FINANCIAL AND ACCOUNTING INFORMATION SYSTEMS INTEROPERABILITY

Iuliana Ionescu[1]
Bogdan Ionescu[2]
Florin Mihai[3]
Silviu Cojocaru[4]

ABSTRACT: The current economic developments have led to substantial changes in terms of how the financial and accounting activities are carried. The business environment can be characterized by a widespread consolidation of companies and their grouping in holding companies. Thus, the focus is on consolidation the financial data, and the integration and interoperability of the financial and accounting applications and also the integrated information systems have a major significance. With the development of the Internet, the architecture of the informatics applications has really been affected, this resulted in substantial changes in how the solutions are distributed and accessed as well.

The present paper analyses and provides viable solutions for the use of the technologies which are applied in the interoperability of the financial and accounting information systems, in the context of economic activity globalization. The dynamism and economic processes development in the new context have generated important researches in the informatics field which had resulted in simplifying the access way to applications (by simply using a Web browser) and also developing the interconnecting technologies for the financial and accounting information systems – their interoperability represents, in fact, the key word.
On this economic scenery, the financial and accounting information systems become very important solutions to integrate different business applications and also to offer a complete perspective for all the business.

Introduction

The current economic developments have led to substantial changes of how the financial and accounting activity is conceived. The business environment can be characterized by a widespread consolidation of companies and their grouping in holding companies. The mergers and acquisitions of investment funds affect the way the financial and accounting activity is done. Thus, the focus is on the consolidation of the financial data, on the integration and interoperability of the financial and accounting applications and also on the integrated information systems have a major significance.

With the development of the Internet, the architecture of the informatics applications has really been affected, this resulted in substantial changes in how the solutions are distributed and

---

[1] Bucharest Academy of Economic Studies, ionescu.iuliana@gmail.com
[2] Bucharest Academy of Economic Studies ionescub@gmail.com
[3] Bucharest Academy of Economic Studies, fmihai@gmail.com
[4] University of Bucharest, silviu@itchannel.ro

accessed as well. Taking into account all these considerations, one can say that accessing applications by using a simple Web browser came to meet the needs of a world characterized, among other things, by dynamism and the acceleration of the economic processes (**Ionescu, 2004**).

Thus, in the 90s, the main trend in the application development is the provision of support and accessibility - for all the computers connected to the Internet - to a much wide palette of informational resources (databases or other) and applications (**Stanek, 2005**). As a proof of this work is the multitude of languages and technologies that enable, an easy development of applications to exploit the database by using a simple Web browser. In this sense, the scripting languages run on the server side (ASP and its successor ASP.Net, PHP, JSP etc.), are now already mature technology, which actually offers the possibility to develop some complex applications for the Web.

Despite the existence of these technologies in the last years, there have taken place a lot of changes as concerns the informational needs or the equipments used by different users. Thus, now are connected to the Internet not only the computers, but also many other devices, such as, the mobile phones or hand-held type devices. Taking into account these needs, providing a universal language which is understood by all these very different devices, has become an imperious requirement.

In this context, XML language represents the answer to all these requests, being practically a new step in the evolution of informational age, this one facilitating data transfer among the tools of the most kinds. Beside the possibility of transferring data in the range of heterogeneous platforms, the need to arise to create some means able to represent the way of communication among the various applications. The answer to these priorities is the Web Services technology forming the most effective tool of communication among applications by using the Internet (**Ionescu, 2004**).
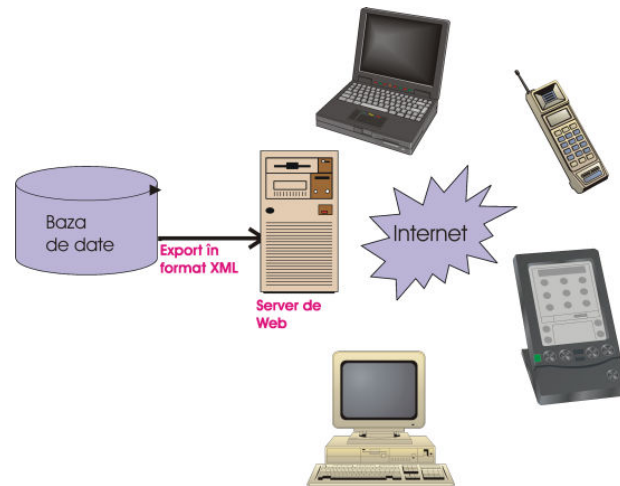
The Web services represent a modern method of communication between applications using the Internet. Using the HTTP protocol and XML language - the fundamental technologies underlying the Internet, the Web services allow the access to applications on absolutely any platform. The PCs, Macs, mainframe systems, mobile phones or hand held devices, all these will be able to communicate efficiently, by simply using the new technologies. It is expected that Web services and XML will have an impact similar to the creation of the Internet, in fact, the introduction of these solutions has consented a new age dominated by computer applications more intelligent and more easily accessible on various platforms (**Cojocaru, 2005**).

