possible, and see which one runs more efficiently. You can learn a lot by trying to figure out why one alternative ran faster than the other.

IV. DATABASES TYPES USED TOGETHER WITH MODERN INFORMATION SYSTEMS

If you were to choose between a fixed phone and a mobile one (both with the same voice features and quality), what would you choose? There is no doubt that most people would choose a mobile phone that offers more opportunity for movement during the telephone conversation. Exactly the same thing happens with databases. Of course, we can use the common databases to replace the information systems in one way or another, "special" databases, but the high costs reflected by the time of processing the information, which is often very long, or by a low management of the information, raise the problem of replacing, where appropriate, the common databases with some "special" ones, dedicated to certain areas.

A. Mobile Databases (Mobile DBs)

As technology keeps developing very fast "day by day", at present office computers are being replaced more and more by much smaller gadgets which are easy to use and transport such as laptops, mobile phones, PDA's (personal digital assistant), smart phones, etc. But „a key contributor to this success is a database's ability to provide a consistent view of shared data across many geographically dispersed users, even in the face of highly concurrent updates at fine granularity.”[14]

However, the using of a mobile gadget brings also, some constraints: „*limited resources, limited energy, disconnection, bandwidth asymmetry.*”[15]

Consequently, it is easy to understand that both applications and databases for mobile gadgets have a number of "specific" features compared to the "classic" ones meant to improve the gadget performance and to increase its efficiency.

“A mobile DB is a database that can be connected to by a mobile computing device over a mobile network. The client and server have wireless connections. A cache is maintained to hold frequent data and transactions so that they are not lost due to connection failure.”[16] „The client application can work with the mobile DB asynchronously, and needs to connect to the central database only when it is necessary to synchronize.”[17] Thus, different from the applications that are using "classic" databases where the connection to the server (where the data base is to be found) is absolutely necessary for the proper functioning of the system, here a permanent connection to the database server is no longer an absolute necessity. This is the main advantage of using applications connected to mobile DBs. “Mobile DBs have benefited through the rapid advancement of global positioning systems (GPS) and geographic information system (GIS), as well as the mobile device itself.”[18]

From a thorough analysis of this "trump" of the mobile DBs

we can easily infer other benefits:

- “*easily protects user privacy* because push applications run mostly at the client machine and client’s profile”[19];
- “*flexibility and reliability*: asynchronous operation makes the application more flexible and tolerant to network failures”[17];
- *saving on network costs*: the fact that an application that uses mobile DBs can work and then disconnect from the network leads to the reduction of costs caused by the use of the network.

The architecture of a system which uses mobile DBs is made up of “three main parts:

a) *fixed hosts*: perform the transaction and data management functions with the help of database servers;

b) *mobile units (MU)*: are portable computers that move around a geographical region that includes the cellular network (or “cells”) that these units use to communicate to base stations (BS);

c) *BS*: are two-way radios, installations in fixed locations, that pass communications with the MU to and from the fixed hosts: mobile phones, portable phones, or wireless routers.”[20] (Fig. 1)

Among the products that are now available on the market for using the mobile DBs technologies, we would like to mention: „Sybase Inc.’s *SQL Anywhere*, IBM’s *DB2 Everyplace*, Microsoft *SQL Server Compact* and Oracle9i *Lite*.”[16]

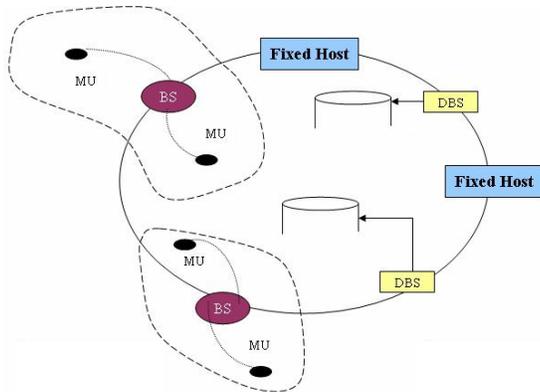


Fig. 1 – Mobile DB System Architecture; source: [20]

B. Spatial Databases (Spatial DBs)

The spatial DBs are a recent development in the databases software. They are the bases of the very complex geographical information systems and could be successfully used in many areas: urban planning [21], market research [22] or even in top business such as the implementation of wind parks [23].

Usually, they are extensions of normal/old databases which contain special objects, procedures, mechanisms in order to store and manipulate geospatial data. Geospatial data means data which describes the location of an object on the surface of the Earth, so a spatially-enabled database must store information about: how the object looks (the representation of the object mainly as points, lines or polygons), where is it

located (the longitude and latitude) and in which spatial reference system the data is given. In order to store this information in one single column and to be able to work with it, the database must have defined a special geometry object with several mandatory attributes and methods defined in the Open GIS Consortium (OGC) standard “Application objects” [24] (Fig. 2). The main methods defined by OGC for the geometry object are: `GetCoordinateReferenceSystem()`, `GetBoundary()`, `GetDistance()`, `GetCentroid()`.

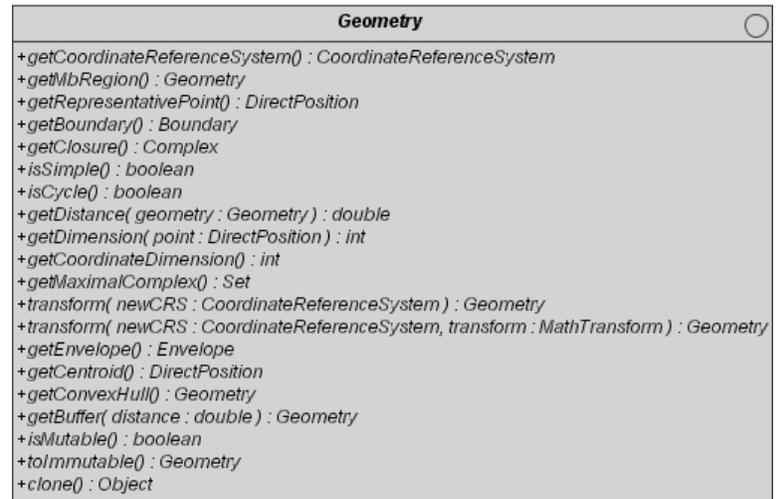


Fig. 2 - OGC Geometry class; source: [24]

Oracle database has a special schema named MDSYS where all the spatial objects are stored. The storage format for the spatial data is the Oracle object `SDO_GEOMETRY`:

```
CREATE TYPE sdo_geometry AS OBJECT (
  SDO_GTYPE NUMBER,
  SDO_SRID NUMBER,
  SDO_POINT SDO_POINT_TYPE,
  SDO_ELEM_INFO DO_ELEM_INFO_ARRAY,
  SDO_ORDINATES DO_ORDINATE_ARRAY);
```

PostGIS extension for the open source PostgreSQL database stores the spatial object in the standard well-known binary (WKB) format, while an other open source database, MySQL stores it in a proprietary way.

