Research Journal of Applied Sciences, Engineering and Technology 7(16): 3273-3279, 2014
DOI:10.19026/rjaset.7.671
ISSN: 2040-7459; e-ISSN: 2040-7467
© 2014 Maxwell Scientific Publication Corp.
Submitted: August 05, 2013 Accepted: September 05, 2013 Published: April 25, 2014

Research Article

Scalable user Interface

[1]Mohammed Al-Maitah, [2]Takialddin A. Al Smadi and [3]Hani Qasem Rashrash Al-Zoubi
[1]King Saud University, Saudi Arabia
[2]Jerash Private University, Jordan
[3]Mutah University, Jordan

Abstract: The study is dedicated to the specific type of user interface based on scaling visual objects images in computer systems. Before considering the issue itself, there had been characterized WIMP paradigm, used for building the majority of typical user interface, analyzed disadvantages and evaluated the potential complexity on its base, with causes the necessity in the development of new approaches to interface building. There had also been presented the basic principles for building the scalable user interface as well as enumerated the main means for the development of the alike systems with the examples of their applications. The last section describes practical aspects of realization of the scalable user interface, its disadvantages and methods for their solution. The research enabled to make a conclusion on prospects in application of scalable user interfaces.

Keywords: Icons, interface, menus and pointer devices, scalable user interface, windows

INTRODUCTION

The majority of modern user interfaces is built on the base of WIMP paradigm, with literary means «Windows, Icons, Menus and Pointer devices». This term was coined by It was coined by Merzouga Wilberts in 1980 to describe the interface of working station (Charlotte, year). The components of this paradigm are even older. Screen windows for example and the application of the pointing device (mouse prototype) had been demonstrated by Douglas Carl Engelbart in 1968 (Jeremy, 2005). The first range of icons had been developed in 70-th and the simplest variants of menu had been applied in the operation system for the working station Xerox Star in 1973 (Hiltzick, 1999). The metaphor of the working table in with each element of computer system was given some graphical image as also pretty old and has been used for building user interfaces for more than 30 years.

This paradigm is reduced to only three principles (Beaudouin-Lafon, 2000):

- Objects of computer system are presented as graphic objects (icons).
- These graphic objects may be free removed both within working table as well as within screen windows and between them.
- Commands are initiate by menu or keyboard panel, sometimes using dialog windows for induction of additional data, which is necessary for execution.

This gave it at the proper time the colossal advantages before other systems. Foremost, orientation on Rectangular Marquee Tools (both home screen and display window appear such is the case) made her most suitable for the thin screen. Secondly, correctly matched graphical representation made appropriation and content of different objects of the system intuitive. Thirdly, all the interactions in alike systems had analogs in the real world and therefore such system was intuitive for the first-time user.

At the same time father development WIMP interface had shown its numerous disadvantages. And peculiarly the limitation of graphical representation among them, because only limited number of objects can be placed on the home screen.

In order to avoid or take away these natural limitations, there was discovered whole lot of different control elements, including virtual home screens (Henderson and Card, 1986), which allow to expand user space essentially. There were invented attached context menus, with the help of specific software tools (for example different file explorers) the opportunities for visualization were expanding. At one point these developments affected user`s interaction directly: for example tags and hard links, which refer to the other objects don't have real analogue as well as hypertext, multiple representation of one and the same object, different interface panels (for example inserts or overlay display windows) and combinations of hot keys.

This lead computer industry to the undetected crisis of interface solutions, which made the complete use of computer systems available only for specialists.

Corresponding Author: Mohammed Al-Maitah, King Saud University, Saudi Arabia

EXPERIMENTAL

Complexity of WIMP-interface: In his work, dedicated to interaction between user and computer interface (Beaudouin-Lafon, 2000) presented four main principles of Schneiderman, used for user interface analysis. They allow to estimate the complexity of interface for user. Let us present these principles briefly, as also WIMP compliance to them:

- Continual presentation of users objects
- Direct interaction with users objects
- Prompt, simple and cancellable operations, directly affecting the condition of users object
- Consistent approach to the system examination

Let's accomplish the complexity of WIMP-interface in the context of these principles.

