Visual Intranet Interfaces and Architecture of Unified Information Space in the Concept of Virtual University at ZSU

Vadim A. Ermolayev
Zaporozhye State University, 66, Zhukovskogo st., 330600, Zaporozhye, Ukraine
E-mail: eva@zsu.zaporizhzhe.ua

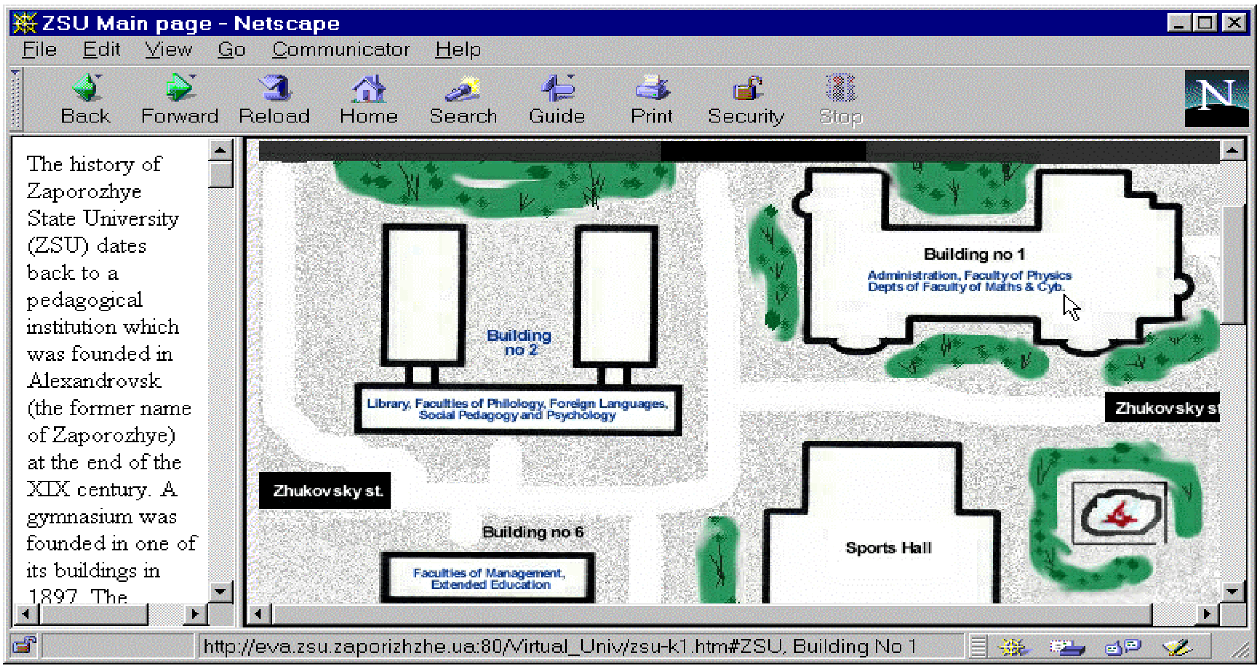
Abstract. Described is the approach to design and implementation the Unified Information Space — the basic engine of Virtual University. Discussed are the aspects of Visual Interface appearance, the approaches to corporate IS architecture design, the ways of enhancing IS flexibility and reusability. A refinement to the architecture and Information System design technology based upon Active Data Dictionary utilization is proposed. The study reported is primarily aimed to extend well formalized methods of Information System and Data Model design to the Enterprise Level Domain and thus enhance the qualities of flexibility, adaptability to the user demand within the phases of enterprise level computer media life cycle.

Introduction

The implementation of enterprise wide software systems and corporate networks today is probably one of the most complex problem for Information System (IS) and Information Technology (IT) specialists. From one side the Universe of Discourse we examine has natural tendency to be rapidly changed within its life cycle. From the other — there is always a severe dichotomy between the conception of system designer and the way the end user understands the functions and the roles of software he is using in his everyday work. The most common user perception of software system may be demonstrated by the following opinion: "I don't understand nothing in computers except the fact it should think for me…". As far as we probably have no chance to change the majority of users to software system designers this semantic gap will exist for a long time. Moreover, this entanglement of different aspects of information retrieval, as Delobel and Casey remarked in [1], "…also imposes stiff economic penalties in additional training costs, in programming delays and in deficiences in program reliability and flexibility." Therefore, the main task for IS and IT developers is to continue deluding their clients and to supply software with virtual intellectual properties. The most evident capabilities we might need for the purpose are: IS should offer the means for getting the resources and solving the problems the user might have an intention to receive or solve, IS should possess the interfaces intuitively clear to the user with navigation native to his environment, IS should transparently and adaptively absorb all the programs, tools and another components which are already "thinking for the user".

The review of even the most important milestones and contributions on this elegant way of cheating the user community (i.e. creating virtual software media) can take pages so we modestly omit it here and try to concentrate on our own research. Its aim is to develop the concept, the architecture, the methods and implementation technologies for the design of the University wide Unified Information Space — the engine of wired Virtual University.

In the present paper we are trying to figure out how the concept of virtual information retrieval is implemented in our Virtual University project by step-by-step diving into the levels of Unified University Information Space. Discussed are the aspects of Visual Interface appearance, the approaches to corporate IS architecture design, the ways of enhancing IS flexibility and reusability.