The Well-known Binary Representation for Geometry is obtained by serializing a geometric object as a sequence of numeric types drawn from the set {Unsigned Integer, Double} and then serializing each numeric type as a sequence of bytes using one of two well defined, standard, binary representations for numeric types (NDR, XDR) [25].

Among the usual database constraints (primary keys, foreign keys, semantic constraints, and user defined constraints), the spatial DBs are also characterized by the topological consistency relations, which are listed in paper [26] as follows:

- “Everything must be bounded by two nodes (start node and end node).
- For every arc, there exist two polygons (left polygon and

right polygon).

- Every polygon has a closed boundary consisting of an alternating sequence of nodes and arcs.
- Around every node, there exists an alternating closed sequence of arcs and polygons.
- Arcs do not intersect except at nodes. “

Store the information is not enough, one must retrieve the information also, in real time. That is why the spatially-enabled databases also have defined a special type of indexes, named RTree indexes, used by all the three mentioned spatial DBs. This index approximates each geometry with a *Minimum Bounding Rectangle (MBR)* and “when there are more geometries, an Rtree index consists of a hierarchical indexing of MBR rectangles” [23]. The Postgres query optimizer will consider using an RTree index whenever an indexed attribute is involved in a comparison using one of the following geometric operators: <<, &<, &>, >>, @, ~=, && which are explained in below table:

Operator	Description
&&	Overlaps?
~=	Same as?
@	Center
<<	Is strictly left of?
>>	Is strictly right of?
&<	Does not extend to the right of?
&>	Does not extend to the left of?

Table I - Geometric operators

Spatial analysis functions are very important feature of these databases and the one that makes a difference between the databases. For example, the most advanced spatial DBs from this point of view is Oracle Spatial, which offers a special package SDO_SAM with many subprograms which can be used in spatial analysis and data mining, such as:

```
AGGREGATES_FOR_GEOMETRY,
BIN_GEOMETRY,
COLOCATED_REFERENCE_FEATURES,
SIMPLIFY_GEOMETRY, SPATIAL_CLUSTERS.
```

PostGIS does not have a special package with functions to be used in the spatial analysis but it can be done by writing some smart SQL queries using the spatial functions ST_DISTANCE, ST_INTERSECTS, ST_OVERLAPS [22]. For example, in PostgreSQL, the below SQL statement gives you the nearest 5 neighbors from a reference row with gid = 1.

```
SELECT g1.gid As gref_gid,
g1.description As gref_description,
g2.gid As gnn_gid, g2.description
FROM some_table As g1, some_table As g2
WHERE g1.gid = 1 and g1.gid <> g2.gid
ORDER BY ST_Distance(g1.the_geom,g2.the_geom)
LIMIT 5
```

Oracle Locator and Oracle Spatial, proprietary spatial

database, does have an SDO_NN and SDO_NN_DISTANCE functions.

```
SELECT g1.name
FROM some_table g1, some_table g2
WHERE SDO_NN(g1.geometry, g2.geometry) = 'TRUE'
AND ROWNUM <= 5;
```

PostGIS function ST_TRANSFORM uses the open source PROJ4 library in order to reproject the geospatial data from one spatial reference system to another. PROJ4 is a cartographic projections library used by many other GIS tools to reproject the geospatial data from one spatial reference system (SRS) to another. This function returns a new geometry with its coordinates transformed to spatial reference system referenced by the SRID integer parameter. The destination SRID must exist in the “spatial_ref_sys” table. For example, to create a view with all the POINT geospatial data, from different SRS, reprojected in Web Mercator SRS, below statement could be used:

```
CREATE OR REPLACE VIEW vw_point_google AS
SELECT some_table.some_column ,
st_transform(some_table.geom, 900913) AS geom
FROM some_table
WHERE geometrytype(some_table.geom) = 'POINT'::text;
```

The Well-Known Text (WKT) representation of a certain SRS is found by using select statements such as:

```
SELECT srtxt FROM spatial_ref_sys
WHERE srid = 4326;
```

```
“GEOGCS[“WGS84”,DATUM[“WGS_1984”,SPHEROID[“WGS
84”,6378137.298,257223563,AUTHORITY[“EPSG”,“7030”]],TOW
GS84[0,0,0,0,0,0],AUTHORITY[“EPSG”,“6326”]],PRIMEM[“Gre
enwich”,0,AUTHORITY[“EPSG”,“8901”]],UNIT[“degree”,0.01745
329251994328,AUTHORITY[“EPSG”,“9122”]],AUTHORITY[“EP
SG”,“4326”]]”
```

The spatial DBs have become very robust, reliable and more user friendly in recent years, but they still need improvements on all these aspects. New extensions in the SQL standard, oriented towards the spatial analysis, such as CLUSTER_BY, proposed in paper [27], would increase their popularity.

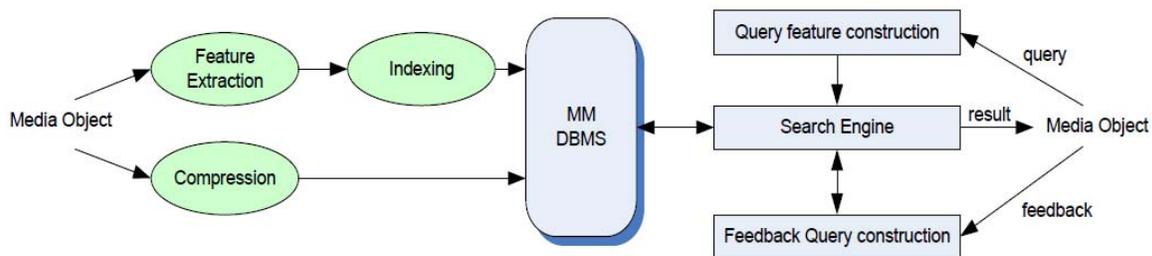
C. Multimedia Databases

In a modern world like “today’s world”, the image, the film or the sound played on digital media have become usual “things” in our everyday life. Nowadays, “information” in the form of multimedia files (be it image under various forms: pictures, scanned documents, etc., be it personal or cinema productions films, be it music, and so on) have widely spread and can be found on almost any computer. And because these files are also used in information systems, the need of retrieving and managing the information contained in

multimedia files has led to the emergence of a new technology, the multimedia database technology.