Modern program packages basically operate with users objects, which are physically more than screen. It applies to text documents and images. In order to allow the user to cooperate with them adequately it is necessary to arrange for scaling duplication, movement, scrolling and they in their turn require control elements, which fill the place on the screen.

While working with objects dialog windows or additional panels often appear and they also fill certain place on the screen. The more complicated the processing is, the more the screen is overloaded with control elements and the less space remains properly for the users object and it complicates the entirety of reflection of the object and provides the user only with his share at the best case. Usually it is compensated by different solutions, which allow to hide or group together the control elements, this, however neutralizes the intuitivity of work and requires the spatial training for the user.

The application of specific control elements also neutralize the principle of direct interaction. Usually the operation on users object is realized with the help of its highlighting and command selection from the context menu or on the keyboard.

But in case of complex processing dialog windows with the numerous options which are often attached (so the second window can be invoked from the dialog window and than the third) or serial (so called wizards) take part in the operation. Therefore, the interaction becomes indirect and the rollback to the times of command-line interface with the necessity of command input with exact data takes place.

The interface overloading in its turn causes the frustration of the third principle-the simplicity and canceling of operations directly on the object. Attached and serial dialog windows create the complex chains of actions, which require either mechanical memorization or the necessity of creation backup copies of the users object-as during the accomplishment of complex sequence of actions (especially little-known), the mistakes, which annulment would be almost impossible, are inevitable. More than that, it questions the possibility for independent learning of the system. It becomes directed on the common actions, which are realized by the previously formulated scheme. It causes the fact, that the system becomes closed and more than a half of its undoubtedly useful functions remain unclaimed. Besides, it is easy-to-see on the modern word processors.

The above shows that the modern WIMP interfaces aren't simple and intuitive anymore, conversely their development becomes more complex. In many spheres, namely in computer graphic, signal processing and document editing, typical interactions are deficient and in order to guarantee the proper operability it is essential to develop special control elements like nested menu, concealable widgets and in some specialized program packages the boards of real devices (namely faders, equalizers and so on) are quite simulated, as current typical elements of control are insufficiently visual and intuitive.

All that leads us to the necessity of development of new interface types, which have big amount of direct interactions or even provide the user with bigger space.

The most simple and efficient variant is the exiting from the commitment to orthographic plain areas and providing the home screen with depth, passing on the three-dimensional interfaces in such a way.

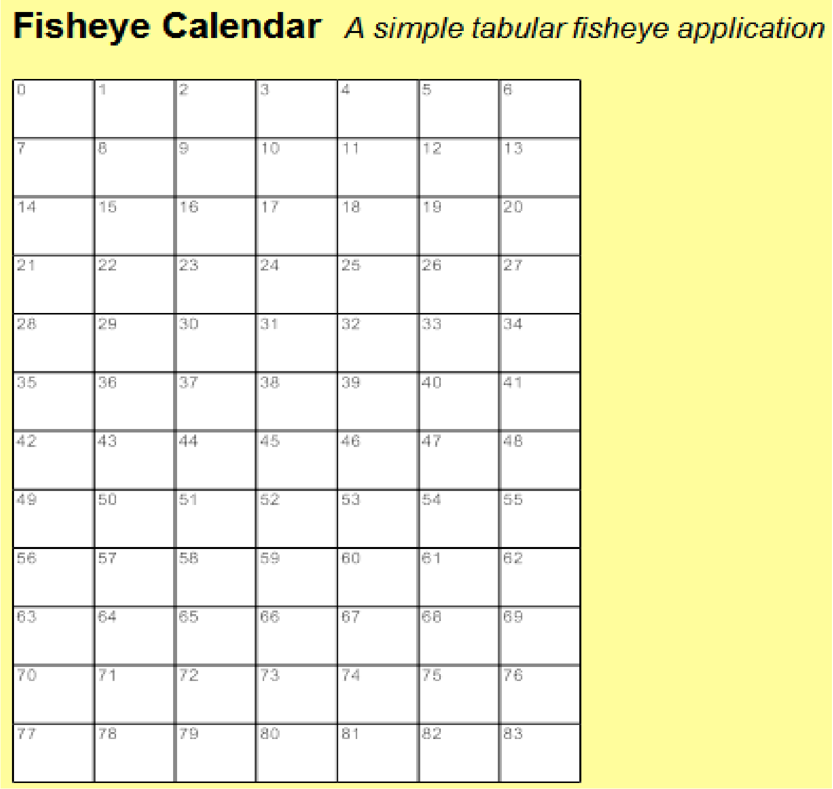
The most promising from this point of view are Zoomable User Interfaces, ZUI:

