e-DEDALUS: A dynamic tool for data publishing on the web

Apostolos Apostolopoulos [2], Christos Bouras [1,2], Vassilis-Javed Khan [2],
Agisilaos Konidaris [1,2], Christos Sintoris [2]

[1] Computer Technology Institute-CTI, Kolokotroni 3, 26221 Patras, Greece

[2] Department of Computer Engineering and Informatics, University of Patras, 26500 Rion, Patras, Greece

E-mail: `bouras@cti.gr`

Abstract— **The World Wide Web (WWW) provides a great opportunity for companies and individuals to publish their ideas, products and everything they would like other people to view. Since the Internet community and the Web are expanding every day, the demand for user friendly Web publishing applications is rising. The procedures that could be followed by Internet users in order to publish content on the web till now, were to either buy and read some books on HTML and have some basic knowledge on Web Servers, or pay an ISP a quite substantial amount of money to do it for them. In this paper we first justify the necessity of the existence of an alternative method for online web publishing. We then describe the method and a tool, called e-DEDALUS, which can be used by many diverse Internet user groups. An Internet user that has no previous experience in programming, or desktop design, will be able to create his/her own Web site, online. The e-DEDALUS tool that will be described in this paper aims at offering simplicity, customisability and multilingual support to a substantial number of WWW users.**

Keywords— **Web Data publishing, PHP, Databases, Web Interface**

Fig. 1. The gradual growth of WWW users (light) in relation to the growth of Internet users (dark) in recent years in US/Canada [16].

I. INTRODUCTION

AS the Internet community is expanding, the World Wide Web is becoming more than another communication facility. In our days the WWW is considered to be a basic tool for communication, business and advertising. The number of Internet hosts as well as that of Internet users, (Figure 1 [?]) has risen rapidly in recent years. Even if we consider every host to contain only one Web site (even though that is not true because each host contains more than one sites) we end up with a vast amount of web sites and consequently a large amount of information.

Together with the growth of the Internet we have also experienced a substantial growth, in the number of Internet users. The term "Internet users" refers to individuals, businesses or special interest groups using the Internet to either publish or acquire information.

Before ten or fifteen years most Internet users could be considered experienced. The number of Internet users was relatively small and usually had special knowledge on computer networks and programming. In our days the situation is very different. The profile of the Internet is not so far from the profile of an everyday person. The special skills that are required to use the Internet have been limited to the use of a mouse. It is easy to conclude by this fact th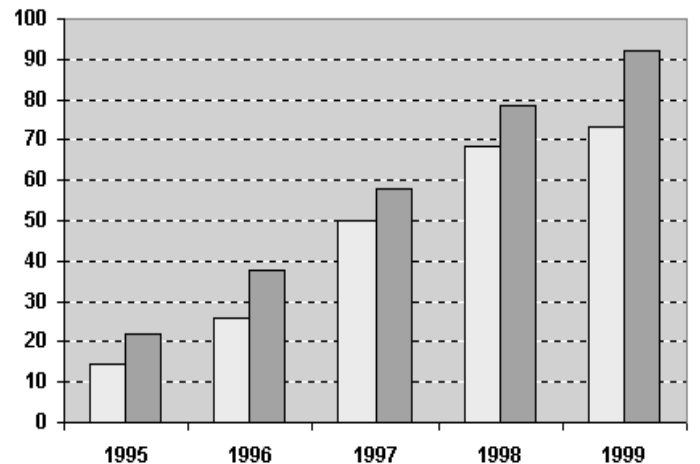at the vast majority of today's Internet users can be considered inexperienced in the fields of computer programming, web administration and computer networks.

It is almost common knowledge that one does not have to understand HTML to "Surf the Web". This is a fact for people that don't want to publish information on the Internet but only want to browse. On the other hand, we can conclude that more and more of the Internet users want to publish information on the Internet. This of course brings up the problem of HTML knowledge and other Internet technologies. The question that can be asked at this point is: "Does an Internet user have to know HTML and other programming languages in order to create his/her own Web Site?".

The answer to this question is different according to what we specifically mean by "Internet user". The individual Internet user and the business Internet user will answer the question differently because they intend to use the Internet in different ways. Even various types of business users will provide different answers, since not all have the same available resources to invest into the Internet. What can be concluded, in any case, is that the Internet users, are a very diverse group of users. They have different aims and needs but most of them want to be actively present on the World Wide Web.