ZSU Main page - Netscape
File
Edit
View
Go
Communicator
Help
Back
Forward
Reload
Home
Search
Guide
Print
Security
Stop
Building no 1
Administration, Faculty of Physics
Depts of Faculty of Maths & Cyb.
Building no 2
Library, Faculties of Philology, Foreign Languages, Social Pedagogy and Psychology
Zhukovsky s
Zhukovsky st.
Building no 6
Faculties of Management, Extended Education
Sports Hall

The history of Zaporozhye State University (ZSU) dates back to a pedagogical institution which was founded in Alexandrovsk (the former name of Zaporozhye) at the end of the XIX century. A gymnasium was founded in one of its buildings in 1897. The

http://eva.zsu.zaporizhzhe.ua:80/Virtual_Univ/zsu-k1.htm#ZSU, Building No 1

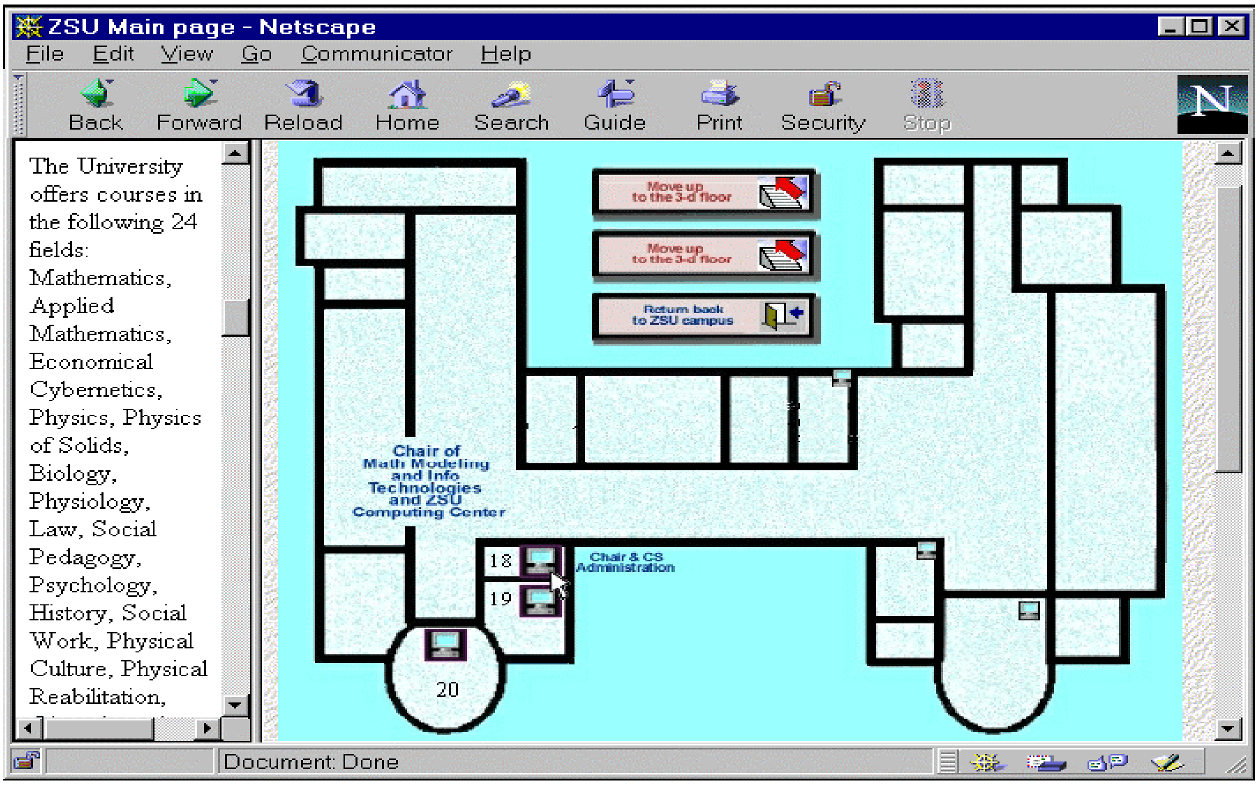
Fig 1 Starting point - University Campus

Concept of Virtual University and Visual Intranet Interfaces

The concept of Virtual University at Zaporozhye State University (ZSU) is formed on the basics of Unified University Information Space which has been described in [2]. The Unified Information Space is developed as a logical build on upon the University Corporate Network. As any other enterprise level computer environment the University Corporate Network is a complex heterogenious distributed system comprising various types of hardware and software components on different levels. The main task of Unified University Information Space is therefore to serve as a homogenizing virtual media providing unified access procedures and navigation tools for the user, desired properties of adaptability, flexibility and reusability for system designer and effective control means for system administrator.

Keeping in mind that the major effort should be applied to the upper side of the iceberg - the closest to the user, we are paying rather a strict attention to the means of user - computer interaction - Unified Visual Intranet Interface (UVII) being one of the solid stones lying in the basement of the Concept of Virtual University. It's obvious that today's computerized enterprise environment are so complicated that attempting to overview all the information despairing task. The way we are trying to solve it is to design visual interfaces which naturally imitate the structure and appearance of the real object - a University, a building, a department or an office, a server or a workstation, a group or a person. As far as this object set is definitely finite, we have an opportunity to design an interface object and link it with the object of the real world. This is the first step of virtualization on our way of narrowing existing semantic gap. To get the impression on how this step is implemented in UVII we offer to get a very short tour along one of the numerous branches of the hierarchical tree representing the University structure: the University - the building - the office - the person.

As shown on Fig 1, we start at the University Campus plan. Next step is to choose the building and click it to go inside moving down along the structural branch. Arriving at next organizational level we get into the sphere of another object of UVII - the building plan (see Fig 2). Obviously, its major properties are the plan chart, navigation controls, and links to the objects of the lower layer: the departments and the offices. In the example shown on the figure the navigation controls allow us to change the floor plans and to return back to the upper level - the Campus. By clicking the office - as shown on Fig 2 - we dive into the office UVII object. Its characteristics are quite different from the above interfaces.