In the present article we tried to come under review the fundamental advantages of using this technology in the financial and accounting applications interoperability. In addition, we presented the details about the tools offered by various companies, for the implementation and development of this new technology.

1. Technologies underlying Web services

XML or eXtended Markup Language constitutes the basis for all elements which represent, in fact, the reason of the Web Services technology. Given the platform independence, XML is the engine that makes it possible to transfer data using the Internet, and also constitutes the foundation of the Web services (**Esposito, 2008**).

One can say that XML language represents a kind of "*envelope*" or a "*container*" in which the databases information is embedded and distributed by Internet, while having the assurance that the data stored in these files will be accessible to any device and on any platform (**Esposito, 2008**).

Figure 1. XML and Web services represent the basis of communication for a wide palette of hardware equipments

Web services are a standardized way of distributing applications that use the Internet and basic technology underlying this network (**Grimes, 2007**). Likewise, the Web services enable the possibility to interconnect a wide palette of applications available on different platforms and in different locations around the world. In this way, one can say that the Web services were set up in an Esperanto of communication between applications, the new technology widely opening the gateway to a new age dominated by applications with a high degree of intelligence, also capable of making decisions and finding information on the Internet, as a support for the decision as well-advised as possible (**Cojocaru, 2005**).

The Web services are built based on some fundamental technologies for the Internet operation. In this respect, to transmit any kind of information using the Internet network, these solutions use standard protocol named HTTP (HyperText Transfer Protocol) and, on the other hand, the transferred data are embedded in XML files. In this way, taking into account that both technologies represent the basic elements for Internet operation, these ones will ensure the accessibility of applications on any available platform.

Unlike Web services, their predecessors in accessing the applications using the Internet were the main disadvantage of not providing a range of equally high portability. To this end, the **DCOM** and **CORBA** technologies were not accepted by all the providers of IT&C solutions, so causing the decrease of their utilization range. Taking into account the heterogeneity of the Internet, a technology should be accepted by a wide variety of users, in order to attain the status of universal Internet technology.

Moreover, the DCOM and CORBA technologies use binary files to transfer data and allow the communication between applications, and this causes a lower portability of these technologies. In contrast to these, the Web services use XML language to transfer data which represents in fact the use of simple text files, thus ensuring a range of maximum portability.

Finally, the Web services use the port 80 on the server which is assigned by default to the HTTP protocol. In comparison, the other two alternatives use their own protocols for data transfer, which means the need to open some additional ports on server of applications. Accordingly, in a world dominated by the need for an ever higher level of security of information systems, this shortcoming of CORBA and DCOM technologies can have a negative impact on the entire architecture of the information systems.

In conclusion, one can consider that the spread in the recent years of the Web services is an outcome of their compatibility with two technologies (generally accepted) represented by the HTTP protocol and XML language. Given these benefits, the experts from IDC (the sampling and analysis

company of IT&C field) consider that the Web services will represent the fundamental way of applications distribution, outclassing all the other alternative technologies for the distribution of applications.

1. Web services architecture

Above all, it is necessary to specify that Web services are a standardized way of the WWW consortium. As we all know, this organization is composed of commercial and educational institutions and has as main objective to ensure uniformity and standardization of technologies in the Internet network. Moreover, given the standardization of Web services by an organization widely accepted as Web Consortium, it was assigned to this technology the title of universal technologies. Web services, regardless of the type of development platform, follow the same set of standards which ensure their adaptability and integration with different software platforms.

In essence, this technology is "built" on three fundamental elements that enable the integration and adaptation of specific functions. The first element represents the protocol **SOAP** (Simple Object Access Protocol), and this protocol is essentially based on XML language with the following functional characteristics:

- ✓ control transfer of data packet between the provider and the user of the Web services, by using the HTTP protocol (GET or POST method) to transfer the data packets between the server and the user;
- ✓ transfer of parameters which are set out by the user and are specific to the functions accessible through the Web service;
- ✓ return of the functions running results to the server which provides the Web service, these results are actually sets of data that were transposed into XML files.

