

Portal management

Chavdar Marchev

Abstract: The paper describes the structure and the opportunities afforded by the new programming framework of Microsoft – Visual Studio .NET. A concrete application has been considered – an abstract model of an Internet portal using XML and RDF data bases. Some advantages of that framework have been described. They are relevant to the stage of the physical creation of the portal.

Key words: Framework, XML specification, RDF model, catalogues

INTRODUCTION

The term “Internet portal” means a site, which can be used as a starting point to other sites. Usually portals have their own searching engines, free of charge services, e-mail, forums etc. The management of an internet portal includes all operations related with its maintenance and functionality. In this paper an internet portal based on XML specification and RDF model has been considered. The tool chosen for its realization is Visual Studio .NET Framework of Microsoft Company.

STRUCTURE AND POTENTIALITIES OF VISUAL STUDIO .NET

Microsoft .NET is a framework working on Windows 2000/XP platform. It gives the programmers an opportunity to create efficient programs quickly. They also can start their programs and components from the framework itself.

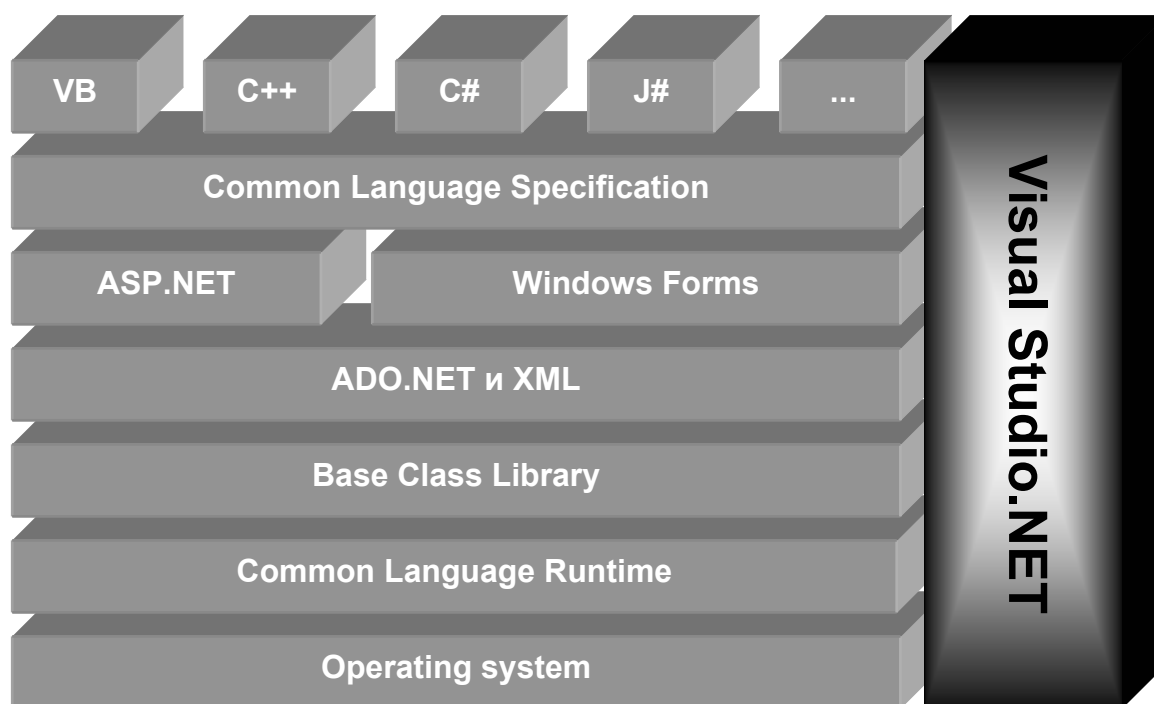


Fig.1 The architecture of .NET framework

The architecture of the Framework is presented on fig.1. It consists of three levels. The following list presents only characteristics, related to the portal development.

1. The Common Language Runtime (CLR) layer is situated on the lowest level (over the Operating System). CLR provides a stable and robust environment while starting programs.