Main principles of ZUI: Considering the main principles of scale user interfaces, one must differentiate them from the functions of scaling, which are present at any program package. Only the system, which provides the user with three-dimensional organization of user objects and three-dimensional navigation between them without additional elements, can be deemed as fully-featured ZUI (Bederson, 2011).

The shining examples of such supplements are demonstration sentences of the package for development of ZUI Piccolo (Bederson, 2004) it will be fully considered below.

The simplest of the these examples, which realizes the scalable calendar is represented on the Fig. 1. But he is quite enough to illustrate the main principle of ZUI-responsiveness. Each box in this calendar represents easily scaling object. Whereby, if it is chosen, it will be extended and the other boxes will descend proportionally. Thus, with the permanent size of working area, there is no necessity in gatefolds or any other special control elements, which guarantee navigation. Mouse click is quite enough. In the same way we can get the endless home screen, removing a restriction for the monitor enlargement and using scaling.

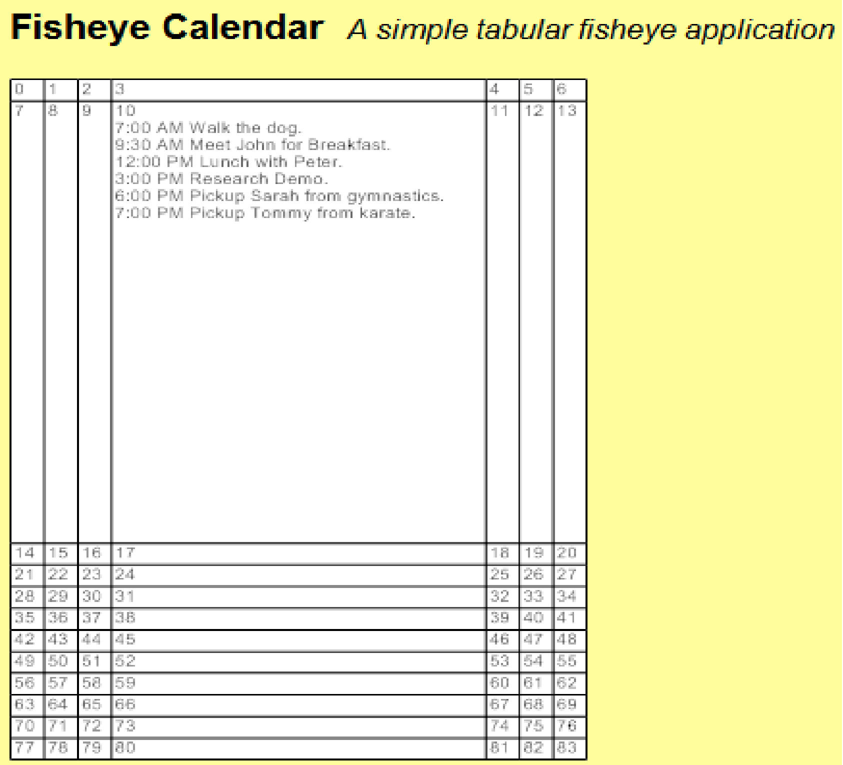
It is even more better to combine the ZUI with the tape element known by web-applications and to create the presentation system.



