

INTEGRATING MULTIPLE SOFTWARE PLATFORMS FOR INTEGRATED MANAGEMENT

BOTEZATU, C.; BOTEZATU, C. P. & CARUTASU, G.

Abstract: *The issue of software integration is difficult and very popular, considering the fact that, most times, the individually developed systems were not conceived considering the possibility of integration and there wasn't an integrated strategy for development. The lack of an integrated system generated the prevention of communication between different IT systems. Each system was based exclusively on the fulfilment of the functions that it was designed for. Then, when porting the data from one system to another, the users are forced to reintroduce them or manually migrates the data. The issue of integrating software at enterprise level became a major issue, as the experts in the field did considerable effort. This paper presents application integration architectures, being part of technical researches made for Development and Implementation of the Integrated Management System (DI - IMS) project.*

Key words: *computer, software integration, application integration, middleware technologies*



Authors' data: Univ.Sen.Lect. Dr.math. **Botezatu** C[ezar]; Univ.Prof. Dr. Ec. **Botezatu**, C[ornelia] P[aulina]; Univ.Sen.Lect. Dr.Eng. **Carutasu**, G[eorge], Romanian-American University, Expozitiei 1B, 012101, Bucharest, Romania, cezarbotezatu@gmail.com, c2botezatu@yahoo.com, georgecarutasu@yahoo.com

This Publication has to be referred as: Botezatu, C[ezar]; Botezatu, C[ornelia] P[aulina] & Carutasu, G[eorge] (2010). Integrating Multiple Software Platforms for Integrated Management, Chapter 49 in DAAAM International Scientific Book 2010, pp. 559-574, B. Katalinic (Ed.), Published by DAAAM International, ISBN 978-3-901509-74-2, ISSN 1726-9687, Vienna, Austria
DOI: 10.2507/daaam.scibook.2010.49

1. Introduction

Application integration represents a strategic approach for unification of several systems, both at services and data level. It offers the possibility to insure data exchange between systems and reuse the business processes in real time. Even if it seems a strictly technical issue, the resulting data and process flux provides for the enterprises a strategic advantage: the opportunity to conduct their businesses in real time, in an event-based environment, with low latency and reusing their IT support in optimal conditions. In *Development and Implementation of the Integrated Management System (DI - IMS)* project (Botezatu et al. 2008) a study was made to ensure the prerequisites for managing unstructured data, having the results the policies and strategies presented in this paper.

In February 2009, 76 Romanian companies were questioned regarding several issues, about Integrated Management Systems:

- General management policies;
- Information systems requirements for integrated management systems;
- Statistical management methods and tools;
- Health and security of labour;
- Quality Assurance. Standards and certifications;
- Environmental policies.

One of the key points of this study was information technology penetration at enterprise level, regarding different types of network access, integrated business solutions, data exchange and integrated support systems for delocalised activities.

As it shown in figure 1 and accordingly with data presented in table 1, the percent of neutral respondents (28, 36 and 29%) is significant, seeming that the networks role in a company is not yet well known. On the other hand, if we take into account only firms opinion (Approval/Denial) all four types of networks are appreciated as necessary to support day-to-day enterprise activity. For a better understanding, the computer networks functions are grouped in sharing resources and information exchange between network members.

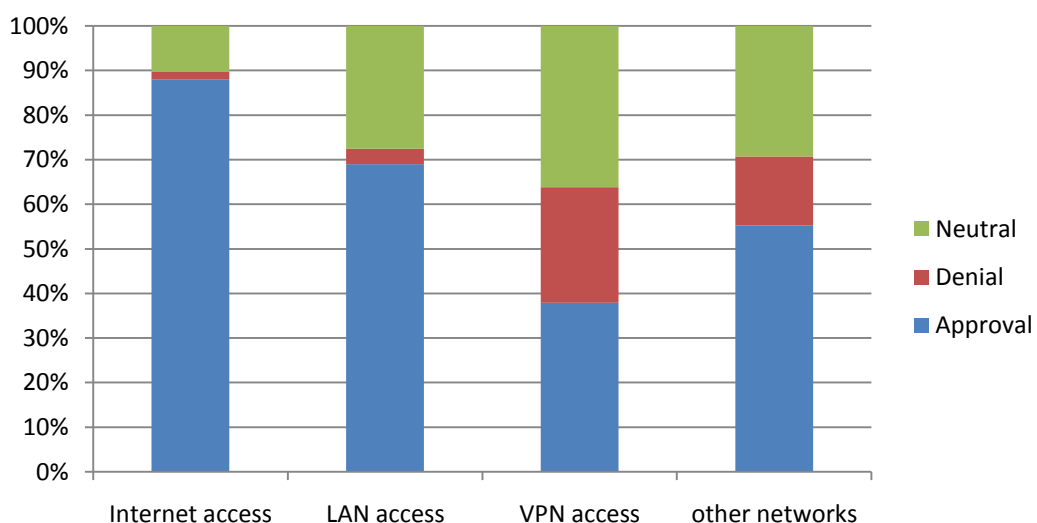
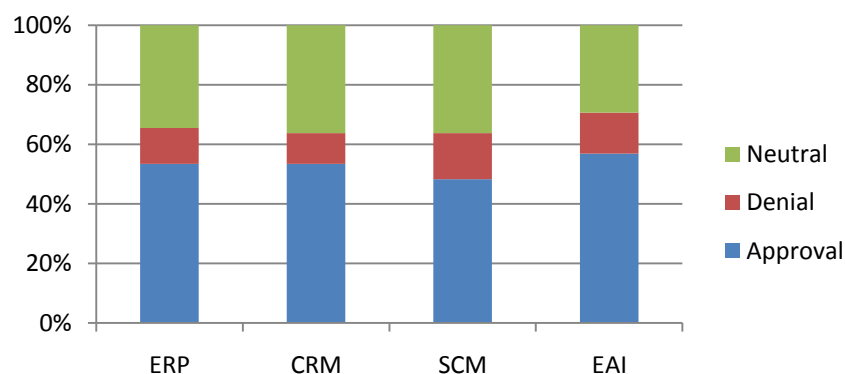