“A multimedia database is a database that hosts one or more primary media file types such as: .txt (documents), .jpg (images), .swf (videos), .mp3 (audio), etc.”[28] Like any database, the multimedia databases also provide consistency, concurrency, integrity, security and availability of data.” [29]

The multimedia databases are used by a number of software applications, such as: “digital libraries, manufacturing and retailing, art and entertainment, journalism and so on. [...] Multimedia databases are required to provide unified frameworks for storing, processing, retrieving, transmitting and presenting a variety of media data types in a wide variety of formats. At the same time, they must adhere to numerical constraints that are normally not found in traditional databases.”[29]



The most common types of multimedia databases are: *relational multimedia databases (Relational DBMS)*, *object-oriented multimedia databases (Object-oriented DBMS)* and *object-relational multimedia databases (Object-relational DBMS)*. According to [30], the main features of these types of databases are:

a) Relational DBMS store everything as First Normal Form tables:

- all data items are atomic and are held in rectangular tables;
- data can only be related if they are in one or in two records connected by a common value (foreign key);
- records are identified only by content;
- it is difficult (if not impossible) to extend the set of domains.

b) Object-oriented DBMS store everything as classes of objects:

- all data is held as components of objects (like Java variables);
- data is related by object reference (i.e. one class variable has a type which is another class and the values of that variable are instances of that class);
- the set of classes is extensible and so you can freely create domains.

c) Object-relational DBMS are fundamentally relations but are not First Normal Form:

- the values in cells can be object references as well as atomic values;
- new types can be defined.

But what can be done with a multimedia database? A simple answer might be that it is possible to carry out some operations which are relatively similar to those performed in a common database, but on media type object / multimedia, not text. These operations can be :

- “find similar images from the database;
- identify the position of a short query video in a long video;
- retrieve the most similar video clip with reference to the query clip”[31].

But given the fact that the multimedia databases are very large, the process would be much more difficult. A generic architecture that points out the process a media/multimedia object has to undergo in order to be used by an information system is represented in Fig. 3.

Fig. 3 - A generic architecture of a multimedia database; source: [32]

In order to be able to store multimedia data in common/commercial databases, besides the usual data types (string, int, real etc.) the existence of the possibility to define fields of image, video, audio types, in tables, is required. In fact, an extensible multimedia database (*extensible DBMS*) is necessary. “*Extensible DBMSs offer facility to provide new data types, along with functions that operate (e.g., “display”, “comparing”, etc.) on them.*”[32].

The media / multimedia data representation in the multimedia databases can be done as follows: “most current DBMSs provide *three different kinds of domain for multimedia data*:

- *Large object (LOB)* data types used to store sequences of unstructured data up to 4GB; two types:
 - *Binary Large Objects (BLOBs)* which are an unstructured sequence of bytes;
 - *Character Large Objects (CLOBs)* which are an unstructured sequence of characters.
- *File references*, instead of holding the data, a file reference contains a link to the data (*OLE in Access*).
- *Genuine multimedia data types*(e.g., Oracle, IBM DB2, and Jasmine).”[32]

In the following example, [32], we will show how to create a table that can store a multimedia type object (a) and then a *select* type application using as a query condition the similarity between an image from the database and a user-selected image (b).

- (a) Create table EMPLOYEE (
empN integer PRIMARY KEY;

name char(50);
age integer;
salary float;
face image);

- (b) Select name
From EMPLOYEE
Where **EMPLOYEE.face similar desirableFace**

D. "Special" Databases in practice

If we were to customize the influence and utility of "special" databases use on a particular type of information system, we would surely choose safety, that is e-government systems. Firstly, because this type of information system currently represents a trend in the flow of carrying out public administrative activities worldwide. Another reason is that, although research concerning the improvement of e-government systems is continuously being carried on, they have been used by citizens (in 2009 according to [33]) at their true value only in a few countries (at least at European level). And last but not least, because e-government systems require (on different modules) connections to "special" databases for increased efficiency. Therefore, we will further present how "special" databases can influence the performance of e-government systems:

- a) *Mobile databases and e-government information systems:*
How could mobile database help increase the performance of an e-government system? Undoubtedly, the answer is: m-government. The mobile databases technology made it possible to extend the use of information systems from the level of common computers to mobile devices (laptops, mobile phones, PDAs, smart phones, etc.). M-government systems are less common in the world today, but the benefits of their implementation and use (only by the fact that they can be accessed from anywhere, anytime, even offline due to the mobile database technologies) turn these systems into a strong "tool" for working with public administration.
- b) *Spatial databases and e-government information systems.*
When working with computer systems that are "meant" to represent various geographical areas under the form of maps, the spatial databases are the answer to a quick and efficient processing of information. In terms of e-government systems, this technology (spatial databases) can prove its utility in the representation of geographical areas (under various aspects / according to certain criteria) related to the local administration to which that the e-government system belongs.
- c) *Multimedia databases and e-government information systems.* Modules of e-government systems connected to the multimedia databases can be used successfully, for example in processing the exchange of scanned document between citizens and public administration.

Be it either "classic" databases or "special" database, they have undeniably proved their utility in working with computer systems over time. In the absence of a database, basically, we cannot speak of an information system for business or for the state institutions. „With the advent of the information age, more and more information applications based on database

system has been developed and deployed. Due to the historical reasons of development of database technology, various enterprises adopt different database system, or even various business sectors in one enterprise using different database systems. Now the Integrated application system in enterprise has to be faced the heterogeneous database problem.”[34]

V. ORACLE WAREHOUSE BUILDER (OWB)

OWB is a full-featured data integration, data warehousing, data quality and metadata management solution designed for the Oracle database (Oracle DB). OWB is an integral part of Oracle DB and it is installed as part of every database installation. You need to run the installer to install Oracle Workflow though.

A. OWB Background

OWB provides data quality, data auditing, fully integrated relational and dimensional modeling, and full life cycle management of data and metadata.