These different aims and needs can be served by grouping them in an abstract manner. This approach can provide more general and more flexible solutions. The main disadvantage in this case is the production of web sites which have a more or less similar structure. But, as we stated above, this disadvantage is minor for the majority of Internet users, since rapid web site development can be considered as their main objective.

In this paper we first present the current situation and then compare similar tools to e-DEDALUS (section 2). In section 3 we continue with the reasons and the motivation behind the attempt to develop a web tool for automated and online web site creation. Then, in section 4, we present the architecture of our tool accompanied with an example that shows an implementation instance of a business web site. In section 5 we point out the advantages and benefits of the tool and, finally, in section 6 we make some considerations about future enhancements which could extend the usage of the tool.

TABLE I

Comparison table between similar tools that work with wizards

	e-DEDALUS	Homestead	SiteBlazer	WebCity	NetObjects
Characteristics concerning the user					
Simplicity	√√	√	√	√	⊗
Number of yielded pages	6	many	3 - 10	3 - 10	unlimited
Multilingual	Yes	No	No	No	No
Variety of styles & templates	√√	√√	√	⊗	√√
Remote administration	√	√	⊗⊗	⊗⊗	⊗⊗
Characteristics concerning the service provider					
Tool Translation	Yes	No	No	No	No
Platform Independence	Yes	No	No	No	No
Legend	√√ very good	√ good	⊗ poor	⊗⊗ very poor	

II. Present situation - similar tools

Online web editors exist already and are widely used by companies providing free web space. Most of them are very plain and inflexible tools which create a fixed number of static, HTML web pages, only in English. A well known tool that can be found at http://www.siteblazer.com is SiteBlazer. SiteBlazer creates small business sites but it's model requires the fitting of the company into a predefined category (e.g. automobile, bookstore). After extensive research, no general purpose online tool was found on the web, although there were several projects under way. Those either require experience in web developing and they have limited capabilities. On the other side, development tools like Microsoft FrontPage can be used to construct any kind of website one may wish, but extensive knowledge of web technologies is fundamental in order to have satisfying results produced.

There is also a number of offline tools that assist inexperienced users in creating web sites. Some of them offer a considerable amount of capabilities, while others are very simple and can be compared with the online tools mentioned above. The first category includes tools such as the widely used NetObjects Fusion [?] which offers dynamic database connectivity and e-commerce capabilities with the use of a user friendly environment. The other category contains tools like the Web Wizard [?] which have limited capabilities and produce very simple web sites.

Some of the abilities of e-DEDALUS are compared with similar characteristics of a sample of other web publishing tools (Table 1). Three of them, homestead, SiteBlazer and WebCity, are on-line tools like e-DEDALUS, while NetObjects Fusion is an off-line tool. In the case of the latter, it is obvious that there is no distinction between "Characteristics concerning the user" and "Characteristics concerning the service provider".

Compared with the other on-line tools, e-DEDALUS has some clear advantages. From the user's point of view, e-DEDALUS is a simple, multilingual and easy to administer tool which can offer a great variety of styles and templates. As for the service provider, platform independence and the ability to translate the tool are two significant features.

III. Motivation and goals

The motivation behind the idea to implement a tool that can be useful to a large number of Internet users, in order to assist them in publishing their web sites, lies in the current situation of the Web. Statistics show that the number of web pages that exist today is over 1.630.000.000 and is growing at a pace of almost 0.2% every day [?]. If we combine the number of web pages with the increase rate of Internet hosts we can easily conclude that more and more people want to publish information on the web every day. It is also easy to conclude that as the number of Internet users and publishers is rising the quality of published information is getting worse. We can also assume that the computer skills and knowledge of the average Internet user are not what they used to be some years ago but are actually a lot worse.

Apart from the exceptional growth rates, another major characteristic of the Internet is the rapid aging of the information itself. Out of date information can be considered as no information at all. The need for updating information is therefore crucial for any kind of web presence. Any attempt to create a web publishing tool should take this fact into consideration in order to incorporate easy to use and efficient updating mechanisms.