Fig. 1. 1st Criterion: Connectivity options

	Approval (value/%)	Denial (value/%)	Neutral (value/%)
Different types network access, as information infrastructure	145/63	27/12	60/26
Internet access	51/88	1/2	6/10
LAN access	40/69	2/3	16/28
VPN access	22/38	15/26	21/36
Other networks	32/55	9/16	17/29
Economic activity support integrated solutions	123/53	30/13	79/34
ERP (Enterprise Resource Planning)	31/53	7/12	20/34
CRM (Customer Relationship Management)	31/53	6/10	21/36
SCM (Supply Chain Management)	28/48	9/16	21/36
Automated exchange data system between customer and supplier based on EDI (Electronic Data Interchange), XML (Extensible Mark-up Language)	33/57	8/14	17/29
Delocalised business support systems			
e-business, e-auction, e-commerce, CMS (Content Management Systems) sites CMS	35/60	7/12	16/28

Tab. 1. Study centralised respondent data

For the second criterion, economic activity support integrated solutions, was draw figure 2. In this case, positive opinion regarding economic activity support integrated solutions is counting between 48 and 57% (regarding commonly oriented solutions ERP, CRM and SCM). Even if integration solutions (EAI – Enterprise Application Integration) is well appreciated, we believe that, on companies level, small steps are made for automate different business software applications.

Fig. 2. 2nd Criterion: Economic activity support integrated solutions

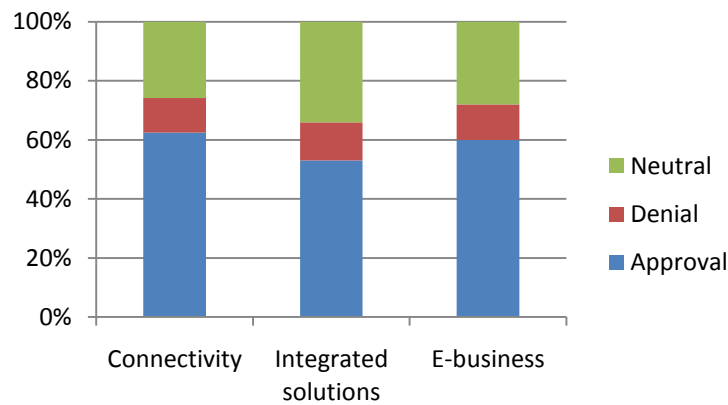


Fig. 3. Overview of information technology support for enterprises activity

Over viewing those mentioned above, having in mind the three claimed criterions: connectivity, integrated solutions and delocalised systems, information technology is positive appreciated with simple majority by companies' managers (figure 3).

Starting with these prerequisites, designing and developing an Integrated Management System (IMS) impose to use already registered data in other software applications. Many of complex software applications (e.g. ERP, CRM or SCM type) include part of business processes which lately will be use by IMS. Even in this happy case, when a complex solution is implemented at enterprise level, take the legacy of past used solution, which keeps old register data. Is a cheaper solution to keep existent proofed modules (e.g. accounting, order management) and integrate with new ones. A good advice, even the prices are higher, is to use, if it is possible, a reduced number of applications, old data being imported once in implementation process. So, an integration policy must be developed for each company. Speaking in generally terms, developing IT structure in a company must have in sight standard principles and procedures of integration described below.

Results obtained from the questionnaire in conjunction with other external data, are analyzed in the following directions:

- Characteristics field general dynamics;
- Increasing economic importance of the field;
- Changing the content of work and preparedness of staff;
- Outsourcing of products and services;
- Proliferation of communication systems based on Internet and Intranet;
- Implementation of new systems of organization in enterprises.

1.1 Characteristics field general dynamics

As apparent from the data presented above, the quantitative growth is assessed by 100% quality with 104.22% and the degree of complexity of 122.56%, the prognosis is three years (DI-SIM Research Report). These data are in line with those presented by Romanian Ministry of Information Technology and Communication (Statistics for Romanian IT industry), expecting both to increase in turnover of firms in the industry, but also an increase in endowment spending to hardware and software

(about 11%). Consequently most polls and analysis shows an increase in recent years, with value growth between 5 and 15% of the field of information systems, but also of importance in supporting business processes at enterprise level.

1.2 Economic increased importance of the field

After an estimated increase to around 17% in 2007, the IT sector will reduce growth this year. IDC forecasts expect IT market of Romania in 2008 to reach the threshold of 1.37 billion EUR, registering an increase of approximately 11% over last year, due to special conditions of this area (legislation, working conditions, etc.). In 2009 because of financial crises it will be a 10% shrink of market. Due the technological progress made, the share of ICT (information and communications technology) in GDP, and expenditure required to achieve the ICT products and services, is growing. In this respect, the expenditure and revenue recorded at the company in 2008 are classified into:

- Hardware equipment, which acquire or renew park equipment (computers and communications infrastructure) – 27%;
- Software acquisition – 12%;
- Services (maintenance, repair and training) – 25%;
- Processed materials – 27%;
- Other expenditures – 9%.

1.3 Modifying the content and preparation of working staff

The more massive IT implementation in various fields of achievement required mutations in relation to the working staff. In this sense, provides current job descriptions and skills of operating the processing of various being distinguished several levels of jurisdiction: general operating, as ECDL or other similar levels and specialized operation and data processing.

Simultaneously, changes occur in the preparation of staff. Acquire a minimum of computer knowledge and skills are a major component of training programs and professional. According to the results of the questionnaire, the average annual expenditure on IT training is about 880 RON (200 EUR) per person. It is estimated that these costs will have a share of the increasingly important due to the need for awareness training of companies' management staff, but due to the diversification of software tools used.

1.4 Outsourcing of products and services

Starting from the premise efficient work on the major business processes, an important part of companies, especially small and medium size have decided to outsource some activities (aided design, installation and maintenance of software and hardware, custom software implementation).