OWB supports data integration and management activities including:

- *providing extraction, transformation, and loading (ETL);*
- *consolidating data from disparate data sources;*
- *migrating data from legacy systems;*
- *data modeling of relational and dimensional structures;*
- *designing and managing corporate metadata;*
- *cleaning data to provide quality information;*
- *profiling and auditing data quality.*

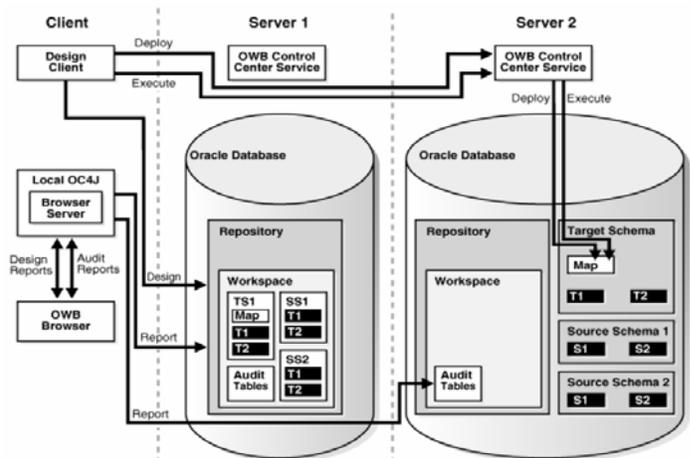


Fig. 4 - OWB Components; source: [35]

OWB supports sources from: Oracle DB releases 8.1 and later, databases accessible through Oracle Heterogeneous Services (Gateways), data stores accessible through the Code Templates (which use JDBC) as DB2, DRDA, Informix, SQL Server, Sybase, and Teradata.

The data in the data system that you implement with OWB is stored in target schemas. This data is in the form of data objects such as tables, views, dimensional objects, and cubes.

OWB can be extended to manage metadata specific to any application, and can integrate with new data source and target

types, and implement support for new data access mechanisms and platforms.

OWB adds value as a solution for data integration, data movement, and data quality.

You can use OWB to:

- Model data and perform extraction, transformation, and loading (ETL);
- Manage the full life cycle of data and metadata;
- Maintain data quality using data profiling and cleansing;
- Integrate with Oracle Business Intelligence Enterprise Edition for reporting.

OWB provides data quality, data auditing, fully integrated relational and dimensional modeling, and full life-cycle management of data and metadata.

The major steps in modeling, building, and populating a data warehouse are:

1. *Design target objects.*

Model the structure (metadata) of the target data warehouse objects.

2. *Design sources and extract source metadata.*

Model the structure (metadata) of the source objects.

3. *Map sources to targets.*

Model the mapping of data from the source objects to the target objects. Specify transformations to transform and clean the data.

4. *Validate, generate, and deploy the code.*

Validate and generate scripts from your design. Deploy those scripts for building physical warehouse objects and extracting and transforming data.

5. *Derive and deploy the BI reporting infrastructure for OBI SE or OBI EE.*

Integration with reporting tools enables easier development of business intelligence information systems.

You can use OWB to derive and manage the BI metadata and then use OBI EE tools to build reports on the existing data warehouse.

6. *Extract, transform, and load data.*

Execute the mapping scripts to extract source data and load it into the warehouse or data mart.

B. OWB Quality and Security

The core capability of OWB is to deliver quality information at the right time to the analytic user.

Data can only be transformed into actionable information when you are confident of its reliability. Before you load data into your target system, you must first understand the structure and the meaning of your data, and then assess the quality.

Using OWB's option Data Profiling is good to better understand the quality of your source data. With the Data Profiling Option, you can correct the source data and establish a means to detect and correct errors that may arise in the loading of transformed data.

In the quality assessment phase, you determine the quality of the source data. The first step in this phase is to import the source data into OWB. You can import metadata and data

from both Oracle and non-Oracle sources.

After you load the source data, you use data profiling to assess its quality. Data profiling is the process of uncovering data anomalies, inconsistencies, and redundancies by analyzing the content, structure, and relationships within the data.

The quality design phase consists of designing your quality processes. You can specify the legal data within a data object or legal relationships between data objects using data rules.

As part of the quality design phase, you also design the transformations that ensure data quality. These transformations could be mappings that are generated by OWB as a result of data profiling or mappings you create. The quality transformation phase consists of running the correction mappings you designed to correct the source data.

After data has been transformed into quality information, OWB integrates with business intelligence enterprise tools, so that you can easily create reports that transform this information into knowledge. *The organization applies this knowledge to business opportunities and problems to receive the benefits.*

Data transformation is the term for converting data from a source data format into a destination data format. Data transformations typically require two steps:

a) *data mapping* (from source to target) to capture any transformations that must occur, and

b) *code generation* to create the actual transformation process.

After you import your source data and define the target, you decide how to transform the source data into the output desired for the target.

Transformations are PL/SQL functions, procedures, packages, and types that enable you to transform data. You use transformations when designing mappings and process flows that define ETL processes.

Mappings provide a visual representation of the flow of the data and the operations performed on the data. Based on the ETL logic that you define in a mapping, OWB generates the code required to implement your design. OWB can generate code for the following languages: PL/SQL, SQL*Loader, SAP ABAP, Code Templates (CT mappings).

OWB provides a dynamic workspace for your projects and one, common look-and-feel for all editors, including automatic layout, dockable panels, and zoom capabilities.

OWB enables you to design security on the metadata you store in the design repository. The design repository is an Oracle DB with users, roles, and access privileges already defined. OWB metadata security operates in addition to the Oracle DB security. The Oracle DB provides security for data while OWB provides security for the metadata.

OWB provides a way to create data auditors, which are processes that provide data monitoring by validating data against a set of data rules to determine which records comply and which do not. Data auditors gather statistical metrics on how well the data in a system complies with a rule by auditing and marking how many errors are occurring against the

audited data.

C. OWB Architecture

A high-level overview on OWB architecture shows a progressively work from design to the run-time environment.

In the design time, you use the design client to get metadata from various sources. That is reverse engineering in development cycle terminology. Then you perform dimensional modeling to define your relational and dimensional objects and create data flows and transformation logic in the form of mappings and process flows.

At run time, you use the Control Center Manager to deploy the various objects, mappings, and process flows to probably a test environment. You run the mappings or process flows to test that the data is transformed and loaded as per the requirements. Finally, using the configuration and configuration templates, you configure the objects and access the production environment and redeploy and rerun the data flows to create the data warehouse.