The second element underlying the Web services is the **WSDL** (Web Services Description Language). In fact, the WSDL language is, in turn, based on XML language serving to inform the potential users of the Web service in connection with its specific elements. Thus, the WSDL language ensures finding information about the functions exposed by the Web service and, also, the related parameters that can be assigned to them.

Finally, the last element the Web services are based on is the **UDDI** concept (Universal Discovery Description and Integration). UDDI has a function bearing some resemblance with the browsers available on the Web, allowing users to search the Web services for their own needs. In fact, UDDI uses the information for describing the Web service, established through WSDL language, in order to offer potential users an efficient means of searching, supplemented by a set of information to use the Web service. The participants involved in the use of Web services are the following:

- ➢ **Web services provider** is the first participant in this process; the provider places a functionality represented by the Web service at the potential users disposal;
- ➢ **Web services consumer** is represented by the user who has access to that Web service and integrates it in its applications;
- ➢ Between the two participants it is possible to appear even an intermediary, represented by the **UDDI register** that allows consumer to find the service provider that offers the set of functionalities he needs.
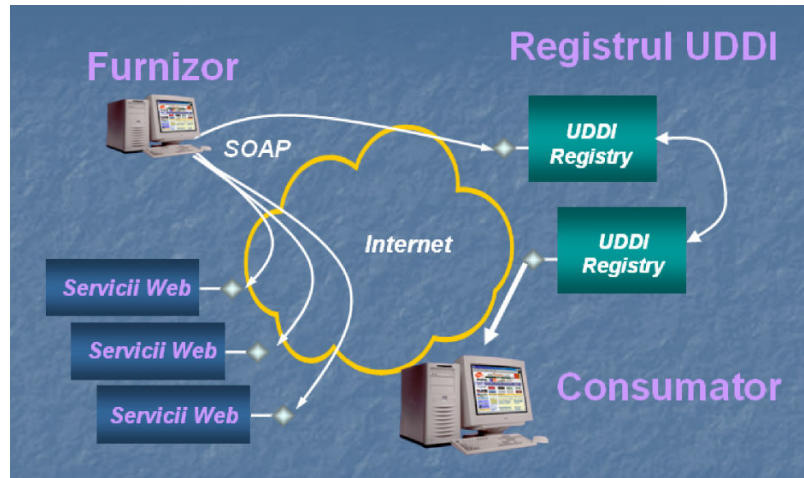
Figure 2. Operation means of Web services

2. Web service for presenting the product offerings to business clients

In terms of effective programming, a Web service involves on the part of the provider of such service, the defining of a special class and then the setting out of some functions by parameters. After service compiling, the provider will forward those interested, how to have access to this service, the class and the available functions as well. Those who will use this service basically won't know how to implement it, not even the language in which this one has been implemented, but only the elements necessary to use such service on their own applications and define where it will be used. From the moment when the offer is established, the company proceeds to detail this offer in two sections:

✓ Based on the client status, we must define the offer, including the date from which this one becomes valid. Moreover, depending on the status of the clients, the company details the discount rates and also the prices for each type of status. For instance, in some situations a company can provide a significant discount for the clients with preferential status. It is logical that the details about these discounts are absolutely confidential, not publicly accessible on the Internet, because the function that implements the Web service has defined the parameters by which the client has to provide: the client code, username and password (as we can notice in Figure 3).
In addition, the working environment is secured by using **SSL** (Secure Socket Layer);

✓ General offer for the public, in fact for the potential clients who are not yet in the company database; to ensure the access to this offer is free and is directly accessible from the company's website.