2. The Base Class Library is at the intermediate level. Some of the classes (in System.XML namespace) afford enough functionality to process XML documents. The nodes in an XML file are encapsulated by means of objects (C# is an object-oriented programming language). To approach the relevant elements (nodes of the hierarchical tree) in the XML document, C# uses XPath syntax.
3. There are several programming languages on the top of the .NET structure. The application described here has been developed by C#. It is a very powerful programming language. Its advantages are as follows:
 - simple syntax;
 - Object-oriented;
 - based on the well-known C/C++ syntax;
 - it combines the high productivity of Visual Basic and the unlimited possibilities of C++;
 - Parts of the framework itself have been coded in C# [2].

MANAGEMENT OF RDF BASED INTERNET PORTAL

Here the concrete realization of a portal uses the RDF model for cataloguing resources. By manipulating the portal the user has the ability to input and search information about real processes (publishing, selling etc.). At this stage the model includes only the process “publishing”, i.e. the visitors of the portal may search information and the authorized users may input information about books, magazines, papers, Internet pages, Word documents etc. (fig.2)

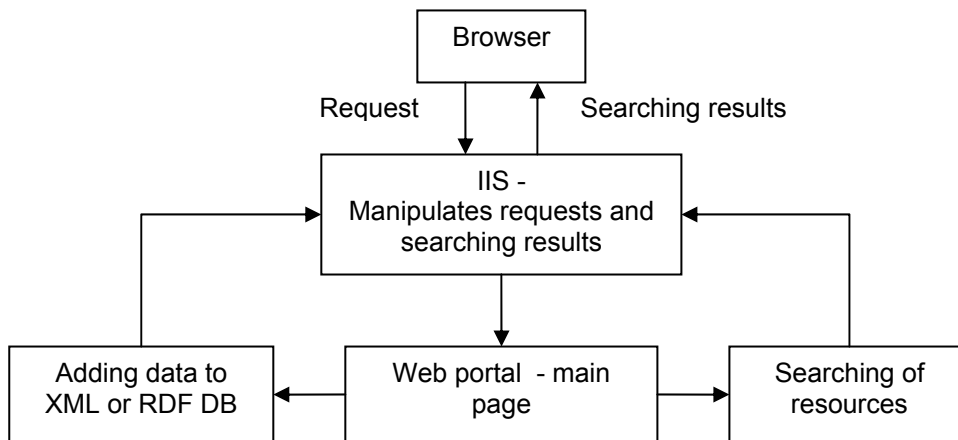


Fig.2 General Sketch of data flows

When the user's browser sends a request to .NET server, it is caught and processed by IIS (Internet Information Services). This is the place where the analysis of the request is performed. The .NET Server manages the data flows. When a page is browsed by the user .NET server renders the page according to the capabilities of the browser. The process is fully automated and there is no need of considering different versions of user's agents when coding the applications. There are three directions which lead out of the main page.

Adding description of a new resource (document) in RDF DB

One of the afforded opportunities in that portal is - describing new documents (fig.3). The descriptions of documents follow the RDF syntax.

Briefly, every RDF document follows the XML specification. Each document is divided into sections – every section describes separate resource. Each resource has a

unique identifier which follows URN specification and several properties with relevant values. Most of the values are primitives (strings, dates). In the RDF terminology they are called literals. Also there are more complex values, which may be other resources too [3].

A part of the necessary information about the document (area, direction and theme) can be retrieved from the XML classifier. When there is no such information, the user has the opportunity to add new items to the classifier. When the user adds a higher element into the structure (for instance - area), he is obligated to define a relevant direction and a theme. That is necessary for preserving the data integrity.

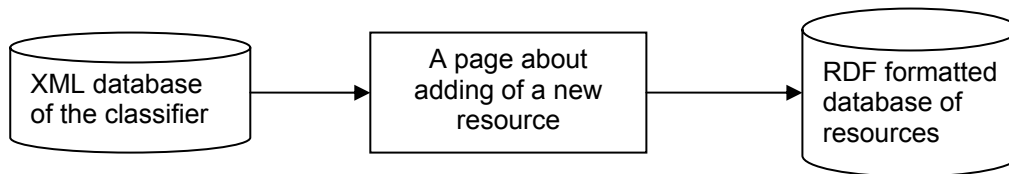


Fig.3 Adding a resource's description

Adding new elements to the XML catalogue (classifier)

Essentially an XML DB is a document which has been created according to XML syntax rules. It has a root element and a hierarchical structure corresponding to the areas that are available in the classifier. There are no other areas (directions and themes) except those which are already used in the RDF DB. This structure has three levels in depth (area, direction and theme). The following figure presents the opportunities to maintain the catalogue (fig.4).