Fisheye Calendar *A simple tabular fisheye application*

0	1	2	3	4	5	6
7	8	9	10	11	12	13
14	15	16	17	18	19	20
21	22	23	24	25	26	27
28	29	30	31	32	33	34
35	36	37	38	39	40	41
42	43	44	45	46	47	48
49	50	51	52	53	54	55
56	57	58	59	60	61	62
63	64	65	66	67	68	69
70	71	72	73	74	75	76
77	78	79	80	81	82	83

(a)



Fisheye Calendar *A simple tabular fisheye application*

0	1	2	3	4	5	6
7	8	9	10 7:00 AM Walk the dog. 9:30 AM Meet John for Breakfast. 12:00 PM Lunch with Peter. 3:00 PM Research Demo. 6:00 PM Pickup Sarah from gymnastics. 7:00 PM Pickup Tommy from karate.	11	12	13
14	15	16	17	18	19	20
21	22	23	24	25	26	27
28	29	30	31	32	33	34
35	36	37	38	39	40	41
42	43	44	45	46	47	48
49	50	51	52	53	54	55
56	57	58	59	60	61	62
63	64	65	66	67	68	69
70	71	72	73	74	75	76
77	78	79	80	81	82	83

(b)

Fig. 1: Scalable calendar

None the worse is to combine ZUI with tape element, known by web-applications and to create the presentation system. Bederson (2004) created precisely this system as part of his scientific work, demonstrating one more principle of scale user interface-visual exhaustibility. Instead of dealing with static and less informative icons, the user deal only with proper miniatures of the objects.

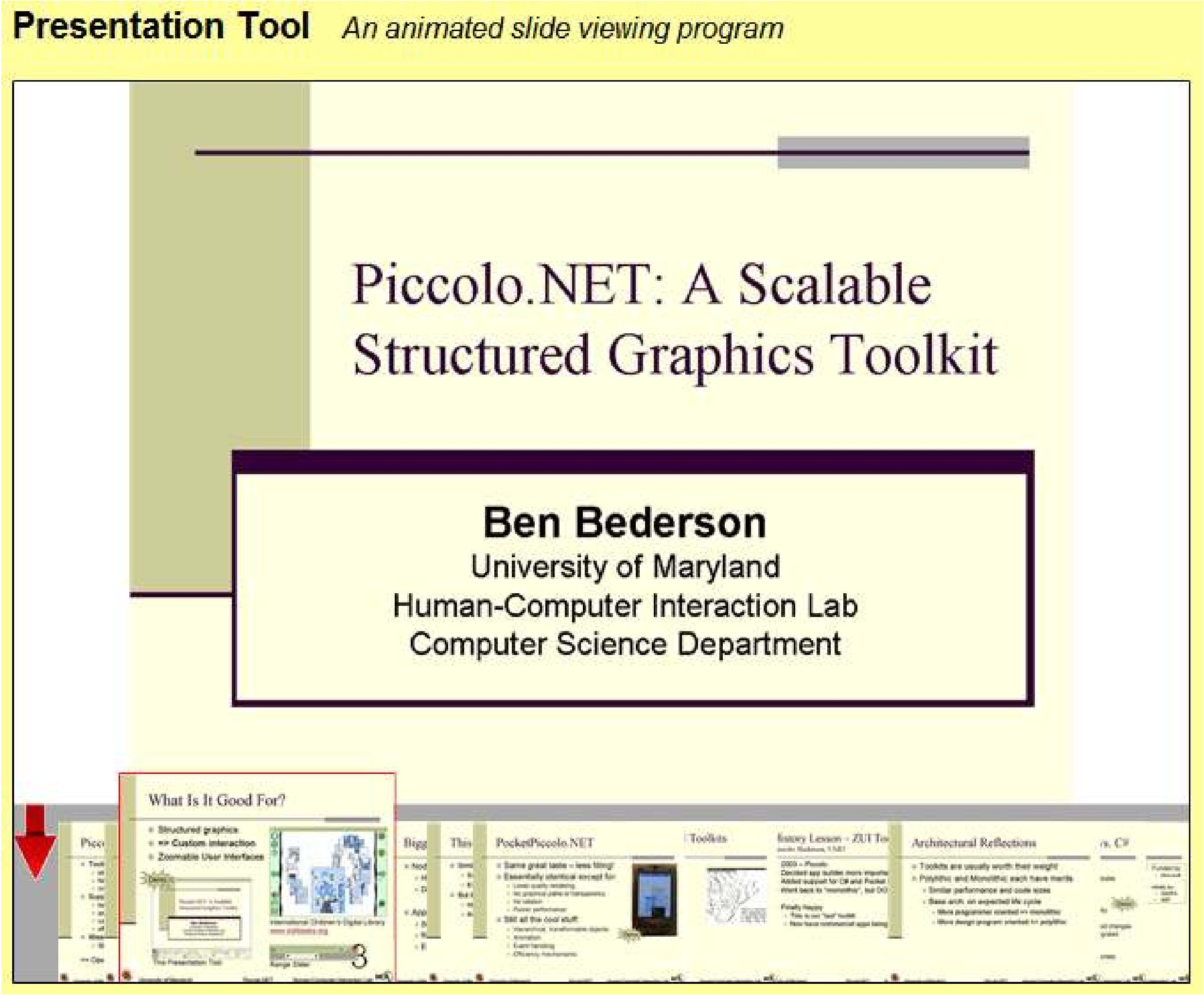
Besides, Steven Jobs had chosen approximately the same model for MacOSX, where the control element «doc» exists for many years (Modine, 2008).