These facts led us to believe that an application for web site creation could be beneficial to the Internet community in two ways. The first goal that could be accomplished is the goal of standard quality. Our application frees the publisher from having to worry about the presentation of his material and allows him to focus on the content itself. Of course, as we all know, the presentation part plays a

major role in order to attract potential users of the site. That's why we have put in an extra effort to provide many different presentation templates that will surely satisfy a large variety of users. Having done this we can ensure that the quality of the pages contained in the publisher's web site will be high. The other goal that we set out to accomplish was to be able to satisfy a wide range of web sites and web publishing groups. This was our main goal and we worked towards it from the beginning of our work. Finally, we aimed at providing an interface which simplifies the procedure of updating information. Thus, the whole application is implemented in such a way that it ensures multilingualism, diverse web site creation and future expansion. The methods used to accomplish these goals will be explained in the following paragraphs.

In the next section we will describe the principles of our design and the architecture of e-DEDALUS. We will also present the basic steps that are required for the user in order to have a web-site produced.

IV. Architecture

The tool consists of four basic parts, the wizard, the administrator, the database and the final site (Figure 2). The wizard is used to collect the information provided by the publisher in order to create the web site in the final step. This information is collected by input forms and is forwarded to the final step where the actual insertion in the database takes place. The reason for this implementation is that we had to ensure that no data would be inserted in the database prior to the completion of all the steps. Thus, in order to avoid having incomplete datasets in the database and considering the fact that PHP did not provide session management at the time we begun the implementation of the tool, we decided that all the data should be inserted in the final step. The implementation of database scripts that could erase incomplete datasets (e.g. PL/SQL) would force us to be depended upon a specific database system.

The database design makes a distinction between translatable and non-translatable fields (e.g. emails and phone numbers are non-translatable, while page titles are translatable). For each entity we have two tables: table_data and table_text. The table_data tables contain non-translatable fields and table_text tables contain translatable fields with an attribute that indicates the language used. The translation of the fields is done through a translation form. This concept allows multilingual presentations of the site.

The final site is created during the last step of the wizard. During the final step, the user gets an automatically created password which is to be used with the user's email in order to get access to the administration wizard. The only static part in the final web site is its layout, which depends upon the template that the user has chosen. All other data, style-data and user-input, are dynamically retrieved from the database.
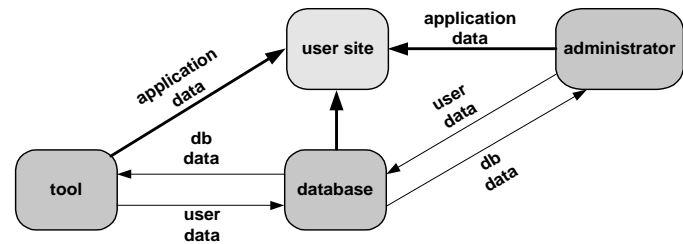

Fig. 2. The application's architecture.

A. The wizard

The wizard interface is a sequence of basic steps in the form of insertion forms. These forms are divided into three logical groups depending on the kind of information that the user provides. We distinguish between data that refer to the user himself, like name, surname, email etc, which are used only for logistic purposes. Next we have data that will actually be presented in the site, like company data for example. Finally, the user provides data that determine the way the site will look, e.g. colours. Before the user can enter any data he has to choose a language from the available in the database and the wizard will be presented in that language. The wizard can be multilingually presented through a concept that enables the dynamic loading from the database of any text displayed to the user according to his language preference.

To access the wizard visit: http://iapetos.ceid.upatras.-gr:2000/wizard/start_page_of_wiz.php3

An implementation instance of the wizard interface is presented in the following section.

A.1 An implementation instance of the wizard

Initially, the wizard allows the selection of the type of site that the user would like to create, e.g. commercial (Figure 3) or hobbyist site. Depending on that choice, the wizard asks for the required data, which are of two kinds:
• The data that will be stored in the user's data module of the database.
• The data that will determine the look of the final site, like templates, colors, fonts etc. This data has already been inserted in the application data module of the database. After the user has selected which of them he will use, those are also copied to the user data module of the database.

