[12] . Golenkov, N. Guliakina, V. Golovko, V. Krasnoproshin, "Methodological problems of the current state of works in the field of artificial intelligence," in *Otkrytye semanticheskie tekhnologii proektirovaniya intellektual'nykh system [Open semantic technologies for intelligent systems]*, ser. 5, V. Golenkov, Ed. BSUIR, Minsk, 2021, pp. 17–24.

[13] . Ryen, A. Soylu, and D. Roman, "Building semantic knowledge graphs from (semi-) structured data: a review," *Future Internet*, vol. 14, no. 5, p. 129, 2022.

[14] . Barnaghi, A. Sheth, and C. Henson, "From data to actionable knowledge: Big data challenges in the web of things [guest editors' introduction]," *IEEE Intelligent Systems*, vol. 28, no. 6, pp. 6–11, 2013.

[15] . Kellogg, "From data management to knowledge management," *Computer*, vol. 19, no. 01, pp. 75–84, 1986.

[16] . Shunkevich, "Universal model of interpreting logical-semantic models of intelligent computer systems of a new generation," in *Otkrytye semanticheskie tekhnologii proektirovaniya intellektual'nykh system [Open semantic technologies for intelligent systems]*, V. Golenkov, Ed. BSUIR, Minsk, 2022, p. 285–296.

[17] olenkov, V. V., "Graphodynamic models of parallel knowledge processing: principles of construction, implementation, and design," in *Open semantic technologies for designing intelligent systems (OSTIS-2012): Proceedings of the II International Scientific and Technical Conference, Minsk, February 16-18, 2012*, Belarusian State University of Informatics and Radioelectronics; editorial board: V. V. Golenkov (chief editor) [et al.]. Minsk, 2012, pp. 23–52.

[18] . Ivashenko, "General-purpose semantic representation language and semantic space," in *Otkrytye semanticheskie tekhnologii proektirovaniya intellektual'nykh system [Open semantic technologies for intelligent systems]*, ser. Iss. 6. Minsk : BSUIR, 2022, pp. 41–64.

[19] . V. Golenkov and N. A. Gulyakina, "Structuring the semantic space," in *Open semantic technologies for designing intelligent systems (OSTIS-2014): Proceedings of the IV International Scientific and Technical Conference*, V. V. Golenkov, Ed. Minsk: BSUIR, 2 2014, pp. 65–78, chief editor Golenkov, V. V. [and others].

[20] ——, "Principles of building mass semantic technology for component design of intelligent systems," in *Open semantic technologies for designing intelligent systems (OSTIS-2011): Proceedings of the international scientific and technical conference*, V. V. Golenkov, Ed. Minsk: Belarusian State University of Informatics and Radioelectronics, 2 2011, pp. 21–58, chief editor Golenkov, V. V. [et al.].

[21] . Shunkevich, "Hybrid problem solvers of intelligent computer systems of a new generation," *Otkrytye semanticheskie tekhnologii proektirovaniya intellektual'nykh system [Open semantic technologies for intelligent systems]*, no. 6, pp. 119–144, 2022.

[22] "Neo4j graph Database Platform | Graph Database Management System [Electronic resource]," April 2024. [Online]. Available: https://neo4j.com/

[23] . Kahveci and A. K. Singh, "An efficient index structure for string databases," in VLDB, vol. 1, 2001, pp. 351–360.

[24] . Barsky, U. Stege, and A. Thomo, "Structures for indexing substrings," in *Full-Text (Substring) Indexes in External Memory*. Springer, 2012, pp. 1–15.

[25] . V. Zotov, "Model of process management in shared semantic memory of intelligent systems," in *Information Technologies and Systems 2023* (ITS 2023), L. Y. Shilin, Ed. Minsk: Belarusian State University of Informatics and Radioelectronics, 11 2023, pp. 53–54, proceedings of the International Scientific Conference.

[26] . L. W. Kessels, "An alternative to event queues for synchronization in monitors," *Communications of the ACM*, vol. 20, no. 7, pp. 500–503, 1977.

[27] . A. R. Hoare, "Monitors: An operating system structuring concept," *Communications of the ACM*, vol. 17, no. 10, pp. 549– 557, 1974.

[28] . Zagorskiy, "Principles for implementing the ecosystem of nextgeneration intelligent computer systems," in *Otkrytye semanticheskie tekhnologii proektirovaniya intellektual'nykh system [Open semantic technologies for intelligent systems]*. BSUIR, Minsk, 2022, p. 347–356.