1.5 Proliferation of communication systems based on internet and intranet

Software architecture requires a high degree of scalability, provided by the communication infrastructure and logical data servers. These modules function as interconnected with the bus interconnection implemented by software communication

protocols and adapters. Requirements can be met by this level of system architecture to achieve the objectives are: to represent a complete solution for high technology, designed to ensure program management, system having an architecture of client / server, providing a management unit circulating information, information entered is transmitted in real time to all parties that the relationship is interdependent ensuring optimum level of security applications and the speed necessary to run a professional product, is a modular system which allows a uniform management information by geographically distributed users eventually, to the development of reports for specific activities and how their integration in a general way to allow native storage and presentation of multilingual information.

1.6 Implementation of new systems in the enterprise organization

Information systems and communications will take the following goal, policy outsourcing:

- Increased focus on own business and not on technology-through specialization and delegation, out-sourcing. The customer can devote to solving problems related to their field of activity is not required to understand and implement their own means through all aspects of computerization;
- Access to new markets. The organization on the Internet automatically allows access to new markets, integration of circuits in a virtual commercial suppliers - manufacturers - distributors – customers;
- Continuous adaptation to changing technology. Clients use automatically and transparently newest technology;
- Rapid implementation. Besides eliminating all problems related to installing and configuring hardware and network, ASP providers tend to offer standardized solutions, already validated, enabling shorter implementation period and bringing in real time operation system;
- Low cost of implementation and operation, because the customer is freed from the problem of ensuring the functioning applications. The client must not invest in any infrastructure hardware equipment or personnel in specialized management applications;
- Predictability of costs and low initial investment. Services use software costs are fixed (monthly or annual subscriptions, rental arrangements for services, etc.). So that the customer may very precisely the size needed computerization;
- Ongoing support. Service provider on the Internet should provide ongoing support and monitoring of business customers: Help Desk On-Line, dedicated phone lines, etc.

All the details are relevant – all the individual aspects have to be considered when developing an integrated system. Considering the complexity of the integrated IT system at the level of multiple legal entities communicating between them, they can choose a distributed system, through the development of multiple application servers, each of them containing an IT subsystem. Different types of users are involved that use a certain subsystem, so at least an application server will be used for each institution. In this case, when multiple servers are used, it must be mapped the chart with the allocation of modules for each server with be developed, as well as

the performance requirements of these servers, the communication chart between application servers, the development of disaster recovery procedures.

According to Nelson King (King, 1999), there are 3 main classes of integration technologies, which can be used in correlation at enterprise level:

- Data Integration Technologies;
- Enterprise Application Integration;
- Business to Business Application Integration.

John Schmidt (Schmidt, 2002), the chairman of the Industrial Consortium EAI Committee, published in his articles the 5 laws that IT systems integration has to fulfil:

- Whole is better than the sum of parts – any integrated system provides more facilities than the sum of the individual systems facilities, through data correlation and aggregation;
- There is no final status – the integrated system undergoes permanently through a series of modifications, due to multiple organizational and functional aspects and technological evolution;
- There are no universal standards – use of standards is different, according to the needs of the enterprise;
- Technologies adapt in order to meet the needs of the local requests - any methodology or technology must be adjusted at enterprise level;

So, in the next pages will be detailed the techniques for software integration resulted as best practices and self-study made by authors, establishing a set of guidelines, depending of software platforms and the type of organization.

2. Software integration architectures

One of the classification criteria for the software integration architectures is the way the user interacts with the systems, as well as with the integrated application. The architectures presented include web components too, as access through Internet is already a common aspect. They also contain links with the data storage deposits and links with other enterprises (Botezatu & Botezatu, 2006).

There are 3 types of architectures:

- Forward type Application Integration Architecture
- Back-end type Application Integration Architecture
- Mix type Application Integration Architecture

2.1 Forward type Application Integration Architecture

The forward type emphasizes user experience and puts the EAI server close to the user interface. In fact, a business process started by a user generates multiple transactions in multiple applications, and the answer is provided almost in real time (Glazier, 2000). It's the case when a user accesses the integrated system, the integrating interface or the so-called EAI server, either by direct means or through internet.

The EAI server makes the updates in the individual systems. In figure 4, the user accesses through internet the web server that sends his request to the integrating

application/system. The EAI application uses rules in order to identify the application that needs to be used and the parameters for it. The individual applications can be both legacy applications, as well as new ones. The integrated system accepts requests from internal users as well as from the business partners (Hammer & Timermman, 2007).

This type of architecture is used only if there are well defined rules for mapping data for individual applications.

The model can route the users' transactions through a single interface, connecting them to multiple applications; this type is very efficient in a B2C (business to customer) environment. The advantage is that the transactions are achieved almost in real time.

2.2 Back-end type Application Integration Architecture

This architecture type improves automatic data exchange between applications. The EAI server does not interact with the user, as it gathers data from the individual applications.

In figure 5 the user accesses the application directly or through a web server. In the user interface can be integrated all the facilities provided by the individual applications, but in reality each application manages its own transactions.