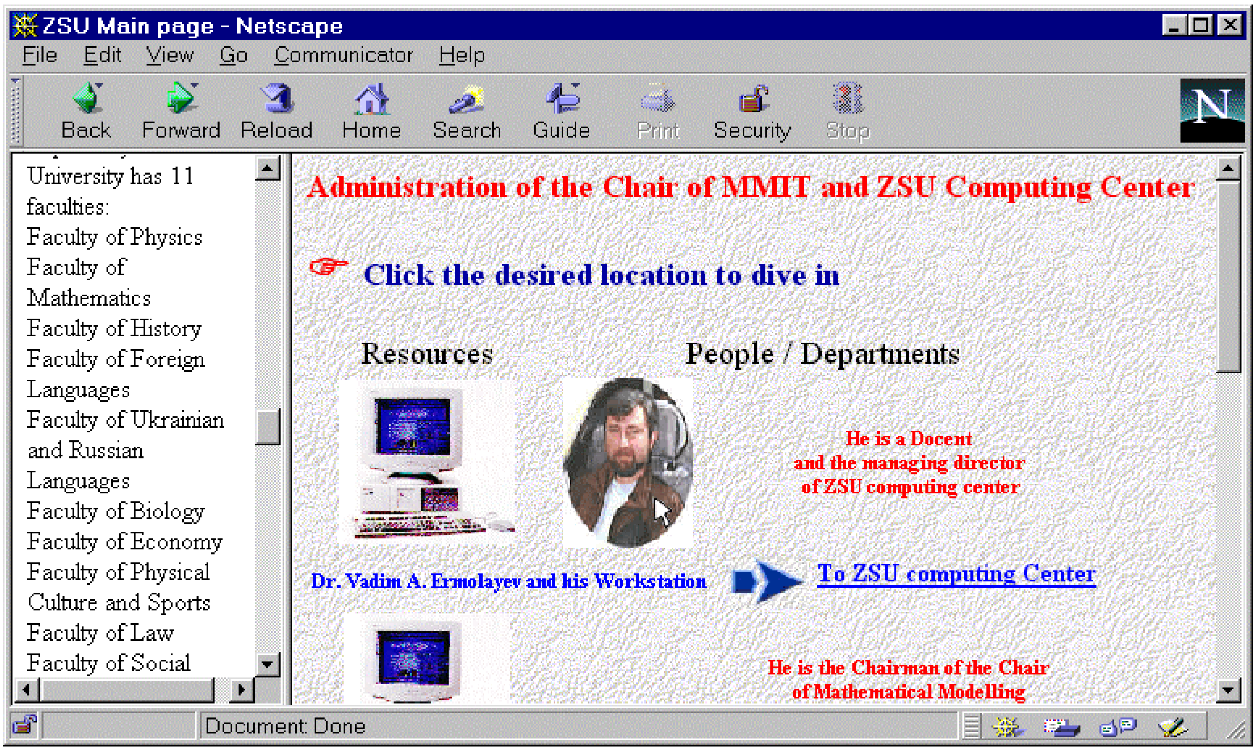


ZSU Main page - Netscape
File Edit View Go Communicator Help
Back Forward Reload Home Search Guide Print Security Stop
Move up to the 3-d floor
Move up to the 3-d floor
Return back to ZSU campus
Chair of Math Modeling and Info Technologies and ZSU Computing Center
18
19
20
Chair & CS Administration
Document: Done

The University offers courses in the following 24 fields: Mathematics, Applied Mathematics, Economical Cybernetics, Physics, Physics of Solids, Biology, Physiology, Law, Social Pedagogy, Psychology, History, Social Work, Physical Culture, Physical Reabilitation,

Fig 2. Navigation inside the building

Unified office object represents two groups of real world structure elements: the wired resources - servers and workstations and the people inhabiting the room with references to the departments. Thus, from the fragment of the office object shown on Fig 3, we can see that the office of the University Computing Center and the Chair of MMIT administration is presented by at least two workstations providing public network resources and at least one person (on the visible part of the office plan). Next object in our hierarchy is that very person and appropriate UVII object. The example of corresponding web page is



ZSU Main page - Netscape
File Edit View Go Communicator Help
Back Forward Reload Home Search Guide Print Security Stop
Document: Done

University has 11 faculties:
Faculty of Physics
Faculty of Mathematics
Faculty of History
Faculty of Foreign Languages
Faculty of Ukrainian and Russian Languages
Faculty of Biology
Faculty of Economy
Faculty of Physical Culture and Sports
Faculty of Law
Faculty of Social