[29] . Love, *Linux system programming: talking directly to the kernel and C library*. O'Reilly Media, Inc., 2013.

[30] "Software implementation of semantic networks processing storage [Electronic resource]," 2024, mode of access: https://github. com/ostis-ai/sc-machine. — Date of access: 30.03.2024.

[31] . Bayer, "Prefix b-trees," ACM *Transactions on Database Systems (TODS)*, vol. 2, no. 1, pp. 11–26, 1977.

[32] . Ferragina and R. Grossi, "The string b-tree: A new data structure for string search in external memory and its applications," *Journal of the ACM* (JACM), vol. 46, no. 2, pp. 236–280, 1999.

[33] . Belazzougui, "Fast prefix search in little space, with applications," in *European Symposium on Algorithms*, 2010, pp. 427– 438.

[34] . I. Cole, *Algorithmic skeletons: structured management of parallel computation*. Pitman London, 1989.

[35] . Gonnord, L. Henrio, L. Morel, and G. Radanne, "A survey on parallelism and determinism," *ACM Computing Surveys*, vol. 55, no. 10, pp. 1–28, 2023.

[36] . V. Zotov, "Quantitative indicators of operations efficiency over shared semantic memory of intelligent systems," in *Information Technologies and Systems 2023 (ITS 2023)*, L. Y. Shilin, Ed. Minsk: Belarusian State University of Informatics and Radioelectronics, 11 2023, pp. 51–52, proceedings of the International Scientific Conference.

[37] . P. Miret, "Consistency models in modern distributed systems. an approach to eventual consistency," *Master. MA thesis. Universitat Politecnica de Valencia, Spain*, 2014.

ФОРМАЛЬНАЯ МОДЕЛЬ ОБЩЕЙ СЕМАНТИЧЕСКОЙ ПАМЯТИ ДЛЯ ИНТЕЛЛЕКТУАЛЬНЫХ СИСТЕМ НОВОГО ПОКОЛЕНИЯ

Зотов Н.В.

В работе подробно рассматривается формальная модель семантической памяти для интеллектуальных систем, структура, её элементы, соответствия между ними, правила и алгоритмы. Описывается реализация на основе данной модели, приводятся количественные показатели её эффективности.

Received 15.03.2024

Adaptive User Interfaces for Intelligent Systems: Unlocking the Potential of Human-System Interaction

Mikhail Sadouski, Pavel Nasevich, Maksim Orlov, Alexandra Zhmyrko
Belarusian State University of Informatics and Radioelectronics
Minsk, Belarus
Email: sadovski@bsuir.by, fallenchromium@gmail.com, orlovmaksimkonst@gmail.com, aleksashazh@gmail.com

Abstract—The paper analyzes the capabilities of computer systems and the level of development of tools for interacting with them (user interfaces). Based on the analysis, an approach to the design of adaptive user interfaces of intelligent systems based on the OSTIS Technology is proposed. The semantic model of such interfaces, proposed earlier, has been clarified and extended, the proposed architecture of such systems is given. Adaptive user interfaces designed on the basis of the proposed approach will provide new scenarios of user interaction with computer systems.

Keywords—adaptive user interface, intelligent systems, user interface of ostis-systems

I. Introduction

In the modern world, people use computer systems of various purposes daily. A key component of such systems that directly influences their efficiency is the user interface, which in a broad sense is a set of tools that provide interaction between a person and the system.

A large part of the cost of developing computer systems is in the design, testing, and development of the user interface [1].

A poorly designed user interface limits the potential of the system by increasing the threshold of entry and reducing the efficiency of interaction with users or making some interaction scenarios impossible [2].

With the development of information society, the need of users for computer systems capable of solving various classes of problems, including tasks that are difficult to formalize, has led to an increased pace of development of computer technologies, the creation of a large number of models, methods and tools for the design and development of computer systems, including intelligent computer systems with increased requirements for interoperability, component compatibility and flexibility of scenarios of interaction with the user because of their ability to self-learning and solving complex problems, as well as problems, in the initial data and algorithms of solution of which there is an influence of non-factors.

However, even computer systems that we use on a daily basis are severely limited in their functionality due to the limited means of interaction with these systems. There is a mismatch between the current level of development of user interfaces and the problem-solving capabilities of computer systems.

Each user has unique needs and the application of adaptive user interfaces for intelligent systems becomes essential. The ease and flexibility of dynamically changing user interfaces based on user tasks that are not predetermined in the design of the system becomes key and allows for greater potential for interaction with intelligent computer systems. This article analyzes the capabilities of computer systems and the level of development of tools for interacting with them (user interfaces).