During the last step, the wizard produces the user's site. At this point we were presented with an implementation choice. The site that would be the outcome of the wizard interface could be stored into the web server's directories in two ways. As static pages or as dynamic pages. In the dynamic version, no actual site would be created, just some dynamic code. Every time a client requested the "site", this code would produce dynamic content according to the information inserted by the creator of the "site". On the contrary, in the static version, the whole site would be actually stored in the web space. The dynamic, "on-the-fly" web site creation, provides an abstract approach and is more elegant, though the performance disadvantages are
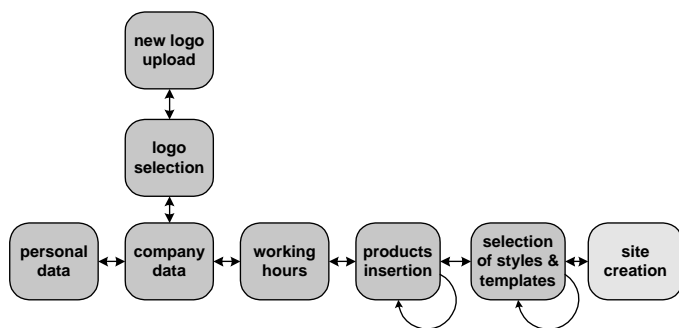
Fig. 3. Steps required to produce a business site.

substantial. In the static version the conclusions are exactly the opposite. We selected the static version because the overhead, for the web server, by the "on the fly" creation of web pages was quite substantial and performance degraded.
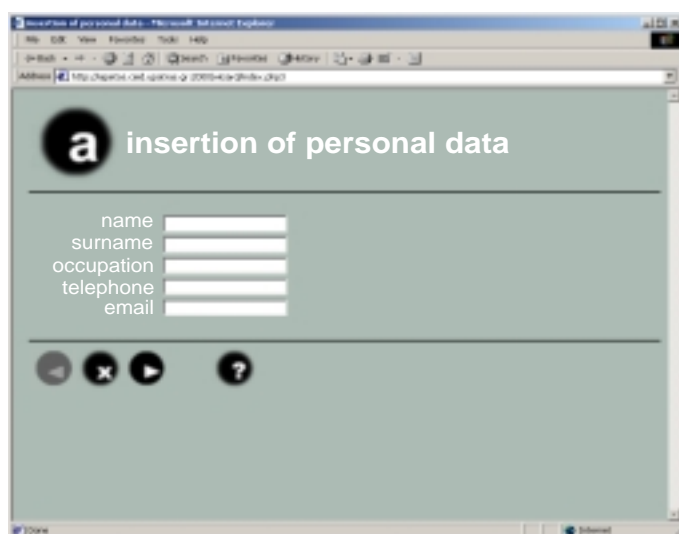

Fig. 4. The first step of the wizard.

Figure 4 shows a screenshot of an implementation instance of the "wizard" when used to create a commercial web site. In the first step information is collected about the user himself (Figure 4). The data entered in this step (name, surname, occupation, phone number and e-mail), are stored in the database to keep a record of the user, as well as to use his e-mail for login, in the administration wizard (described later on). During the next steps the wizard gathers data about the company (such as: name, email, address, timetable and choice of logo among some images offered from the database, or even uploading user's own logo). In the last step of the wizard, the user has an option of plenty of styles and templates to choose from, for the appearance of his web site. In every step of the wizard there is a help button available with which the user can read instructions for the completion of the forms.

In the final step of the wizard, the user can have an overview of the final result of the appearance of his entire web site. Additionally, he can change - in this step - the text presented on the web site toolbar buttons.

B. The database

The database is used to store the data that the user enters and additional data needed for the tool to operate. It is divided into two independent parts, one holding application data and the other holding the user's data. The application data contain information like the available styles and templates, or, more substantial language data, which are used to provide a customized interface according to the language the user chooses. The extension of the tool with more languages is a very simple task. One, only needs to determine the new language, the text in that language and the position in the wizard in which this text will appear.