Various components of the OWB architecture can be:

1. OWBSYS Repository

This is a preseeded schema containing database objects. It is created when you install the Oracle Database. The repository stores and maintains the metadata of OWB database objects, for example tables, views, PL/SQL packages, and so on.

In typical production situations, the repository can be present on two machines.

One repository will store the workspace containing the design metadata; the other repository will contain a workspace that stores only the run-time information for the deployed objects in audit tables.

2. Workspace

A workspace comprises a workspace owner, and probably one or more workspace users. The workspace stores metadata definitions for your design metadata. In addition to containing design metadata, a workspace contains the run-time data generated by Control Center Manager and the Control Center Service. As part of the initial installation of OWB, you use the Repository Assistant to define one or more workspaces.

Each workspace is associated with at least one workspace owner. The workspace owner has all the administrative privileges for administering the workspace and its users. Multiple identifiable workspace users can access the OWBSYS repository schema when they are registered by the workspace owner. Traditionally, the workspace user has privileges to design, deploy, and execute various OWB objects and could also be enabled as a target user.

3. Control Center Service

The Control Center Service enables you to register locations and deploy objects from Control Center Manager, and provides the ability to execute ETL mappings and process flows in the context of a database. A Control Center

Service is like an “execution agent,” outside the database, which can execute and monitor things that cannot be run directly in the database, such as SAP ABAP scripts, SQL*Loader, and shell/batch scripts. The Control Center Service dispatches requests such as “Deploy object X. Execute object Y.” to the workspace and also manages the request—for example, waits for the request to complete and then might update an audit record in the run-time workspace. The Control Center Service on the execution database server will be used to deploy and execute the code.

4. Control Center

A control center manages a set of source and target locations. A location can be a target for only one control center but can be a source for many control centers.

5. Configuration and Configuration template

Configurations help you easily manage multiple deployments of same set of objects with different physical configurations to test or production environments. A default configuration is automatically created to manage the default configuration settings on the objects and is associated with a default control center, which is also created. Each configuration points to exactly one control center. A configuration indicates which control center it is using, and consequently at which location. The active configuration is set on a per-user, per-project basis. You can associate a configuration template with a configuration. A configuration template is a way of setting defaults for a set of configurations. That is, it is a way of overriding the seeded defaults for many object types in the repository. The templates help you establish default values, which can easily be repeated and slightly modified.

6. Repository Browser

Repository Browser is a Web browser interface for reporting on design and run-time repository information. You can view the metadata, create reports, audit run-time operations, and perform lineage and impact analysis. You can also start processes and mappings from the browser, and drill down to see why certain processes might have failed.

Repository Browser can run as a stand-alone client without Application Server, using local OC4J containers. From the local OC4J, you can connect to the server hosting the repository. There is no need to run the browser on the server.

D. Managing Metadata in OWB

The lineage and impact analysis tools of OWB enable us to uncover and fix the ripple effect of changes in metadata, anywhere in our system, before these changes break our system. We can use these tools to proactively managing change in our system.

Changes may come into the repository because of various reasons: source re-import, changes required by end users, or because the system was not ready yet. In any case, use lineage and impact analysis to resolve the impact of the change.

OWB assists you in reconciling changes (as you make the changes to your design) to dependent objects by providing a change propagation tool, as well as lineage and impact analysis.

You could examine the results of impact analysis diagrams, and then manually edit each dependent object downstream of an object whose attribute changes, or you can use the Propagate Change dialog box to automatically change the dependent objects that are downstream.

In the Propagate Change dialog box, you specify metadata changes, which OWB propagates to all dependent objects, as indicated in the Lineage Impact Analysis diagram.

E. OWB and Oracle Business Intelligence Solution Enterprise Edition and Standard Edition (OBI EE and OBI SE)

By implementing business intelligence objects, OWB can be integrated with reporting environments seamlessly.

Integration with reporting tools enables easier development of business intelligence information systems. The Business Intelligence node in the Projects Navigator panel enables you to define sets of OWB objects as business intelligence objects. These business intelligence definitions are deployed to the target reporting environment after they are configured, validated, generated, and deployed by the OWB client.

In OWB, the repository is the center for metadata. Consolidating all metadata in a single place allows OWB to deliver attribute-level lineage and impact analysis from the source to various query tools. You can use OWB to derive and manage the Business intelligence metadata and then use OBI EE tools to build reports on the existing data warehouse. It also becomes easier to determine the impact of changes all the way from the source to the end-user targets because there is a single metadata source.

OWB enables you to create object definitions that represent objects in the OBI SE End User Layer (earlier known as Discoverer).

The objects can be designed or derived and then deployed into OBI SE.

OWB eliminates the duplication of effort and metadata by deriving the end-user metadata from the design. This not only reduces the work required, but also the risk of errors and metadata synchronization requirements, thus increasing metadata quality (see Fig. 5).

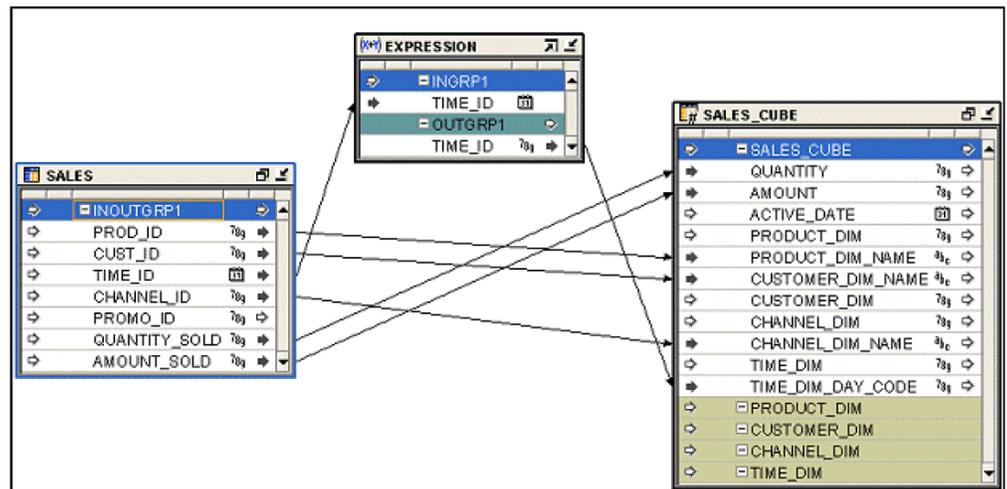


Fig. 5 – OWB Design Center

By having metadata definitions in one place, you can deliver end-to-end impact analysis. You can also determine upfront the impact of the changes on your database, and the impact on your business intelligence environment.