The user interface is considered in this context as the language of communication between the system and the user, together with the means for such communication, emphasizing the content of the interaction more than the specific technical aspects of the implementation of the interaction. This analysis is based on a description of the historical development of computer systems (and, as a consequence, the development of the many classes of problems that computer systems solve).

Based on the analysis, an approach to the design of adaptive user interfaces of intelligent systems based on the OSTIS Technology is proposed. The proposed [3] semantic model of such interfaces is clarified and extended, and the proposed architecture of such systems is given. Adaptive user interfaces designed on the basis of the proposed approach will provide new scenarios of user interaction with computer systems.

II. Analysis of existing approaches to solving the problem

A. *Challenges of modern user interfaces*

The challenges of modern user interfaces that cause users to fail to take advantage of the full potential of computer system problem solvers can be categorized as follows:

- A mismatch between the user's skills and means of interacting with the system and the actual means of interaction provided by the system. An example is an over-complicated user interface for inexperienced users, or a user interface that does not take into account that the user is fully informed about the algorithm for solving a problem and takes time and attention away from unnecessary explanations. This leads to the fact that the user cannot predict the navigation through the user interface — his cognitive load increases, the time required to solve the problem increases [4].

- Mismatch between the task and environment for which the user interface was designed with the actual task and environment of the system. This refers to the case where the system's problem solver is capable of solving a broader and more general class of problems than the system's user interface allows, since only in such a case can the user interface alone be said to be the limiting factor in the applicability of the system — [5].
- Lack of user's ability to make changes to the interface for integration with other user interfaces and/or systems. The impossibility of building an integrated working environment (i.e. a new user interface, qualitatively different from the multitude of interfaces designed to solve each of the sub-tasks of the complex task) makes it impossible to build scenarios for automatic integration of computer system subsystems and severely limits the user's ability to integrate computer systems, because without the ability to change the user interface, the program behavior can only be changed programmatically [6], [7].

There are a lot of examples of functionalities limited on the user interface side, which are nevertheless technically realizable, but we will limit ourselves to a few significant ones:

- Lack of integration of system components: your word processor does not prompt you to open the last quarter's financial report, even though you were emailed it yesterday and are scheduled to check it on your electronic calendar today.
- Lack of automatic adaptation to the environment: the online store does not offer you to make a pickup to the branch you are closest to at the moment.
- Lack of ability to personalize the user interface: your fitness app does not allow you to fully customize the start screen with health metrics that are irrelevant to you and with workout suggestions that do not match your goal and access to fitness equipment
- Solving complex tasks: for an average user the interface for automated solution of a multi-step task is not standard and is always available. For example, to process a call or a letter from a customer, enter relevant information from the letter into the project accounting system, assign a person with the necessary competence to be responsible for the project and send the customer a letter with the contact of the person responsible for the project.

It is worth noting that for each of these examples it is possible to find a counterexample: a computer system that did take into account the described use case, but it is worth recognizing that in general the described limitations exist in the vast majority of computer systems.

B. *Overview of the evolution of computer systems and their user interfaces*

Let us consider the development paths of computer systems and their user interfaces - the major milestones, and what they were driven by (discoveries or user needs).

Over the last 60 years, a large number of approaches to user interface design have been developed. The approaches focused on different directions, such as "which side of the interaction is primary"(the emotional state of the user, the purpose of the system, feedback from the user, the design of the program that implements the algorithm for solving the problem) and "how to convey the meaning of the objects of the user interface"— confrontation of metaphorical and idiomatic approach [8], [9].

At the same time, the paradigms used to implement the user interface have evolved. For example, the transition from a model where each interaction has an underlying function call to perform the process and provide feedback to an object-oriented interface where the UI elements correspond to the properties of the system entities (reactive approach).

Currently, the approach when the user interface is described by a set of states and transitions in the algorithm of problem solving is popular (i.e. the paradigm of so-called "wizards which leads the user step-by-step from the need to its resolution). Each of the approaches has a place, but a special place in the design of user interfaces for intelligent computer systems, based on the dynamic nature of the tasks that can be set before it, takes the paradigm of interface design based on the problem being solved and the algorithm for its solution. To date, there are no comprehensive means of building adaptive user interfaces (taking into account the environment, the characteristics of the user and his device), while dynamically taking into account the information about the algorithm for solving the problem.

C. *Conclusions*

Based on the representation of the user interface described in I, we can conclude that the problems described in II-A are the result of a methodological error in the design of interfaces. At the moment, the factors affecting how the user interface should look like are taken into