Therewith, there is a psychological aspect. When seeing not just abstract graphical objects but miniatures (they are called «overviews» in ZUI) of targeted objects, it is much easier for to get accustomed to the system.

It is worth saying separately about control elements. Using the methods of ZUI, we can seriously reduce the complexity of work and refuse the complex system of attached menus and dialog windows. For this it is necessary just to make keyboards scalable, realizing them like docs, or to combine the concept of container with scaling and thereby attached commands will appear with reduction/increase of the keys. This method is called semantic scaling and increasingly finds application in modern software systems.

Software tools for ZUI development: Its mistakenly to think that the development of ZUI is possible only for mobile units, although namely it is more often used in PDA, smartphones and electronic organizer. There exists the range of special tools, which allow to include ZUI elements into the usual application or even to work out the sterling ZUI.

The most large package for the development of scaling interfaces is Piccolo2D, which was mentioned above. It was developed under the guidance of one of the best specialists in ZUI branch Ben Bedderson and currently implements practically all what is necessary for the ZUI construction.



Presentation Tool An animated slide viewing program

Piccolo.NET: A Scalable Structured Graphics Toolkit

Ben Bederson
University of Maryland
Human-Computer Interaction Lab
Computer Science Department

Fig. 2: System of presentation of B. Bedderson

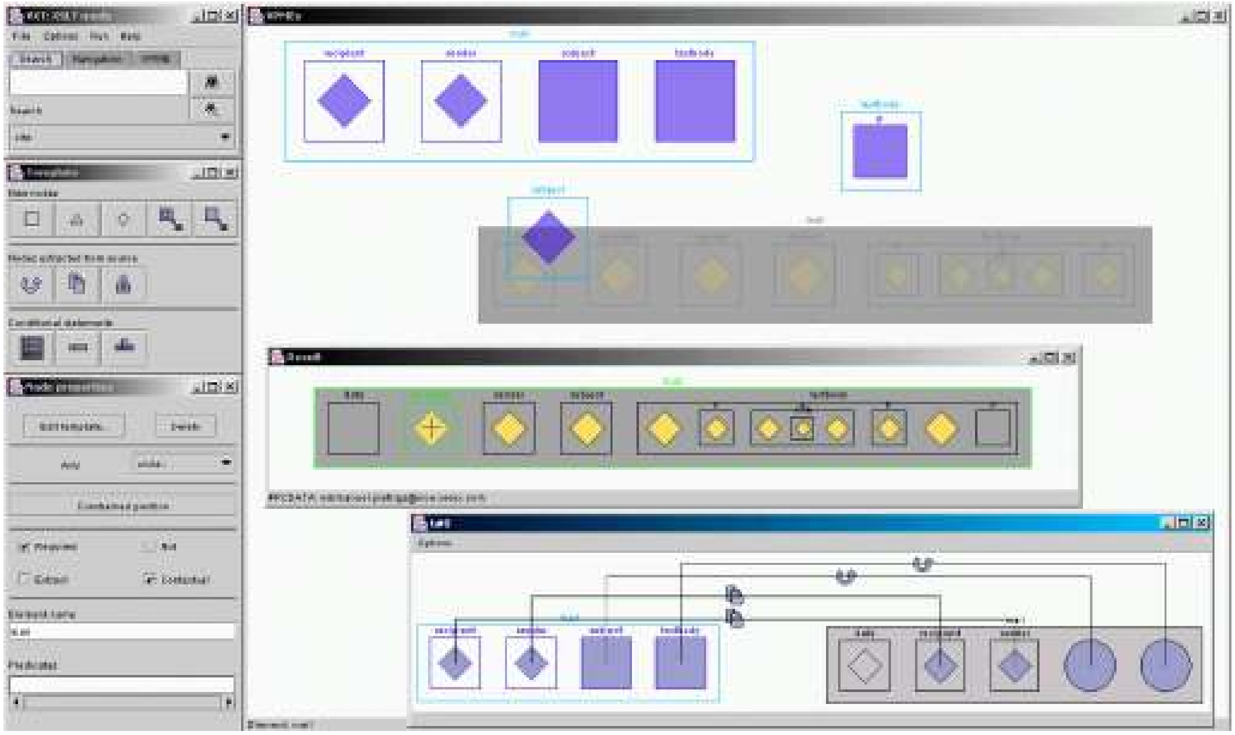


Fig. 3: Package VXT

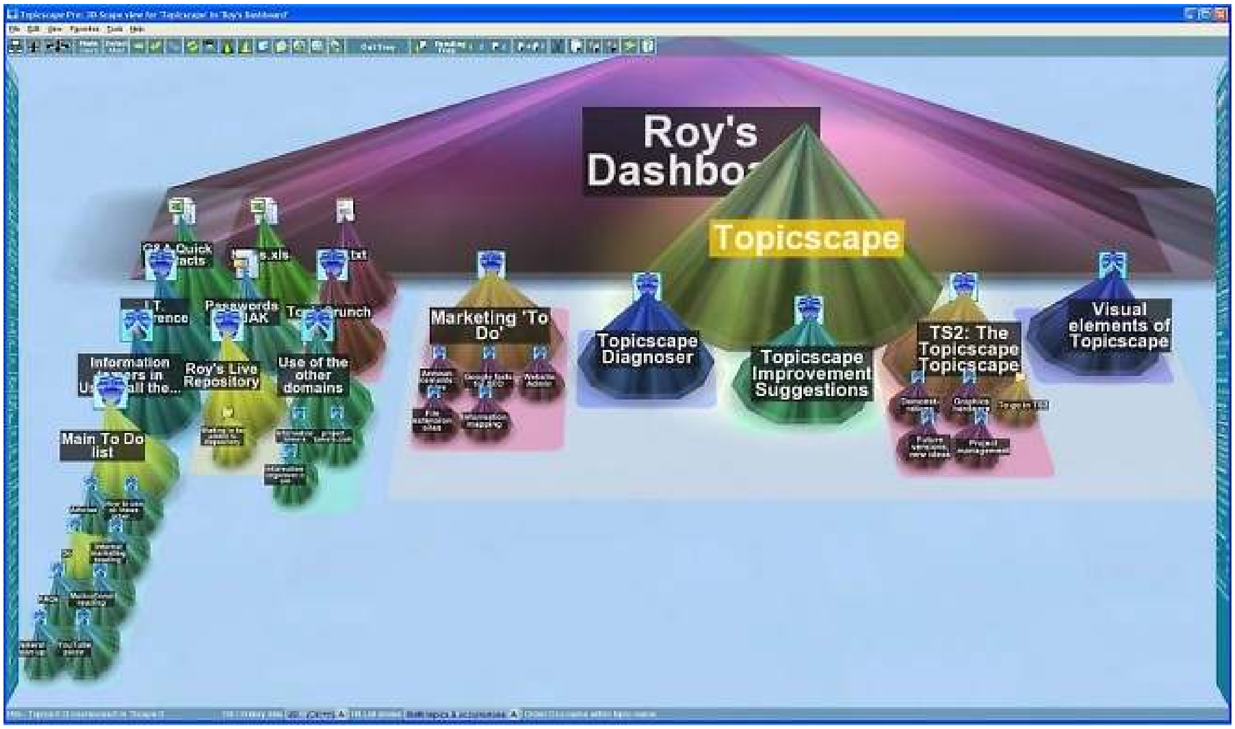


Fig. 4: Sphere of TopicScape (year) (office screenshot)

The package exists in three redactions-.NET, Java and Pocket Net (Bederson, 2004).

Its advantages should include the deep approbation of the example, which demonstrates all aspects and application variants of ZUI. The main defect is extremely rare updating, the last stable version was released in 2005. Examples of programs, created with the help of this library, is shown on the Fig. 1 and 2.