VI. CONCLUSION

The complexity of the information systems used in a company has grown along with its expansion and increase in its volume of sales or along with the increase in their number of employees. So, the gathered information is useful only if it is of dependable quality and is delivered at the right time. In the same time, the need of software integration comes as a must for disconnected businesses that need to interact and the EAI is the tendency to create complex, robust, efficient and finally, complete software solutions. On the other hand: the spatial DBs, multimedia DBs or the mobile databases, usually they are extensions of normal/old databases which contain “special” objects, procedures, mechanisms in order to store and manipulate data, are used for increasing the efficiency of the information systems.

REFERENCES

- [1] E. Bendoly, E. D. Rosenzweig, and J. K. Stratman, “The efficient use of enterprise information for strategic advantage: A data envelopment analysis”, *Journal of Operations Management*, no.27, 2009, pp. 310-323, (ISI Thomson).
- [2] A. N. K. Chen, S. Sen, and B. B. M. Shao, “Strategies for effective Web services adoption for dynamic e-businesses”, *Decision Support Systems*, no.42, 2006, pp. 789-809, (ISI Thomson).
- [3] S. Pastore, “The service discovery methods issue: A Web services UDDI specification framework integrated in a Grid environment”, *Journal of Network and Computer Applications*, no.31, 2008, pp. 93-107, (ISI Thomson).
- [4] S. Chandra, J. de Vries, J. Field, H. Hess, M. Kalidasan, K. V. Raghavan, F. Nieuwerth, G. Ramalingam, and J. Xue, “Using logical data models for understanding and transforming legacy business applications”, *IBM Systems Journal*, vol. 45, no.3, 2006, pp. 647-655, (ISI Thomson).
- [5] X. Jin, “Research on the Model of Enterprise Application Integration with Web Services”, *Proc. of the 3rd WSEAS International Conference on Computer Engineering and Applications (CEA 2009)*, 2009, pp.105-109.

- [6] Integration Consortium website [Online]. Available: www.integrationconsortium.org
- [7] G. Hohpe, and B. Woolf, "Enterprise Integration Patterns: Designing, Building and Deploying Messaging Solutions", Addison-Wesley, ISBN 0321200683, 2004.
- [8] "Query optimizer", *Wikipedia, the free encyclopedia* [Online]. Available: http://en.wikipedia.org/wiki/Query_optimizer
- [9] J. Panus, J. Pirkil, "Influence of Oracle hints on query execution", *10th WSEAS International Conference on Applied Computer Science*, Japan, 2010, pp. 490-495.
- [10] News and updates on latest Internet technologies website [Online]. Available: <http://techgurulive.com/2008>
- [11] Bluehost Company website [Online]. Available: <https://www.bluehost.com>
- [12] J. Stephens, and C. Russell, "Beginning MySQL Database Design and Optimization: From Novice to Professional", Apress, 2004.
- [13] N. T. Nguyen, M. T. Le, and J. „Swiatek, *Intelligent Information and Database Systems*", Springer-Verlag, Berlin Heidelberg, 2010.
- [14] N. Tolia, and M. Satyanarayanan, "A. Wolbach, Improving mobile database access over wide-area networks without degrading consistency", *MobiSys'07*, Puerto Rico, USA, 2007, pp. 71-84.
- [15] E. Oliver, and D. R. Cheriton, "A Survey of Mobile Database Caching Strategies", August, 2007 [Online]. Available: http://blizzard.cs.uwaterloo.ca/eaoliver/papers/2007/mobile_database_cache.pdf
- [16] "Mobile database", *Wikipedia, the free encyclopedia* [Online]. Available: http://en.wikipedia.org/wiki/Mobile_database
- [17] G. Drosatos, P.S. Efraimidis, and A. Karakos, "Secure Mobile Database Applications: A Case Study, Technical Report LPDP-2007-01" [Online]. Available: <http://utopia.duth.gr/~pefraimi/research/data/SecMobDB.pdf>
- [18] K. Xuan, G. Zhao, D. Taniar, W. Rahayu, M. Safar, and B. Srinivasan, "Voronoi-based range and continuous range query processing in mobile databases", *Journal of Computer and System Sciences*, Article in Press (Received in revised form 11 Jan. 2010), (ISI Thomson).
- [19] V. Kumar, *Mobile Database Systems*, John Wiley & Sons, Inc., Hoboken, New Jersey, 2006.
- [20] B. K. Mohanty, "Mobile Database Systems", *Computer Science and Engineering*, 2008.
- [21] A. Tsouchlaraki, G. Achilleos, Z. Nasioula, and A. Nikolidakis, "Designing and creating a database for the environmental quality of urban roads, using GIS", *7th WSEAS Internat. Conf. on Environment, Ecosystems and Development, Tenerife*, Spain, 2009, pp. 109-113.
- [22] D. Litan, A. M. Mocanu (Virgolici), S. Olaru, and A. Apostu, "Modern information technologies used in market research", *Proc. of the 9th WSEAS Internat. Conf. on Computational Intelligence, Man-Machine Systems and Cybernetics, Merida, Venezuela*, 2010, pp. 245-250.
- [23] A. Bara, A. Velicanu, I. Lungu, and I. Botha, "Natural factors that can affect wind parks and possible implementation solutions in a Geographic Information System", *Proc. of the Internat. Conf. on Development, Energy, Environment, Economics, Tenerife*, Spain, 2010, pp. 50-54.
- [24] Open Geospatial Consortium (OGC) website [Online]. Available: www.opengeospatial.org
- [25] "OpenGIS® Implementation Specification for Geographic information - Simple feature access - Part 1: Common architecture", Open Geospatial Consortium website [Online]. Available: <http://www.opengeospatial.org/standards/sfa>
- [26] K. Priyanath Udagepola, L. Xiang, Y. Xiaozong, and A.W. Wijeratne, "MSRIC: A Model for Spatial Relations and Integrity Constraints in Topographic Databases", *Proceedings of the 5th WSEAS International Conference on Artificial Intelligence, Knowledge Engineering and Data Bases*, Spain, 2006, pp. 366-371.
- [27] C. Zhang, and Y. Huang, "Cluster By: a new sql extension for spatial data aggregation", *The 15th annual ACM internat. symposium on Advances in geographic information systems ACM*, New York, 2007.
- [28] Multimedia database [Online]. Available: <http://www.tech-faq.com/multimedia-database.html>
- [29] "Multimedia database", *Information Technology Portal (India)* [Online]. Available: <http://www.peterindia.net/MultimediaDatabase.html>
- [30] "Possible solutions to the relational problem – Multimedia Databases (Lecture 3)", *Multimedia Storage & Retrieval*, School of Computing, Engineering & Information Sciences, Northumbria University, Newcastle, UK, Feb., 2006 (course notes).
- [31] S. H. Park, "Multimedia Databases (Lecture 2)", Yonsei University, 2nd semester, 2009 [Online]. Available: <http://www.scribd.com/doc/49456182/multimedia-dbms> (course notes).
- [32] I. Bartolini-DEIS, "Similarity Measures for MM Data Retrieval and MM DBMS", *Multimedia Databases: Fundamentals, Retrieval Techniques, and Applications*, University of Bologna, Sept., 2010 [Online]. Available: <http://www.lia.deis.unibo.it/phd/materials/courses/Multimedia%20Databases%20Fundamentals,%20Retrieval%20Techniques,%20and%20Applications/02MMDBs-Bartolini.pdf> (a short course for doctoral students).
- [33] Eurostat (The statistical office of the European Union) [Online]. Available: http://epp.eurostat.ec.europa.eu/statistics_explained/index.php/E-government_statistics#Further_Eurostat_information
- [34] Y. Xiao-Tong, "The Research of Heterogeneous Database Middleware Model", *Proc. of the 3rd WSEAS International Conference on Computer Engineering and Applications (CEA'09)*, China, Jan. 10-12, 2009, pp. 243-246.
- [35] Oracle Company website [Online]. Available: www.otn.oracle.com
- [36] B. Griesemer, "Oracle Warehouse Builder 11g Getting Started", Packt Publishing, 2009.
- [37] M. Teohari, and L. Copcea, "XML Authoring Tool", *Proc. of the 4th European Computing Conf. (ECC'10, University Politehnica of Bucharest)- WSEAS Conference*, Romania, Apr. 20-22, 2010, pp. 143-147.