Administration of the Chair of MMIT and ZSU Computing Center

☞ Click the desired location to dive in

Resources People / Departments

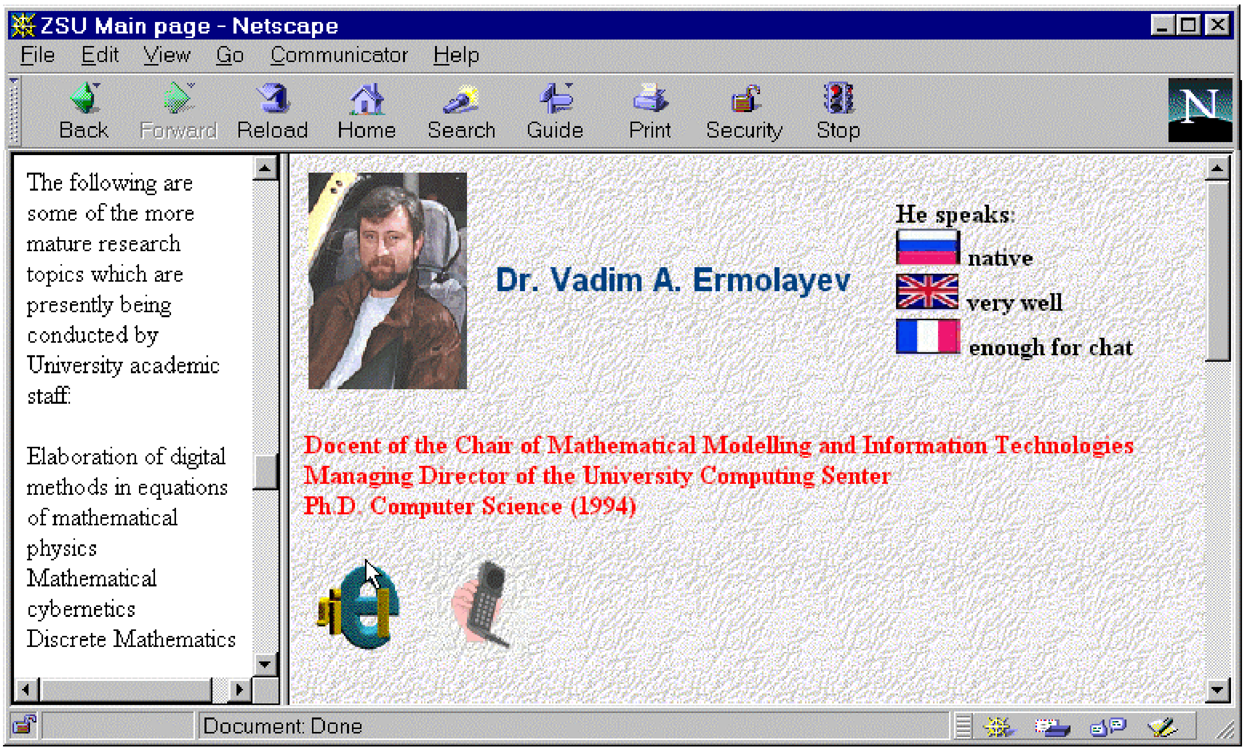
He is a Docent and the managing director of ZSU computing center

Dr. Vadim A. Ermolayev and his Workstation

To ZSU computing Center

He is the Chairman of the Chair of Mathematical Modelling

Fig 3. Resources and people inside a department/office



The following are some of the more mature research topics which are presently being conducted by University academic staff.

Elaboration of digital methods in equations of mathematical physics
Mathematical cybernetics
Discrete Mathematics

Dr. Vadim A. Ermolayev

He speaks:
🟥 native
very well
enough for chat

Docent of the Chair of Mathematical Modelling and Information Technologies
Managing Director of the University Computing Senter
Ph.D. Computer Science (1994)

Document: Done

Fig 4. An example of personal entry

given on Fig 4. The controls at this layer provide the means of communication (e-mail, internal phone call) and links to the items related to the person forming the functional model of the real world object. Analogously computer or wired resource object's task is to present the functional model of the network node - workstation or server - linking the user to the tasks the node is able to fulfil and/or resources it provides.

To conclude the description of the first level of Unified Information Space - the basics of our Virtual University concept - the following should be mentioned:

- ✓ the main idea of the usage of visual interface component as a part of Unified Information Space is to operate with a finite set of standard interface components which can be easily generated automatically
- ✓ the idea of such a visual interface is not new - Unicenter TNG by Computer Associates with its Real World Interface can be mentioned as a good example
- ✓ the interface part of Unified Information Space from our point of view may be considered as not used before in the implementation of a unified information resource media at universities and moreover it's brand original for Ukraine

Some interesting and original points in Unified Information Space will be described in the following sections. As far as the main idea of the first level is to generate HTML code for visual interface automatically, the second step on this way is quite natural and consists in taking data for HTML code generation from Unified Information Space database.

The Architecture of Unified Information Space

To describe the organization and interaction of Unified Information Space components - i.e. its architecture - we'll follow the method of determining the levels and the interfaces between the levels (refer for instance to [3]). The first level of Unified Information Space architecture is therefore the User or the External Level and it was partially described in the previous section. The interfaces of this level are the request the user passes to Unified Information Space and the result Unified Information Space returns in the form of HTML code to the user's browser. Next level is the one of Corporate Information System. The task of this level is to transform the user's request and to generate the virtual query to Corporate IS. After the query is processed and the result of the query is returned back the turn of one more level component comes. Query Result Processor generates appropriate HTML code from the results of the