The other package-Zoomable Visual Transformation Machine (ZVTM) is similar to Piccolo2D in its functional (Zoomable Visual Transformation Machine, 2013), Fig. 3. It is more focused on geo-information systems, but is much more flexible than Piccolo and updates more often. The last version was released in present 2012 year. Visualization of transformation system XML VXT was based on this system (Pietriga *et al*., 2001).

The most complex environment for the development of scale user interfaces is TopicScape (TopicScape, year) commercial system designed for three-dimensional modeling of different types of data. There are also editions for students and non commercial use, Fig. 4.

RESULTS AND DISCUSSION

Practical aspects of ZUI: Since the scaling user interfaces are have not long ago become popular, there is no much information as for the practice of its creation and application and the existing information is of advertising character, not that of the academic one. Commercial applications use the hybrid model, comprising the elements of classical WIMP-interface and ZUI elements.

This is caused by the fact that despite the evident advantages and possibilities in eliminating the restrictions from WIMP, the scaling user interface has its boundaries.

First, the space location of plethora of visual objects on the specific stage ceases to simplify the work and starts complicating it. This is caused by the fact that the man can remember the location of maximum 10-20 objects. When their number increases, a man gets confused and the infinity of the working table in ZUI complicates the visual search. Therefore the scaling interfaces are acceptable for relatively small or greatly structured sets of objects. The latter requires much attention, sometimes down to the special development of space data models-not only the essence of the object, but the way they will be realised by the final user must be considered. This also requires the special searching principles which consider different scale of visual objects (Pietriga *et al*., 2007).