Daniela Litan has graduated the Academy of Economic Studies (Bucharest, Romania), Faculty of Cybernetics, Statistics and Economic Informatics in 2006. She holds a Master diploma in Databases - Support for business from 2009 and in present she is a Ph.D. Candidate in Economic Informatics with the Doctor's Degree Thesis: The integration of new technologies in the information systems of e-Government.

Her research activity can be observed in the following achievements: 12 proceedings (10 papers ISI proceedings), among with:

- "Ways to Increase the Efficiency of Information Systems", *Proc. of the 10th WSEAS International Conference on Artificial Intelligence, Knowledge Engineering and Databases (AIKED '11, University of Cambridge)*, February 20-22, 2011, Cambridge, UK;
 - "Modern Information Technologies Used in Market Research", *Proc. of the 9th WSEAS International Conference on Computational Intelligence, Man-Machine Systems and Cybernetics (CIMMACS '10)*, December 14-16, 2010, Merida, Venezuela;
 - "Business Intelligence and Data warehouse – Technological Support for Decisional Management in Geographical Information Systems", *Proc. of the 3rd International Conference on Communications and Information Technology (CIT'09)*, December 29-31, 2009, Athens, Greece;
- and 7 articles published in scientific reviews , among with:
- "Market Research – a significant step in propelling a business or a solution for the e-government systems to the top", *International Journal of Applied Mathematics and Informatics*, issue 2, vol. 5, 2011.
 - "Modern Database Machines", *Journal: Informatics Economics*, no. 2, 2010;
 - "Information systems in the knowledge based economy", *Journal: WSEAS TRANSACTIONS on BUSINESS and ECONOMICS*, issue:1, vol. 7, 2010.

Her scientific fields of interest include: Databases, Database Management Systems, Programming, Information Systems and Economics.



Manole Velicanu is a Professor at the Economic Informatics Department at the Faculty of Cybernetics, Statistics and Economic Informatics from the Academy of Economic Studies of Bucharest, Romania. He has graduated the Faculty of Economic Cybernetics in 1976, holds a Ph.D. diploma in Economics from 1994 and starting with 2002 he is a PhD coordinator in the field of Economic Informatics.

He is a member of INFOREC professional

association, a CNCIS expert evaluator and a MCT expert evaluator for the program Cercetare de Excelenta - CEEEX (from 2006). From 2005 he is co-manager of the master program Databases for Business Support. He participated (as director or as team member) in more than 40 research projects that have been financed from national research programs.

His research activity can be observed in the following achievements: 18 books in the domain of economic informatics, 64 published articles (among which 2 articles ISI indexed), 55 scientific papers published in conferences proceedings (among which 5 papers ISI indexed and 7 included in international databases) and 36 scientific papers presented at conferences, but unpublished.

His scientific fields of interest include: Databases, Design of Economic Information Systems, Database Management Systems, Artificial Intelligence, Programming languages.



Larisa Copcea (Teohari) has graduated the Academy of Economic Studies (Bucharest, Romania), Faculty of Cybernetics, Statistics and Economic Informatics in 2006. She holds a Master diploma in Databases - Support for business from 2008 and in present she is a Ph.D. Candidate in Economic Informatics with the Doctor's Degree Thesis: Master data management in data warehouses.

Her research activity can be observed in the following achievements: 4 proceedings (papers ISI proceedings), among witch:

- "Ways to Increase the Efficiency of Information Systems", *Proc. of the 10th WSEAS International Conference on Artificial Intelligence, Knowledge Engineering and Databases (AIKED '11, University of Cambridge)*, February 20-22, 2011, Cambridge, UK;
 - „Some Information Technologies to Improve the Performance of an ERP System”, *Proc. of the 5th WSEAS International Conference on Computer Engineering and Applications (CEA '11)*, January 29-31, 2011, Puerto Morelos, Mexico;
 - "XML Authoring Tool", *Proc. of the 4th European Computing Conference (ECC'10, University Politehnica of Bucharest)*, April 20-22, 2010, Bucharest, Romania;
- and 2 articles published in scientific reviews , among witch:
- "Technologies for Development of the Information Systems: from ERP to e-Government", *International Journal of Applied Mathematics and Informatics*, issue 2, vol. 5, 2011.