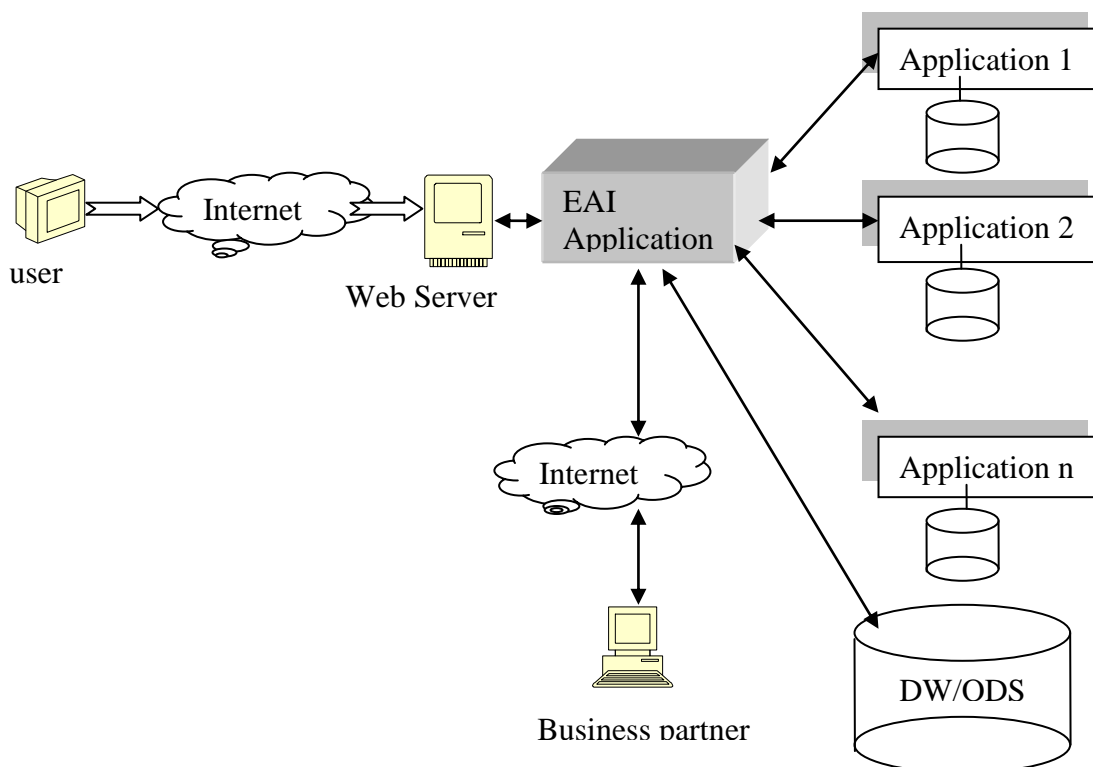


Fig. 4. Forward type Application Integration Architecture

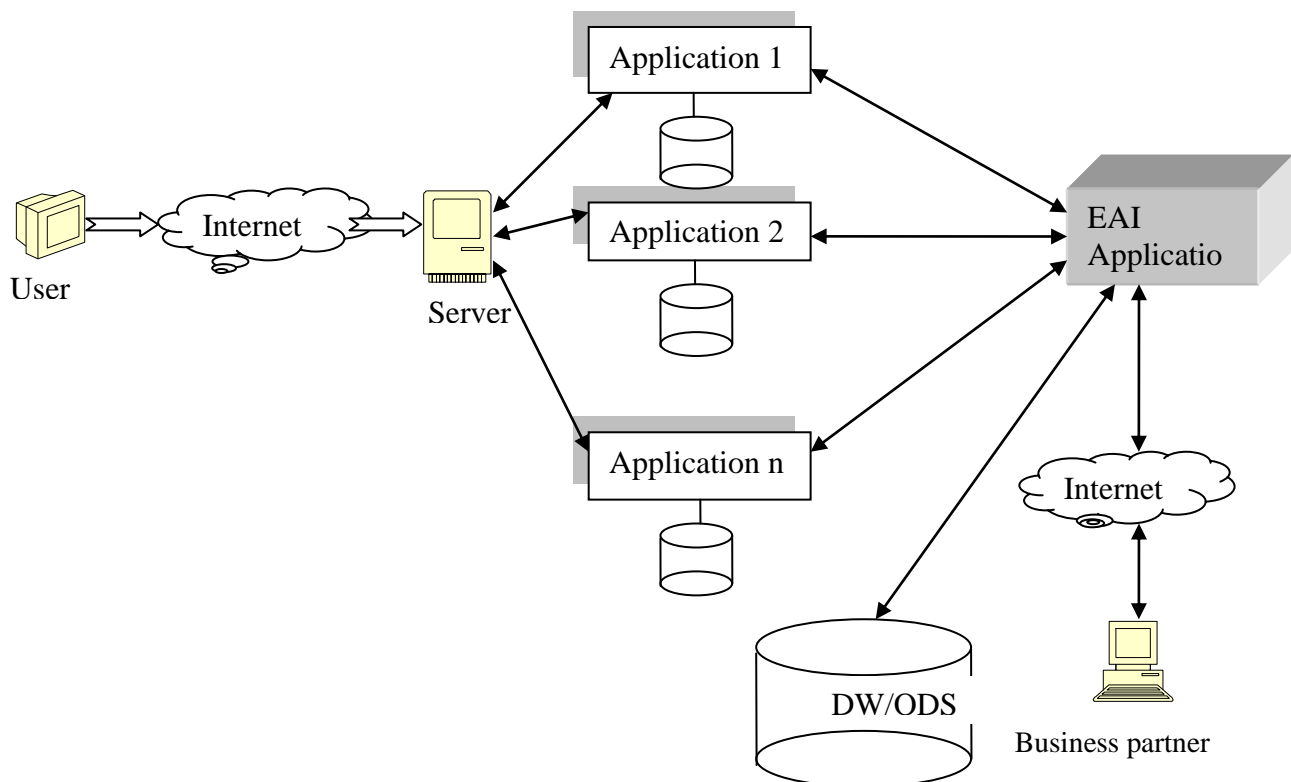


Fig. 5. Back-end type Application Integration Architecture

When an application needs data managed by a different application, it uses the EAI server.

This architecture type is easier to implement, as it has a lower number of rules to be implemented. It is used for B2B applications (business to business), which require data exchange between business partners (Bramer, M. & Terziyan, V. 2005).

2.3 Mix type Application Integration Architecture

This type (figure 3) represents a combination of the two previous architectures. User accesses different applications, as the EAI application. This architecture type is used when the forward model is needed, with front-end applications, but with some back-end applications as well that might be needed.

2.4 Middleware technologies

Application integration is very popular through the use of middleware technologies, which includes communication protocols, API integration interfaces, application servers, web servers. *Middleware* is defined as an intelligent component that works as an interface between user and a large number of systems with different functions. Through middleware the way to communicate between systems is controlled.