It is also necessary to consider the types of the data which are under processing. ZUI works best when the miniatures for each visual object are present. This means that it is referred to graphical formats (photo and video). The text under the character of the miniatures look bad and the specific formats sort of the archives or sound files look even worse. They have to be assigned special unique diagrammatic representation or go back to static icons. This either requires the additional expenses or it is necessary to change the visual representation in correspondence with the scale. The researches carried out in the University of Maryland (Good, 2003), within the frameworks of which there had been developed the system of contextual visualization and demonstrational program Niagara are of significant importance. And, since we are talking of practically the infinite working table, the design of the space becomes also the important issue. As was mentioned above, a man can remember the location of strictly determined number of objects. If their number is bigger-there is the need the model within which it is possible to structurize the objects according with their categories and to search them by the semantic scaling. These may then be special zones, specified for different types of the objects or specific structuring control elements. Such a system was created in the University of Maryland on the base of the dynamic tree (Lee *et al*., 2006).

Apart from that, navigation with scaling, despite its simplicity and intuitional apprehensibility is an unusual for the majority of users and is associated with maps and geo-information systems but not with the other data types. Therefore the users do not pay much attention to the software with ZUI, with the exception of mobile devices, in which interface scaling is the only way to ensure the necessary functionality.

CONCLUSION

On the basis of conducted research it can be stated that scale user interfaces despite of its attractiveness and opportunities, which allow to simplify the work with visual data for the user are likely not to find wide application in steady-state systems, while scaling will become as habitual way of work with visual objects, as moving on the screen.

Maybe distribution of smartphones, pads and handheld computers, in which methods Of ZUI are widely used will be a significant factor in the sense, but for the present such tendency is not observed.

Nevertheless analysis of the sources shows that scale user interface is popular enough in academic researches and consequently deserves close attention. Researchers are held in the sphere of standardization of navigation methods, corporate design, which includes visual objects methods of visual structuring and object search as well as combining classic interfaces with ZUI. The range of software tools, based exactly on this paradigm already exists and these tools little by little win a place on the market.

It makes this theme one of the perspective directions in the sphere of ergonomics and software development.

REFERENCES

Beaudouin-Lafon, M., 2000. Instrumental interaction: An Interaction model for designing post-WIMP user interface. Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, pp: 446-453.

Bederson, B., 2004. A Structured 2D Graphics Framework. Retrieved form: http://www.piccolo2d.org/.

Bederson, B.B., 2011. The promise of zoomable user interface. Behav. Inform. Technol., 30(6): 853-866.

Charlotte, B., (year). Alan Kay and Graphical User Interface. Retrieved form: http://www.lottiebooth.com/pdf/essay.pdf.

Good, L., 2003. Zoomable User interfaces for the authoring and delivery of slide presentations. Ph.D. Thesis, University of Maryland.

Henderson, A. and S.K. Card, 1986. Rooms: The use of multiple virtual workspaces to reduce space contention in a window-based graphical user interface. ACM T. Graph., 5(3): 211-243.

Hiltzick, M.A., 1999. Dealers of Lightning: Xerox PARC and the Dawn of the Computer Age. Harper-Collins, New York.

Jeremy, R., 2005. A History of GUI. Ars Technica. Retrieved form: http://arstechnica.com/features/2005/ 05/gui/.

Lee, B., C.S. Parr, C. Plaisant, B.B. Bederson, V.D. Veksler, W.D. Gray and C. Kotfila, 2006. TreePlus: Interactive exploration of networks with enhanced tree layouts. IEEE T. Vis. Comput. Gr., 12(6): 1414-1426.

Modine, A., 2008. Apple Patents OS X Dock. Retrieved form: http://www.theregister.Co.uk/2008/10/08/apple_patents_osx_dock/.

Pietriga, E., V. Quint and J.Y. Vion-Dury, 2001. VXT: A visual approach to XML transformations. Proceeding of the ACM Symposium on Document Engineering, pp: 1-10.

Pietriga, E., C. Appert and M. Beaudouin-Lafon, 2007. An operationalization and preliminary evaluation of multi-scale searching. Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, pp: 1215-1224.

TopicScape, (year). Retrieved form: http://www.topicscape.com/onlineDemo.php.

Zoomable Visual Transformation Machine, 2013. Retrieved form: http://zvtm.sourceforge.net/.