Her scientific fields of interest include: Data warehouse, Databases, Database Management Systems, High availability solutions, Information Systems and Economics.



Mihai Teohari has graduated the Academy of Economic Studies (Bucharest, Romania), Faculty of Cybernetics, Statistics and Economic Informatics in 2004. He is a Ph.D. Candidate in Economic Informatics with the Doctor's Degree Thesis: XML technologies for databases.

His research activity can be observed in the following achievements: 4 proceedings (papers ISI proceedings), among witch:

- "Ways to Increase the Efficiency of Information Systems", *Proc. of the 10th WSEAS International Conference on Artificial Intelligence, Knowledge Engineering and Databases (AIKED '11, University of Cambridge)*, February 20-22, 2011, Cambridge, UK;
 - „Some Information Technologies to Improve the Performance of an ERP System”, *Proc. of the 5th WSEAS International Conference on Computer Engineering and Applications (CEA '11)*, January 29-31, 2011, Puerto Morelos, Mexico;
 - "XML Authoring Tool", *Proc. of the 4th European Computing Conference (ECC'10, University Politehnica of Bucharest)*, April 20-22, 2010, Bucharest, Romania;
- and 2 articles published in scientific reviews , among witch:
- "Technologies for Development of the Information Systems: from ERP to e-Government", *International Journal of Applied Mathematics and Informatics*, issue 2, vol. 5, 2011.

His scientific fields of interest include: XML, Databases, Data warehouse, Information Systems and Economics.



Aura Mihaela Mocanu (Virgolici) has graduated the Academy of Economic Studies (Bucharest, Romania), Faculty of Cybernetics, Statistics and Economic Informatics in 2007. She holds a Master diploma in Databases - Support for business from 2009 and in present she is a Ph.D. Candidate in Economic Informatics with the Doctor's Degree Thesis: Software technologies to build a Geographical Information System for a public institution.

Her research activity can be observed in the following achievements: 8 proceedings (6 papers ISI proceedings), among witch:

- "Ways to Increase the Efficiency of Information Systems", *Proc. of the 10th WSEAS International Conference on Artificial Intelligence, Knowledge Engineering and Databases (AIKED '11, University of Cambridge)*, February 20-22, 2011, Cambridge, UK;
 - "Modern Information Technologies Used in Market Research", *Proc. of the 9th WSEAS International Conference on Computational Intelligence, Man-Machine Systems and Cybernetics (CIMMACS '10)*, December 14-16, 2010, Merida, Venezuela;
 - "Business Intelligence and Data warehouse – Technological Support for Decisional Management in Geographical Information Systems", *Proc. of the 3rd International Conference on Communications and Information Technology (CIT'09)*, December 29-31, 2009, Athens, Greece;
- and 4 articles published in scientific reviews , among witch:
- "Market Research – a significant step in propelling a business or a solution for the e-government systems to the top", *International Journal of Applied Mathematics and Informatics*, issue 2, vol. 5, 2011.
 - "Modern Database Machines", *Journal: Informatics Economics*, no. 2, 2010;
 - "Information systems in the knowledge based economy", *Journal: WSEAS TRANSACTIONS on BUSINESS and ECONOMICS*, issue:1, vol. 7, 2010.

Her areas of interest are: Geographical Information Systems, Databases, Information Systems integration and Programming languages.



Iulia Surugiu has graduated the Academy of Economic Studies (Bucharest, Romania), Faculty of Cybernetics, Statistics and Economic Informatics in 2007. She is a Ph.D. Candidate in Economic Informatics with the Doctor's Degree Thesis: The integration of information technologies in enterprise applications development.

Her research activity can be observed in the following achievements: 5 proceedings, among which:

- "Information Systems Integration, a New Trend in Business", *Proc. of the 10th WSEAS International Conference on Applications of Computer Engineering (ACE '11)*, March 24-26, 2011, Meloneras, Gran Canaria, Canary Islands, Spain;
- "Ways to Increase the Efficiency of Information Systems", *Proc. of the 10th WSEAS International Conference on Artificial Intelligence, Knowledge Engineering and Databases (AIKED '11, University of Cambridge)*, February 20-22, 2011, Cambridge, UK;
- "Advanced e-Government Information Service Bus (eGov-Bus)", *Proc. of the 9th International Conference on Informatics in Economy*, May 5-7, 2009, Bucharest, Romania,

and 2 articles published in scientific reviews:

- "Academic Research Information System", *Journal: JAQM (Journal Of Applied Quantitative Methods)*, issue: 1, vol. 3, 2008;
- "Medical Virtual Public Services", *Journal: Informatics Economics*, no. 1(45), 2008.

Her scientific fields of interest include: Computer Programming, Information Systems, Databases, Database Management Systems and Economics.



Ovidiu Raduta has graduated the Academy of Economic Studies (Bucharest, Romania), Faculty of Cybernetics, Statistics and Economic Informatics in 2006. He holds a Master diploma in Informatics Security (Master Thesis: IT Software in banks. Security Issues) from 2008 and currently, he is a Ph.D. Candidate in Economic Informatics with the Doctor's Degree Thesis: Bank System's Process Optimizing.

In present, he is ISTQB – Advanced Test Analyst

certified and he works as Senior Test Analyst with 3+ years testing experience in Raiffeisen Bank Romania (6+ years banking projects experience).

His research activity can be observed in international proceedings (papers ISI proceedings), among which:

- “Ways to Increase the Efficiency of Information Systems”, *Proc. of the 10th WSEAS International Conference on Artificial Intelligence, Knowledge Engineering and Databases (AIKED '11, University of Cambridge)*, February 20-22, 2011, Cambridge, UK;
- “Information Systems Integration, a New Trend in Business”, *Proc. of the 10th WSEAS International Conference on Applications of Computer Engineering (ACE '11)*, March 24-26, 2011, Meloneras, Gran Canaria, Canary Islands, Spain;
- „Information Technology Standards – a Viable Solution to Reach the Performance”, *Proc. of the 12th WSEAS International Conference on Automation & Information (ICAI '11)*, Apr. 11-13, 2011, Brasov, Romania.

His scientific fields of interest include: Test management, Test Techniques, Databases processes, Middleware Products, Information Systems and Economics.