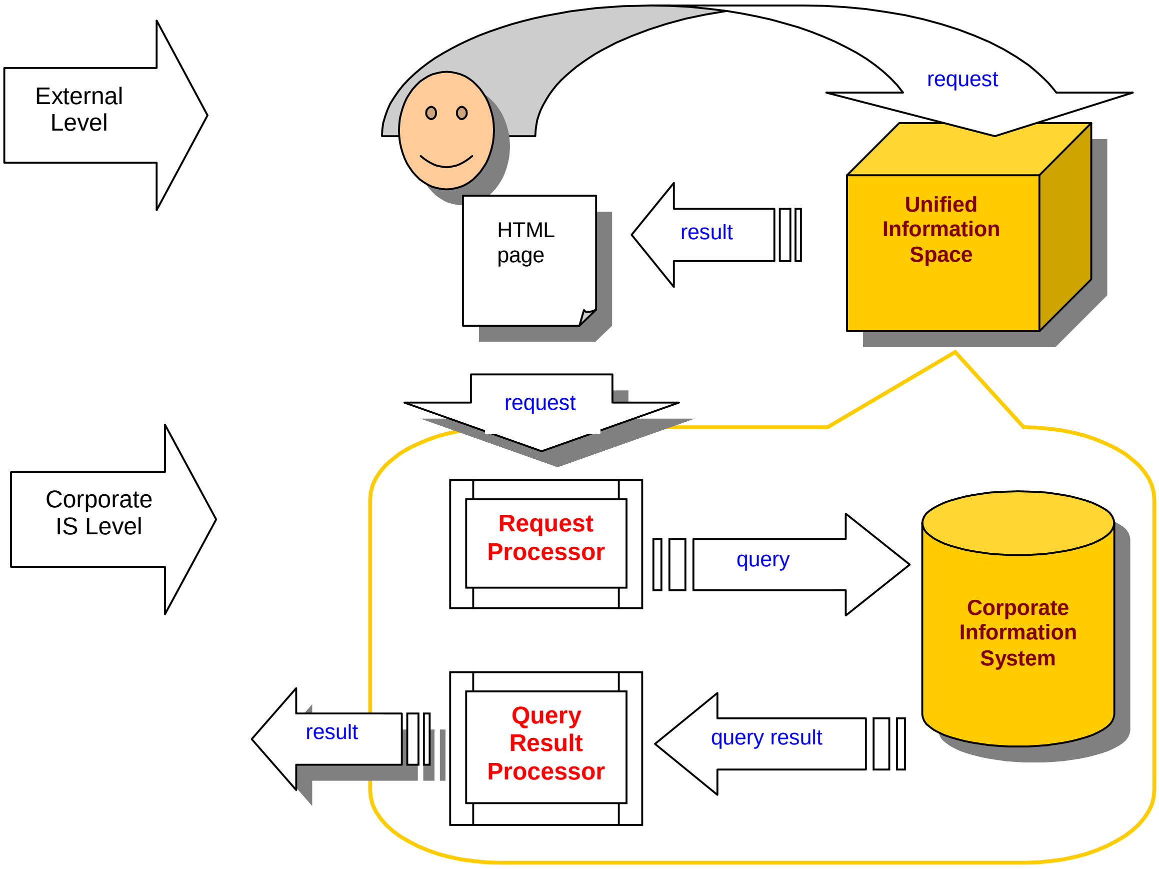


Fig 5. External and Corporate IS levels of Unified Information Space Architecture

query and returns it to the upper (User) level of the architecture. The architecture of External and Corporate IS Levels is schematically shown on Fig 5.

After we've converted user's request to query we immediately gain the opportunity to describe UVII in the terms of data model, data objects and relationships and process the requests as queries by means of DBMS. This transition from HTML and request to query gives us pretty much for formalizing the interfaces, data retrieval and maintenance routines as far as we receive Data Model and DBMS as quite powerful tools of automation. For the implementation of this level of architecture we need just data model for the description of interface elements, a DBMS, user request processor and query result processor.

Unfortunately more problems arise when we start analyzing virtual query processing. Corporate IS is actually quite a complex aggregation of the set of Local Information Systems and Functional Servers distributed over the Enterprise Network and handling local information, applications and resources. The data and functional characteristics of the applications and the resources often have semantic overlaps. One of the known methods of resolving the problem of this semantic intersection is the use of Federative Data Models and DBMS. The methods of federalizing data are quite well developed for the moment and are widely discussed. In Unified Information Space we are planning to use the method and the model developed by G. Saake and his colleagues (se for instance [4] as well as for extra bibliography) with the enhancement by our own Active Data Dictionary approach ([5] and in the following section). Thus the Federation Level of Unified Information Space architecture may be schematically presented by the picture given on Fig 6. This level comprises two layers with corresponding interfaces - the Federation Layer with functions very much like the External Level of Federative Data Model [] and the set of local information resource servers, interacting with the Federative Layer by means of query based interface. As outlined on Fig 6, main function of the Federation Layer are the control of transparent aggregation of data models on the Local Level, incoming query translation distribution and redirection as well as processing local query results, which are in general case coming from more than one Local Information System and/or Local resource Server. Thus, the Federation Level of Unified Information Space