C. The administration wizard

The administration wizard is used for maintaining and updating the web site and the information that the web site contains. It allows the manipulation of the data already inserted and the addition of new data. The user is free to change every possible detail of the site as if he is creating an entirely new site. The concept of the administration wizard (Figure 5) is simple: the user can update all the relevant data in the database or he can add new data, e.g. new products, so he only has to interact with the database. The actual html code is updated only when the presentation of the web site is to be changed.
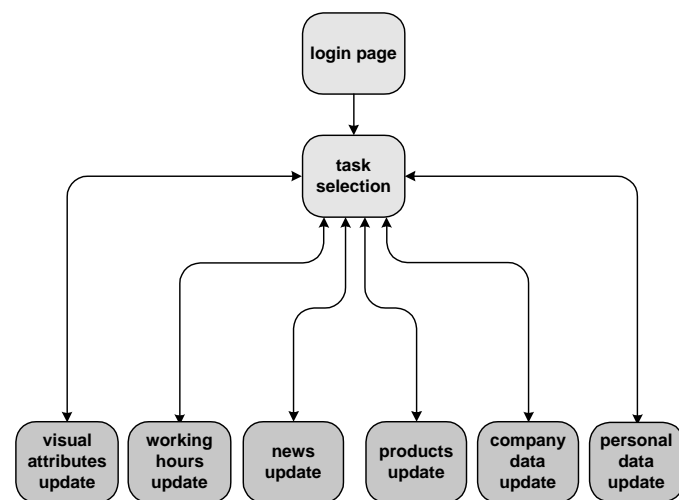

Fig. 5. An outline of the administration interface.

An implementation instance of the administration interface is presented in the following section.

C.1 An implementation instance of the administration wizard

The administration wizard allows the updating of all data inserted by the user in the creation process of the site. It has also been designed to be as easy to use as possible ([?]). The only entries that the user can't change are his name and surname. If the user forgets his password

([**?**]), he will receive it automatically just by entering his email. With the administration wizard the user can delete and create product categories, insert, delete and update products for each product category, update the company's opening hours, maintain his news section etc.
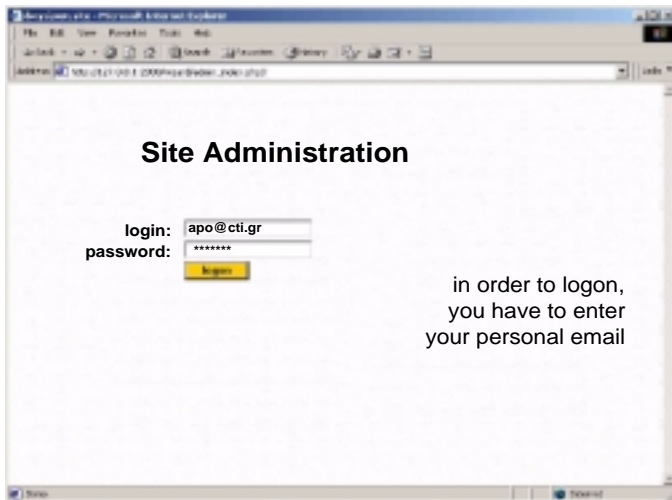

Fig. 6. The page used to log into the administration wizard.

D. Creation of the site

After the last step of the wizard is completed, all the data that was in the form of variables, is inserted into the database. The technique of "carrying" the data inserted by the user through all the steps of the wizard was chosen in order to ensure that no data would be inserted into the database until the final step. For the insertion of the data, the incorporated ODBC-functions of PHP ([**?**], [**?**]) are used. Then, a new folder is created in the directory of the Web-Server, named "site_¡id of the user¿" (example site-s). This folder contains the site's images, as well as the appropriate PHP3 files. The final site that uses the data from the database to present the information the user wants to publish. This on-the-fly approach provides many advantages as shown in [**?**].

In the following paragraphs we will describe an implementation instance of all three Web interfaces of our tool. The implementation instance is that of a commercial web site. The processes of site creation and editing will be thoroughly described as well as the final outcome of the procedure, which is the resulting web site.

D.1 Final Result - An implementation instance

In our implementation instance the final site consists of seven PHP pages. The pages contained in the web server are:

• An introduction page that is a page containing information about the company (address, phone number, opening hours etc).

• A contact page through which users can send email messages to the company.

• A news page which can be updated at any time by the owner of the page through the administration wizard.

• A company timetable page that holds information about the company working hours.

• A product category that contains the categories of products that the company provides together with some short descriptions.

• Two specific products page that contain specific product information such as product presentation, purchase costs etc.

The product categories pages contain links to the product page for each product category. These work like templates, retrieving dynamically all the data they present from the database. This allows easy updating of the web site through the administration wizard, which only needs to update the database entries and not the actual web pages.

V. ADVANTAGES AND BENEFITS OF THE TOOL

The advantages of our tool, when compared to other similar online and offline applications, are significant. We have grouped these advantages into two major categories. The advantages that facilitate the user and the advantages that facilitate the provider of the online application. Before we go on to describe the specific advantages of the tool we will present the basic advantage that is a result of the architecture that was presented in Figure 2.