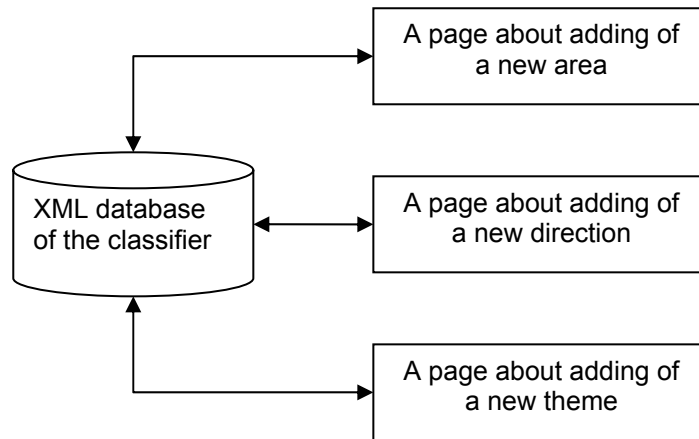


Fig.4 Adding a new element to the XML classifier

Searching resources in RDF DB

Searching is the third opportunity (fig. 5). It is performed within the RDF DB. Searching is based on several criteria. It is important to emphasize that searching forms, relevant to XML classifier, are realized by combo boxes. Thus the possibility of making mistakes when the users input data are avoided. It is impossible to add "pending" areas (without directions) to the XML catalogue. The adding of the directions is realized in the same way. The combo boxes have "interactive" behavior, i.e. when somebody makes his choice in a combo box two processes are performed:

1. Searching the RDF database and filtering the information on the basis of chosen elements from activated combos. Although this delay appears to be

senseless, it has the advantage that the user could stop the searching and restart it using other criteria, at every time.

2. Defining the contents of the successive combos in the sequence. The choice that has been made in the previous combo defines the contents of the following one. Thus the number of the elements in the combo boxes is reduced to its minimum and the afforded opportunities are adequate.

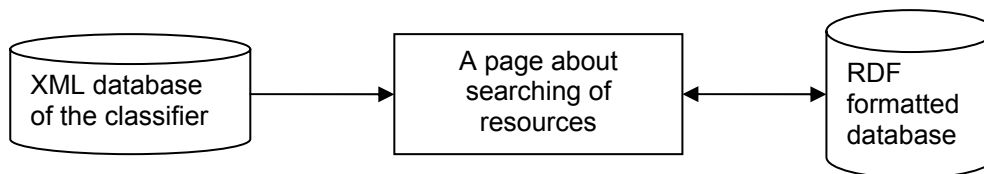


Fig.5 Searching the resource

The results are shaped as a table, in which the first column contains address of the resource according to URL specification, such as entered by the user. He has the ability to follow the hyperlink.

CONCLUSION

From contextual point of view the RDF model may be considered as a definitional semantic network. At first, this type of semantic networks introduces more common conceptual types. Consequently these types are described by simpler types. Their structure is hierarchical. A graphical representation of the definitional networks is almost identical with the representation of RDF model. It is expected that the main part of the future Common Web Space will be constructed of semantic networks. Probably some of them will store their data in RDF databases. Designed to be easily maintained, they will help users in working with Internet. Searching of information will not result such enormous lists of useless addresses. Of course, that depends on the acceptance of common standards of describing data.

REFERENCES

- [1] Jones, Russell. Mastering ASP.NET with C#. Sybex, 2002
- [2] Platt, D.S. Introducing Microsoft .NET. Microsoft Press, 1996
- [3] RDF Primer. <http://www.w3.org/TR/2004/REC-rdf-primer-20040210/>

ABOUT THE AUTHOR

Chavdar Marchev; PhD student in department "Computer systems and technologies", "St. Cyril and St. Methodius" University of Veliko Tarnovo, home phone: +359 62 35363; mobile: 0888 78 65 64; e-mail: choky@programmer.net