In practice, the user does not feel this level that sends the information to every system that needs that data. At query, the middleware section gathers data from different applications correlates them and aggregates them. In this way the transparency of the systems distribution and data sources is ensured.

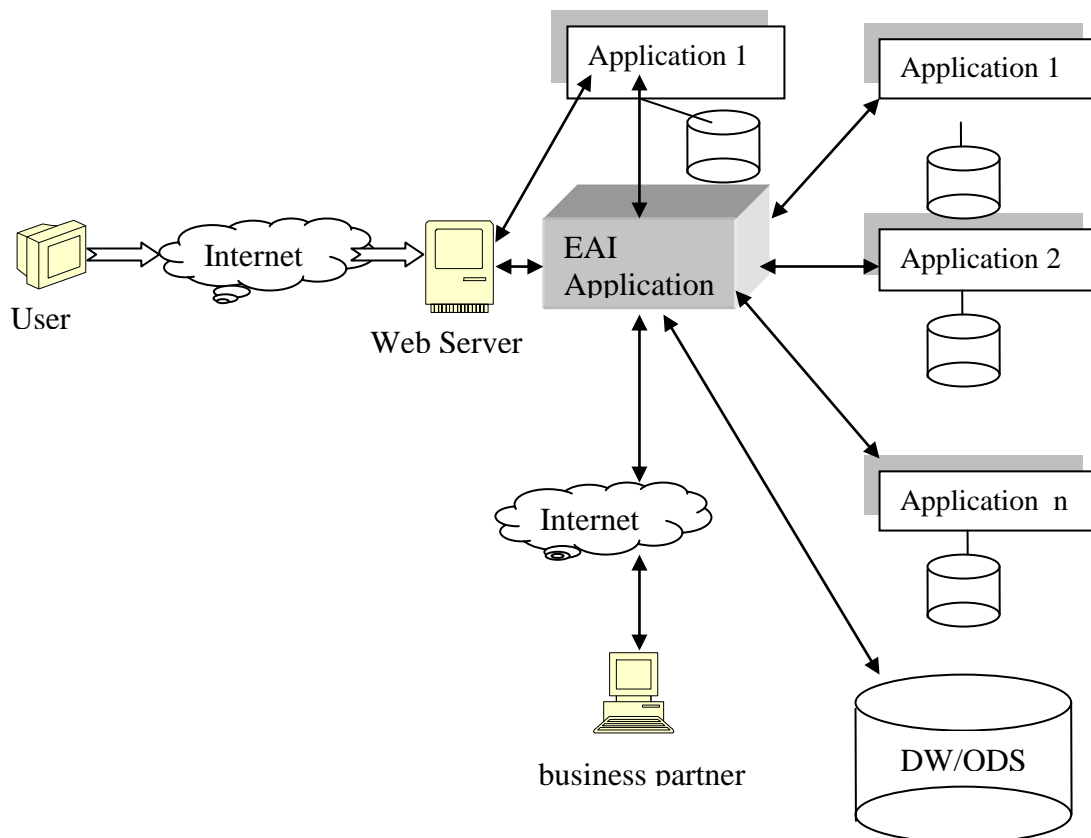


Fig. 6. Mix type Application Integration Architecture

Integration of systems using middleware can be achieved in different ways:

- Message oriented – through messaging;
- Transaction oriented – communication through transactions;
- Through distributed objects - the link between applications is done through distributed objects;
- For databases – uses databases;
- Remote calls.

3. Integration Technologies

Use of adaptors. In large enterprises there are used many types of applications, with different interfaces, for example: .NET, COM (Component Object Model), CORBA (Common Object Request Broker Architecture), JAVA etc. At the development of a new application that has to relate with existing applications, the API has to be translated from the source technology in the technology of the new application. Each new application has to use the interface connection with the existing applications. The way to solve this is given by adaptors. The interfaces for the application and databases are different, including from well defined API to the complex output procedures. The software developers started to introduce adaptors, as a response after long periods of time of manual code writing. The adaptors are small applications that intervene between the source applications and destinations, middleware, APIs, database interfaces, or even different types of middleware components (Chang et al., 2008).

Use of messaging. The principle of message oriented middleware is similar in approach with sending e-mails. When an application has to send or receive some data, the data is transformed in electronic form, as a message, that is set in a queue. In some cases middleware can supply a notice concerning the receipt of the message. The process, as in e-mails case, is asynchronous, as the application can continue with a different message. The messages can be sent through local network or Internet. Between applications can be developed central node that's receiving messages and later send forward. It's a more practical approach, because the adaptors can be integrated only in the central node, and there is no more need for adaptors for each application. This type of middleware is referred to in the specialized literature as the MOM (Message Oriented Middleware), as one of the technologies with a growing rate of use.

Application servers. The application servers are based on a component model, as they are a container type environment for keeping applications, which includes development tools. Here are included typical concepts for container based systems like Enterprise JavaBeans (EJB) or Microsoft Transaction Server (MTS).

The technology eliminates the need to develop complex middleware services of low level. The developers have the possibility to focus on programming business rules, achieving a low business implementation time.

The technology was developed using standard specifications J2EE (Java 2 Enterprise Edition) or Microsoft .NET. The best known producers are: IBM, Sun and Oracle.

The application servers based on J2EE (Figure 7) include support for distributed transactions, security, persistence, clustering at application level. More than that, they eliminate the need to write codes for certain issues (Land et al., 2007).

The composing elements are:

- Visual tools for development of the configuration;
- Framework with prebuilt code sequences, partially generated, that will be combined with code sequences written by the beneficiary;
- Application data storage environment.

Figure 4 presents the architecture of the application servers for J2EE. The programmers can write Java codes directly or they can use visual tools that generate Java code. Applications that access the databases are generated. The application servers have surpassed from a long time ago the initial purpose of providing a complete environment for application development.