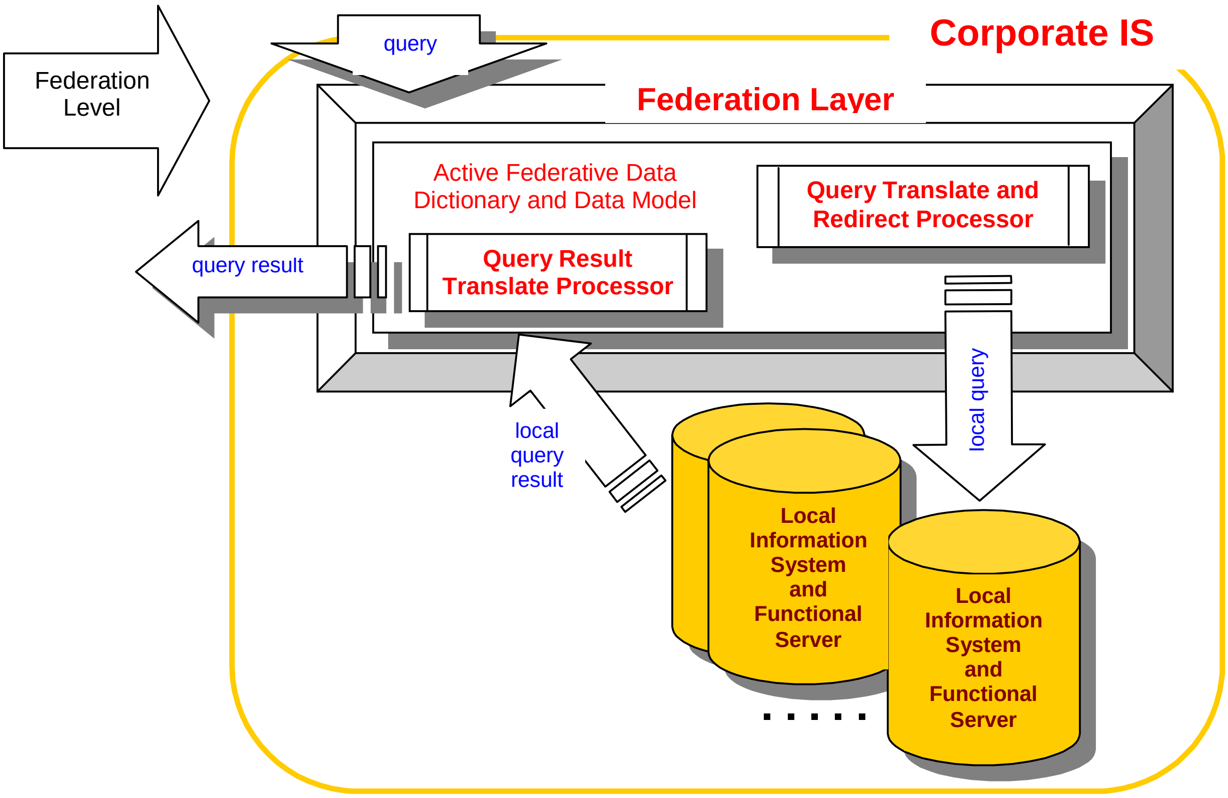

Corporate IS
Federation Level
query
Federation Layer
Active Federative Data Dictionary and Data Model
Query Translate and Redirect Processor
query result
Query Result Translate Processor
local query
local query result
Local Information System and Functional Server
Local Information System and Functional Server
. . . . .

Fig 6. Federation level of Unified Information Space Architecture

architecture interprets the incoming virtual query into the set of local queries and appropriately redirects them to the components of the Local Level of the architecture.

The main architectural advantage we have at the Local Level is the method of Local IS control. We use Active Data Dictionary as IS control shell and Data Model as IS control engine. Local Level comprises therefore three layers: Program Code Layer, Data Model Layer and Data Layer (refer to Fig 7) Data Dictionary with its active behavior function controlled by Data Model serves as an intelligent flexible interface between the upper levels and local data and/or resource. This changes in Local IS architecture definitely affect the technology of IS Design. This modified design procedure is based upon certain properties of our Active Data Dictionary which are discussed in the following section.

Active Data Dictionary

Data Dictionaries have always been considered as tools for data description maintenance. Our approach is to supply a data dictionary with an active behavior function. Active data dictionary is a program media for data model to serve as information system control engine. Active data dictionary having the functions outlined can serve as a tool for an ideal data model driven information system design procedure. It was already mentioned that Database Applications are subjected to face numerous changes while passing through the stages of their life cycle. Unfortunately traditional methods and tools for design and implementation of Information Systems do not possess the means flexible enough to face this changes without modifying the program code. Attempts to solve the problem are presented mainly by Object Oriented Approach to Information System Design. The systems implemented by means of Object Databases embed appropriate data manipulation code into Database as Methods. This interpretative technique is quite flexible and the code is well structured due to the principles of Object Orientation. However, the question is: *Is it reasonable and possible to use compilative technique and at the same time to overcome the flexibility restrictions - some definite part of program code will anyway be dependent from the Universe of Discourse.* The approach we develop is based on the idea of creation of Database Application Shell which, been once compiled, may serve as a Data Model controlled media or interface for data access and manipulation as well as for Data Model design.