The basic advantage of our tool is reusability. With minor changes in the design of the database or the web interfaces, the tool is ready to provide functionality to a variety of Internet users. In other words a company and a home user can both use the tool to produce their web sites. This is possible because of the design architecture that was followed, and is explained in Figure 7.

After defining some potential user target groups we mapped all wizard fields of every potentially created web site into one database column. This is easily understood by the following example. The path of the image representing the logo of the company in the company web site is stored into the database. The same is done with the path of the picture of a user when creating a personal home page. We mapped both paths to the same database column. Even though they represent different entities, the database does not distinguish between them. In other words the database holds all data for all potentially created sites in the same tables and the distinction between them, as well as how they will be shown on the web page, partially relies on the user and partially relies on our implementation strategies. It is easy to conclude that by mapping wizard fields one to one we can create any type of web site and a corresponding wizard without having to change the database or web interface structure. Furthermore, due to the underlying structure of the tool, enhancements in the direction of the style and feeling of the finally produced site are very easy to do. This is because we make a distinction between the data the site presents and the way they are presented. Moreover, the presentation is based on two modules: a style module and a template module. The style module defines attributes such as colors, font styles and sizes and the template mod-
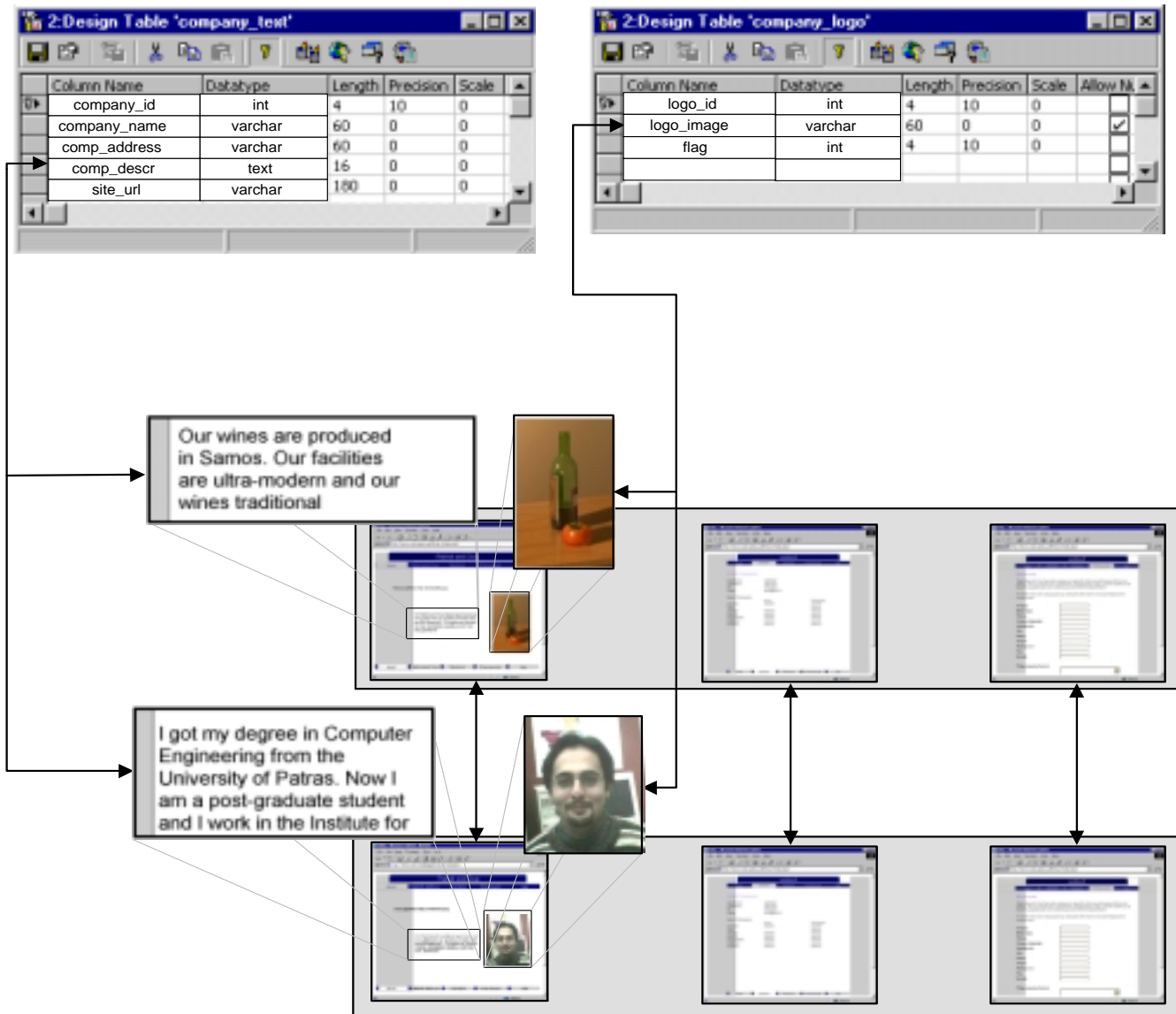
Fig. 7. The expansion architecture of the tool.