In the last few years, they were developed in the direction of integrating the legacy systems, application packages or other technologies based on J2EE standards: Java Messaging Service (JMS) or Web services. The application servers' producers extended the facilities provided, by including integration standards, including use of messages, adaptors, management of heterogeneous systems and management of business systems.

Integration brokers. Integration brokers, developed as hub-spoke type systems, where the broker product is the integration hub and the spoke are the applications that need integration. The hub type vision reduces the number of interferences that need integration.

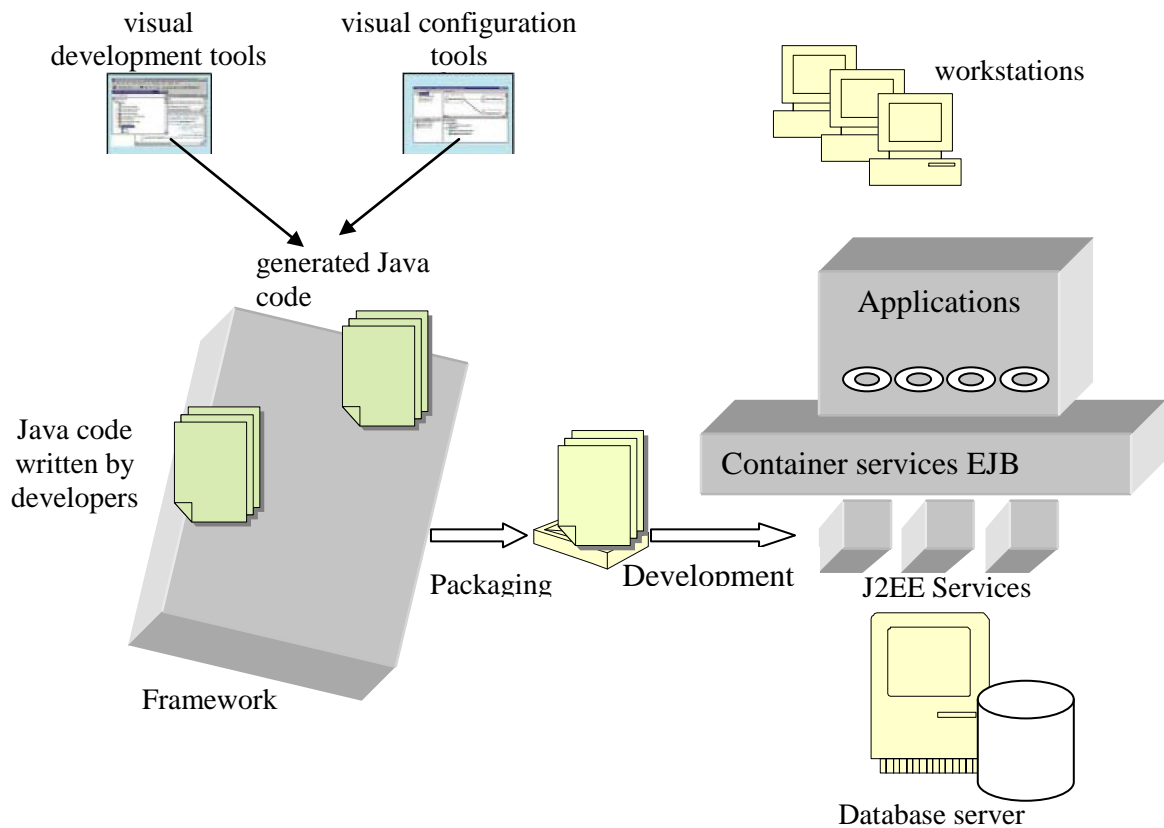


Fig. 7. J2EE Application server's architecture

The composing elements of an integration broker are:

- Adaptors for establishing connections between applications, especially with legacy systems or with software packages;
- Certain forms of MOM, previously presented for distribution of events and dates;
- System of rules for routing and transformation of data.

The best known producers of integration brokers are: IBM, TIBCO, SeeBezond, webMethods. Traditionally, brokers are concentrated on the concept of distributing events between applications, based on well-established rules, most times in the form of messages, as we have mentioned before. The integration broker transforms the application specific schemes using transformation rules, that the developers would define using visual components (Zhao, 2006). As well, the developers can define routing charts of the components through different integration points. The applications can be modified without affecting the integrated application. The brokers offer prebuilt functions in order to reduce the application integration effort and the possibility of reusing components (Liu et al, 2009).

So, the facilities provided by an integration broker are:

- Transformation through caption of the charts and graphic mapping;
- Intelligent routing of messages;
- Graphic instruments for designing transformations, processes and routing rules;
- Distributed architecture;
- Data dictionary that keeps information about applications, with the possibility of reuse;

- XML transformation;
- Encryption possibilities.

Integration brokers were developed including different communication models (synchronous, asynchronous, publish/subscribe type, message queue). The essential components introduced were: the possibility of business processes management, BPM (Business Process Management), workflow charts and support for web services. In this way the interaction with users outside the system was created. The usual architecture used by the integration brokers is presented in Figure 8. The code can be written by developers or generated through visual transformation tools or visual tools for the management of the business processes. This is processed and sent to integration brokers, which manage the routing to different applications (Larson et al., 2008). In this moment the integration brokers offer integration support for a diversity of applications:

- For component models: CORBA (Common Object Request Broker Architecture), COM (Component Object Model) or .NET from Microsoft
- For MOM standards: MSMQ (Microsoft Message Queue), JMS (J2EE Message Service), IBM MQSeries
- For XML format: ebXML, CXML etc.

Actually, as it can be seen, both the application servers and the integration brokers evolve around the takeover and improvement of the facilities of each of them. In the near future, we will probably not be able to speak about different technologies, each of them including a mix of instruments and facilities provided.

3.1 Use of XML technology

There is a much faster way to incrementally develop the integrated software products. XML was already recognized as an integration standard, providing a way of viewing the data in a common format. For example, in the financial field, using XML and web services were recognized as a standard, as it was agreed by industrial groups and field standards. The international organizations from the financial area have already adopted XML type messaging systems.