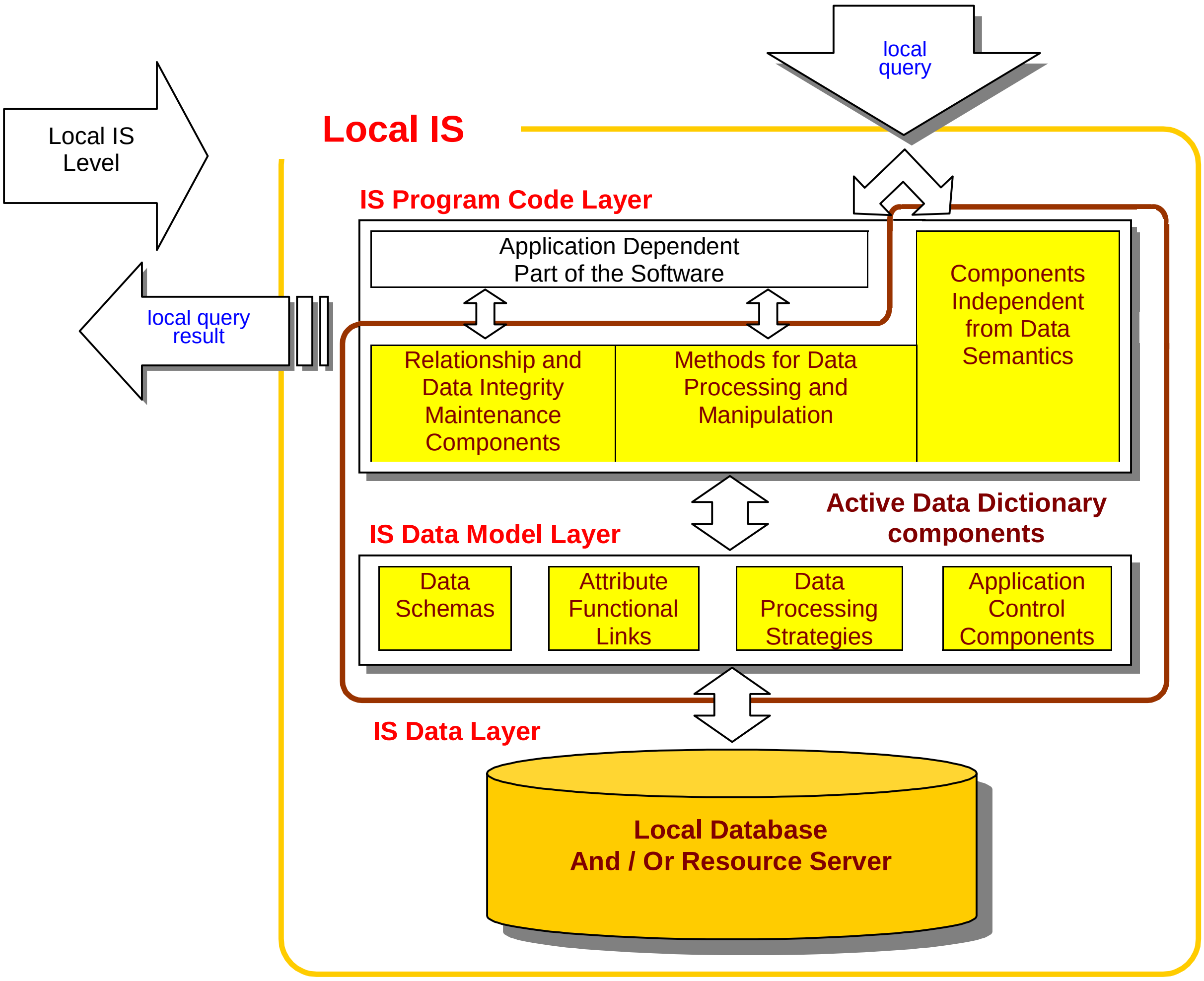

local query
Local IS Level
Local IS
IS Program Code Layer
Application Dependent Part of the Software
Components Independent from Data Semantics
local query result
Relationship and Data Integrity Maintenance Components
Methods for Data Processing and Manipulation
Active Data Dictionary components
IS Data Model Layer
Data Schemas
Attribute Functional Links
Data Processing Strategies
Application Control Components
IS Data Layer
Local Database And / Or Resource Server

Fig 7. Local IS level of Unified Information Space Architecture

For the moment we have the prototype of Active Data Dictionary (ADD) - the tool for solving the problem.

Strategic goal of ADD utilization is to obtain a database application shell possessing the following properties:

- ✓ Flexible enough to be invariant to specific features of the application domain and, therefore, 100 per cent reusable in various database applications
- ✓ Containing data model driven means which, been embedded into the application, handle data access, manipulation as well as data integrity
- ✓ Possessing the tools for data model description and design.

The structure of the Local IS given on Fig 7 presents the essence of the method used: to define the structure of an abstract Information System and to figure out the parts being application dependent and those being invariant to application domain. We assume that the principal structure of an abstract Information System comprises three major parts: *data, data model* and *program code* with data model been the active component providing interface and control means both for data and program code.

Further analyses of this structure outlines:

- ✓ The following parts can be extracted from information system program code: application dependent portion implementing the algorithms specific to the application domain and code objects invariant to specific character of application and corresponding data model
- ✓ Data model is an essential part of this structure and its major function is to contain data descriptions as well as data access and manipulation methods descriptions. The structure of data model is invariant to application domain and may be described by means of data model itself