The listing (the version presented is a minimal one) of the Visual Basic.NET service, which implements the offer supply to all the registered clients, is presented as follows:

```
 1   Imports System.Web.Services
 2   Imports System.Data
 3   Imports System.Data.SqlClient
 4   Public Class Oferta : Inherits WebService
 5       <WebMethod(Description:="Oferta de pret clienti")> _
 6           Public Function OfertaPret(ByVal CodClient As String, ByVal Utilizator As String, _
 7           ByVal Parola As String, ByVal Data As Date) As DataSet
 8           Dim Conexiune As SqlConnection
 9           Dim Comanda1 As SqlDataAdapter
10           Dim ds As New DataSet()
11           'conexiunea la baza de date si procedura stocata ce se va executa
12           Conexiune = New SqlConnection("server=localhost;uid=AccesInternet;pwd=a!s@d#f$;database=Total")
13           Comanda1 = New SqlDataAdapter("OfertaPret", Conexiune)
14           Comanda1.SelectCommand.CommandType = CommandType.StoredProcedure
15           'parametrii procedurii stocate
16           Dim pCodPartener As New SqlParameter("@pCodPartener", SqlDbType.NChar, 20)
17           Dim pUtilizator As New SqlParameter("@pUtilizator", SqlDbType.NChar, 10)
18           Dim pParola As New SqlParameter("@pParola", SqlDbType.NChar, 10)
19           Dim pData As New SqlParameter("@pData", SqlDbType.SmallDateTime)
20           Comanda1.SelectCommand.Parameters.Add(pCodPartener)
21           Comanda1.SelectCommand.Parameters.Add(pUtilizator)
22           Comanda1.SelectCommand.Parameters.Add(pParola)
23           Comanda1.SelectCommand.Parameters.Add(pData)
24           'valorile parametrilor
25           pCodPartener.Value = CodClient
26           pUtilizator.Value = Utilizator
27           pParola.Value = Parola
28           pData.Value = Data
29           'executarea si transferul in DataSet
30           Comanda1.Fill(ds, "OfertaPret")
31           Return ds
32           Conexiune.Close()
33       End Function
34   End Class
```

As can be noticed in the listing above, the offer is provided by using a SQL Server stored procedure, called **OfertaPret**; the use of this procedure increases the degree of security, because it is stored in the database, not in the service code. However, the client of the service has no way of knowing that the offer is the result of a SQL Server stored procedure, thus he gets to use only one programmable component. The **OfertaPret** procedure code is the following:

```
CREATE PROCEDURE OfertaPret
@pCodPartener char(20), @pData as smalldatetime, @pUtilizator as char(10), @pParola as char(10)
AS

SELECT convert(char(10), Cp.Data , 105) as [Data Inceput Oferta], ART.CodArticol [Cod produs], ART.DenumireArticol Produs,
        UM, PretVCatalog [Pret fara TVA], CotaTVA [Cota TVA],
        Discount [Procent discount], (PretVCatalog-PretVCatalog*Discount) [Pret final fara TVA],
        (PretVCatalog-PretVCatalog*Discount) +(PretVCatalog-PretVCatalog*Discount) *[CotaTVA] as [Pret final cu TVA]
FROM CatalogPret CP INNER JOIN
--ultima oferta in functie de data pe tipul de statut
(SELECT    IdStatutPartener, CodArticol, MAX(Data) AS DataMaxOf
FROM      CatalogPret
WHERE Data<=@pData
GROUP BY IdStatutPartener, CodArticol) UOF ON CP.IdStatutPartener=UOF.IdStatutPartener AND CP.Data=UOF.DataMaxOf AND CP.CodArticol=UOF.CodArticol

INNER JOIN
--aici se verifica utilizatorul si parola
(SELECT P.CodPartener, ST.IdStatutPartener from Partener P inner join
--partenerul si idstatut
(SELECT    spp.CodPartener, spp.IdStatutPartener, spp.DataStatut
    FROM       stocuri.dbo.StatutPartener spp INNER JOIN
        (SELECT    dsp.CodPartener, MAX(dsp.DataStatut) AS DataMax
        FROM       stocuri.dbo.StatutPartener dsp
        WHERE     (dsp.DataStatut <= @pData and dsp.CodPartener=@pCodPartener)
        GROUP BY dsp.CodPartener) pss
    ON spp.CodPartener = pss.CodPartener AND spp.DataStatut = pss.DataMax) ST
ON P.CodPartener=ST.CodPartener
WHERE UtilizatorOnLine=@pUtilizator AND ParolaOnLine=@pParola) T2
ON  CP.IdStatutPartener=T2.IdStatutPartener
INNER JOIN Articol ART ON CP.CodArticol=ART.CodArticol

WHERE CP.OfertaActiva=1
GO
```
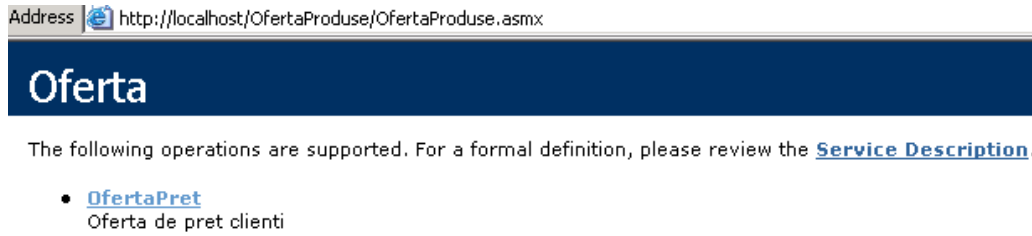
Likewise, the offer is presented depending on the type of the client status and the date provided by the client to be taken into account in offer supply. For information, the following figures show how to test the services developed by its provider:

After activating the hypertext link OfertaPret, test values for the parameters of the function OfertaPret must be indicated, as seen in the Figure 3.
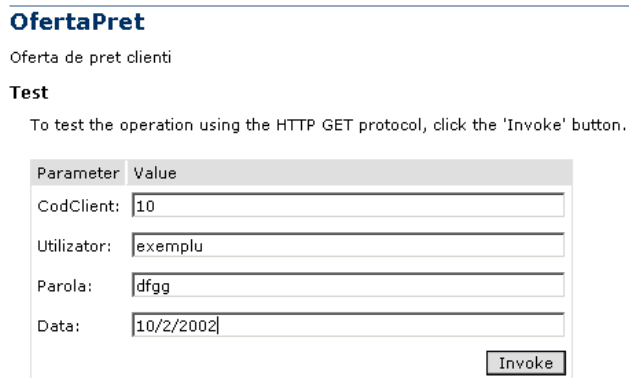

Figure 3. Parameters of the function OfertaPret

After setting the values of parameters, the effective testing is done by simply pressing the control button named **Invoke**, and the result will be presented as XML file. A sequence of this file is shown in the following figure:

The clients of this service will receive to use the following details:

- ✓ **method of access**: https:// 127.0.0.1/OfertaProduse/OfertaProduse.asmx
- ✓ **class:** Oferta
- ✓ **function to be called:** OfertaPret
- ✓ **function parameters and their type:** CodClient - string, Utilizator - string, Parola - string, Data - date
- ✓ **function result:** set of records **(**DataSet**)**
- ✓ **documentation** which is automatically generated in the case of service compiling in Visual Studio. NET, as can be observed in Figure 4.
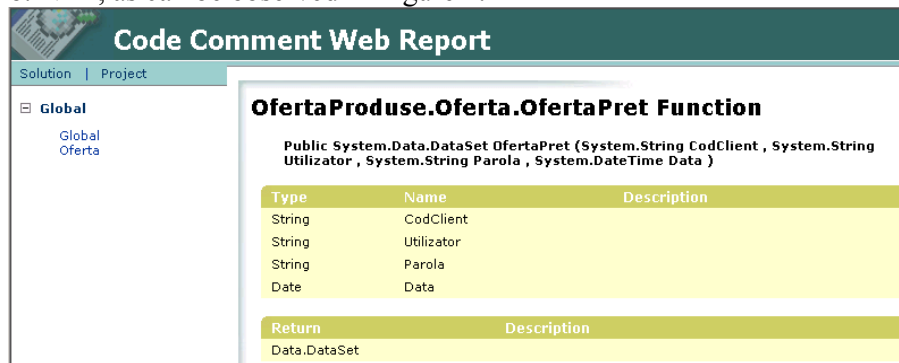

Figure 4. Providing a simple documentation for the service OfertaPret by the Visual Studio. NET environment

For an effective test of the service, in terms of authentication and security based on SSL, we have created, by way of example, a client for this service, an ASP. NET application to highlight the simplicity of its use. The listing of this client using Visual Basic. NET is the following:

```
Private Sub Button1_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) _
    Handles Button1.Click
    Try 'rutina de tratare a eventualelor erori cauzate de tastarea incorecta a parametrilor
            'instanta a clasei Oferta pusa la dispozitie de serviciul WEB
            Dim Oferta As New WebReference1.Oferta()
            'incarc obiectul DataSet cu datele transmise prin servicul WEB
            Dim ds As DataSet = Oferta.OfertaPret(Me.CodPartener.Text,
        Me.Utilizator.Text, _
            Me.Parola.Text, Cdate(Me.DataOferta.Text))
                SalvareDS(ds)
            'asocierea setului de date transmis de serviciul WEB obiectul DataGrid
                'al aplicatiei client
            Grid1.DataSource = ds.Tables("OfertaPret")
            Grid1.DataBind()
            Me.Button2.Enabled = True
            Me.CaleSalvare.Enabled = True
    Catch x As Exception
        Beep()
    Finally

    End Try
End Sub
```