ule defines the placement of the site's components e.g. the position of the navigation bar.

A. Advantages for the user

There are many advantages and benefits for the user of the tool. These advantages are the key features that differentiate our tool from every other similar application ([?],[?],[?],[?],[?],[?]), either online or offline. The advantages are:

• Simplicity: The obvious advantage for the user of the tool is simplicity and user friendliness. The user does not need to have any specific knowledge of any kind of programming language or HTML to use it.

• Modifiability: With the administration wizard the user can modify all of the pages of his web site. There are many options in the administration wizard. The user is able to change all of his personal data, as well as all the data inserted.

• Variety of styles and Templates: except for the interface used for inserting, editing and deleting data (referred to previously as the Wizard and administration interfaces) a lot of attention has been given to the actual interface of the resulting web site ([?],[?]).

• Remote administration: Another very important advantage for the publisher is that they are able to administer their web site from everywhere in the world by using a PC with an Internet connection and no special software, other than a browser.

• Multilingual support: As we have mentioned earlier in the paper the tool supports multilingual web sites but at the moment it only supports Greek and English wizards.

• Prompt creation and publication of the site: There is no need of uploading the material created as well as practically no knowledge of the web server technology.

- The tool could also have an educational usage in the future too.

B. Advantages for the service provider

Except for the advantages that the tool provides to the potential publishers, it also provides some advantages to the Service provider that will decide to host the tool. The most important advantages for the Service provider are the following:

- Platform independence: Considering that our tool has access through the web, there is no need for the host to provide a special platform.
- Database independence: All database calls have been implemented with PHP-ODBC functions which frees the service provider from the need to use a specific database system.
- Automatic creation of web sites: This advantage refers to the fact that once the tool is installed, the service provider does not have to worry about maintenance or other costs.
- Tool Translation: The service provider can easily add new languages.

VI. FUTURE ENHANCEMENTS

The tool that we have described in this paper has many unique features but can also be considered as a prototype. The aim is to provide a much more extended variety of features. The tool is designed in such a way that its future feature expansion will require minimum redesign of the database and the web interface. Some of the future enhancements that we intend to offer through the tool are:

- A user oriented "wizard": At the moment, the wizard contains many "static" fields and options. It produces a fully operational web site but the user is still "manipulated" by the wizard to a certain path of web site creation. Potentially, and with very minor additions to the database, as well as a new, initial step in the wizard, the tool will be useful to almost every Internet user, who wants to create a web site of any kind. The idea is to let the user first define the wizard and its steps and then let them use that wizard to create their site.
- Used as a Learning tool: In future versions we intend to provide HTML and PHP learning capabilities to our tool. We consider the knowledge of certain web technologies to be very important to all potential web publishers. Since we automatically create the HTML and the required PHP3 files we would like to provide the user with a step by step preview of what he has created (in terms of HTML or PHP code).

VII. CONCLUSIONS

In this paper we have described an online web based tool that can create diverse and multilingual web sites requiring minimum "technical" knowledge by its users. The tool is under experimental use and evaluation at the University of Patras, Greece and will be given for public use in the near future. Considering the facts that there aren't many similar applications available and that none provide features like multilingual support, we believe that the tool will be very

useful to potential Internet content publishers. Of course there are still many enhancements that can be applied to the tool but even with its present features it can compete with other similar applications.