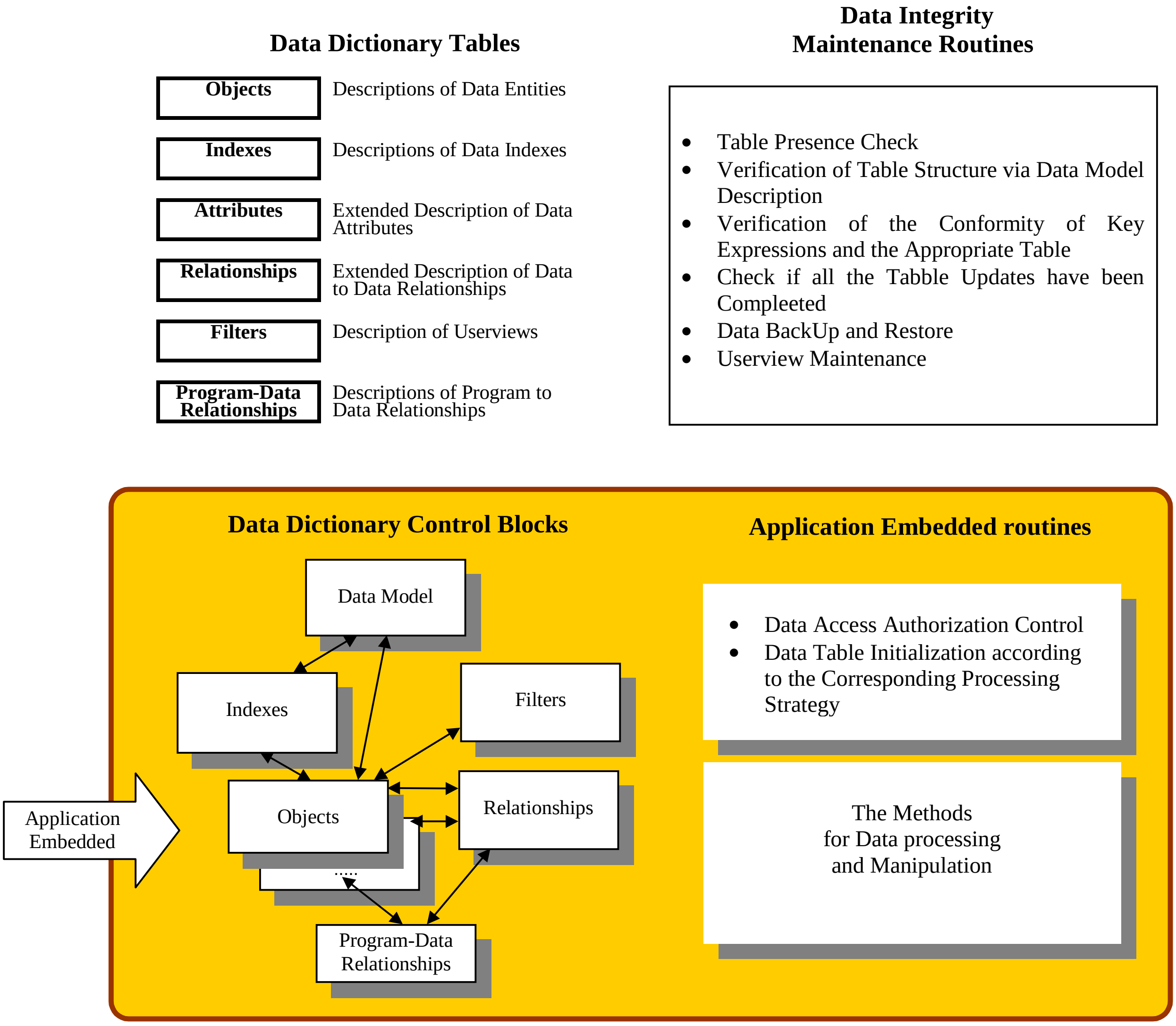

Data Dictionary Tables
Objects
Descriptions of Data Entities
Indexes
Descriptions of Data Indexes
Attributes
Extended Description of Data Attributes
Relationships
Extended Description of Data to Data Relationships
Filters
Description of Userviews
Program-Data Relationships
Descriptions of Program to Data Relationships
Data Integrity Maintenance Routines
Table Presence Check
Verification of Table Structure via Data Model Description
Verification of the Conformity of Key Expressions and the Appropriate Table
Check if all the Tabble Updates have been Compleeted
Data BackUp and Restore
Userview Maintenance
Data Dictionary Control Blocks
Application Embedded routines
Data Model
Indexes
Filters
Objects
Relationships
.....
Program-Data Relationships
Application Embedded
Data Access Authorization Control
Data Table Initialization according to the Corresponding Processing Strategy
The Methods for Data processing and Manipulation

Fig 8. Active Data Dictionary Components

✓ Data model (as an engine) together with application independent part of program code combined together may serve as a compilable program shell controlled by data model and is called Active Data Dictionary (refer to Fig. 7).

ADD is theoretically based upon some extension (summarized in [5]) of basic notions taken from Relational Data Model theory and is quite a complex object comprising the major components shown on Fig 8 in its staff.

Thus, Using ADD with Data Model as an active control component brings up the benefits of improved Data Model Driven IS design and implementation methodology both to Local and Federation Levels of Unified Information Space Architecture. Evidently, the components designed under this technology are much more flexible and reusable due to the quality of ADD Application Domain independence.

Conclusions

In this paper we have briefly described our approach to design and implementation the Unified Information Space which is actually the information model of wired Virtual University. The study reported here is primarily aimed to extend well formalized methods of Information System and Data Model design to the Enterprise Level Domain and thus enhance the qualities of flexibility, adaptability to the user demand and inspire users to feel that their computers are thinking for them.

Acknowledgments

The author is indebted to prof. G. Saake and his collaborators at Otto-von-Guericke University, Magdeburg for their ideas and models of Database Federation. Special thanks should be addressed to Computer Associates Inc. for their conceptual idea of Real World Interface we use in our Unified Visual Intranet Interface.

References

1. C. Delobel, R. G. Casey, Decomposition of a Data Base and the Theory of Boolean Switching Functions, IBM J. Res. Develop., Vol 17, No 5 (Sep. 1973), pp. 374-386

2. V. A. Tolok, S. U. Borue, V. A. Ermolayev, A. I. Kubushkaites, Development of the Consept and Implementation of the First Line of the Integrated Network at ZSU, Research Work Intermediate Report, State Reg. No 0197y012776, Ministry of Education of Ukraine, Zaporozhye State Univ., Zaporozhye, 1997, 28 p. (in russian)

3. G. J. Myers, Advances in Computer Architecture, John Wiley & Sons, 1982, Vol. 1, 384 p.

4. I. Schmitt, G. Saake, Merging Inheritance Hierarches for Schema Integration Based on Concept Lattices, Fakultät für Informatik, Otto-von-Guericke-Universität, Magdeburg, Preprint Nr. 2, 1997, 29 p.

5. V. A. Ermolayev, Object Oriented Dynamic Data Modelling and Active Data Dictionaries - Some Crosspoints . - to appeare in "Journal of Metrology and Certification" Vol 1, No 1, (Jul.-Dec. 1997)

6. I. Schmitt, Flexible Integration and Derivation of Heterogenious Schemata in Federated Database Systems, Fakultät für Informatik, Otto-von-Guericke-Universität, Magdeburg, Preprint Nr. 10, 1995, 32 p.