From the code presented above it may easily seen that it cannot be inferred that the company providing the service has a database in SQL Server, the stored procedure which is performed, or the fact that the service is written in VB. NET. The client invokes the function called OfertaPret:

Dim ds As DataSet = Oferta.OfertaPret(Me.CodPartener.Text, Me.Utilizator.Text, Me.Parola.Text, CDate(Me.DataOferta.Text))

The Web window in which is performed the authentication, the display of corresponding offer and the possibility of saving it, is shown in Figure 5.

| Address | https://localhost/ClientOfertaProduse/ClientServiciuOfertaProduse.aspx |

Interfata aplicatiei client va fi conceputa in functie de necesitatile clientului

Codul de partener din baza de date a firmei ABC	10
Numele de utilizator din baza de date a firmei ABC	exemplu
Parola din baza de date a firmei ABC	
Data la care vreau sa vad oferta firmei ABC	2/2/2002

Afiseaza oferta　　**Salveaza oferta**　　c:\oferta.xml

Data Inceput Oferta	Cod produs	Produs	UM	Pret fara TVA	Cota TVA	Procent discount	Pret final fara TVA	Pret final cu TVA
01-01-2002	10	CARNE AMESTEC I	KG	100000	0.19	0	100000	119000
01-01-2002	11	CARNE PORC CAL II	KG	150000	0.19	0.05	142500	169575
01-01-2002	12	CARNE VITA CAL I	KG	200000	0.19	0	200000	238000
01-01-2002	13	AMLIORATOR	KG	300000	0.19	0	300000	357000
01-01-2002	14	EMULGATOR	KG	100000	0.19	0	100000	119000
01-01-2002	15	ORGANE PASARE	KG	200000	0.19	0	200000	238000
01-02-2002	16	CARNE PASARE I	KG	350000	0.19	0.01	346500	412335

Figure 5. The window of client application for OfertaProduse service

As it can be seen in the application code and the Web window shown in Figure 5, the content of the offer can be saved at any time in an XML file; then this offer could be integrated into its own database or used for other purposes, even on other platforms.

Conclusions and advantages

We live in a world dominated by the need for interoperability among different information systems available on various platforms (**Grimes, 2008**). Accordingly, the Web services represent a technology able to efficiently solve this problem. The simplicity of the Web service is actually a resultant of compatibility with the basic Internet technologies; this simplicity ensures the universal acceptance of this solution as a way of communication among applications. In addition, the relatively simple development of these elements will undoubtedly contribute to the spread of the said elements (**Prossise, 2007**).

The integration of the financial and accounting applications with Web (**Mihai, 2004**) is a necessity which is not questioned anymore by anyone. The syntagm "*you're not online, so you don't exist*", affects a company's informational system in all aspects of communication, with, to and from it. The financial and accounting information, provided within certain limits on the Web, acquires new values, in terms that clients can easily see anytime the current offer of products addressed to

them, so they can check out their own transactions with a particular company and may place orders online or can pay online, etc.

The present application is quite relevant in this respect, even if only a small part of what can be done. The integration of the financial and accounting application with Web should not be viewed only in terms of the relations with the outside part of the company, but also with the internal side of its informational system. A company that has developed a well-formed Intranet can transfer a part of applications into the Intranet network, and if the company has some subsidiaries in the territory, the advantages of such an approach are clear. The integration with Web applications could be seen in order to win new clients, new markets and low costs advertising.

References
1. Ionescu, B. (2004) et. all, *Informatica utilizatorului*, Bucureşti, Infomega
2. Mihai, F. (2004), *Sisteme informaţionale financiar-contabile,* Bucureşti, Infomega
3. Cojocaru, S. (2005), *Un site web la îndemâna tuturor!* , Bucureşti, Compania
4. Stanek, W. (2005), *SharePoint Products and Technologies*, Microsoft Press
5. Richter, J. (2008), *Applied Microsoft .NET Framework Programming*, Microsoft Press
6. Grimes, R. (2008), *Dezvoltarea aplicatiilor cu Visual Studio .NET* , Bucureşti, Teora
7. Esposito, D. (2008), *Applied XML Programming for Microsoft® .NET* , Microsoft Press
8. Prossise, J. (2007), *Programming Microsoft® .NET,* Microsoft Press
9. Grimes, F. (2007), *Microsoft .NET for programmers*, Manning
10. Lowy, J. (2008), *Programming .NET Components*, O'Reilly
11. Chappel, D. (2009), *Understanding .NET*, 2nd Edition, (Independent Technology Guides), David Chappell Press