Another example that is worth mentioning is the recommendation of the US Government for the federal agencies to use XML for information exchange. Actually, heterogeneous information systems can be integrated, as for example Java, COM, relational databases, message based systems like WebSphere de la IBM or ActiveEnterprise from TIBCO, management systems, SAP type applications and PeopleSoft, Oracle.

Most enterprises use database based applications, and that is why the strongpoint of a XML solution has to take into consideration solving the data communication between heterogeneous databases. The solution has to be able to detect all the modifications done at database level and generate XML files without destroying or modifying existing data, and especially without compromising the performances of the database.

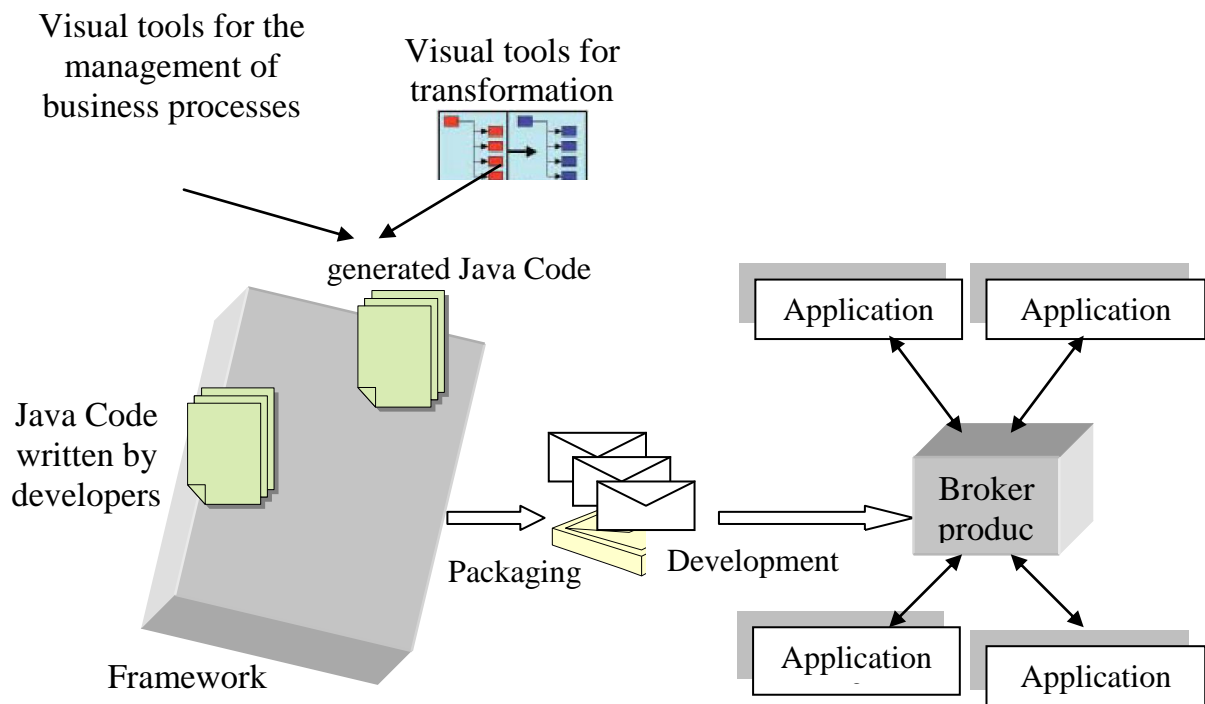


Fig. 8. Architecture used by integration brokers

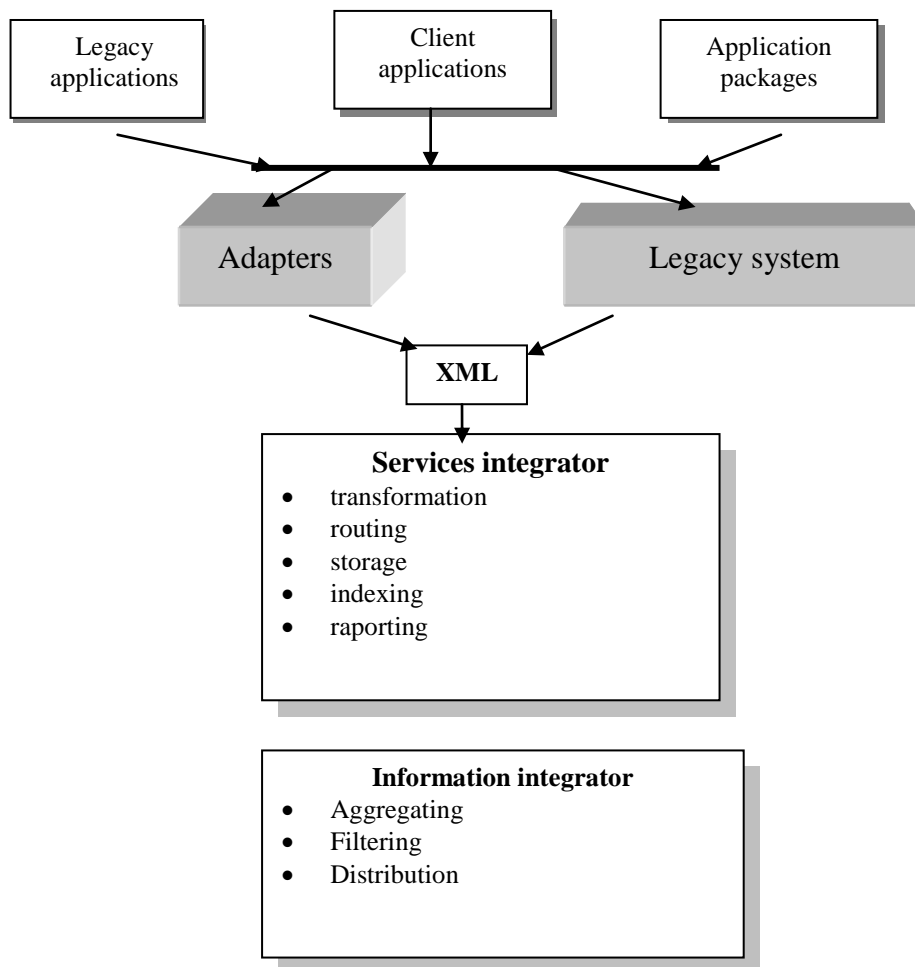


Fig. 9. Example of integration architecture using XML

4. Conclusions

Integration of information systems at enterprise levels provides the competitive edge. Data of great value exist already in different storage forms within an enterprise. Those enterprises that manage to use these data will be the first ones to gain an advantage in the competition with other enterprises, as they will have access to quality information regarding marketing, performances, customer care, etc. Also, the fact that they have these advantages will allow them to generate better predictions, be more flexible and react better to the changing market, and eventually to have better decisions than competition.

The paperwork presents a part of the new technologies and architectures for integration of the information systems that allow communication of data between heterogeneous databases and provide an integrated management for processes and documents, integration of new and legacy applications, management of the data storage space and their security, being detailed the policies prescript by authors after a study made on 76 Romanian companies for assuring the prerequisites for implementing integrated management systems, project being in course, funded by Romanian Ministry of Education, Research and Innovation. In this project we have as goal to determine, model and design new management approach, having in mind the integration of subsystems such as: HR, AQ, environmental preserve etc. In most cases emphasized by the study mentioned above, each subsystem is supported by one or more software applications so integration of those is an imperative. Another issue is that the applications are very diverse and each case imposes a unique solution of software integration. The actual software business solutions implemented at large companies level, because of their high selling and implementation prices are incomplete (customers are not buying entire application with all facilities and features) combining new solutions with modules legacies from past. Sometimes, even is cheaper to combine a new solution with an old one integrated solutions are recommended.

Future research will cover integration between types of application, (CAE, business support applications, delocalised solutions etc.) and integration between customer and suppliers process, with most commonly and market best sold solutions.

5. References