We must point out though, that we consider the knowledge of basic HTML and other web technologies to be a basic asset for anyone that wishes to publish content on the Web. This is why we intend to use the tool as a learning tool in the way that we described above. On the other hand since it is not possible to require special knowledge from every potential publisher, and since the Internet should remain open to all users, we strongly feel that applications such as the one described in this paper will be widely used in the future.

REFERENCES

[1] Internet Software Consortium, *Internet Domain Survey*, found at http://www.isc.org/ds/WWW-200001/report.html, January 2000.
[2] The Censorware Project, *Size of the web: A dynamic essay for a dynamic medium*, found at http://censorware.org/web_size
[3] Siteblazer.com, *Build a web site in 10 minutes*, found at http://www.siteblazer.com
[4] Fortunecity.com, *Community web sites*, found at http://www.fortunecity.com
[5] NetObjects, *NetObjects Fusion*, found at http://www.netobjects.com
[6] Trellix, *Free site building software* , found at http://www.trellix.com
[7] CompuServe, found at http://www.compuserve.com
[8] WebWizard, found at http://www.halcyon.com/artamedia/webwizard/
[9] Zope, found at http://yyy.zope.org
[10] Leon Atkinson, *Core PHP programming*, Prentice Hall, 1999
[11] M. Rees, *Exploiting the Full Web User Interface Spectrum*, Second Australian World-Wide Web Conference, 1996
[12] B. Schneiderman, *Designing the User Interface: Strategies for Effective HumanComputer Interaction*, Addison-Wesley, 1997
[13] User Interface Engineering, *When to Develop a Wizard*, found at http://world.std.com/ uieweb/wiz_art.htm
[14] Simson Garfinkel, Gene Spafford, *Web Security and Commerce*, O'Reilly, June 1997
[15] Gary Lebing, Maura Hogan, Wesley Gurney, *On-the-Fly: Web pages Minus the HTML Files*, WebNet96 - San Fransisco, CA, October 15-19 1996
[16] CommerceNet Research Center, found at http://www.commerce.net/research/stats/wwwpop.html

Christos Bouras obtained his Diploma and PhD from the Computer Science and Engineering Department of Patras University (Greece). He is currently an Assistant Professor in the above department, member of BoD in Computer Technology Institute (CTI) and head of Research Unit 6 at the CTI, Patras, Greece. He is a researcher in the area of Networks, Educational Technology, Telematics and New Services. He has extended professional experience in Design and Analysis of Networks, Protocols, Telematics and New Services, Distance Learning and Education, Electronic Publishing, High Speed Networks and Interactive Hypermedia and Multimedia He has published 80 papers in various well-known refereed conferences and journals. He is a co-author of five books in greek. He has been a PC member and referee in various journals and conferences. He has participated in R&D projects such as RACE, ESPRIT, TELEMATICS, EDUCATIONAL MULTIMEDIA, ISPO, EMPLOYMENT, ADAPT, STRIDE, EUROFORM, IST, GROWTH and others.

Agisilaos Konidaris obtained his Diploma and Master's Degree from the Computer Engineering and Informatics Department of the Patras University (Greece). He is currently an R&D Computer Engineer in Research Unit 6 at the Computer Technology Institute, Patras, Greece. He is also attending the second year of studies to obtain his PhD degree at the Computer Engineering and Informatics Department of the Patras University (Greece). His research interests include Telematics and New Services, Database design, implementation of Managed Bandwidth Services and Video on Demand algorithms. He has published 10 papers in various well-known refereed conferences and journals (Greek and International). He has participated in R&D projects such as ELECTRA and ODYSSEAS and is currently involved in the long term research project TODAY'S STORIES and the Internet over Satellite project.

Vassilis-Javed Khan is currently studying in the Computer Engineering and Informatics Department as an undergraduate in his final year. He has also worked as a Computer Engineer in Research Unit 6 at the Computer Technology Institute, Patras, Greece and as Software Developer in Knowledge S.A., Patras, Greece.

Christos Sintoris is currently studying in the Computer Engineering and Informatics Department as an undergraduate in his final year. He has also worked as a Computer Engineer in Research Unit 6 at the Computer Technology Institute, Patras, Greece and as Software Developer in Knowledge S.A., Patras, Greece.

Apostolos Apostolopoulos is currently studying in the Computer Engineering and Informatics Department as an undergraduate in his final year. He is currently involved in a long term business project at a medical company.