- Botezatu, C. P.; Carutasu, G. & Botezatu, C. (2008). Building Integrated Management Systems, *Chapter 11 in DAAAM International Scientific Book 2008*, pp. 125-136, B. Katalinic (Ed.), Published by DAAAM International, ISBN 978-3-901509-69-0, ISSN 1726-9687, Vienna, Austria, DOI: 10.2507/daaam.scibook.2008.11
- Botezatu, C. P. & Botezatu, C. (2006) New aspects of Software Development in Economy, *International Journal of Computers, Communication and Control*, Volume I Supplementary issue, ISSN 1841-9836, pp.100-104
- Bramer, M. & Terziyan, V. (2005) *Industrial Applications of Semantic Web*, Springer Publishing house, ISBN 0387285687

- Chang, C., Lu, C., & Chu, W. C. (2008). Improving Software Integration from Requirement Process with a Model-Based Object-Oriented Approach, *Proceedings of the 2008 Second international Conference on Secure System integration and Reliability Improvement - Volume 00* (July 14 - 17, 2008). SSIRI. IEEE Computer Society, Washington, DC, pp.175-176. DOI=<http://dx.doi.org/10.1109/SSIRI.2008.39>, ISBN:978-0-7695-3266-0
- Glazier, S. C. (2000) *Patent Strategies for Software, e-Commerce, the Internet, Telecom Services, Financial Services, and Business Methods (with Case Studies and Forecasts)* BI Law & Business Institute; ISBN: 0966143779.
- Hammer, K. & Timmerman, T. (2007). *Fundamentals of Software Integration*. 1st. Jones and Bartlett Publishers, Inc., ISBN:0763741337
- King N. (1999). The New Integration Imperative, *Intelligent Enterprise Magazine* available from: www.intelligententerprise.com, Accessed:2009-03-01
- Land, R. & Crnkovic, I. (2007). Software systems in-house integration: Architecture, process practices, and strategy selection. *Information Software Technology* 49, 5 (May. 2007), pp. 419-444. DOI=<http://dx.doi.org/10.1016/j.infsof.2006.07.002> ISSN:0950-5849
- Larsson, S., Myllyperkiö, P., Ekdahl, F., and Crnkovic, I. (2009). Software product integration: A case study-based synthesis of reference models. *Information Software Technology* 51, 6 (Jun. 2009), pp. 1066-1080. DOI=<http://dx.doi.org/10.1016/j.infsof.2009.01.001>, ISSN:0950-5849
- Li, H., Shi, A., & Meng, Y. (2008). "Parallel Development and Independent Integration Model" for Scientific Research Software engineering. *Proceedings of the 2008 international Conference on Computer Science and Software Engineering - Volume 02* (December 12 - 14, 2008). CSSE. IEEE Computer Society, Washington, DC, 244-246. DOI=<http://dx.doi.org/10.1109/CSSE.2008.1372> ISBN:978-0-7695-3336-0
- Liu, S., Tamai, T., & Nakajima, S. (2009). Integration of formal specification, review, and testing for software component quality assurance. *Proceedings of the 2009 ACM Symposium on Applied Computing* (Honolulu, Hawaii). SAC '09. ACM, New York, NY, pp. 415-421. DOI=<http://doi.acm.org/10.1145/1529282.1529375>, ISBN:978-1-60558-166-8
- Schmidt, J. (2002) EAI Principles Part I, Part II, Part III, Part IV, Part V, *Business Integration Journal*, TCI Publication Publishing house
- Zhao, W. (2006). *Model-Driven Integration of Software and Service Components*. Doctoral Thesis. UMI Order Number: AAI3253670., University of Alabama at Birmingham. AAI3253670
- *** DI-SIM Research Report (2009). Romanian Ministry of Education and Research, Phase II, No.2387
- *** Statistics for Romanian IT industry (2009). *Romanian Ministry of Information Technology and Communication* Available from: <http://www.mcti.ro/index.php?id=465&L=0> Accessed: 2009-06-08